Oracle® Fusion Middleware
Upgrading Oracle Business Intelligence

12*c* (12.2.1.3.0)
E96170-03
July 2020

ORACLE®

Oracle Fusion Middleware Upgrading Oracle Business Intelligence, 12*c* (12.2.1.3.0)

E96170-03

Contents

3   Upgrading Oracle Business Intelligence from 11g (Out-of-Place Migration)

4   Post-Migration Tasks for Oracle BI EE

5   Post-Migration Tasks for Oracle BI Publisher

6   Post-Migration Tasks for Essbase

7   Upgrading Oracle Business Intelligence from a Previous 12c Release

8   Upgrading Scaled-Out Systems from a Previous 12c Release

9   Post-Upgrade Tasks

A   Known Differences: Oracle Business Intelligence 12c

B   Troubleshooting a Migrated Oracle BI Instance

ORACLE®

C   Migrating Security Certificates when Using Secure LDAP

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

E   Stopping Servers and Processes

F   Starting Servers and Processes

Preface

This document describes how to upgrade an existing Oracle Business Intelligence environment to 12*c* (12.2.1.3.0).

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

- Documentation Accessibility

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

- Conventions

Audience

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

This document is intended for system administrators who are responsible for installing, maintaining, and upgrading Oracle Business Intelligence. It is assumed that readers have knowledge of the following:

- Oracle Fusion Middleware system administration and configuration.

- Configuration parameters and expected behavior of the system being upgraded.

Documentation Accessibility

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

Access to Oracle Support

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

Related Documents

Upgrade documentation is organized by tasks in the 12*c* 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 Oracle Business Intelligence information, see Oracle Business Intelligence Enterprise Edition 12.2.1.3.0.

- For installation information, see Fusion Middleware Installation Documentation.

- For upgrade information, see Fusion Middleware 12*c* Upgrade Documentation.

- For administration-related information, see Fusion Middleware 12*c* Administration Documentation.

- For release-related information, see Fusion Middleware 12*c* 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.
italic	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
`monospace`	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

1
Introduction to Upgrading Oracle Business Intelligence to 12*c*

When you are considering a new release of Oracle Business Intelligence, the tools and documentation required vary, depending on whether you are performing an upgrade or applying a patchset, a bundle patch, or a one-off patch. Upgrading an Oracle Business Intelligence system to 12*c* requires careful preparation, planning, and testing. Oracle provides tools and technology to automate much of the upgrade process.

Migrating the metadata and configuration from Oracle Business Intelligence 11*g* (11.1.1.7, 11.1.1.8, and 11.1.1.9) to 12*c* is an out-of-place process performed by using the BI Migration Tool. This tool creates an Oracle BI 11*g* metadata archive which contains system security configuration information, the Oracle BI repository, and the Oracle BI Presentation Services Catalog of the 11*g* system.

Upgrading Oracle Business Intelligence from a previous 12*c* release to 12.2.1.3.0 is an in-place upgrade performed by using the Upgrade Assistant. You install the 12.2.1.3.0 product distribution in a new Oracle home and then use the Upgrade Assistant to upgrade the existing 12*c* schemas and domain configuration. You also use the Reconfiguration Wizard to reconfigure the existing 12*c* domain.

> ✎ **Note:**
>
> For general information about Fusion Middleware upgrade planning and other upgrade concepts and resources, see the following sections in *Planning an Upgrade of Oracle Fusion Middleware*:
>
> • Planning an Upgrade to Oracle Fusion Middleware
>
> • Understanding In-Place versus Out-of-Place Upgrades
>
> • Understanding the Basic 12*c* Upgrade Tasks

The following topics describe the concepts related to upgrading Oracle Business Intelligence:

• About the Starting Points for Oracle Business Intelligence Upgrade
  You can upgrade to Oracle Business Intelligence 12*c* (12.2.1.3.0) from supported 11*g* and 12*c* releases.

• About the Oracle Business Intelligence Standard Topology
  The steps to upgrade Oracle Business Intelligence to 12*c* (12.2.1.3.0) depend on your existing production topology.

• About Upgrade Restrictions
  If you are using two or more Oracle Fusion Middleware products of the same or different versions in a single, supported, Oracle Fusion Middleware configuration,

you must consider the interoperability and compatibility factors before planning the upgrade.

About the Starting Points for Oracle Business Intelligence Upgrade

You can upgrade to Oracle Business Intelligence 12*c* (12.2.1.3.0) from supported 11*g* and 12*c* releases.

Supported release starting points are:

- Oracle Business Intelligence 11*g* Release 1 (11.1.1.7, 11.1.1.8, and 11.1.1.9)

- Oracle Business Intelligence 12*c* (12.1.2.0.0, 12.1.3.0.0, 12.2.1.0.0, 12.2.1.1.0, and 12.2.1.2.0)

If your existing version of Oracle Business Intelligence is earlier than 11*g* Release 1 (11.1.1.6), you must first upgrade your software to one of the following supported versions before you can upgrade to 12*c* (12.2.1.3.0):

- To upgrade to 11*g* Release 1 (11.1.1.6), see *Oracle Fusion Middleware Upgrade Guide for Oracle Data Integrator* in the 11*g* Release 1 (11.1.1.6) documentation library.

- To upgrade to 11*g* Release 1 (11.1.1.7), see *Oracle Fusion Middleware Upgrade Guide for Oracle Data Integrator* in the 11*g* Release 1 (11.1.1.7) documentation library.

To upgrade from 11*g*, see Upgrading Oracle Business Intelligence from 11*g* (Out-of-Place Migration).

To upgrade from a previous 12*c* release, see Upgrading Oracle Business Intelligence from a Previous 12*c* Release.

About the Oracle Business Intelligence Standard Topology

The steps to upgrade Oracle Business Intelligence to 12*c* (12.2.1.3.0) depend on your existing production topology.

As a result, it is difficult to provide exact upgrade instructions for every possible Oracle Business Intelligence installation. This upgrade documentation provides detailed instructions for upgrading BI and is referred to as standard upgrade topology.

Your actual topology may vary, but the topology described in this guide provide an example that can be used as a guide to upgrade BI.

Figure 1-1 shows the standard upgrade topology for Oracle Business Intelligence.

> **✎ Note:**
>
> Because you are upgrading from a previous 12*c* release, the standard topology remains unchanged.

Figure 1-1   Oracle Business Intelligence Standard Upgrade Topology
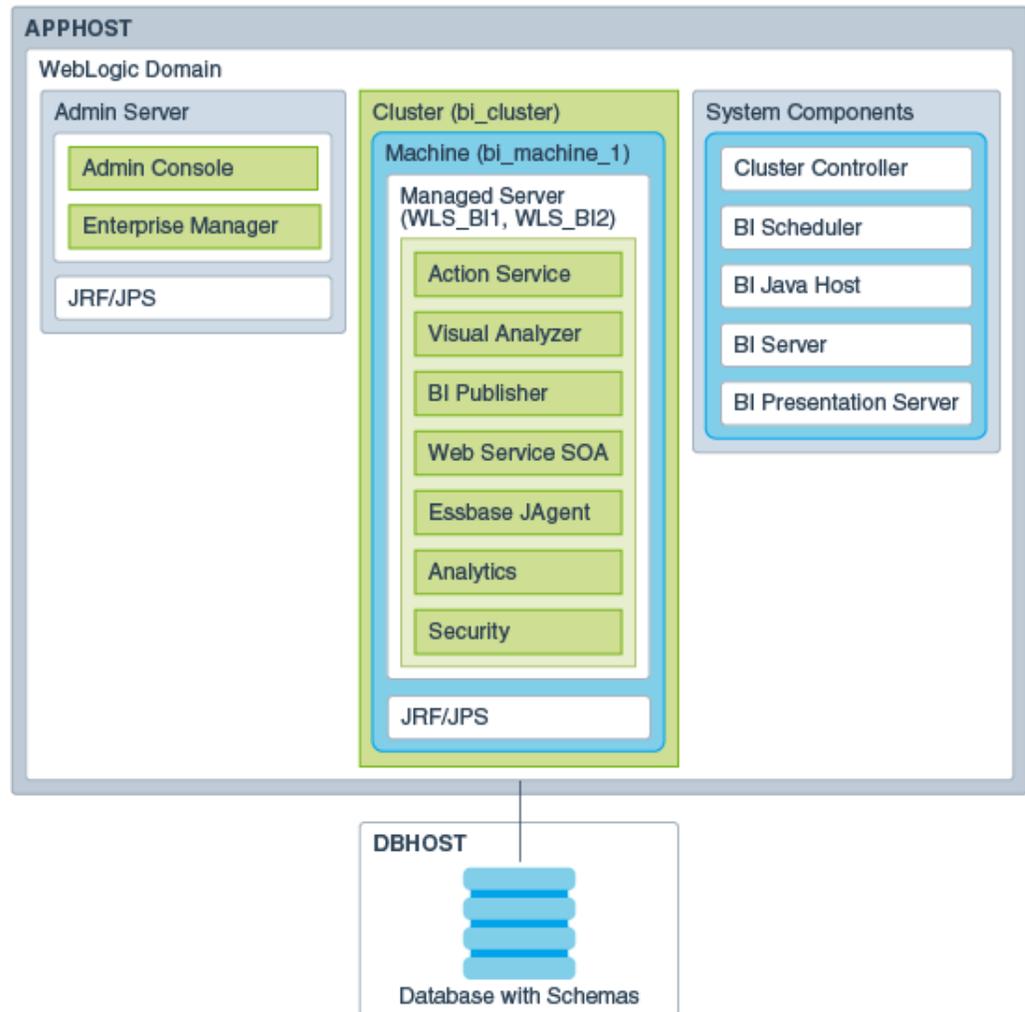
About Upgrade Restrictions

If you are using two or more Oracle Fusion Middleware products of the same or different versions in a single, supported, Oracle Fusion Middleware configuration, you must consider the interoperability and compatibility factors before planning the upgrade.

Interoperability

In the context of Oracle Fusion Middleware products, interoperability is defined as the ability of two Oracle Fusion Middleware products or components of the same version (or release) to work together (interoperate) in a supported Oracle Fusion Middleware configuration. Specifically, interoperability applies when the first 4 digits of the release or version number are the same. For example, Oracle Fusion Middleware 12*c* (12.2.1.0) components are generally interoperable with other 12*c* (12.2.1.3.0) components.

Compatibility

In the context of Oracle Fusion Middleware products, compatibility is defined as the ability of two Oracle Fusion Middleware components of different versions (or releases) to interoperate.

For a list of products and features available in Oracle Fusion Middleware Release 12.2.1.3.0, see Products and Features Available in Oracle Fusion Middleware in *Understanding Interoperability and Compatibility*.

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

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

2

Pre-Upgrade Requirements

Before you begin the upgrade to Oracle Business Intelligence 12*c* (12.2.1.3.0), you must perform pre-upgrade tasks such as backing up, cloning your current environment, and verifying that your system meets certified requirements.

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

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

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

- Cloning Your Production Environment for Testing
  Create a copy of your actual production environment, upgrade the cloned environment, verify that the upgraded components work as expected, and then (and only then) upgrade your production environment.

- Verifying Certification and System Requirements
  Review the certification matrix and system requirements documents to verify that your environment meets the necessary requirements for installation.

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

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

- Creating 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.

- Disabling Internal SSL
  You must disable SSL on the internal communication links before you start the upgrade process.

- Creating a Baseline Report with the Oracle BI Baseline Validation Tool
  The Oracle BI Baseline Validation Tool lets you compare two system installations. Run this tool on your pre-upgrade environment to extract result sets from your existing BI deployment and generate comparison files in order to determine whether any changes are needed before making a modified system available to your user community.

Pre-Upgrade Checklist

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

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

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

This table describes the Pre-Upgrade Checklist. It lists all the required components and describes them in detail.

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

Task	Description
Required Create a complete backup of your existing environment.	Back up all system-critical files and databases that contain any schemas that are to be upgraded. If the upgrade fails, you must restore your pre-upgrade environment and begin the upgrade again. See Creating a Complete Backup. • Make sure that your backup includes the schema version registry table. See Backing Up the Schema Version Registry Table. • If you have modified or customized any of the startup scripts or any of the configuration files in your existing domain (for example, setting a value for the cookie-path property), you need to copy them to the temporary directory location (outside of the existing domain) during the upgrade, and redeploy them after the upgrade.
Optional Clone your production environment to use as an upgrade testing platform.	In addition to creating a complete backup of your system files, Oracle strongly recommends that you clone your production environment. This environment can be used to test the upgrade. See Cloning Your Production Environment for Testing.

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

Task	Description
Required Verify that you install and upgrade your product on a supported hardware and software configuration. **CAUTION**: Do not attempt an upgrade if you are unable to use the latest supported operating system. As with all supported configurations, failure to comply with these requirements may cause your upgrade to fail.	Verify that your hardware and software configurations (including operating systems) are supported by the latest certifications and requirements documents. Also make sure to use a supported JDK version before you install the product distributions. Oracle recommends that you verify this information right before you start the upgrade as the certification requirements are frequently updated. **Make sure that you have applied the latest patches to your components before you upgrade.** See Verifying Certification and System Requirements.
Required for 32–bit Operating Systems Only Migrate to a 64-bit operating system before you can upgrade.	This is required only if you are currently running an unsupported 32–bit operating system.
Optional Update the security policy files if you are using enhanced encryption (AES 256).	Some of the security algorithms used in Fusion Middleware 12*c* require additional policy files for the JDK. If you plan to use enhanced encryption, such as AES 256, Oracle recommends that you apply the latest required policy files to the JDK before you upgrade. See Updating Policy Files when Using Enhanced Encryption (AES 256).
Optional Purge any outdated or unused data before you upgrade.	To optimize performance, Oracle strongly recommends that you purge the data and objects that are not used in the upgraded environment. See Purging Unused Data.
Required for Oracle Database Users Only Before you upgrade an Edition-Based Redefinition (EBR) enabled schema, you must connect to the database server and create an edition on the database server for 12*c* (12.2.1.3.0).	If you are using an Edition-Based Redefinition (EBR) database, you must create the edition before you start the upgrade.
Optional Create a Non-SYSDBA user to run the Upgrade Assistant.	Oracle recommends that you create the FMW user to run the Upgrade Assistant. The FMW user can run the Upgrade Assistant without any system administration privileges. See Creating a Non-SYSDBA User to Run the Upgrade Assistant.
Optional Identify the schemas that are currently in your domain before you begin.	It is important that you know the schemas that are in your pre-upgrade domain before you start the upgrade. You should know the schema owner names and passwords, as well as the versions of each schema.

Creating a Complete Backup

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

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

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

See:

- Backing Up Your Environment in *Administering Oracle Fusion Middleware*

- Upgrading and Preparing Your Oracle Databases for 12*c* in *Planning an Upgrade of Oracle Fusion Middleware*

- Backing Up the Schema Version Registry Table
  Your system backup must include the `SYSTEM.SCHEMA_VERSION_REGISTRY$` table or the `FMWREGISTRY.SCHEMA_VERSION_REGISTRY$` table.

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

Backing Up the Schema Version Registry Table

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

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

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

Maintaining Customized Domain and Environment Settings

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

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

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

setUserOverrides.sh (UNIX) and configure it to add custom libraries to the WebLogic Server classpath, specify additional command-line options for running the servers, or specify additional environment variables. When using the pack and unpack commands, any custom settings that you add to this file are preserved during the domain upgrade operation and are carried over to the remote servers.

The following example illustrates startup customizations in a setUserOverrides file:

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

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

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

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

Cloning Your Production Environment for Testing

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

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

Upgrades cannot be reversed. In most cases, if an error occurs, you must stop the upgrade and restore the entire environment from backup and begin the upgrade process from the beginning. Identifying potential upgrade issues in a development environment can eliminate unnecessary downtime.

ORACLE

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

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

- Uncover and correct any upgrade issues.

- Practice completing an end-to-end upgrade.

- Understand the upgrade performance and how purge scripts can help.

- Understand the time required to complete the upgrade.

- Understand the database resource usage (such as temporary tablespace; PGA, and so on).

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

Verifying Certification and System Requirements

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

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

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

ORACLE

- Verify Your Environment Meets Certification Requirements
  Oracle has tested and verified the performance of your product on all certified
  systems and environments. Make sure that you are installing your product on a
  supported hardware or software configuration.

- Verify System Requirements and Specifications
  It is important to verify that the system requirements such as disk space, available
  memory, specific platform packages and patches, and other operating system-
  specific items are met.

- Verify That the Database Hosting Oracle Fusion Middleware is Supported
  You must have a supported Oracle database configured with the required
  schemas before you run Oracle Fusion Middleware 12*c* (12.2.1.3.0).

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

Verify Your Environment Meets Certification Requirements

Oracle has tested and verified the performance of your product on all certified systems
and environments. Make sure that you are installing your product on a supported
hardware or software configuration.

Whenever new certifications occur, they are added to the appropriate certification
document right away. New certifications can occur at any time, and for this reason
the certification documents are kept outside of the documentation libraries and
are available on Oracle Technology Network. See the Certification Matrix for 12*c*
(12.2.1.3.0).

Verify System Requirements and Specifications

It is important to verify that the system requirements such as disk space, available
memory, specific platform packages and patches, and other operating system-specific
items are met.

Use the *Oracle Fusion Middleware System Requirements and Specifications*
document to verify that the requirements of the certification are met. For example,
if the Certification Matrix for 12*c* (12.2.1.3.0) indicates that your product is certified
for installation on 64-Bit Oracle Linux 7, the System Requirements and Specifications
document should be used to verify that your Oracle Linux 7 system has met the
required minimum specifications such as disk space, available memory, specific
platform packages and patches, and other operating system-specific items. This
document is updated as needed and resides outside of the documentation libraries
on the Oracle Technology Network (OTN).

> **Note:**
>
> When you install the Oracle Fusion Middleware Release 12c software
> in preparation for upgrade, you should use the same user account that
> you used to install and configure the existing, pre-upgrade Oracle Fusion
> Middleware software. On UNIX operating systems, this ensures that the
> proper owner and group is applied to new Oracle Fusion Middleware 12c
> files and directories.

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

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

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

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

Make sure to validate the migration to ensure all your Oracle Fusion Middleware 12*c* (12.2.1.2.0) software is working properly on the 64-bit machine, and only then perform the upgrade to Oracle Fusion Middleware 12*c* (12.2.1.3.0).

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

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

Upgrading an operating system typically involves the following:

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

- Procure the Hardware That Supports the Upgrade's 64-bit Software Requirement
  Make sure that you have supported target hardware in place before you begin the upgrade process.

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

- Back Up All Files from the 32-bit Host Machine
  Make sure that you have created a complete backup of your entire 12*c* (12.2.1.2.0) deployment before you begin the upgrade process. These files can be used if there is an issue during the migration and you have to restart the process.

- Set Up the Target 64-bit Machine with the 12c (12.2.1.2.0) Host Name and IP Address
  The host name and IP address of the target machine must be made identical to the host. This requires you to change the IP address and name of the source machine or decommission the source machine to avoid conflicts in the network.

- Restore the 12c (12.2.1.2.0) Backup from 32-bit Host to 64-bit Host
  Restore the files you backed from the 32-bit host using the same directory
  structure that was used in 12*c* (12.2.1.2.0). The directory structure on the target
  machine must be identical to the structure of the host machine.

- Install the 12c (12.2.1.3.0) Product Distributions on the Target Machine
  Oracle recommends an Out-of-Place approach for upgrade. Therefore, you must
  install the 12*c* (12.2.1.3.0) product distributions in a new Oracle home on the
  target machine.

- Upgrade the Target 64-bit Environment Using the Standard Upgrade Procedure
  After installing the product on the target machine, you must upgrade each product
  component individually using an Upgrade Utility specified in the component-
  specific upgrade guide and complete any post-upgrade tasks.

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

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

Stop All Processes

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

Stop the Managed Servers

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

- (UNIX) *EXISTING_DOMAIN_HOME*`/bin/stopManagedWebLogic.sh`
  `managed_server_name admin_url`

- (Windows) *EXISTING_DOMAIN_HOME*`\bin\stopManagedWebLogic.cmd`
  `managed_server_name admin_url`

When prompted, enter your user name and password.

Stop the Administration Server

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

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

- (UNIX) *EXISTING_DOMAIN_HOME*`/bin/stopWebLogic.sh`

- (Windows) *EXISTING_DOMAIN_HOME*`\bin\stopWebLogic.cmd`

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

Stop Node Manager

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

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

Back Up All Files from the 32-bit Host Machine

Make sure that you have created a complete backup of your entire 12*c* (12.2.1.2.0) deployment before you begin the upgrade process. These files can be used if there is an issue during the migration and you have to restart the process.

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

See Backing Up Your Environment in *Oracle Fusion Middleware Administrator's Guide*.

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

- *12c_DOMAIN_HOME*

- *12c*/nodemanager directory located in *12c_ORACLE_HOME*/wlserver/ common/

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

Set Up the Target 64-bit Machine with the 12*c* (12.2.1.2.0) Host Name and IP Address

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

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

Restore the 12*c* (12.2.1.2.0) Backup from 32-bit Host to 64-bit Host

Restore the files you backed from the 32-bit host using the same directory structure that was used in 12*c* (12.2.1.2.0). The directory structure on the target machine must be identical to the structure of the host machine.

See Recovering Your Environment in *Oracle Fusion Middleware Administrator's Guide*.

Install the 12*c* (12.2.1.3.0) Product Distributions on the Target Machine

Oracle recommends an Out-of-Place approach for upgrade. Therefore, you must install the 12*c* (12.2.1.3.0) product distributions in a new Oracle home on the target machine.

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

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

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

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

> **✎ Note:**
>
> The Node Manager upgrade procedure requires access to the original Node Manager files. Use the 11*g* Node Manger files that you backed up from the 32-bit source machine as part of Back Up All Files from the 32-bit Host Machine.

Verify That the Database Hosting Oracle Fusion Middleware is Supported

You must have a supported Oracle database configured with the required schemas before you run Oracle Fusion Middleware 12*c* (12.2.1.3.0).

Review the Fusion Middleware database requirements before starting the upgrade to ensure that the database hosting Oracle Fusion Middleware is supported and has sufficient space to perform an upgrade. See the Certification Matrix for 12*c* (12.2.1.3.0).

> **✎ Note:**
>
> If your database version is no longer supported, you must upgrade to a supported version before starting an upgrade. See Upgrading and Preparing Your Oracle Databases for 12*c* (12.2.1.3.0) in *Planning an Upgrade of Oracle Fusion Middleware*.

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

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

Refer to the Oracle Fusion Middleware Supported System Configurations information on the Oracle Technology Network (OTN) to verify that the JDK you are using is supported.

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

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

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

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

Updating Policy Files when Using Enhanced Encryption (AES 256)

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

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

Some of the security algorithms used in Fusion Middleware 12*c* require additional policy files for the JDK. See Java Cryptography Architecture Oracle Providers Documentation.

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

Purging Unused Data

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

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

Creating a Non-SYSDBA User to Run the Upgrade Assistant

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

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

> **✎ Notes:**
>
> The non-SYSDBA user FMW is created solely for the purpose of running the Upgrade Assistant. After this step is complete, drop the FMW user. Note that privileges required for running the Upgrade Assistant may change from release to release.
> By default, the `v$xatrans$` table does not exist. You must run the `XAVIEW.SQL` script to create this table before creating the user. Moreover, the `grant select` privilege on the `v$xatrans$` table is required only by Oracle Identity Governance. If you do not require Oracle Identity Governance for configuration, or if you do not have the `v$xatrans$` table, then remove the following line from the script:
>
> ```
> grant select on v$xatrans$ to FMW with grant option;
> ```

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

```
create user FMW identified by password;
grant dba to FMW;
grant execute on DBMS_LOB to FMW with grant option;
grant execute on DBMS_OUTPUT to FMW with grant option;
grant execute on DBMS_STATS to FMW with grant option;
grant execute on sys.dbms_aqadm to FMW with grant option;
grant execute on sys.dbms_aqin to FMW with grant option;
grant execute on sys.dbms_aqjms to FMW with grant option;
grant execute on sys.dbms_aq to FMW with grant option;
grant execute on utl_file to FMW with grant option;
grant execute on dbms_lock to FMW with grant option;
grant select on sys.V_$INSTANCE to FMW with grant option;
```

```
grant select on sys.GV_$INSTANCE to FMW with grant option;
grant select on sys.V_$SESSION to FMW with grant option;
grant select on sys.GV_$SESSION to FMW with grant option;
grant select on dba_scheduler_jobs to FMW with grant option;
grant select on dba_scheduler_job_run_details to FMW with grant option;
grant select on dba_scheduler_running_jobs to FMW with grant option;
grant select on dba_aq_agents to FMW with grant option;
grant execute on sys.DBMS_SHARED_POOL to FMW with grant option;
grant select on dba_2pc_pending to FMW with grant option;
grant select on dba_pending_transactions to FMW with grant option;
grant execute on DBMS_FLASHBACK to FMW with grant option;
grant execute on dbms_crypto to FMW with grant option;
grant execute on DBMS_REPUTIL to FMW with grant option;
grant execute on dbms_job to FMW with grant option;
grant select on pending_trans$ to FMW with grant option;
grant select on dba_scheduler_job_classes to FMW with grant option;
grant select on sys.DBA_TABLESPACE_USAGE_METRICS to FMW with grant
option;
grant select on SYS.DBA_DATA_FILES to FMW with grant option;
grant select on SYS.V_$ASM_DISKGROUP to FMW with grant option;
grant select on v$xatrans$ to FMW with grant option;
grant execute on sys.dbms_system to FMW with grant option;
grant execute on DBMS_SCHEDULER to FMW with grant option;
grant select on dba_data_files to FMW with grant option;
grant execute on UTL_RAW to FMW with grant option;
grant execute on DBMS_XMLDOM to FMW with grant option;
grant execute on DBMS_APPLICATION_INFO to FMW with grant option;
grant execute on DBMS_UTILITY to FMW with grant option;
grant execute on DBMS_SESSION to FMW with grant option;
grant execute on DBMS_METADATA to FMW with grant option;
grant execute on DBMS_XMLGEN to FMW with grant option;
grant execute on DBMS_DATAPUMP to FMW with grant option;
grant execute on DBMS_MVIEW to FMW with grant option;
grant select on ALL_ENCRYPTED_COLUMNS to FMW with grant option;
grant select on dba_queue_subscribers to FMW with grant option;
grant execute on SYS.DBMS_ASSERT to FMW with grant option;
grant select on dba_subscr_registrations to FMW with grant option;
grant manage scheduler to FMW;
```

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

```
grant execute on SYS.DBMS_FLASHBACK to fmw with grant option;
grant execute on sys.DBMS_SHARED_POOL to fmw with grant option;
grant execute on SYS.DBMS_XMLGEN to FMW with grant option;
grant execute on SYS.DBMS_DB_VERSION to FMW with grant option;
grant execute on SYS.DBMS_SCHEDULER to FMW with grant option;
grant execute on SYS.DBMS_SQL to FMW with grant option;
grant execute on SYS.DBMS_UTILITY to FMW with grant option;
grant ctxapp to FMW with admin option;
grant execute on SYS.DBMS_FLASHBACK TO FMW with grant option;
grant create MATERIALIZED VIEW to FMW with admin option;
grant all on SCHEMA_VERSION_REGISTRY TO FMW with grant option;
grant create SYNONYM to FMW with admin option;
```

```
grant execute on CTXSYS.CTX_ADM to FMW with grant option;
grant execute on CTXSYS.CTX_CLS TO FMW with grant option;
grant execute on CTXSYS.CTX_DDL TO FMW with grant option;
grant execute on CTXSYS.CTX_DOC TO FMW with grant option;
grant execute on CTXSYS.CTX_OUTPUT TO FMW with grant option;
grant execute on CTXSYS.CTX_QUERY TO FMW with grant option;
grant execute on CTXSYS.CTX_REPORT TO FMW with grant option;
grant execute on CTXSYS.CTX_THES TO FMW with grant option;
grant execute on CTXSYS.CTX_ULEXER TO FMW with grant option;
grant create JOB to FMW with admin option;
```

Disabling Internal SSL

You must disable SSL on the internal communication links before you start the upgrade process.

To disable internal SSL:

1. Stop the system by entering the following command:

   (UNIX) *EXISTING_DOMAIN_HOME*/bitools/bin/stop.sh

   (Windows) *EXISTING_DOMAIN_HOME*\bitools\bin\stop.cmd

2. Enter the following command to disable SSL on WebLogic internal channels and internal components:

   (UNIX) *EXISTING_DOMAIN_HOME*/bitools/bin/ssl.sh internalssl false

   (Windows) *EXISTING_DOMAIN_HOME*\bitools\bin\ssl.cmd internalssl false

3. Restart the system by entering the following command:

   (UNIX) *EXISTING_DOMAIN_HOME*/bitools/bin/start.sh

   (Windows) *EXISTING_DOMAIN_HOME*\bitools\bin\start.cmd

Creating a Baseline Report with the Oracle BI Baseline Validation Tool

The Oracle BI Baseline Validation Tool lets you compare two system installations. Run this tool on your pre-upgrade environment to extract result sets from your existing BI deployment and generate comparison files in order to determine whether any changes are needed before making a modified system available to your user community.

> **Note:**
>
> You can download the Oracle BI Validation Tool along with the other Oracle Business Intelligence download on the Oracle Technology Network. See the Readme document that is included in the Oracle BI Validation Tool download for more information. For specific information on the distributions you want to download for each product, see Oracle Fusion Middleware Download, Installation, and Configuration Readme Files page.

To know more about using the Oracle BI Baseline Validation Tool, see *Comparing Oracle Business Intelligence Deployments Using the Oracle Business Intelligence Baseline Validation Tool*. This documentation is available with the Oracle BI Baseline Validation Tool.

3

Upgrading Oracle Business Intelligence from 11*g* (Out-of-Place Migration)

Oracle has introduced many enhancements to existing features in the 12*c* release for Business Intelligence. Sometimes, these enhancements replace the existing functionality or reimplement it in a different way. Wherever possible, the existing 11*g* functionality and configuration is migrated to the corresponding 12*c* system. Although the appearance and behavior of the 12*c* system can be different, the end result is expected to be functionally equivalent. Review the flowchart and roadmap for an overview of the migration process for Oracle Business Intelligence from a 11*g* release.

An efficient migration strategy allows you to migrate your metadata and configuration from Oracle BI 11*g* to the 12*c* environment. The goal of this process is to not exactly replicate the appearance and behavior of the original 11*g* system in the 12*c* environment. Replicating the 11*g* appearance and behavior is time-consuming and sometimes difficult, if not impossible. For example, the dashboards and prompts are represented differently in 12*c* and would require significant manual intervention to recreate the 11*g* appearance. It also undermines the rationale behind moving on to Oracle BI 12*c*; namely, to take advantage of the enhancements that are introduced in 12*c*.

- **About the Oracle Business Intelligence Migration Process**
  During migration, you migrate your 11*g* metadata and configuration of the BI components that you have installed on your system. Migration is a two-step process. In the first step, you create an export bundle from a read-only 11*g* certified Release (11.1.1.7.0 or later) by using the BI Migration Tool (bi-migration-tool.jar). In the second step, you import the export bundle in to the 12*c* system by using the BI Migration Script (migration-tool.sh).

- **11g to 12c Migration Prerequisites**
  Ensure that you configure the environment as per Oracle recommendations in readiness for the migration.

- **Installing the Product Distributions**
  Before you begin your upgrade, download Oracle Fusion Middleware Infrastructure and Oracle Business Intelligence 12*c* (12.2.1.3.0) distributions on the target system and install them by using Oracle Universal Installer.

- **Creating the Required 12c Schemas**
  When upgrading from 11*g*, you must create the required 12*c* schemas. You can use the Repository Creation Utility (RCU) to create customized schemas or, optionally, you can use the Upgrade Assistant to create schemas using the default schema settings. This procedure describes how to create schemas using the RCU. Information about using the Upgrade Assistant to create schemas is covered in the upgrade procedures.

- **Configuring Oracle BI Domain with the Configuration Assistant**
  The Oracle BI 12*c* Configuration Assistant is a `.sh` (UNIX) or `.cmd` (Windows) file. Start the Configuration Assistant by starting the config executable from the

`bi/bin` directory inside the Oracle home. Follow the procedure in this topic to complete the configuration step-by-step.

- Stopping Servers and Processes
  Before you migrate your 11*g* metadata and configuration to 12*c*, stop the Oracle BI 12*c* instance, OBIS1, OBIPS1, Administration Servers, Managed Servers, and the Node Manager.

- Generating the BI Migration Tool (bi-migration-tool.jar)
  The BI Migration Tool (bi-migration-tool.jar) is self-executing and self-contained. You must generate the BI Migration Tool jar file by using the BI Migration Script (migration-tool.sh|cmd). The BI Migration Script is made available after you configure the 12*c* domain. This step packages the components of the BI Migration Tool into a single, self-executing jar file, so that it can be easily transported on to an 11*g* system. To do this, you must have access to a 12*c* system with a configured domain. You must set up the 12*c* domain to provide sufficient infrastructure to run the BI Migration Tool and allow it to repackage itself. The 12*c* domain can be empty at this stage.

- Creating the Export Bundle
  The export bundle is a ".jar " file and consists of the metadata information from the 11*g* Oracle home.

- Importing with the BI Migration Script (migration-tool.sh|cmd)
  You must import the export bundle using the BI Migration Script (migration-tool.sh|cmd). The BI Migration Script automatically determines the Oracle home and the Domain home directories.

- Starting Servers and Processes
  After you migrate your 11*g* metadata and configuration to 12*c*, start the Oracle BI 12*c* instance, OBIS1, OBIPS1, Administration Servers, Managed Servers, and the Node Manager.

- Validating the Oracle BI Deployments
  The Oracle BI Baseline Validation Tool enables you to identify differences during life cycle operations, such as migrating from the Oracle BI 11*g* release to the 12*c* release. After you complete the migration procedure, you can use this tool to compare the two deployments and verify whether the results from the 11*g* environment are the same as the results from the 12*c* environment.

About the Oracle Business Intelligence Migration Process

During migration, you migrate your 11*g* metadata and configuration of the BI components that you have installed on your system. Migration is a two-step process. In the first step, you create an export bundle from a read-only 11*g* certified Release (11.1.1.7.0 or later) by using the BI Migration Tool (bi-migration-tool.jar). In the second step, you import the export bundle in to the 12*c* system by using the BI Migration Script (migration-tool.sh).

> **✎ Note:**
>
> The export process is read-only. Data, configuration, or existing binaries are not modified or deleted on the source system. During export, the metadata and configuration (specifically the data model and connection pools), the catalog content, and the security store authorization policy are retained. You must reconfigure the following:
>
> - The WebLogic authentication configuration
>
>   WebLogic does not support migration from 11*g* to 12*c*. Therefore, you must reconfigure the security realm in 12*c* as it is not a part of the 11*g* to 12*c* migration. If your users and groups are in an external LDAP, you must configure your Oracle BI 12*c* to point to the external LDAP. If your Oracle BI 11*g* users were hosted in the WebLogic LDAP, you can use the WebLogic Server to export users from 11*g* in to 12*c* WebLogic LDAP. However, 12*c* does not support a BI System User. Therefore, you must delete the BI System User after you import it in to the 12*c* system.
>
> - Mid-tier database content (such as BI Publisher schedules, the job history of agents, scorecard annotations, and usage tracking tables)
>
>   The Agents are migrated from 11*g* to 12*c* in a disabled state. You must reenable them after the migration. Reenabling the agents recreates the entries in the scheduler database. However, the Agent history is not migrated from 11*g* to 12*c*. You must also reconfigure Usage Tracking.
>
> - Application-specific data such as TimesTen aggregates, the global cache, required database schemas, and Essbase applications, data, outlines, rules, and calculations
>
> The import process is offline. During import, metadata content is deployed to customize the specified service instance and it overwrites the existing configuration settings.

The following components are migrated:

- Oracle BI Server: This includes the NQSConfig.INI and the opmn.xml files.

- Oracle BI Presentation Services: This includes the instanceconfig.xml file.

- Oracle BI Publisher: This includes the scanner.xml, xdo.cfg, xmlp-server-config.xml, datasources.xml, xdodelivery.cfg, cupsprinters.cfg files, the Map directory, and the Plugins directory.

The following metadata is migrated:

- Content (WebCat), model (RPD), and the authorization policy securing it (Java AuthoriZatioN (JAZN).

The following OPMN properties contained in the opmn.xml file are not migrated to 12*c* during configuration migration:

- obips

- obiccs

- obijh

- obisch

However, the OBIS property is migrated.

> **Note:**
>
> The 11*g* system is left unchanged during and after the migration process is complete.

The following flowchart shows the flow of the migration process.

Figure 3-1   Migration Process Flowchart for Oracle Business Intelligence from 11*g* to 12*c*
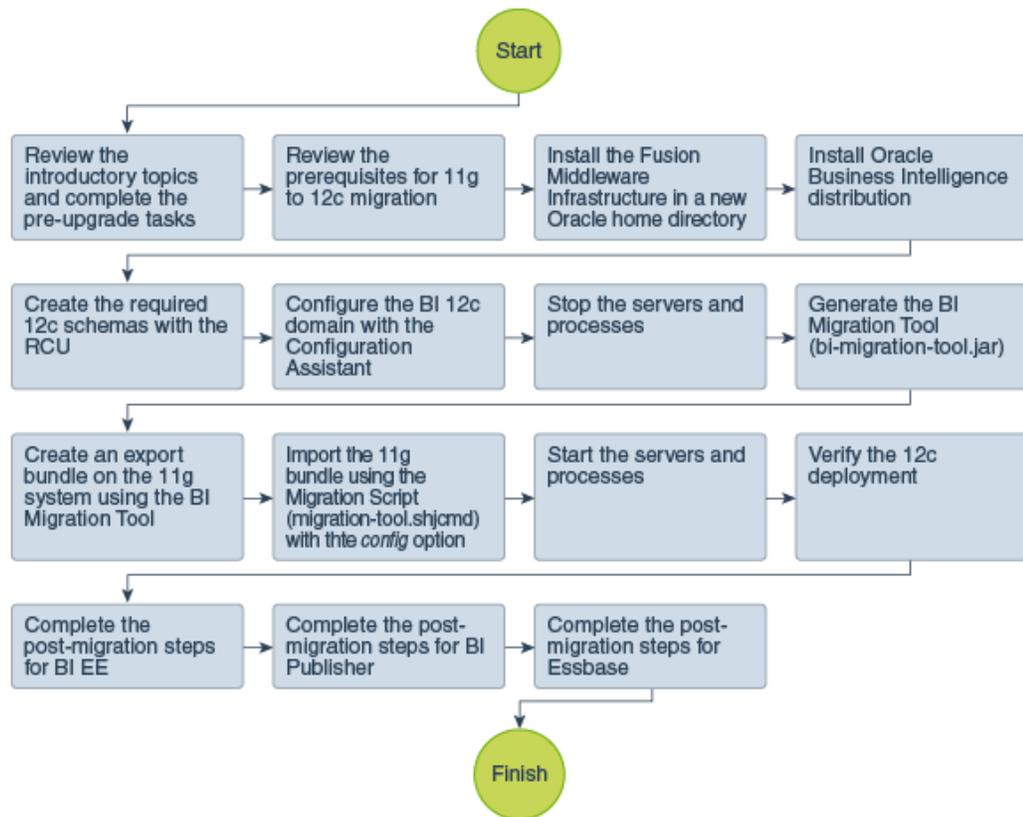
Table 3-1 provides a high-level summary of the steps that you must perform to migrate from Oracle BI 11*g* to 12*c*.

Table 3-1   Tasks for Migrating Oracle Business Intelligence from 11*g* Release

Task	Description
Required If you have not done so already, review the introductory topics in this guide and complete the required pre-upgrade tasks.	The pre-upgrade tasks include cloning your production environment, verifying system requirements and certifications, purging unused data, and creating a non-SYSDBA user. For a complete list of pre-upgrade tasks, see Pre-Upgrade Requirements.
Required Review the prerequisites for 11*g* to 12*c* migration.	Make sure that you meet the necessary requirements before you migrate your metadata and configuration to 12*c*. Additionally, analyze and optimize your 11*g* system by removing redundant content and merging and consolidating similar content. For more information, see 11*g* to 12*c* Migration Prerequisites.
Required Download and install the 12.2.1.3.0 Fusion Middleware Infrastructure and Oracle Business Intelligence distributions.	The Infrastructure distribution combines the WebLogic Server and the Java Required Files (JRF) that are required to set up the foundation to install other Fusion Middleware products. As per the upgrade topology defined in this guide, you must install the Infrastructure in a new Oracle home. The Oracle Business Intelligence distribution packs Oracle BI EE, Publisher, and Essbase. You must install Oracle BI in the Oracle home that is created when you installed the 12.2.1.3.0 Infrastructure. To install the product distributions, follow the procedure that is described in Installing the Product Distributions.
Required Create the required 12*c* schemas.	The schemas that you create will vary depending on your existing schema configuration. To create schemas with the Repository Creation Utility (RCU), see Creating the Required 12*c* Schemas.
Required Configure the 12*c* domain.	To configure the 12*c* domain with the Configuration Assistant, see Configuring Oracle BI Domain with the Configuration Assistant.
Required Stop the servers and processes.	You must stop the 12*c* Oracle BI instance, OBIS1, OBIPS1, Administration Server, Managed Servers, and the Node Manager before you start the migration. For the complete procedure, see Stopping Servers and Processes.

Table 3-1   (Cont.) Tasks for Migrating Oracle Business Intelligence from 11*g* Release

Task	Description
Required Generate the BI Migration Tool (bi-migration-tool.jar) and copy it from the 12*c* Oracle home to the 11*g* system.	You must generate the BI Migration Tool jar file by using the BI Migration Script (migration-tool.sh|cmd). For the complete procedure, see Generating the BI Migration Tool (bi-migration-tool.jar).
Required Create an export bundle by using the BI Migration Tool on the 11*g* system.	The export bundle is a jar file and consists of the metadata information from the 11*g* Oracle home. To create the export bundle, see Creating the Export Bundle.
Required Import the export bundle by using the BI Migration Script (migration-tool.sh|cmd).	You must import the export bundle by using the BI Migration Script (migration-tool.sh|cmd). The BI Migration Script automatically determines the Oracle home and the Domain home directories. To import the bundle, see Importing with the BI Migration Script (migration-tool.sh|cmd).
Required Start the servers and processes.	To start the 12*c* Oracle BI instance, OBIS1, OBIPS1, Administration Server, Managed Servers, and the Node Manager, see Starting Servers and Processes.
Required Verify the 12*c* deployment.	After you complete the migration procedure, run the Oracle BI Baseline Validation Tool to compare the 11*g* and 12*c* deployments. To verify whether the data from the 11*g* environment is correctly copied to the 12*c* environment, see Validating the Oracle BI Deployments.
Required Complete the post-migration steps.	For Oracle BI EE post-migration steps, see Post-Migration Tasks for Oracle BI EE. For Oracle BI Publisher post-migration steps, see Post-Migration Tasks for Oracle BI Publisher. For Oracle Essbase post-migration steps, see Post-Migration Tasks for Essbase.

11*g* to 12*c* Migration Prerequisites

Ensure that you configure the environment as per Oracle recommendations in readiness for the migration.

Make sure that you have met the following requirements before you proceed to the migration procedure:

- You have file system permission on both the 11*g* and 12*c* systems.

- You have configured the WebLogic authentication chain to enable the 11*g* users to sign in to the 12*c* domain. See Configuring Authentication Providers in *Administering Security for Oracle WebLogic Server*.

Make sure that you analyze and optimize the existing 11*g* system by removing redundant content and merging and consolidating similar content.

While analyzing the 11*g* system, note the following details:

- Name and size of the repository

- Name and size of the Oracle BI Presentation Catalog

- Existing security model details

- Data sources

- Number of scheduled jobs

- Any links to external systems

To optimize the existing 11*g* system, see:

- Optimizing the 11g System
  Migrating metadata and configuration from an existing Oracle BI 11*g* system requires time and resources. A poorly optimized 11*g* deployment can disrupt the migration process and can affect the performance of the 12*c* system. Oracle recommends that you analyze and optimize the existing 11*g* system by removing redundant content and merging and consolidating similar content.

Optimizing the 11*g* System

Migrating metadata and configuration from an existing Oracle BI 11*g* system requires time and resources. A poorly optimized 11*g* deployment can disrupt the migration process and can affect the performance of the 12*c* system. Oracle recommends that you analyze and optimize the existing 11*g* system by removing redundant content and merging and consolidating similar content.

To optimize the existing 11*g* system in readiness for migration:

1. Run the Consistency Checker to check the validity of the 11*g* repository, and to identify and fix the syntax or semantic errors and warnings that can cause the queries to fail on the Oracle BI 12*c* Administration Tool. See Checking the Consistency of a Repository or a Business Model in *Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition*.

2. Disable the initialization blocks that are no longer being used.

3. Identify and remove users and groups that are no longer required, and therefore do not need to be migrated.

4. Identify and remove objects in the repository and the Oracle BI Presentation Catalog that are no longer required, and therefore do not need to be upgraded.

5. If you have been using the Oracle BI Server usage tracking functionality, then review the usage tracking data to identify unused objects.

6. If you have been using the Oracle BI Server summary advisor functionality, then review the summary advisor for aggregated data and aggregation script.

Installing the Product Distributions

Before you begin your upgrade, download Oracle Fusion Middleware Infrastructure and Oracle Business Intelligence 12*c* (12.2.1.3.0) distributions on the target system and install them by using Oracle Universal Installer.

> **Note:**
>
> When Infrastructure is required for the upgrade, you must install the Oracle Fusion Middleware distribution first before you install other Fusion Middleware products.

To install the 12*c* (12.2.1.3.0) distributions:

1. Sign in to the target system.

2. Download the following from Oracle Technology Network or Oracle Software Delivery Cloud to your target system:

   • Oracle Fusion Middleware Infrastructure (`fmw_12.2.1.3.0_infrastructure_generic.jar`)

   • Oracle Business Intelligence (UNIX: `fmw_12.2.1.3.0_bi_platform_linux64.bin`), (Windows: `setup_fmw_12.2.1.3.0_bi_platform_win64.exe`)

3. Change to the directory where you downloaded the 12*c* (12.2.1.3.0) product distribution.

4. Start the installation program for Oracle Fusion Middleware Infrastructure:

   • (UNIX) *JDK_HOME*`/bin/java -jar fmw_12.2.1.3.0_infrastructure_generic.jar`

   • (Windows) *JDK_HOME*`\bin\java -jar fmw_12.2.1.3.0_infrastructure_generic.jar`

5. On UNIX operating systems, the Installation Inventory Setup screen appears if this is the first time you are installing an Oracle product on this host.

   Specify the location where you want to create your central inventory. Make sure that the operating system group name selected on this screen has write permissions to the central inventory location, and click **Next**.

   > **Note:**
   >
   > The Installation Inventory Setup screen does not appear on Windows operating systems.

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

7. On the Auto Updates screen, select an option:

   • **Skip Auto Updates**: If you do not want your system to check for software updates at this time.

- **Select patches from directory**: To navigate to a local directory if you downloaded patch files.

- **Search My Oracle Support for Updates**: To automatically download software updates if you have a My Oracle Support account. You must enter Oracle Support credentials then click **Search**. To configure a proxy server for the installer to access My Oracle Support, click **Proxy Settings**. Click **Test Connection** to test the connection.

   Click **Next**.

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

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

9. On the Installation Type screen, select the following:

   - For Infrastructure, select **Fusion Middleware Infrastructure**.

   - For Oracle Business Intelligence, select **BI Platform Distribution with Samples**.

   Click **Next**.

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

    To view the list of tasks that are verified, select **View Successful Tasks**. To view log details, select **View Log**. If any prerequisite check fails, then an error message appears at the bottom of the screen. Fix the error and click **Rerun** to try again. To ignore the error or the warning message and continue with the installation, click **Skip** (not recommended).

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

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

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

13. The Installation Complete screen displays the Installation Location and the Feature Sets that are installed. Review this information and click **Finish** to close the installer.

14. After you have installed Oracle Fusion Middleware Infrastructure, enter the following command to start the installer for your product distribution and repeat the steps above to navigate through the installer screens:

    (UNIX) `./fmw_12.2.1.3.0_bi_platform_linux64.bin`

    (Windows) `setup_fmw_12.2.1.3.0_bi_platform_win64.exe`

Creating the Required 12*c* Schemas

When upgrading from 11*g*, you must create the required 12*c* schemas. You can use the Repository Creation Utility (RCU) to create customized schemas or, optionally, you can use the Upgrade Assistant to create schemas using the default schema settings. This procedure describes how to create schemas using the RCU. Information about using the Upgrade Assistant to create schemas is covered in the upgrade procedures.

To create the required schemas:

1. (Optional) If you are upgrading from 11*g*, and you wish to confirm the schemas which are present in your existing domain, then connect to the database as a user with DBA privileges, and run the following code from SQL*Plus:

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

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

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

    *   (UNIX) `setenv JAVA_HOME=/home/Oracle/Java/jdk1.8.0_131`

    *   (Windows) `set JAVA_HOME=C:\home\Oracle\Java\jdk1.8.0_131`

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

3. Go to the `oracle_common/bin` directory:

    *   (UNIX) *NEW_ORACLE_HOME*`/oracle_common/bin`

    *   (Windows) *NEW_ORACLE_HOME*`\oracle_common\bin`

4. Start the RCU:

    *   (UNIX) `./rcu`

    *   (Windows) `rcu.bat`

5. On the Welcome screen, click **Next**.

6. On the Create Repository screen, select **Create Repository** and then select **System Load and Product Load**.

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

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

Table 3-2   Connection Credentials for Oracle Databases and Oracle Databases with Edition-Based Redefinition

Option	Description and Example
Host Name	Specify the name of the server where your database is running in the following format: `examplehost.exampledomain.com` For Oracle RAC databases, specify the VIP name or one of the node names in this field.
Port	Specify the port number for your database. The default port number for Oracle databases is `1521`.
Service Name	Specify the service name for the database. Typically, the service name is the same as the global database name. For Oracle RAC databases, specify the service name of one of the nodes in this field. For example: `examplehost.exampledomain.com`
Username	Enter the user name for your database. The default user name is `SYS`.
Password	Enter the password for your database user.
Role	Select the database user's role from the drop-down list: **Normal** or **SYSDBA**

Table 3-3   Connection Credentials for MySQL Databases

Option	Description and Example
Host Name	Specify the host name, IP address, or complete server name in $host\backslash server$ format of the server where your database is running.
Port	Specify the port number for your database.
Database Name	Specify the name of your database.
Username	Specify the name of a user with administrator privileges.
Password	Enter the password for your database user.

Table 3-4   Connection Credentials for Microsoft SQL Server Databases

Option	Description and Example
Unicode Support	Select **Yes** or **No** from the drop-down list.
Server Name	Specify the host name, IP address, or complete server name in $host\backslash server$ format of the server where your database is running. MSSQL named instances: A named instance is identified by the network name of the computer and the instance name that you specify during installation. The client must specify both the server name and the instance name when connecting.

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

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

Table 3-5   Connection Credentials for IBM DB2 Databases

Option	Description and Example
Server Name	Specify the host name, IP address, or complete server name in $host\backslash server$ format of the server where your database is running.
Port	Specify the port number for your database.
Database Name	Specify the name of your database.
Username	Specify the name of a user with DB Owner privileges. The default user name for IBM DB2 databases is `db2admin`.
Password	Enter the password for your database user.

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

8. On the Select Components screen:

   a. Specify the custom prefix you want to use to identify the Oracle Fusion Middleware schemas.

   The custom prefix is used to logically group these schemas together for use in this domain. For the purposes of this guide, use the prefix `FMW12213`.

   Note the custom prefix that you have specified on this screen. You will need this later, during the domain creation process.

   b. Select **AS Common Schemas** and **BIPLATFORM**.

   When you select **AS Common Schemas** and **BIPLATFORM**, all of the schemas in this section are automatically selected. If the schemas in this section are not automatically selected, then select the required schemas.

   Common Infrastructure Services schema, also known as STB schema, is automatically created. You cannot select or deselect an STB schema. STB schema enables you to retrieve information from the RCU during domain configuration. For more information about the Service Table schema, see Understanding the Service Table Schema in *Creating Schemas with the Repository Creation Utility*.

   For more information about how to organize your schemas in a multi-domain environment, see Planning Your Schema Creation in *Creating Schemas with the Repository Creation Utility*.

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

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

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

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

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

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

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

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

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

Configuring Oracle BI Domain with the Configuration Assistant

The Oracle BI 12*c* Configuration Assistant is a `.sh` (UNIX) or `.cmd` (Windows) file. Start the Configuration Assistant by starting the config executable from the `bi/bin` directory inside the Oracle home. Follow the procedure in this topic to complete the configuration step-by-step.

> **Note:**
>
> Oracle recommends that you use the Oracle BI 12*c* Configuration Assistant to configure your 12*c* system.

To configure the BI standard topology:

> **Note:**
>
> If you are extending the WebLogic domain with Oracle Business Intelligence by using the Configuration Assistant, make sure that the Administration Server for the domain is named *AdminServer*. Other names for the Administration Server are not supported.

1. Change to the following directory:

   (UNIX) `ORACLE_HOME/bi/bin`

   (Windows) `ORACLE_HOME\bi\bin`

2. Start the Configuration Assistant by entering the following command:

(UNIX) `./config.sh`

(Windows) `config.cmd`

3. Select the components to install and click **Next**.

> **Note:**
>
> To ensure that a consistent set of suites are deployed, the Configuration Assistant automatically adjusts your selection.

- Essbase: Includes components such as Essbase Server, Cube Deployment Server, and Analytic Provider Services.

- Business Intelligence Enterprise Edition: Includes components such as Presentation Services, Visual Analyzer, BI Composer, web services, proactive intelligence (Delivers and Actions), Web Services for SOA (WS4SOA), and Mobile Application Designer.

- Business Intelligence Publisher: Includes Oracle BI Publisher.

4. On the Prerequisite Checks screen, after the prerequisite checks conclude with no errors, click **Next**.

   If any of the prerequisite checks fail, then a short error message appears at the bottom of the screen. Fix the error and click **Rerun** to try again. To ignore the error or warning messages and continue with the installation, click **Skip**, although this approach is not recommended.

   > **Note:**
   >
   > The configuration might not function normally if you continue without resolving the issue that caused an error or warning message during the prerequisite checks.

5. On the Define New Domain screen, specify the following, and click **Next**:

Table 3-6   Field-name descriptions for the Define New Domain screen

Field	Description
Domains Directory	Specify the path where you want to set up the domain directory.
Domain Name	Specify domain.
Username	Specify a username for the System Administrator. > **Note:** > This user is created in the embedded LDAP and is granted WebLogic Administrator permissions.
Password	Specify a password for the System Administrator.

Table 3-6 (Cont.) Field-name descriptions for the Define New Domain screen

Field	Description
Confirm Password	Confirm the password by reentering it.

6. On the Database Schema screen, either create a new schema or use an existing schema by clicking the appropriate option.

   If you select to create a new schema, the Configuration Assistant creates a schema for you. Specify the following and click **Next**.

Table 3-7 Field-name descriptions for the Database Schema screen

Field	Description
Schema prefix	Specify a unique schema prefix.
Schema password	Specify a password for your schema.
Confirm password	Confirm the password by reentering it.
Database type	Select the database that you are using from the list of values. Supported databases are: • Oracle Database • Microsoft SQL Server • IBM DB2
Username	Enter the privileged username to create the schema.
Password	Enter the password for the above username.
Simple connect string	The connect string that you specify varies depending on the type of database you are using. • (Oracle Database) `host:port/service` or `host:port:SID` or `TNS` connect string • (Microsoft SQL Server) `// host:port;DatabaseName=dbname` • (IBM DB2) `// host:port;DatabaseName=dbname`

If you select to use an existing schema, you must create STB, BIPLATFORM, IAU, MDS, OPSS, and WLS schemas using the RCU. Specify the following and click **Next**.

> **Note:**
>
> If you are using Oracle Database version 12.2.0.1 or higher, you must create a pluggable database within a container database to create repository schemas. You cannot create schemas within a container database without using a pluggable database.

7. On the Port Range screen, specify the port range and click **Next**.

> **✎ Note:**
>
> The default, allocated port range is from 9500 to 9999, both inclusive. You can keep the default values or specify different values within this range.

8. On the Initial Application screen, select one of the following options as per your requirement, and click **Next**:

   • Oracle sample application (SampleAppLite).

   • Your own existing BI Application from export bundle (`.jar`).

   > **✎ Note:**
   >
   > This option is applicable if you are migrating from Oracle BI 11*g* to 12*c*. See Upgrading Oracle Business Intelligence from 11*g* (Out-of-place Migration) in *Upgrading Oracle Business Intelligence*.

   • Clean Slate (no predefined application).

9. On the Summary screen, verify the values that you specified on each screen.

   Click **Save** to generate a response file used for silent installation (optional).

   Click **Configure**.

   The configuration process starts and the Configuration Progress screen is displayed.

10. After the configuration concludes without any errors, click **Next** to go to the Configuration Complete screen.

11. On the Configuration Complete screen, review the configuration summary.

    Click **Save** to save the information displayed on this screen in a file.

    Click **Finish** to close the Configuration Assistant.

    The BI Application opens in the browser. Use the login credentials that you specified while configuring to sign in to the BI application home.

You have configured the standard BI topology. The Configuration Assistant automatically starts the newly configured BI instance after successful completion. However, if you want to restart a domain that has been shut down manually, see Starting Servers and Processes.

Stopping Servers and Processes

Before you migrate your 11*g* metadata and configuration to 12*c*, stop the Oracle BI 12*c* instance, OBIS1, OBIPS1, Administration Servers, Managed Servers, and the Node Manager.

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

To stop your Fusion Middleware environment:

1. Change to the following directory:

   (UNIX) `12c_DOMAIN_HOME`/bitools/bin

   (Windows) `12c_DOMAIN_HOME`\bitools\bin

2. To stop the Oracle BI instance and servers, enter the following command:

   (UNIX) `stop.sh`

   (Windows) `stop.cmd`

> **Note:**
>
> When prompted to enter the password, specify the Node Manager password that you entered while configuring the Oracle BI domain.

Generating the BI Migration Tool (bi-migration-tool.jar)

The BI Migration Tool (bi-migration-tool.jar) is self-executing and self-contained. You must generate the BI Migration Tool jar file by using the BI Migration Script (migration-tool.sh|cmd). The BI Migration Script is made available after you configure the 12*c* domain. This step packages the components of the BI Migration Tool into a single, self-executing jar file, so that it can be easily transported on to an 11*g* system. To do this, you must have access to a 12*c* system with a configured domain. You must set up the 12*c* domain to provide sufficient infrastructure to run the BI Migration Tool and allow it to repackage itself. The 12*c* domain can be empty at this stage.

To generate the BI Migration Tool jar:

1. The BI Migration Script (`migration-tool.sh|cmd`) is available at the following location:

   (UNIX) `12c_DOMAIN_HOME`/bitools/bin/

   (Windows) `12c_DOMAIN_HOME`\bitools\bin\

2. Run the following command to generate the BI Migration Tool:

ORACLE

(UNIX) *12c_DOMAIN_HOME*/bitools/bin/migration-tool.sh package bi-migration-tool.jar

(Windows) *12c_DOMAIN_HOME*\bitools\bin\migration-tool.cmd package bi-migration-tool.jar

Where,

Table 3-8   Command Description

Option	Description
12c_DOMAIN_HOME/bitools/bin	Specifies the location of the BI Migration Script.
package	Specifies the BI Migration Script to perform the packaging operation.
bi-migration-tool.jar	Specifies the file name of the migration tool jar file where the output is written. In this documentation, the bi-migration-tool.jar file is referred as the "BI Migration Tool" and the migration-tool.sh script is referred as the "BI Migration Script".

3. Copy the BI Migration Tool to the host system from where you want to export data.

Creating the Export Bundle

The export bundle is a ".jar " file and consists of the metadata information from the 11*g* Oracle home.

To create an export bundle:

1. Change to the directory where you created the bi-migration-tool.jar file.

2. Run the BI Migration Tool without passing parameters. Enter the following command:

    (UNIX) *JDK_HOME*/bin/java -jar bi-migration-tool.jar

    (Windows) *JDK_HOME*\bin\java -jar bi-migration-tool.jar

    This command displays the list of parameters that you can include for the BI Migration Tool to locate various parts of the 11*g* system.

3. Run the BI Migration Tool with the following parameters this time to create an export bundle.

Table 3-9   Parameter Description: Creating Export Bundle

Parameter	Description
out	Indicates the BI Migration Tool to run in Export mode.
<oracle 11g home>	Specifies the Oracle home directory. This is typically the directory Oracle_BI inside Middleware home.
DOMAIN_HOME	Specify the Domain home directory. This is typically the directory user_projects/domains/bi/ inside the Middleware home.
<output export bundle path>	Specifies the file name of the export bundle where the output is written. The output is not a BAR file. The file name of the export bundle must include the "jar" extension.

Following is a sample command for creating an export bundle:

(UNIX) *JDK_HOME*/bin/java -jar bi-migration-tool.jar out
ORACLE_HOME/Oracle_BI1 *DOMAIN_HOME*/tmp/migration-tool-test/
test_export.jar

(Windows) *JDK_HOME*\bin\java -jar bi-migration-tool.jar
out *ORACLE_HOME*/Oracle_BI1 *DOMAIN_HOME*\tmp\migration-tool-
test\test_export.jar

Where,

Table 3-10   Parameter Values: Creating Export Bundle

Parameter	Description
ORACLE_HOME/Oracle_BI1	The path where the Oracle home directory is located.
DOMAIN_HOME	The path where the Domain home directory is located.
/tmp/migration-tool-test/ test_export.jar	The location where the export bundle is created.

> ✏️ **Note:**
>
> Make sure to replace these file paths with the respective paths on your
> system.

The following message indicates a successful export:
Export succeeded

Copy the export bundle in to the 12*c* system.

Importing with the BI Migration Script (migration-tool.sh|cmd)

You must import the export bundle using the BI Migration Script (migration-tool.sh|
cmd). The BI Migration Script automatically determines the Oracle home and the
Domain home directories.

You can include the following parameters while running the BI Migration Script to
import the export bundle in to the 12*c* system:

Table 3-11   Parameter Description: Importing the export bundle

Parameter	Description
in	Indicates the BI Migration Script to import the bundle.
config	Indicates the BI Migration Script to overwrite the configuration files during the import process. This allows you to also migrate the configuration along with the data.
<export bundle>	The path where the export bundle is located.
<service instance name>	Specifies the name of the service instance, which is ssi.

> **✎ Note:**
>
> The default service instance name is "ssi". However, if you install Oracle BI in a silent mode using a response file, you can specify a service instance name during the product installation. In that case, you must specify the service instance name you have specified in the response file.

To import the metadata and configuration in to the 12*c* system:

1. Run the BI Migration Script with the following parameters:

   (UNIX) *12c_DOMAIN_HOME*`/bitools/bin/migration-tool.sh in config <export bundle> <service instance name>`

   For example,

   12c_DOMAIN_HOME`/bitools/bin/migration-tool.sh in config /tmp/migration-tool-test/test_export.jar ssi`

   (Windows) *12c_DOMAIN_HOME*`\bitools\bin\migration-tool.cmd in config <export bundle> <service instance name>`

   For example,

   12c_DOMAIN_HOME`\bitools\bin\migration-tool.cmd in config /tmp/migration-tool-test/test_export.jar ssi`

   Where,

Table 3-12   Parameter Values: Importing the export bundle

Parameter	Description
in	Indicates the BI Migration Script to import the bundle.
config	Indicates the BI Migration Script to overwrite the configuration files during the import process.
`/tmp/migration-tool-test/test_export.jar`	The path where the export bundle is located.
ssi	Specifies the name of the service instance.

2. If the migration is successful, you see the output such as the following:

   ```
   Import succeeded
   About to close down logging to: DOMAIN_HOME/bilogs/migration/
   migration-2016-05-05-06-13-05.log
   This is so that the log file can be archived into the diagnostics zip
   Any remaining log entries will go to '/tmp/migration.log', and will not
   appear in the diagnostics zip
   Migration action succeeded
   ```

You can now start the 12*c* system. For more information about starting the 12*c* system, see About Managing Oracle Business Intelligence Processes in *System Administrator's Guide for Oracle Business Intelligence Enterprise Edition*.

> **Note:**
>
> It takes about 10 more minutes for the application roles and policies to migrate after the migration process is complete.

After completing the import process, you must start OBIS1 and OBIPS1 which you stopped before beginning the migration process.

Starting Servers and Processes

After you migrate your 11*g* metadata and configuration to 12*c*, start the Oracle BI 12*c* instance, OBIS1, OBIPS1, Administration Servers, Managed Servers, and the Node Manager.

> **Note:**
>
> The procedure in this section describe how to start servers and processes using the WLST command-line utility or a script. You can also use the Fusion Middleware Control and the Oracle WebLogic Server Administration Console. See Starting and Stopping Administration and Managed Servers and Node Manager

To start your Fusion Middleware environment:

1. Change to the following directory:

    (UNIX) `12c_DOMAIN_HOME`/bitools/bin

    (Windows) `12c_DOMAIN_HOME`\bitools\bin

2. To start the Oracle BI instance and servers, enter the following command:

    (UNIX) `start.sh`

    (Windows) `start.cmd`

    > **Note:**
    >
    > When prompted to enter the password, specify the Node Manager password that you entered while configuring the Oracle BI domain.

Validating the Oracle BI Deployments

The Oracle BI Baseline Validation Tool enables you to identify differences during life cycle operations, such as migrating from the Oracle BI 11*g* release to the 12*c* release. After you complete the migration procedure, you can use this tool to compare the two deployments and verify whether the results from the 11*g* environment are the same as the results from the 12*c* environment.

You can download the Oracle BI Validation Tool from Oracle BI Baseline Validation Tool Downloads.

For more information about using the Oracle BI Validation Tool, see Comparing Oracle Business Intelligence Deployments Using the Oracle Business Intelligence Baseline Validation Tool.

> ✏️ **Note:**
>
> You can download the Oracle BI Validation Tool along with the other Oracle Business Intelligence download on the Oracle Technology Network. See the document that is included in the Oracle BI Validation Tool download for more information. For specific information on the distributions you want to download for each product, see Oracle Fusion Middleware Download, Installation, and Configuration Readme Files page.

4

Post-Migration Tasks for Oracle BI EE

After migrating the Oracle BI EE metadata, manually migrate the configuration settings for catalog groups and other components such as security for WebLogic and Enterprise Manager, BI Server, and BI Presentation Services.

- Migrating Catalog Groups
  Catalog groups were a feature of Oracle BI EE that allowed administrators to organize users and application roles for security administration purposes.

- Migrating Configuration of Oracle BI EE
  After migrating the data, manually migrate the configuration information from the 11*g* system to the 12*c* system, which includes WebLogic security, Enterprise Manager security, the managed BI configurations, BI logs, BI Server, and BI Presentation Server configuration settings.

- Configuring the Database to Use DataDirect Drivers
  You must configure the database to use the appropriate DataDirect drivers. If you are using a database that is already configured to use the DataDirect drivers, modify the database configuration's `odbc.ini` file to use the correct DataDirect drivers.

- Configuring Usage Tracking
  The Oracle BI Server supports the collection of usage tracking data. When usage tracking is enabled, the BI Server collects usage tracking data for each query. The statistics are then written to a usage tracking log file or are inserted directly in to a database table. After you complete migrating and configuring the 12*c* system, you must enable usage tracking in the NQSConfig.INI file.

- Configuring the SQL Server
  After configuring the 12*c* system, you must configure the SQL Server settings in the odbc.ini file.

- Adding Roles and Permissions
  After migrating the data, you must add roles and permissions for BI Administrator, BI Author, and BI Consumer groups.

- Configuring MySQL for Oracle BI
  You must configure the MySQL database to use the DataDirect driver. If you are using a database that is already configured to use the DataDirect drivers, modify the database configuration's odbc.ini file to use the correct DataDirect drivers.

- Checking Oracle BI JavaHost Configuration
  Make sure that the XMLP and Oracle BI JavaHost settings match with the settings specified for the configuration of the 11*g* system.

- Enabling Clusters
  After migrating the data, you must manually enable the ClusterEnabled parameter in the ClusterConfig.xml file to turn on the cluster instances on the 12*c* system.

- Enabling Oracle Hardware Acceleration and Compatibility Mode
  Hardware acceleration affects the autocompletion of features such as dashboard prompts, trellis charts and microcharts, Summary Advisor functionality, and aggregate persistence for the TimesTen In-Memory database. Therefore, you

must manually enable the bi:hw-acceleration flag in the 12*c* bi-config.xml file post-migration. You must also enable the bi:compat-mode-11g flag, so that the state of the 11*g* system is preserved at run time.

- Setting the Compatibility Framework for Oracle BI Server
  The compatibility framework allows the BI Server to add new features or bug fixes in Oracle BI 12*c* (12.2.1.1 and later) that are not compatible with the Oracle BI 11*g* releases while providing a flexible framework that enables the BI Server to operate in a compatibility mode with the earlier major release. Individual features and bug fixes can be enabled or disabled independently using compatibility flags. Alternatively, you can set all the compatibility flags to the default values using a single COMPATIBLE_RELEASE flag for the Oracle BI system to be compatible with the earlier release, which is 11.1.1.9. Add the COMPATIBLE_RELEASE parameter to the NQSConfig.INI file to ensure that the migrated 12*c* environment behaves as closely as possible to Oracle Business Intelligence 11*g* Release 1 (11.1.1.9) environment.

- Migrating the Fusion Middleware MapViewer Configuration
  Oracle Fusion Middleware Mapviewer (MapViewer) is a programmable tool for rendering maps using spatial data managed by Oracle Spatial and Graph or Oracle Locator (also referred to as Locator). MapViewer provides tools that hide the complexity of spatial data queries and cartographic rendering, while providing customizable options for more advanced users. These tools can be deployed in a platform-independent manner and are designed to integrate with map-rendering applications. After migrating the data, you must manually modify the 12*c* MapViewer configuration file to contain the same contents as the 11*g* file except the Logging section.

- Resolving Authentication Issues After Migration
  To avoid authentication issues post-migration, you must uncheck the **Required for Authentication** option in the DYNAMIC_OLAP_LOGIN initialization block.

- Copying Configuration Files
  You must manually copy the following configuration files after the migration: writebacktemplate.xml, userpref_currencies.xml, and bicustom.ear.

- Removing the Display of HTML Codes in a Customized "No Results" Message
  In an analysis, content developers can control the text that is displayed when the results of the analysis return no data. Content developers can accept the default message, or they can customize the text of the message, including by inserting HTML formatting codes in the message.

Migrating Catalog Groups

Catalog groups were a feature of Oracle BI EE that allowed administrators to organize users and application roles for security administration purposes.

In Oracle BI Release 12*c* (12.2.1.1 and higher), catalog groups are obsolete. The catalog groups are migrated by the upgrade process to application roles.

Migrating Configuration of Oracle BI EE

After migrating the data, manually migrate the configuration information from the 11*g* system to the 12*c* system, which includes WebLogic security, Enterprise Manager security, the managed BI configurations, BI logs, BI Server, and BI Presentation Server configuration settings.

- **Migrating the Security Configuration for Oracle BI with Oracle WebLogic Server**
  Oracle BI 12*c* is tightly integrated with the Oracle Fusion Middleware Security architecture and delegates core security functionality to components of that architecture. By default, an Oracle BI installation is configured with an authentication provider that uses the Oracle WebLogic Server embedded LDAP server for user and group information. The Oracle BI default policy store provider and credential store provider stores credentials, application roles, and application policies in files in the domain. You must manually migrate the security configuration for Oracle BI from the 11*g* system to the 12*c* system with Oracle WebLogic Server.

- **Migrating the Oracle Enterprise Manager Fusion Middleware Control Security Configuration**
  Most of the security configuration related to application roles, security grants, and application policies are migrated to the Oracle Enterprise Manager Fusion Middleware Control 12*c* during the migration process. You must examine and manually apply any special Security Provider configuration in the Fusion Middleware Control 12*c*.

- **Migrating the Oracle Enterprise Manager Fusion Middleware Control Managed Business Intelligence Configurations**
  The Fusion Middleware Control is used to manage some of the important configurations for Oracle BI. You must manually examine the configuration of the General, Presentation, Performance, and Mail sections of the Fusion Middleware Control 11*g* and apply the same settings to the 12*c* system.

- **Migrating the Oracle Enterprise Manager Fusion Middleware Control Managed Business Intelligence Log Configuration**
  The Oracle Enterprise Manager Fusion Middleware Control is used to manage the log configuration for the Oracle BI components. You must manually examine the log configuration settings in the 11*g* system and apply the same settings to your 12*c* system.

- **Migrating the Oracle BI Server Configuration**
  Various files for the BI Server contain configuration information: odbc.ini and tnsnames.ora. The odbc.ini file contains non-Oracle data source (DSN) connection information. The tnsnames.ora file contains network service names mapped to connect descriptors for the local naming method, or net service names mapped to listener protocol addresses. You must examine these 11*g* files and manually copy the configuration information to the respective 12*c* files.

- **Migrating the Oracle BI Presentation Server Configuration**
  The Oracle BI Presentation Services process hosts most of the business logic of the Web server and provides the framework and interface for the presentation of business intelligence data to web clients. The userpref_currencies.xml file defines the currency options that are displayed in the Currency box of the Preferences tab of the My Account dialog. The bridgeconfig.properties file contains the Presentation Services Plug-in information. Examine these 11*g* files and manually copy the configuration information to the respective 12*c* files.

Migrating the Security Configuration for Oracle BI with Oracle WebLogic Server

Oracle BI 12*c* is tightly integrated with the Oracle Fusion Middleware Security architecture and delegates core security functionality to components of that architecture. By default, an Oracle BI installation is configured with an authentication

provider that uses the Oracle WebLogic Server embedded LDAP server for user and group information. The Oracle BI default policy store provider and credential store provider stores credentials, application roles, and application policies in files in the domain. You must manually migrate the security configuration for Oracle BI from the 11*g* system to the 12*c* system with Oracle WebLogic Server.

To migrate the security configuration for Oracle BI:

1. Examine your Oracle BI 11*g* WebLogic security configuration by accessing the Oracle BI 11*g* WebLogic Console to examine the current security configuration for security providers.

2. Manually configure the same security configuration in WebLogic Console of Oracle BI 12*c*.

Migrating the Oracle Enterprise Manager Fusion Middleware Control Security Configuration

Most of the security configuration related to application roles, security grants, and application policies are migrated to the Oracle Enterprise Manager Fusion Middleware Control 12*c* during the migration process. You must examine and manually apply any special Security Provider configuration in the Fusion Middleware Control 12*c*.

To migrate the Oracle Enterprise Manager Fusion Middleware Control security configuration:

1. Examine the Security Provider configuration in the Oracle Enterprise Manager Fusion Middleware Control 11*g*.

2. Validate the Oracle Enterprise Manager Fusion Middleware Control 12*c* security setup by searching application policies and appropriate users.

3. Apply any special Security Provider configuration in the Oracle Enterprise Manager Fusion Middleware Control 12*c*.

4. Ensure that the user, group, and application roles appear correctly in the 12*c* system.

Migrating the Oracle Enterprise Manager Fusion Middleware Control Managed Business Intelligence Configurations

The Fusion Middleware Control is used to manage some of the important configurations for Oracle BI. You must manually examine the configuration of the General, Presentation, Performance, and Mail sections of the Fusion Middleware Control 11*g* and apply the same settings to the 12*c* system.

To migrate the Fusion Middleware Control managed BI configurations:

1. In the Fusion Middleware Control 11*g*, navigate to the **Business Intelligence Configuration** link and examine the Configuration tab.

2. Apply the same configuration settings in the **Business Intelligence Configuration** tab of the Fusion Middleware Control 12*c*.

   Ensure that the configuration settings of the General, Presentation, Performance, and Mail sections are applied correctly to the Fusion Middleware Control 12*c*.

Migrating the Oracle Enterprise Manager Fusion Middleware Control Managed Business Intelligence Log Configuration

The Oracle Enterprise Manager Fusion Middleware Control is used to manage the log configuration for the Oracle BI components. You must manually examine the log configuration settings in the 11*g* system and apply the same settings to your 12*c* system.

To migrate the Fusion Middleware Control managed Oracle BI log configuration:

1. In the Fusion Middleware Control 11*g*, navigate to the **Business Intelligence** link and examine the settings in the Diagnostics tab.

2. Note the configuration settings and apply the Oracle BI component log configuration and component log levels in the Fusion Middleware Control 12*c*.

Migrating the Oracle BI Server Configuration

Various files for the BI Server contain configuration information: odbc.ini and tnsnames.ora. The odbc.ini file contains non-Oracle data source (DSN) connection information. The tnsnames.ora file contains network service names mapped to connect descriptors for the local naming method, or net service names mapped to listener protocol addresses. You must examine these 11*g* files and manually copy the configuration information to the respective 12*c* files.

To migrate the BI Server configuration details:

1. View the BI Server information in the 11*g* odbc.ini file present at the following location:

   (UNIX) `11g_DOMAIN_HOME`/config/fmwconfig/bienv/core

   (Windows) `11g_DOMAIN_HOME`\config\fmwconfig\bienv\core

2. Copy the BI Server configuration from the 11*g* odbc.ini file to the 12*c* file.

3. In the existing 12*c* odbc.ini file. integrate the ODBC entries from 11*g*.

Migrating the Oracle BI Presentation Server Configuration

The Oracle BI Presentation Services process hosts most of the business logic of the Web server and provides the framework and interface for the presentation of business intelligence data to web clients. The userpref_currencies.xml file defines the currency options that are displayed in the Currency box of the Preferences tab of the My Account dialog. The bridgeconfig.properties file contains the Presentation Services Plug-in information. Examine these 11*g* files and manually copy the configuration information to the respective 12*c* files.

To migrate the Oracle BI Presentation Server configuration:

1. View the configuration information in the userpref_currencies.xml file present at the following location:

   (UNIX) `11g_DOMAIN_HOME`/config/fmwconfig/biconfig/OBIPS/

   (Windows) `11g_DOMAIN_HOME`\config\fmwconfig\biconfig\OBIPS\

2. Copy the appropriate configuration information from the 11*g* userpref_currencies.xml file to the 12*c* file.

3. View the configuration information in the bridgeconfig.properties file present at the following location:

   (UNIX) *11g_DOMAIN_HOME*/config/fmwconfig/biconfig

   (Windows) *11g_DOMAIN_HOME*\config\fmwconfig\biconfig

4. Copy the appropriate configuration information from the 11*g* bridgeconfig.properties file to the 12*c* file.

Configuring the Database to Use DataDirect Drivers

You must configure the database to use the appropriate DataDirect drivers. If you are using a database that is already configured to use the DataDirect drivers, modify the database configuration's `odbc.ini` file to use the correct DataDirect drivers.

To configure the database to use the DataDirect drivers:

1. Open the odbc.ini file located at the following location:

   (UNIX) *12c_DOMAIN_HOME*/config/fmwconfig/bienv/core/odbc.ini

   (Windows) *12c_DOMAIN_HOME*\config\fmwconfig\bienv\core/odbc.ini

   Replace the value of *DOMAIN_HOME* with the actual path to the 12*c* Domain home you created when you installed the 12*c* software.

2. Update the ODBC entry to use the DataDirect drivers, as shown in the following example:

   ```
   [DSN name in RPD]
   Driver=MW_HOME/bi/common/ODBC/Merant/7.1.4/lib/<7.1.4.so>
   Description=DataDirect 7.1.4 Sybase Wire Protocol
   LogonID=DB username
   Password=DB password
   NetworkAddress=DB hostname, DB port
   Database=DB name
   ```

   Where, RPD indicates rapidfile database. RPD is a binary file used by the BI Server to retrieve data from a source database.

3. Update all existing data source names (DSNs) that are configured with the 11*g* DataDirect version. For example, the data source name for SYBASE should point to DataDirect 7.1.4 as shown in the following example:

   ```
   [ODBC Data Sources]
   AnalyticsWeb = Oracle BI Server
   Cluster = Oracle BI Server
   SSL_Sample = Oracle BI Server
   DSN name in RPD = DataDirect 7.1.4
   ```

4. Save and close the odbc.ini file.

Configuring Usage Tracking

The Oracle BI Server supports the collection of usage tracking data. When usage tracking is enabled, the BI Server collects usage tracking data for each query. The statistics are then written to a usage tracking log file or are inserted directly in to a database table. After you complete migrating and configuring the 12*c* system, you must enable usage tracking in the NQSConfig.INI file.

To configure usage tracking:

1. Open the repository in online mode.

2. Import the S_NQ_ACCT and the S_NQ_DB_ACCT table from the RCU schema.

3. In the NQSConfig.INI file, set `USAGE_TRACKING=YES` and update the PHYSICAL_TABLE_NAME and the CONNECTION_POOL.

4. Import the following tables to track query statistics related to the initialization block execution and the summary advisor feature:

   • S_NQ_INITBLOCK

   • S_NQ_SUMMARY_ADVISOR

5. To enable initialization block logging, update the `INIT_BLOCK_TABLE_NAME`and INIT_BLOCK_CONNECTION_POOL in the NQSConfig.INI file.

6. To enable summary advisor logging, set `SUMMARY_STATISTICS_LOGGING=YES` and update the SUMMARY_ADVISOR_TABLE_NAME in the NQSConfig.INI file.

7. Restart the BI Server.

For more information about usage tracking, see Managing Usage Tracking in *System Administrator's Guide for Oracle Business Intelligence Enterprise Edition*.

> **Note:**
>
> Ensure that the queries of the BI Server are being tracked in the table by view data.

Configuring the SQL Server

After configuring the 12*c* system, you must configure the SQL Server settings in the odbc.ini file.

To configure the SQL Server:

1. Open the odbc.ini file located at the following location:

   (UNIX) `12c_DOMAIN_HOME`/config/fmwconfig/bienv/core/odbc.ini

   (Windows) `12c_DOMAIN_HOME`\config\fmwconfig\bienv\core\odbc.ini

   Replace the value of *DOMAIN_HOME* with the actual path to the 12*c* Domain home you created when you installed the 12*c* software.

2. Update the ODBC entry to use the DataDirect drivers, as shown in the following example:

```
[DSN name in RPD]
Driver=$ORACLE_HOME/bi/common/ODBC/Merant/7.1.4/lib/<7.1.4.so>
Description=DataDirect 7.1.4 SQL Server Wire Protocol
Address=DB HOST\SQLSERVER NAME
Port=PORT NUMBER
AlternateServers=
AnsiNPW=Yes
ConnectionRetryCount=0
ConnectionRetryDelay=3
Database=DB name
LoadBalancing=0
LogonID=DB username
Password=DB password
QuotedId=Yes
SnapshotSerializable=0
ReportCodePageConversionErrors=
```

3. Update all existing data source names (DSNs) that are configured with the 11*g* DataDirect version. For example, the data source name for SQL Server should point to DataDirect 7.1.4 as shown in the following example:

```
[ODBC Data Sources]
AnalyticsWeb = Oracle BI Server
Cluster = Oracle BI Server
SSL_Sample = Oracle BI Server
DSN name in RPD = DataDirect 7.1.4
```

4. Save and close the `odbc.ini` file.

Adding Roles and Permissions

After migrating the data, you must add roles and permissions for BI Administrator, BI Author, and BI Consumer groups.

To add roles and permissions:

1. Sign in to the WebLogic Console.

2. Go to **Myrealm** under **Security Realms** and click **Users and Groups**.

3. Create the following groups:

   On the **Groups** tab, create the BIAdministrators group.

   On the **Membership** tab, create the BIAuthors and the BIConsumers groups.

4. Assign the BIAdministrators group to the Administrator or the WebLogic user.

Configuring MySQL for Oracle BI

You must configure the MySQL database to use the DataDirect driver. If you are using a database that is already configured to use the DataDirect drivers, modify the database configuration's odbc.ini file to use the correct DataDirect drivers.

To configure MySQL for Oracle BI:

1. Open the odbc.ini file located at the following location:

   (UNIX) *12c_DOMAIN_HOME*/config/fmwconfig/bienv/core/odbc.ini

   (Windows) *12c_DOMAIN_HOME*\config\fmwconfig\bienv\core\odbc.ini

   Replace the value of *DOMAIN_HOME* with the actual path to the 12*c* Domain home you created when you installed the 12*c* software.

2. Update the ODBC entry to use the DataDirect drivers, as shown in the following example:

   ```
   [DSN name in RPD]
   Driver=NEW_ORACLE_HOME/bi/common/ODBC/Merant/7.1.4/lib/<7.1.4.so>
   Description=DataDirect 7.1.4 MySQL Wire Protocol
   ApplicationUsingThreads=1
   ConnectionRetryCount=0
   ConnectionRetryDelay=3
   Database=DB name
   DefaultLongDataBuffLen=1024
   EnableDescribeParam=0
   InteractiveClient=0
   LoadBalancing=0
   LogonID=DB username
   Password=DB password
   PortNumbeR=DB PORT
   ReportCodepageConversionErrors=0
   TreatBinaryAsChar=0
   ```

3. Update all existing data source names (DSNs) that are configured with the 11*g* DataDirect version. For example, the data source name for MySQL should point to DataDirect 7.1.4 as shown in the following example:

   ```
   [ODBC Data Sources]
   AnalyticsWeb = Oracle BI Server
   Cluster = Oracle BI Server
   SSL_Sample = Oracle BI Server
   DSN name in RPD = DataDirect 7.1.4
   ```

4. Save and close the odbc.ini file.

Checking Oracle BI JavaHost Configuration

Make sure that the XMLP and Oracle BI JavaHost settings match with the settings specified for the configuration of the 11*g* system.

To check the XMLP and the BI JavaHost configuration:

ORACLE

1. View the config.xml file on the 11*g* and the 12*c* systems.

   The file is present at the following location on the 11*g* system:

   (UNIX) *EXISTING_DOMAIN_HOME*`/config/OracleBIJavaHostComponent/`
   `coreapplication_obijh1/config .xml`

   (Windows)
   EXISTING_DOMAIN_HOME`\config\OracleBIJavaHostComponent\coreapp`
   `lication_obijh1\config .xml`

   The file is present at the following location on the 12*c* system:

   (UNIX) *NEW_DOMAIN_HOME*`/config/fmwconfig/biconfig/OBIJH/`
   `config.xml`

   (Windows)
   NEW_DOMAIN_HOME`\config\fmwconfig\biconfig\OBIJH\config.xml`

2. Verify that the XMLP configuration on both the 11*g* and the 12*c* systems is as
   shown in the following code block:

   ```
   <XMLP>
   <InputStreamLimitInKB>8192</InputStreamLimitInKB>
   <ReadRequestBeforeProcessing>true</ReadRequestBeforeProcessing>
   </XMLP>
   ```

3. Verify that the `OBIJH_ARGS="-server -Xmx1024M -Xrs"` parameter in the
   12*c* setOBIJHEnv.sh file matches with the `"start-args" value="-server`
   `-Xmx1024M -Xrs` parameter in the 11*g* opmn.xml file.

   The opmn.xml file is located at the following location on the 11*g* system:

   MW_HOME`/instances/config/OPMN/opmn/opmn.xml`

   ```
   <process-type id="OracleBIJavaHostComponent" module-id="CUSTOM">
   <module-data>
   <category id="start-parameters">
   <data id="start-executable" value="$ORACLE_HOME/jdk/bin/java"/>
   <data id="start-args" value="-server -Xmx1024M -Xrs
   .
   .
   .
   ```

   The setOBIJHEnv.sh file is located at the following location on the 12*c* system:

   (UNIX) *NEW_ORACLE_HOME*`/bi/modules/oracle.bi.cam.obijh/`
   `setOBIJHEnv.sh`

   (Windows)
   NEW_ORACLE_HOME`\bi\modules\oracle.bi.cam.obijh\setOBIJHEnv.sh`

Enabling Clusters

After migrating the data, you must manually enable the ClusterEnabled parameter in
the ClusterConfig.xml file to turn on the cluster instances on the 12*c* system.

To enable the clusters:

1. Open the ClusterConfig.xml file for editing, located at the following location:

    On UNIX operating system:

    `DOMAIN_HOME/config/fmwconfig/biconfig/core`

    On Windows operating system:

    `DOMAIN_HOME\config\fmwconfig\biconfig\core`

2. Locate the following:

    `<ClusterEnabled>false</ClusterEnabled>`

    Change it to the following:

    `<ClusterEnabled>true</ClusterEnabled>`

3. Save and close the file.

Enabling Oracle Hardware Acceleration and Compatibility Mode

Hardware acceleration affects the autocompletion of features such as dashboard prompts, trellis charts and microcharts, Summary Advisor functionality, and aggregate persistence for the TimesTen In-Memory database. Therefore, you must manually enable the bi:hw-acceleration flag in the 12*c* bi-config.xml file post-migration. You must also enable the bi:compat-mode-11g flag, so that the state of the 11*g* system is preserved at run time.

To enable the bi:hw-acceleration flag:

1. Locate the bi-config.xml, available at the following location:

    (UNIX) `12c_DOMAIN_HOME/config/fmwconfig/biconfig/core/bi-config.xml`

    (Windows) `12c_DOMAIN_HOME\config\fmwconfig\biconfig\core\bi-config.xml`

    Replace the value of *DOMAIN_HOME* with the path of the Oracle BI domain on the 12*c* system.

2. Locate the following:

    `<bi:hw-acceleration>false</bi:hw-acceleration>`

    Replace it with the following:

    `<bi:hw-acceleration>true</bi:hw-acceleration>`

3. Locate the following:

    `<bi:compat-mode-11g>false</bi:compat-mode-11g>`

Replace it with the following:

```
<bi:compat-mode-11g>true</bi:compat-mode-11g>
```

4. Save and close the file.

Setting the Compatibility Framework for Oracle BI Server

The compatibility framework allows the BI Server to add new features or bug fixes in Oracle BI 12*c* (12.2.1.1 and later) that are not compatible with the Oracle BI 11*g* releases while providing a flexible framework that enables the BI Server to operate in a compatibility mode with the earlier major release. Individual features and bug fixes can be enabled or disabled independently using compatibility flags. Alternatively, you can set all the compatibility flags to the default values using a single COMPATIBLE_RELEASE flag for the Oracle BI system to be compatible with the earlier release, which is 11.1.1.9. Add the COMPATIBLE_RELEASE parameter to the NQSConfig.INI file to ensure that the migrated 12*c* environment behaves as closely as possible to Oracle Business Intelligence 11*g* Release 1 (11.1.1.9) environment.

> **Note:**
>
> If the COMPATIBLE_RELEASE parameter is not set, then the system defaults to the behavior of the current release, such as 12.2.1.3.0.

The framework does not distinguish individual bundle patches (only the first four digits of the version number are significant for determining compatibility).

When applying new features, enhancements, or bug fixes that are not inherently backward-compatible, ensure that you specify a compatibility flag using the guidelines described in this topic.

> **Note:**
>
> The NQSConfig.INI file does not contain the COMPATIBLE_RELEASE parameter by default. The global bi:compat-mode-11g is the default flag and it applies to all Oracle BI EE components. When the bi:compat-mode-11g element in the bi-config.xml file is set to `true`, the BI Server compatibility framework behaves as if the COMPATIBLE_RELEASE were set to 11.1.1.9 and it disables any features or bug fixes that would cause major compatibility issues during migration from 11*g* to 12*c*. However, if you explicitly edit the NQSConfig.INI file to set the COMPATIBLE_RELEASE parameter or any other compatibility flag, then the setting in the NQSConfig.INI file takes precedence over the global bi:compat-mode-11g flag specified in the bi-config.xml file.

In addition to setting the general COMPATIBLE_RELEASE parameter, you can set compatibility flags for specific features or bug fixes, which begin with the prefix

"OBIS_". Instructions to set these specific compatibility flags are provided by the support team to solve certain issues on a need basis. For example,

```
[ COMPATIBILITY ]
COMPATIBLE_RELEASE=11.1.1.9;
OBIS_ENABLE_DIMENSIONALITY=0;
```

Setting the COMPATIBLE_RELEASE parameter or specific compatibility flags in the NQSConfig.INI file applies the changes to the entire system. You can also set the COMPATIBLE_RELEASE parameter or the specific feature flag as a session variable or an environment variable so that the compatibility mode is set to specific reports or dashboards. You can do this by adding a query prefix to an analysis request. The following is an example of a variable setting:

```
set variable COMPATIBLE_RELEASE='11.1.1.9':
```

Migrating the Fusion Middleware MapViewer Configuration

Oracle Fusion Middleware Mapviewer (MapViewer) is a programmable tool for rendering maps using spatial data managed by Oracle Spatial and Graph or Oracle Locator (also referred to as Locator). MapViewer provides tools that hide the complexity of spatial data queries and cartographic rendering, while providing customizable options for more advanced users. These tools can be deployed in a platform-independent manner and are designed to integrate with map-rendering applications. After migrating the data, you must manually modify the 12*c* MapViewer configuration file to contain the same contents as the 11*g* file except the Logging section.

To migrate the MapViewer configuration:

1. Go to the 12*c* Fusion Middleware MapViewer Administration Console.

2. Create a copy of the default 12*c* MapViewer configuration file.

3. Overwrite the contents in the copy of the 12*c* MapViewer configuration file with the content from the mapViewerConfig.xml file from the 11*g* MapViewer install.

4. Replace the logging section of the MapViewer configuration file copy with the settings from the default 12*c* MapViewer configuration file.

5. Update the logging settings log levels of the MapViewer configuration file as needed for the 12*c*.

6. Click **Save** on the 12*c* MapViewer Administration Console and then click **Restart**.

Resolving Authentication Issues After Migration

To avoid authentication issues post-migration, you must uncheck the **Required for Authentication** option in the DYNAMIC_OLAP_LOGIN initialization block.

> ✎ **Note:**
>
> Follow this procedure only if you are migrating from Oracle BI Release 11.1.1.7 to 12*c*.

To resolve the authentication issues after migration:

1. Open the repository in offline mode.

   a. Change to the following directory on the 12*c* system:

      (UNIX) `12c_DOMAIN_HOME/bitools/bin/`

      (Windows) `12c_DOMAIN_HOME\bitools\bin\`

      Replace the value of *DOMAIN_HOME* with the actual Domain home on your 12*c* system.

   b. Run the following command:

      ```
      sh data-model-cmd.sh downloadrpd -u uname -p pwd -o
      downloaded.rpd -w rpd_password
      ```

      Where,

      -o is the output RPD
      -w *rpd_password* is the RPD password, say Admin123

   c. Open the downloaded repository offline in the BI Administration Tool, which you installed with the client installer.

2. Complete the following steps:

   a. From the Manage menu in the BI Administration Tool, select **Variables**.

   b. In the Variable Manager dialog, from the Action menu, select **Session**, then **Initialization Blocks**.

   c. In the Variable Initialization Block dialog, locate DYNAMIC_OLAP_LOGIN.

   d. Open the DYNAMIC_OLAP_LOGIN properties, and uncheck the **Required for Authentication** checkbox.

   e. Click **Apply** and save the repository.

3. Open the repository in online mode using the `uploadrpd` command.

   ```
   12c_DOMAIN_HOME/bitools/bin/data-model-cmd.sh uploadrpd -u
   uname -p pwd -i downloaded.rpd -w rpd_password
   ```

Copying Configuration Files

You must manually copy the following configuration files after the migration: writebacktemplate.xml, userpref_currencies.xml, and bicustom.ear.

Copy the following configuration files to the new BI instance:

1. Change to the following directory:

   (UNIX) `DOMAIN_HOME/bi/bifoundation/web/msgdb/messages`

   (Windows) `DOMAIN_HOME\bi\bifoundation\web\msgdb\messages`

2. Copy the writebacktemplate.xml file to the 12*c* instance.

3. Change to the following directory:

   (UNIX) `APPLICATION_HOME/bi/bidata/components/OBIPS/`

   (Windows) `APPLICATION_HOME\bi\bidata\components\OBIPS\`

4. Copy the bicustom.ear file to the 12*c* instance.

ORACLE®

5. Change to the following directory:

    (UNIX) *DOMAIN_HOME*`/config/fmwconfig/biconfig/OBIPS`

    (Windows) *DOMAIN_HOME*`\config\fmwconfig\biconfig\OBIPS`

6. Copy the userpref_currencies.xml file to the 12*c* instance.

7. Open the userpref_currencies.xml file for editing and add the following lines:

    ```
    <Config>
    <UserCurrencyPreferences currencyTagMappingType="static">
      <UserCurrencyPreference sessionVarValue="gc1" displayText="Global
    Currency 1" currencyTag="int:USD" />
      <UserCurrencyPreference sessionVarValue="gc2" displayText="Global
    Currency 2" currencyTag="int:euro-l" />
      <UserCurrencyPreference sessionVarValue="gc3" displayText="Global
    Currency 3" currencyTag="loc:ja-JP" />
      <UserCurrencyPreference sessionVarValue="orgc" displayText="Org
    Currency" currencyTag="loc:en-BZ" />
      <UserCurrencyPreference sessionVarValue="lc1"
    displayTag="int:DEM" currencyTag="int:DEM" />
    </UserCurrencyPreferences>
    </Config>
    ```

    Save and close the file.

8. Change to the following directory:

    (UNIX) *DOMAIN_HOME*`/config/fmwconfig/biconfig/OBIPS`

    (Windows) *DOMAIN_HOME*`\config\fmwconfig\biconfig\OBIPS`

9. Open the instanceconfig.xml file for editing and add the following line:

    ```
    <UserprefCurrenciesConfigFile>DOMAIN_HOME/config/fmwconfig/biconfig/
    OBIPS/userpref_currencies.xml</UserprefCurrenciesConfigFile>
    ```

    Save and close the file.

10. Restart the services.

Removing the Display of HTML Codes in a Customized "No Results" Message

In an analysis, content developers can control the text that is displayed when the results of the analysis return no data. Content developers can accept the default message, or they can customize the text of the message, including by inserting HTML formatting codes in the message.

If content developers customized the message and included HTML codes in Release 11*g*, then the message is displayed with the codes showing as readable text in Release 12*c*, rather than performing their functions to format the message. To work around this issue, open the Analysis Properties dialog for each analysis whose custom message includes HTML codes and select the **Contains HTML Markup** option.

For information on the use of the **Contains HTML Markup** option, see EnableSavingContentWithHTML in *Security Guide for Oracle Business Intelligence Enterprise Edition*.

5
Post-Migration Tasks for Oracle BI Publisher

After completing the data migration, you must migrate the BI Publisher configuration, scheduler jobs, and job history. You must also remove the BISystemUser policy from the JMSResource security configuration.

- Migrating Scheduler Jobs and Job History
  After migrating the 11g configuration to a new 12c environment, you must migrate the scheduler jobs and the job history data from the 11g BIPLATFORM schema to a 12c BIPLATFORM schema. You can connect to 12c BIPLATFROM schema through SQL*PLUS or any other tool and execute the script.

- Removing the BISystemUser Policy from the JMSResource Security Configuration
  BISystemUser is not supported in the 12*c* environment. When the users and policies are migrated from an 11*g* instance, the JMSResource policy blocks the scheduled jobs. Therefore, you must remove this policy from the JMSResource security configuration. You can manually remove the BISystemUser policy from WebLogic console. Alternatively, you can execute a WLST (Python) script to accomplish this task.

Migrating Scheduler Jobs and Job History

After migrating the 11g configuration to a new 12c environment, you must migrate the scheduler jobs and the job history data from the 11g BIPLATFORM schema to a 12c BIPLATFORM schema. You can connect to 12*c* BIPLATFROM schema through SQL*PLUS or any other tool and execute the script.

The bip_12c_scheduler_migration.sql script requires the user ID, the password, and the connection string for 11g BIPLATFROM schema. The script creates a database link called BIP11g_DBLINK and imports all schedule objects using that link. You must have Database Administrator privileges to run the script.

> **Note:**
>
> You cannot migrate scheduler data from DB2 and SQL Server databases. You can migrate scheduler data only from Oracle databases.

To migrate scheduler jobs and job history:

1. Change to the directory appropriate for your platform:

   (UNIX) *NEW_ORACLE_HOME*/bi/modules/oracle.bi.publisher/upgradeutil

   (Windows)
   NEW_ORACLE_HOME\bi\modules\oracle.bi.publisher\upgradeutil

Replace *NEW_ORACLE_HOME* with the actual path to the 12c Oracle home you created when you installed the 12c software.

2. Connect to the target 12c BIPLATFORM schema from SQLPLUS or SQL Developer by entering the following command:

```
sqlplus userid/password@connectionString -- for 12c RCU
Schema
```

For example:

```
sqlplus username/password@hostname.yourcompany.com:1521/
pdborcl.yourcompany.com
```

3. Run the `bip_12c_scheduler_migration.sql` script, passing the command-line parameters `11g_RCU_SCHEMA`, `11G_PASSWORD`, `11G_CONNECTION_STRING`, and `12C_BIPLATFORM_SCHEMA`.

Usage:

```
SQL> @bip_12c_scheduler_migrationg.sql 11g_userid
11g_password 11g_connection_string
```

Sample command:

```
SQL> bip_12c_scheduler_migration.sql 11g_RCU_USERNAME
11g_PASSSWORD 11G_hostname.yourcompany.com:1521/
orcl.yourcompany.com 12C_BIPLATFROM_SCHEMA_NAME
System output:
old 1: &&1
new 1: adc00ccq_biplatform
old 2: &&2
new 2: your_password
old 3: '&&3'
new 3: 'bipdev4.yourcompany.com:1521/ora11g.yourcompany.com'
old 4: '&&4'
new 4: 12C_BIPLATFROM_SCHEMA_NAME
Database link created.
9979 rows created.
9769 rows created.
9739 rows created.
4159 rows created.
6 rows created.
6 rows created.
6 rows created.
Commit complete.
Database link dropped.
SQL>
```

After the import operation completes, the database link is deleted.

Removing the BISystemUser Policy from the JMSResource Security Configuration

BISystemUser is not supported in the 12*c* environment. When the users and policies are migrated from an 11*g* instance, the JMSResource policy blocks the scheduled jobs. Therefore, you must remove this policy from the JMSResource security

configuration. You can manually remove the BISystemUser policy from WebLogic console. Alternatively, you can execute a WLST (Python) script to accomplish this task.

- Manually Removing the BISystemUser Policy
  This topic describes the procedure to remove the BISystemUser policy from the WebLogic Console.

- Removing the BISystemUser Policy using a WLST Script
  This topic describes the procedure to remove the BISystemUser policy using a WLST script.

Manually Removing the BISystemUser Policy

This topic describes the procedure to remove the BISystemUser policy from the WebLogic Console.

To manually remove the BISystemUser policy:

1. Sign in to the WebLogic Console.

2. Under **Services**, click **Messaging** and then **JMSModules**, and select **BipJmsResource**.

Figure 5-1   JMS Modules

3. Under the Security tab, click **Policies**.

4. Under Policy Conditions, select the **User: BISystemUser** checkbox and click **Remove**.

Figure 5-2   Settings for BipJmsResource

5. Restart the BI Publisher and check the scheduled jobs.

Removing the BISystemUser Policy using a WLST Script

This topic describes the procedure to remove the BISystemUser policy using a WLST script.

To remove the BISystemUser policy:

1. Copy the BIPRemoveJMSResourcePolicy.py script from the following location:

   (UNIX) *NEW_ORACLE_HOME*/bi/modules/oracle.bi.publisher

   (Windows) *NEW_ORACLE_HOME*\bi\modules\oracle.bi.publisher

2. Go to *NEW_ORACLE_HOME*/server/bin directory.

3. Set the WLS environment. Enter

   ```
   source ./setWLSEnv.sh
   ```

4. Run the script to remove JMS Resource policy. Enter

   ```
   java weblogic.WLST BIPRemoveJMSResourcePolicy.py <WL Admin
   Username> <WL Admin Password> <Admin Server URL: t3://
   localhost:port>
   ```

5. Sign in to the WebLogic Console and verify whether the policy is removed.

6
Post-Migration Tasks for Essbase

After completing data migration, manually migrate the configuration settings from the 11*g* system in to the 12*c* system and complete the post-migration tasks.

See Understanding Essbase Deployed in BI 12.2.1 in Oracle® Essbase Database Administrator's Guide for the procedure to migrate the following configuration settings from the 11*g* system to the 12*c* system and to complete the post-migration tasks:

- Thread Pool Configuration for Essbase
- ESSLANG Configuration Setting for Essbase Server
- Configuring Modes, Ports, and Cipher Suites
- Configuring wallets for root, Essbase Server, and Essbase clients
- Importing certificates in to the key store

7

Upgrading Oracle Business Intelligence from a Previous 12*c* Release

This chapter contains the procedure to upgrade Oracle Business Intelligence from Release 12.2.1.x to Release 12.2.1.3.0. The upgrade is performed in-place, that is, the upgrade operations are performed on the existing 12.2.1.x domain.

- **About the Oracle Business Intelligence Upgrade Process**
  Review the flowchart and roadmap for an overview of the upgrade process for Oracle Business Intelligence upgrade from a previous 12*c* release.

- **Installing the Product Distributions**
  Before you begin your upgrade, download Oracle Fusion Middleware Infrastructure and Oracle Business Intelligence 12*c* (12.2.1.3.0) distributions on the target system and install them by using Oracle Universal Installer.

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

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

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

- **Backing Up the mapViewerConfig.xml File**
  The mapViewerConfig.xml file is overwritten by the reconfiguration templates when you run the Reconfiguration Wizard. Therefore, you must back up the mapViewerConfig.xml file before reconfiguring your existing domain.

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

- **Restoring the mapViewerConfig.xml File**
  The mapViewerConfig.xml file is overwritten when you run the Reconfiguration Wizard on your domain. You can know that the mapViewerConfig.xml file is overwritten by checking its file size. You must restore the original file that you backed up before upgrading your domain with the Upgrade Assistant.

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

- Verifying the Domain-Specific-Component Configurations Upgrade
  To verify that the domain-specific-component configurations upgrade was
  successful, sign in to the Administration console and the Oracle Enterprise
  Manager Fusion Middleware Control and verify that the version numbers for each
  component is 12.2.1.3.0.

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

- Validating the Oracle BI Deployments
  The Oracle BI Baseline Validation Tool enables you to identify differences during
  life cycle operations, such as migrating from the Oracle BI 11*g* release to the
  12*c* release. After you complete the migration procedure, you can use this tool
  to compare the two deployments and verify whether the results from the 11*g*
  environment are the same as the results from the 12*c* environment.

About the Oracle Business Intelligence Upgrade Process

Review the flowchart and roadmap for an overview of the upgrade process for Oracle
Business Intelligence upgrade from a previous 12*c* release.

**Figure 7-1   Upgrade Process Flowchart for Oracle Business Intelligence from a
Previous 12*c* Release**
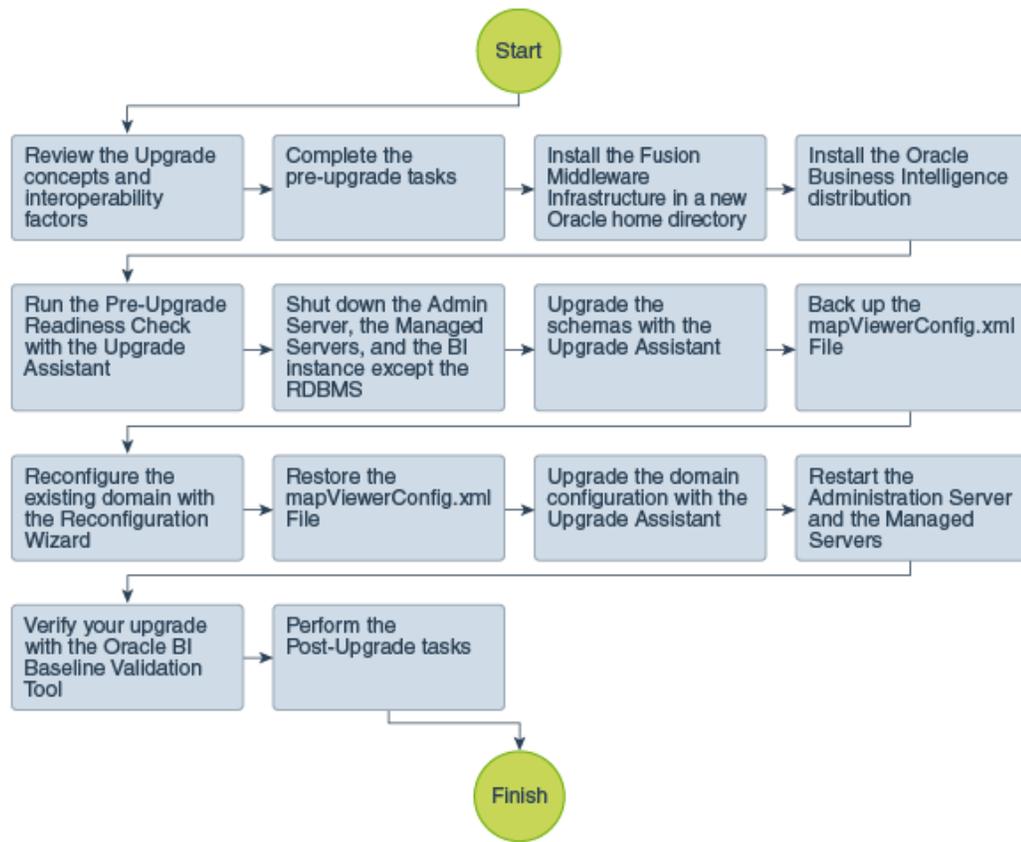
Table 7-1 lists the high-level steps that you need to perform to upgrade to Oracle BI Release 12.2.1.3.0:

Table 7-1   Tasks for Upgrading Oracle Business Intelligence from a Pervious 12*c* Release

Task	Description
Optional Learn about the interoperability and compatibility factors that could affect how you upgrade to Oracle BI 12.2.1.3.0.	It is important to understand how two or more Oracle Fusion Middleware products of the same version or different versions work together (interoperate) in a supported Oracle Fusion Middleware configuration. You can learn more about interoperability and compatibility in Oracle® Fusion Middleware Understanding Interoperability and Compatibility.
Required If you have not done so already, review the introductory topics in this guide and complete the required pre-upgrade tasks.	The pre-upgrade tasks include cloning your production environment, verifying system requirements and certifications, purging unused data, and creating a non-SYSDBA user. For a complete list of pre-upgrade tasks, see Pre-Upgrade Requirements
Required Download and install the 12.2.1.3.0 Fusion Middleware Infrastructure and Oracle Business Intelligence distributions.	The Infrastructure distribution combines the WebLogic Server and the Java Required Files (JRF) that are required to set up the foundation to install other Fusion Middleware products. As per the upgrade topology defined in this guide, you must install the Infrastructure in a new Oracle home. The Oracle Business Intelligence distribution packs Oracle BI EE, Publisher, and Essbase. You must install Oracle BI in the Oracle home that is created when you installed the 12.2.1.3.0 Infrastructure. To install the product distributions, follow the procedure described in Installing the Product Distributions.
Optional Run the Readiness Check.	See Running a Pre-Upgrade Readiness Check.
Required Shut down the servers and Oracle BI instance.	Before starting the upgrade process, shut down the Administration Server, the Managed Servers, and your existing Oracle BI instance. However, keep the database (RDBMS) running. See Stopping Servers and Processes.
Required Upgrade the existing schemas with the Upgrade Assistant.	The schemas that you had created during the earlier 12*c* versions are supported in 12.2.1.3.0. Therefore, you don't need to create the schemas again. You must upgrade all the schemas within your domain using the Upgrade Assistant. For the complete procedure, see Upgrading Product Schemas Using the Upgrade Assistant.
Required Backup the mapViewerConfig.xml File.	The mapViewerConfig.xml file is overwritten by the reconfiguration templates when you run the Reconfiguration Wizard. Therefore, you must back up the mapViewerConfig.xml file before reconfiguring your existing domain. See Backing Up the mapViewerConfig.xml File.

Table 7-1   (Cont.) Tasks for Upgrading Oracle Business Intelligence from a Pervious 12c Release

Task	Description
Required Reconfigure the existing domain with the Reconfiguration Wizard.	When you run the Reconfiguration Wizard on your existing domain, it prepares your domain for upgrade by selecting and applying the recongifuration templates. It also tests the JDBC data sources and component schemas that are present within your domain. To reconfigure you domain, follow the procedure described in Reconfiguring the Oracle BI Domain with the Reconfiguration Wizard.
Required Restore the mapViewerConfig.xml File.	You must restore the original file that you backed up before upgrading your domain with the Upgrade Assistant. See Restoring the mapViewerConfig.xml File.
Required Upgrade the existing domain configurations with the Upgrade Assistant.	After you have reconfigured your existing domain, you must run the Upgrade Assistant to upgrade all configurations used by your existing domain. You can see all the components within your domain that will be upgraded on the Component List screen when you run the Upgrade Assistant. For the complete procedure, see Upgrading Domain Components Using the Upgrade Assistant.
Required Restart the servers and 12.2.1.3.0 Oracle BI instance.	The upgrade process is complete. You can now restart the Administration Server, the Managed Servers, and your 12.2.1.3.0 Oracle BI instance. See Starting Servers and Processes.
Required Verify your upgrade.	It is important to compare your existing 12c and 12.2.1.3.0 environments and verify that the data and configuration settings are consistent in the newly upgraded environment. Oracle provides the BI Baseline Validation Tool that you can use to compare and verify the upgrade. See Validating the Oracle BI Deployments.
Required Perform the post-upgrade tasks.	For a list of post-upgrade tasks, see Post-Upgrade Tasks.

Installing the Product Distributions

Before you begin your upgrade, download Oracle Fusion Middleware Infrastructure and Oracle Business Intelligence 12c (12.2.1.3.0) distributions on the target system and install them by using Oracle Universal Installer.

> **✎ Note:**
>
> When Infrastructure is required for the upgrade, you must install the Oracle Fusion Middleware distribution first before you install other Fusion Middleware products.

To install the 12*c* (12.2.1.3.0) distributions:

1. Sign in to the target system.

2. Download the following from Oracle Technology Network or Oracle Software Delivery Cloud to your target system:

   • Oracle Fusion Middleware Infrastructure (`fmw_12.2.1.3.0_infrastructure_generic.jar`)

   • Oracle Business Intelligence (UNIX: `fmw_12.2.1.3.0_bi_platform_linux64.bin`), (Windows: `setup_fmw_12.2.1.3.0_bi_platform_win64.exe`)

3. Change to the directory where you downloaded the 12*c* (12.2.1.3.0) product distribution.

4. Start the installation program for Oracle Fusion Middleware Infrastructure:

   • (UNIX) *JDK_HOME*`/bin/java -jar fmw_12.2.1.3.0_infrastructure_generic.jar`

   • (Windows) *JDK_HOME*`\bin\java -jar fmw_12.2.1.3.0_infrastructure_generic.jar`

5. On UNIX operating systems, the Installation Inventory Setup screen appears if this is the first time you are installing an Oracle product on this host.

   Specify the location where you want to create your central inventory. Make sure that the operating system group name selected on this screen has write permissions to the central inventory location, and click **Next**.

   > **Note:**
   >
   > The Installation Inventory Setup screen does not appear on Windows operating systems.

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

7. On the Auto Updates screen, select an option:

   • **Skip Auto Updates**: If you do not want your system to check for software updates at this time.

   • **Select patches from directory**: To navigate to a local directory if you downloaded patch files.

   • **Search My Oracle Support for Updates**: To automatically download software updates if you have a My Oracle Support account. You must enter Oracle Support credentials then click **Search**. To configure a proxy server for the installer to access My Oracle Support, click **Proxy Settings**. Click **Test Connection** to test the connection.

   Click **Next**.

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

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

9. On the Installation Type screen, select the following:

   • For Infrastructure, select **Fusion Middleware Infrastructure**.

   • For Oracle Business Intelligence, select **BI Platform Distribution with Samples**.

   Click **Next**.

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

    To view the list of tasks that are verified, select **View Successful Tasks**. To view log details, select **View Log**. If any prerequisite check fails, then an error message appears at the bottom of the screen. Fix the error and click **Rerun** to try again. To ignore the error or the warning message and continue with the installation, click **Skip** (not recommended).

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

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

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

13. The Installation Complete screen displays the Installation Location and the Feature Sets that are installed. Review this information and click **Finish** to close the installer.

14. After you have installed Oracle Fusion Middleware Infrastructure, enter the following command to start the installer for your product distribution and repeat the steps above to navigate through the installer screens:

    (UNIX) `./fmw_12.2.1.3.0_bi_platform_linux64.bin`

    (Windows) `setup_fmw_12.2.1.3.0_bi_platform_win64.exe`

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 7-2   Upgrade Assistant Command-Line Parameters

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

Table 7-2   (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 `-readiness` parameter.
`-logLevel attribute`	Optional	Sets the logging level, specifying one of the following attributes: • `TRACE` • `NOTIFICATION` • `WARNING` • `ERROR` • `INCIDENT_ERROR` The default logging level is `NOTIFICATION`. Consider setting the `-logLevel TRACE` attribute to so that more information is logged. This is useful when troubleshooting a failed upgrade. The Upgrade Assistant's log files can become very large if `-logLevel TRACE` is used.
`-logDir location`	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) *NEW_ORACLE_HOME*`/oracle_common/upgrade/logs` *NEW_ORACLE_HOME*`/oracle_common/upgrade/temp` (Windows) *NEW_ORACLE_HOME*`\oracle_common\upgrade\logs` *NEW_ORACLE_HOME*`\oracle_common\upgrade\temp`
`-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 7-3   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 `ORACLE_HOME/ oracle_common/upgrade/ logs`	The directory location of the generated log file.	No action required.
Readiness report location `ORACLE_HOME/ 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 March 30 11:15:52 EDT
2019
Log file is located at: ORACLE_HOME/oracle_common/upgrade/logs/
ua2016-05-30-11-14-06AM.log
Readiness Check Report File: ORACLE_HOME/oracle_common/upgrade/logs/
readiness2016-05-30-11-15-52AM.txt

Starting readiness check of components.

Oracle Metadata Services
   Starting readiness check of Oracle Metadata Services.
     Schema User Name: DEV11_MDS
     Database Type: Oracle Database
     Database Connect String: machinename@yourcompany.com
     VERSION Schema DEV11_MDS is currently at version 12.2.1.4.0.
Readiness checks will now be performed.
   Starting schema test:  TEST_REQUIRED_TABLES  Test that the schema
contains all the required tables
   Completed schema test: TEST_REQUIRED_TABLES --> Test that the schema
contains all the required tables +++ PASS
   Starting schema test:  TEST_REQUIRED_PROCEDURES  Test that the
schema contains all the required stored procedures
     EXCEPTION     Schema is missing a required procedure:
GETREPOSITORYFEATURES
   Completed schema test: TEST_REQUIRED_PROCEDURES --> Test that the
schema contains all the required stored procedures +++ FAIL
   Starting schema test:  TEST_REQUIRED_VIEWS  Test that the schema
contains all the required database views
   Completed schema test: TEST_REQUIRED_VIEWS --> Test that the schema
contains all the required database views +++ PASS
   Starting index test for table MDS_ATTRIBUTES:  TEST_REQUIRED_INDEXES
--> Test that the table contains all the required indexes
   Completed index test for table MDS_ATTRIBUTES: TEST_REQUIRED_INDEXES
--> Test that the table contains all the required indexes +++ PASS
   Starting index test for table MDS_COMPONENTS:  TEST_REQUIRED_INDEXES
--> Test that the table contains all the required indexes
   Completed index test for table MDS_TXN_LOCKS: TEST_REQUIRED_INDEXES
--> Test that the table contains all the required indexes +++ PASS
   Starting schema test:  TEST_REQUIRED_TRIGGERS  Test that the schema
has all the required triggers
   Completed schema test: TEST_REQUIRED_TRIGGERS --> Test that the
schema has all the required triggers +++ PASS
   Starting schema test:  TEST_MISSING_COLUMNS  Test that tables and
views are not missing any required columns
   Completed schema test: TEST_MISSING_COLUMNS --> Test that tables and
views are not missing any required columns +++ PASS
   Starting schema test:  TEST_UNEXPECTED_TABLES  Test that the schema
does not contain any unexpected tables
   Completed schema test: TEST_UNEXPECTED_TABLES --> Test that the
schema does not contain any unexpected tables +++ PASS
```

    Starting schema test: TEST_UNEXPECTED_PROCEDURES   Test that the
schema does not contain any unexpected stored procedures
    Completed schema test: TEST_UNEXPECTED_PROCEDURES --> Test that the
schema does not contain any unexpected stored procedures +++ PASS
    Starting schema test: TEST_UNEXPECTED_VIEWS   Test that the schema
does not contain any unexpected views
    Completed schema test: TEST_UNEXPECTED_VIEWS --> Test that the
schema does not contain any unexpected views +++ PASS
    Starting index test for table MDS_ATTRIBUTES:
TEST_UNEXPECTED_INDEXES --> Test that the table does not contain any
unexpected indexes
    Completed index test for table MDS_ATTRIBUTES:
TEST_UNEXPECTED_INDEXES --> Test that the table does not contain any
unexpected indexes +++ PASS
    Completed index test for table MDS_LABELS: TEST_UNEXPECTED_INDEXES
--> Test that the table does not contain any unexpected indexes +++ PASS
    Starting index test for table MDS_LARGE_ATTRIBUTES:
TEST_UNEXPECTED_INDEXES --> Test that the table does not contain any
unexpected indexes
    Starting schema test: TEST_UNEXPECTED_TRIGGERS   Test that the
schema does not contain any unexpected triggers
    Completed schema test: TEST_UNEXPECTED_TRIGGERS --> Test that the
schema does not contain any unexpected triggers +++ PASS
    Starting schema test: TEST_UNEXPECTED_COLUMNS   Test that tables and
views do not contain any unexpected columns
    Completed schema test: TEST_UNEXPECTED_COLUMNS --> Test that tables
and views do not contain any unexpected columns +++ PASS
    Starting datatype test for table MDS_ATTRIBUTES:
TEST_COLUMN_DATATYPES_V2 --> Test that all table columns have the
proper datatypes
    Completed datatype test for table MDS_ATTRIBUTES:
TEST_COLUMN_DATATYPES_V2 --> Test that all table columns have the
proper datatypes +++ PASS
    Starting datatype test for table MDS_COMPONENTS:
TEST_COLUMN_DATATYPES_V2 --> Test that all table columns have the
proper datatypes
    Starting permissions test: TEST_DBA_TABLE_GRANTS   Test that DBA
user has privilege to view all user tables
    Completed permissions test: TEST_DBA_TABLE_GRANTS --> Test that DBA
user has privilege to view all user tables +++ PASS
    Starting schema test: TEST_ENOUGH_TABLESPACE   Test that the schema
tablespaces automatically extend if full
    Completed schema test: TEST_ENOUGH_TABLESPACE --> Test that the
schema tablespaces automatically extend if full +++ PASS
    Starting schema test: TEST_USER_TABLESPACE_QUOTA   Test that
tablespace quota for this user is sufficient to perform the upgrade
    Completed schema test: TEST_USER_TABLESPACE_QUOTA --> Test that
tablespace quota for this user is sufficient to perform the upgrade
+++ PASS
    Starting schema test: TEST_ONLINE_TABLESPACE   Test that schema
tablespaces are online
    Completed schema test: TEST_ONLINE_TABLESPACE --> Test that schema
tablespaces are online +++ PASS
    Starting schema test: TEST_DATABASE_VERSION   Test that the database
server version number is supported for upgrade

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

If you are running the 12.1.3.0 version of Oracle Fusion Middleware IAU Schemas, and those schemas were upgraded from 11*g* (11.1.1.7 and later) or 12*c* (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 12*c* (12.2.1.3.0) by using the RCU.

Stopping Servers and Processes

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

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

To stop your Fusion Middleware environment:

1. Change to the following directory:

   (UNIX) `12c_DOMAIN_HOME/bitools/bin`

   (Windows) `12c_DOMAIN_HOME\bitools\bin`

2. To stop the Oracle BI instance and servers, enter the following command:

   (UNIX) `stop.sh`

   (Windows) `stop.cmd`

> **Note:**
>
> When prompted to enter the password, specify the Node Manager password that you entered while configuring the Oracle BI domain.

Upgrading Product Schemas

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

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

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

- Upgrading Product Schemas Using the Upgrade Assistant
  Navigate through the screens in the Upgrade Assistant to upgrade the product schemas.

- Verifying the Schema Upgrade
  After completing all the upgrade steps, verify that the upgrade was successful by checking that the schema version in `schema_version_registry` has been properly updated.

Starting the Upgrade Assistant

Run the Upgrade Assistant to upgrade product schemas, domain component configurations, or standalone system components to 12*c* (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 7-4   Upgrade Assistant Command-Line Parameters

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

Table 7-4   (Cont.) Upgrade Assistant Command-Line Parameters

Parameter	Required or Optional	Description
`-logDir` *location*	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)
		NEW_ORACLE_HOME`/oracle_common/upgrade/logs`
		NEW_ORACLE_HOME`/oracle_common/upgrade/temp`
		(Windows)
		NEW_ORACLE_HOME`\oracle_common\upgrade\logs`
		NEW_ORACLE_HOME`\oracle_common\upgrade\temp`
`-help`	Optional	Displays all of the command-line options.

Upgrading Product Schemas Using the Upgrade Assistant

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

If you are using DB2 in your existing BI instance, then perform the following before starting the upgrade process:

1. Sign in to the WebLogic Console.

2. Go to **Datasources** under **Services** and select **wlsservices_datasource**.

3. Change the user for data source 'wlsservices_datasource' from *<PREFIX>*_WLS_RUNTIME to *<PREFIX>*_WLS.

   For example: Change the user from L1_WLS_RUNTIME to L1_WLS.

You can now proceed with the upgrade process.

To upgrade product schemas with the Upgrade Assistant:

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

> **✎ Note:**
>
> > For more information about any Upgrade Assistant screen, click **Help** on the screen.

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

   - **All Schemas Used by a Domain** to allow the Upgrade Assistant to discover and select all components that have a schema available to upgrade in the domain specified in the **Domain Directory** field. This is also known as a *domain assisted schema upgrade*. Additionally, the Upgrade Assistant pre-populates connection information on the schema input screens.

     > **✎ Note:**
     >
     > > Oracle recommends that you select **All Schemas Used by a Domain** for most upgrades to ensure all of the required schemas are included in the upgrade.

   Click **Next**.

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

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

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

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

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

5. On the BIPLATFORM Schema [BIEE_PLATFORM] screen, the system automatically populates data in the other fields from the existing 12*c* schema.

   Enter the database administrator account: DBA username and password for the database containing the BIEE_PLATFORM schema and click **Connect**.

   When you see the message "Connection to database successfully completed", click **Next**.

6. On the IAU Schema [BIEE_IAU] screen, the system automatically populates data in the other fields from the existing 12*c* schema.

Enter the database administrator account: DBA username and password for the database containing the BIEE_IAU schema and click **Connect**.

When you see the message "Connection to database successfully completed", click **Next**.

7. On the OPSS Schema [BIEE_OPSS] screen, the system automatically populates data in the other fields from the existing 12*c* schema.

   Enter the database administrator account: DBA username and password for the database containing the BIEE_OPSS schema and click **Connect**.

   When you see the message "Connection to database successfully completed", click **Next**.

8. On the MDS Schema [BIEE_MDS] screen, the system automatically populates data in the other fields from the existing 12*c* schema.

   Enter the database administrator account: DBA username and password for the database containing the BIEE_MDS schema and click **Connect**.

   When you see the message "Connection to database successfully completed", click **Next**.

9. On the STB Schema [BIEE_STB] screen, the system automatically populates data in the other fields from the existing 12*c* schema.

   Enter the database administrator account: DBA username and password for the database containing the BIEE_STB schema and click **Connect**.

   When you see the message "Connection to database successfully completed", click **Next**.

10. On the WLS Schema [BIEE_WLS_RUNTIME] screen, the system automatically populates data in the other fields from the existing 12*c* schema.

    Enter the database administrator account: DBA username and password for the database containing the BIEE_WLS_RUNTIME schema and click **Connect**.

    When you see the message "Connection to database successfully completed", click **Next**.

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

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

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

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

    Verify that the correct Source and Target Versions are listed for each schema you intend to upgrade.

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

    Click **Next** .

13. On the Upgrade Progress screen, monitor the status of the upgrade.

> **Caution:**
>
> Allow the Upgrade Assistant enough time to perform the upgrade. Do not cancel the upgrade operation unless absolutely necessary. Doing so may result in an unstable environment.

    If any schemas are not upgraded successfully, refer to the Upgrade Assistant log files for more information.

> **Note:**
>
> The progress bar on this screen displays the progress of the current upgrade procedure. It does not indicate the time remaining for the upgrade.

    Click **Next**.

14. If the upgrade is successful: On the Upgrade Success screen, click **Close** to complete the upgrade and close the wizard.

If the upgrade fails: On the Upgrade Failure screen, click **View Log** to view and troubleshoot the errors. The logs are available at *NEW_ORACLE_HOME/* `oracle_common/upgrade/logs`.

> ✏️ **Note:**
>
> If the upgrade fails, you must restore your pre-upgrade environment from backup, fix the issues, then restart the Upgrade Assistant.

Verifying the Schema Upgrade

After completing all the upgrade steps, verify that the upgrade was successful by checking that the schema version in `schema_version_registry` has been properly updated.

If you are using an Oracle database, connect to the database as a user having Oracle DBA privileges, and run the following from SQL*Plus to get the current version numbers:

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

In the query result:

*   Check that the number in the `VERSION` column matches the latest version number for that schema. For example, verify that the schema version number is 12.2.1.3.0.

    > ✏️ **Note:**
    >
    > However, that not all schema versions will be updated. Some schemas do not require an upgrade to this release and will retain their pre-upgrade version number.

*   The `STATUS` field will be either `UPGRADING` or `UPGRADED` during the schema patching operation, and will become `VALID` when the operation is completed.

*   If the status appears as `INVALID`, the schema update failed. You should examine the logs files to determine the reason for the failure.

*   Synonym objects owned by `IAU_APPEND` and `IAU_VIEWER` will appear as `INVALID`, but that does not indicate a failure.

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

Backing Up the mapViewerConfig.xml File

The mapViewerConfig.xml file is overwritten by the reconfiguration templates when you run the Reconfiguration Wizard. Therefore, you must back up the mapViewerConfig.xml file before reconfiguring your existing domain.

Backing Up the mapViewerConfig.xml File on UNIX Operating Systems

To back up the mapViewerConfig.xml file on UNIX systems:

1. The mapViewerConfig.xml file is present in the following location on your system. Use the list command: `ls -ltr` to check its file size.

   EXISTING_DOMAIN_HOME`/config/fmwconfig/mapviewer/conf/`
   `mapViewerConfig.xml`

   `-rw-r----- 1 mwport svrtech` **2xxx7** `May  5 18:45` *EXISTING_DOMAIN_HOME*`/config/`
   `fmwconfig/mapviewer/conf/mapViewerConfig.xml`

2. Copy the mapViewerConfig.xml file as shown in the following example:

   `cp` *EXISTING_DOMAIN_HOME*`/config/fmwconfig/mapviewer/conf/`
   `mapViewerConfig.xml` *EXISTING_DOMAIN_HOME*`/config/fmwconfig/`
   `mapviewer/conf/mapViewerConfig_orig.xml`

   A copy of the mapViewerConfig.xml file is created with the filename 'mapViewerConfig_orig.xml'.

Backing Up the mapViewerConfig.xml File on Windows Operating Systems

To back up the mapViewerConfig.xml file on Windows systems:

1. Open a command prompt window and change to the following directory:

   EXISTING_DOMAIN_HOME`\config\fmwconfig\mapviewer\conf\mapViewe`
   `rConfig.xml`

2. Enter the following command to check the file size of the mapViewerConfig.xml file:

   `dir`

3. Copy the mapViewerConfig.xml file as shown in the following example:

   copy "mapViewerConfig.xml" "mapViewerConfig_original.xml"

   A copy of the mapViewerConfig.xml file is created with the filename 'mapViewerConfig_original.xml'.

About Reconfiguring the Domain

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

When you reconfigure a WebLogic Server domain, the following items are automatically updated, depending on the applications in the domain:

- WebLogic Server core infrastructure
- Domain version

> **✎ Note:**
>
> Before you begin the domain reconfiguration, note the following limitations:
>
> - The Reconfiguration Wizard does not update any of your own applications that are included in the domain.
>
> - Transforming a non-dynamic cluster domain to a dynamic cluster domain during the upgrade process is not supported.
>
>   The dynamic cluster feature is available when running the Reconfiguration Wizard, but Oracle only supports upgrading a non-dynamic cluster upgrade and then adding dynamic clusters. You cannot add dynamic cluster during the upgrade process.
>
> - If the installation that you're upgrading does not use Oracle Access Management (OAM), then you must edit two files to prevent the Reconfiguration Wizard from attempting to update the nonexistent OAM Infrastructure schema, which causes the upgrade to fail.
>
>   Comment out the lines in your `$DOMAIN/init-info/domain-info.xml` that are similar to this example:
>
> ```
> <!--extention-template-ref name="Oracle Identity Navigator"
>   version="11.1.1.3.0"
>   location="/u01/app/
> oracle/product/fmw/iam111130/common/templates/applications/
> oracle.oinav_11.1.1.3.0_template.jar"
>   symbol=""/-->
>
> <!--install-comp-ref name="oracle.idm.oinav"
> version="11.1.1.3.0"
>
> symbol="oracle.idm.oinav_11.1.1.3.0_iam111130_ORACLE_HOME"
>   product_home="/u01/app/oracle/product/fmw/iam111130"/-->
> ```
>
>   and similarly comment out the lines in `$DOMAIN/config/config.xml` that are similar to this example:
>
> ```
> <!--app-deployment>
>   <name>oinav#11.1.1.3.0</name>
>   <target>AdminServer</target>
>   <module-type>ear</module-type>
>
>   <source-path>/u01/app/oracle/product/fmw/iam111130/oinav/
> modules/oinav.ear_11.1.1.3.0/oinav.ear</source-path>
>   <deployment-order>500</deployment-order>
>   <security-dd-model>DDOnly</security-dd-model>
>   <staging-mode>nostage</staging-mode>
> </app-deployment-->
> ```

Specifically, when you reconfigure a domain, the following occurs:

- The domain version number in the `config.xml` file for the domain is updated to the Administration Server's installed WebLogic Server version.

- Reconfiguration templates for all installed Oracle products are automatically selected and applied to the domain. These templates define any reconfiguration tasks that are required to make the WebLogic domain compatible with the current WebLogic Server version.

- Start scripts are updated.

  If you want to preserve your modified start scripts, be sure to back them up before starting the Reconfiguration Wizard.

> **Note:**
>
> When the domain reconfiguration process starts, you can't undo the changes that it makes. Before running the Reconfiguration Wizard, ensure that you have backed up the domain as covered in the pre-upgrade checklist. If an error or other interruption occurs while running the Reconfiguration Wizard, you must restore the domain by copying the files and directories from the backup location to the original domain directory. This is the only way to ensure that the domain has been returned to its original state before reconfiguration.

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

- [Backing Up the Domain](#)
- [Starting the Reconfiguration Wizard](#)
- [Reconfiguring the Oracle BI Domain with the Reconfiguration Wizard](#)
  Navigate through the screens in the Reconfiguration Wizard to reconfigure your existing domain.

Backing Up the Domain

Before running the Reconfiguration Wizard, create a backup copy of the domain directory.

To create a backup of the domain directory:

1. Copy the source domain to a separate location to preserve the contents.

   (Windows) `copy C:\domains\mydomain to C:\domains\mydomain_backup`.

   (UNIX) `cp mydomain /domains/mydomain_backup`

2. Before updating the domain on each remote Managed Server, create a backup copy of the domain directory on each remote machine.

3. Verify that the backed up versions of the domain are complete.

If domain reconfiguration fails for any reason, you must copy all files and directories from the backup directory into the original domain directory to ensure that the domain is returned entirely to its original state before reconfiguration.

Starting the Reconfiguration Wizard

> **Note:**
>
> Shut down the administration server and all collocated managed servers before starting the reconfiguration process. See Stopping Servers and Processes .

To start the Reconfiguration Wizard in graphical mode:

1. Sign in to the system on which the domain resides.

2. Open the command shell (on UNIX operating systems) or open a command prompt window (on Windows operating systems).

3. **Edition Based Database Users Only**: If your schemas are configured with EBR database, a default edition name must be manually supplied before you run the Reconfiguration Wizard.

   Run the following SQL command to set the default edition:
   ```
   ALTER DATABASE DEFAULT EDITION = edition_name;
   ```

   where `edition_name` is the child edition name.

4. Go to the `oracle_common/common/bin` directory:

   - (UNIX) `NEW_ORACLE_HOME/oracle_common/common/bin`

   - (Windows) `NEW_ORACLE_HOME\oracle_common\commom\bin`

5. Start the Reconfiguration Wizard with the following logging options:

   - (UNIX) `./reconfig.sh -log=log_file -log_priority=ALL`

   - (Windows) `reconfig.cmd -log=log_file -log_priority=ALL`

   where `log_file` is the absolute path of the log file you'd like to create for the domain reconfiguration session. This can be helpful if you need to troubleshoot the reconfiguration process.

   The parameter `-log_priority=ALL` ensures that logs are logged in fine mode.

   > **Note:**
   >
   > When you run this command, the following error message might appear to indicate that the default cache directory is not valid:
   >
   > ```
   > *sys-package-mgr*: can't create package cache dir
   > ```
   >
   > You can change the cache directory by setting the environment variable `CONFIG_JVM_ARGS`. For example:
   >
   > ```
   > CONFIG_JVM_ARGS=-Dpython.cachedir=valid_directory
   > ```

Reconfiguring the Oracle BI Domain with the Reconfiguration Wizard

Navigate through the screens in the Reconfiguration Wizard to reconfigure your existing domain.

To reconfigure the domain:

1. On the Select Domain screen, specify the location of the domain you want to upgrade or click **Browse** to navigate and select the domain directory. Click **Next**.

2. On the Reconfiguration Setup Progress screen, view the progress of the setup process. When complete, click **Next**.

   During this process:

   - The reconfiguration templates for your installed products, including Fusion Middleware products, are automatically applied. This updates various domain configuration files such as `config.xml`, `config-groups.xml`, and `security.xml` (among others).

   - Schemas, scripts, and other such files that support your Fusion Middleware products are updated.

   - The domain upgrade is validated.

3. On the Domain Mode and JDK screen, select the JDK to use in the domain or click **Browse** to navigate to the JDK you want to use. The supported JDK version for 12*c* (12.2.1.3.0) is 1.8.0_131 and later. Click **Next**.

   > ✎ **Note:**
   >
   > You cannot change the **Domain Mode** at this stage.

   For a list of JDKs that are supported for a specific platform, see Oracle Fusion Middleware Supported System Configurations.

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

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

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

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

   - Convert to GridLink

   - Convert to RAC multi data source

   - Don't convert

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

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

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

```
Missing Driver

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

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

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

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

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

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

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

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

> **Note:**
>
> For any existing 11*g* datasource, the reconfiguration will preserve the existing values. For new datasources where the schema was created for 12*c* by the RCU, the default connection data will be retrieved from the _STB schema. If no connection data for a given schema is found in the _STB schema, then the default connection data is used.

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

> **Note:**
>
> If you are upgrading from 11*g*, and your database has _OPSS or _IAU 11*g* database schemas, you must manually enter database connection details for those schemas. These schemas were not required in 11*g* and had to be created manually. Users could assign any name to these schemas, therefore the Reconfiguration Wizard does not recognize them. When providing connection information for _IAU, use the IAU_APPEND user information.

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

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

8. On the Advanced Configuration screen, you can select all categories for which you want to perform advanced configuration. For each category you select, the appropriate configuration screen is displayed to allow you to perform advanced configuration.

   > **Note:**
   >
   > The categories that are listed on the Advanced Configuration screen depend on the resources defined in the templates you selected for the domain.

   For this upgrade, select none of the options and click **Next**.

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

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

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

   > **Note:**
   >
   > The location of the domain does not change when you reconfigure it.

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

    During this process:

    • Domain information is extracted, saved, and updated.

    • Schemas, scripts, and other such files that support your Fusion Middleware products are updated.

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

11. The End of Configuration screen indicates whether the reconfiguration process completed successfully or failed. It also displays the location of the domain that

ORACLE

was reconfigured as well as the Administration Server URL (including the listen port). If the reconfiguration is successful, it displays **Oracle WebLogic Server Reconfiguration Succeeded**.

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

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

Restoring the mapViewerConfig.xml File

The mapViewerConfig.xml file is overwritten when you run the Reconfiguration Wizard on your domain. You can know that the mapViewerConfig.xml file is overwritten by checking its file size. You must restore the original file that you backed up before upgrading your domain with the Upgrade Assistant.

Restoring the mapViewerConfig.xml File on UNIX Operating Systems

To restore the original mapViewerConfig.xml file:

1. Change to the following directory:

   EXISTING_DOMAIN_HOME/config/fmwconfig/mapviewer/conf

2. Enter the following command to list the files within this directory and check the file size of the mapViewerConfig.xml file:

   ```
   ls -ltr
   ```

   Following is a sample output:

   ```
   -rw-r----- 1 mwport svrtech 2xxx2 May 5 19:07 EXISTING_DOMAIN_HOME/config/
   fmwconfig/mapviewer/conf/mapViewerConfig.xml
   ```

   > **Note:**
   >
   > The change in the file size implies that the mapViewerConfig.xml file is overwritten by the reconfiguration template.

3. Restore the original file by renaming the mapViewerConfig_original.xml to mapViewerConfig.xml.

4. Enter the following command to list the files again and check the file size of the mapViewerConfig.xml file:

   ```
   ls -ltr
   ```

   The file size of the mapViewerConfig.xml file should now match with the file size you obtained while backing up the mapViewerConfig.xml file.

   In this example, **2xxx7**.

Restoring the mapViewerConfig.xml File on Windows Operating Systems

To restore the original mapViewerConfig.xml file:

1. Change to the following directory:

   EXISTING_DOMAIN_HOME\config\fmwconfig\mapviewer\conf

2. Enter the following command to list the files within this directory and check the file size of the mapViewerConfig.xml file:

   ```
   dir
   ```

   > **Note:**
   >
   > The change in the file size implies that the mapViewerConfig.xml file is overwritten by the reconfiguration template.

3. Restore the original file by renaming the mapViewerConfig_original.xml to mapViewerConfig.xml.

4. Enter the following command to list the files again and check the file size of the mapViewerConfig.xml file:

   ```
   dir
   ```

   The file size of the mapViewerConfig.xml file should now match with the file size you obtained while backing up the mapViewerConfig.xml file.

Upgrading Domain Component Configurations

After reconfiguring the domain, use the Upgrade Assistant to upgrade the domain *component* configurations inside the domain to match the updated domain configuration.

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

- Upgrading Domain Components Using the Upgrade Assistant
  Navigate through the screens in the Upgrade Assistant to upgrade component configurations in the WebLogic domain.

Starting the Upgrade Assistant

Run the Upgrade Assistant to upgrade product schemas, domain component configurations, or standalone system components to 12*c* (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 7-5   Upgrade Assistant Command-Line Parameters

Parameter	Required or Optional	Description
`-readiness`	Required for readiness checks **Note**: Readiness checks cannot be performed on standalone installations (those not managed by the WebLogic Server).	Performs the upgrade readiness check without performing an actual upgrade. Schemas and configurations are checked. Do not use this parameter if you have specified the `-examine` parameter.
`-threads`	Optional	Identifies the number of threads available for concurrent schema upgrades or readiness checks of the schemas. The value must be a positive integer in the range 1 to 8. The default is 4.

Table 7-5   (Cont.) Upgrade Assistant Command-Line Parameters

Parameter	Required or Optional	Description
`-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 *silent mode* (without displaying Upgrade Assistant screens).
`-examine`	Optional	Performs the examine phase but does not perform an actual upgrade. Do not specify this parameter if you have specified the `-readiness` parameter.
`-logLevel` *attribute*	Optional	Sets the logging level, specifying one of the following attributes: • `TRACE` • `NOTIFICATION` • `WARNING` • `ERROR` • `INCIDENT_ERROR` The default logging level is `NOTIFICATION`. Consider setting the `-logLevel TRACE` attribute to so that more information is logged. This is useful when troubleshooting a failed upgrade. The Upgrade Assistant's log files can become very large if `-logLevel TRACE` is used.

Table 7-5   (Cont.) Upgrade Assistant Command-Line Parameters

Parameter	Required or Optional	Description
`-logDir` *location*	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)
		NEW_ORACLE_HOME`/oracle_common/upgrade/logs` *NEW_ORACLE_HOME*`/oracle_common/upgrade/temp`
		(Windows)
		NEW_ORACLE_HOME`\oracle_common\upgrade\logs` *NEW_ORACLE_HOME*`\oracle_common\upgrade\temp`
`-help`	Optional	Displays all of the command-line options.

Upgrading Domain Components Using the Upgrade Assistant

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

After running the Reconfiguration Wizard to reconfigure the WebLogic domain to 12*c* (12.2.1.3.0), you must run the Upgrade Assistant to upgrade the domain *component* configurations to match the updated domain configuration.

To upgrade the 12.2.1.3.0 domain:

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

   > ✎ **Note:**
   >
   > For more information about any Upgrade Assistant screen, click **Help** on the screen.

2. On the next screen:

   • Select **All Configurations Used By a Domain**. The screen name changes to WebLogic Components.

   • In the **Domain Directory** field, enter the WebLogic domain directory path.

Click **Next**.

3. On the Component List screen, verify that the list includes all the components for which you want to upgrade configurations and click **Next**.

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

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

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

5. On the MapViewer Configuration screen, select **12c** and click **Next**.

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

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

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

7. On the Upgrade Summary screen, review the summary of the options you have selected for component configuration upgrade.

   The response file collects and stores all the information that you have entered, and enables you to perform a silent upgrade at a later time. The silent upgrade performs exactly the same function that the Upgrade Assistant performs, but you do not have to manually enter the data again. If you want to save these options to a response file, click **Save Response File** and provide the location and name of the response file.

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

8. On the Upgrade Progress screen, monitor the status of the upgrade.

> ⚠️ **Caution:**
>
> Allow the Upgrade Assistant enough time to perform the upgrade. Do not cancel the upgrade operation unless absolutely necessary. Doing so may result in an unstable environment.

If any components are not upgraded successfully, refer to the Upgrade Assistant log files for more information.

> ✎ **Note:**
>
> The progress bar on this screen displays the progress of the current upgrade procedure. It does not indicate the time remaining for the upgrade.

Click **Next**.

9. If the upgrade is successful: On the Upgrade Success screen, click **Close** to complete the upgrade and close the wizard. The Post-Upgrade Actions window describes the manual tasks you must perform to make components functional in the new installation. This window appears only if a component has post-upgrade steps.

   If the upgrade fails: On the Upgrade Failure screen, click **View Log** to view and troubleshoot the errors. The logs are available at `NEW_ORACLE_HOME/oracle_common/upgrade/logs`.

> ✎ **Note:**
>
> If the upgrade fails you must restore your pre-upgrade environment from backup, fix the issues, then restart the Upgrade Assistant.

Verifying the Domain-Specific-Component Configurations Upgrade

To verify that the domain-specific-component configurations upgrade was successful, sign in to the Administration console and the Oracle Enterprise Manager Fusion Middleware Control and verify that the version numbers for each component is 12.2.1.3.0.

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

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

> **Note:**
>
> After upgrade, make sure you run the administration tools from the new 12*c* Oracle home directory and not from the previous Oracle home directory.
>
> During the upgrade process, some OWSM documents, including policy sets and predefined documents such as policies and assertion templates, may need to be upgraded. If a policy set or a predefined document is upgraded, its version number is incremented by 1.
>
> If you created the FMW user to run the Upgrade Assistant, ensure that you delete the account after verifying your upgrade was successful.

Starting Servers and Processes

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

> **Note:**
>
> The procedure in this section describe how to start servers and processes using the WLST command-line utility or a script. You can also use the Fusion Middleware Control and the Oracle WebLogic Server Administration Console. See Starting and Stopping Administration and Managed Servers and Node Manager

To start your Fusion Middleware environment:

1. Change to the following directory:

   (UNIX) `12c_DOMAIN_HOME/bitools/bin`

   (Windows) `12c_DOMAIN_HOME\bitools\bin`

2. To start the Oracle BI instance and servers, enter the following command:

   (UNIX) `start.sh`

   (Windows) `start.cmd`

> **Note:**
>
> When prompted to enter the password, specify the Node Manager password that you entered while configuring the Oracle BI domain.

Validating the Oracle BI Deployments

The Oracle BI Baseline Validation Tool enables you to identify differences during life cycle operations, such as migrating from the Oracle BI 11*g* release to the 12*c* release. After you complete the migration procedure, you can use this tool to compare the two

deployments and verify whether the results from the 11g environment are the same as the results from the 12c environment.

You can download the Oracle BI Validation Tool from Oracle BI Baseline Validation Tool Downloads.

For more information about using the Oracle BI Validation Tool, see Comparing Oracle Business Intelligence Deployments Using the Oracle Business Intelligence Baseline Validation Tool.

> **Note:**
>
> You can download the Oracle BI Validation Tool along with the other Oracle Business Intelligence download on the Oracle Technology Network. See the document that is included in the Oracle BI Validation Tool download for more information. For specific information on the distributions you want to download for each product, see Oracle Fusion Middleware Download, Installation, and Configuration Readme Files page.

8

Upgrading Scaled-Out Systems from a Previous 12*c* Release

This chapter contains the procedure to upgrade a multi-node, scaled-out Oracle Business Intelligence deployment to the latest 12*c* (12.2.1.3.0) release.

- About the Multi-Node Oracle Business Intelligence Upgrade Process
  Review the flowchart and roadmap for an overview to upgrade a multi-node, scaled-out Oracle Business Intelligence deployment from a previous 12*c* release.

- Checking the Status of the Servers in the Oracle BI Instance
  The following servers must be in the running state on both the nodes: Servers hosting Oracle BI instance (bi_server), OBIPS, OBIJH, OBICCS, OBICH, and OBIS.

- Installing the Product Distributions on Node1 and Node2
  Before beginning your upgrade, download Oracle Fusion Middleware Infrastructure 12.2.1.3.0 and Oracle Business Intelligence 12*c* (12.2.1.3.0) distributions on the target system and install them on both the nodes using Oracle Universal Installer.

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

- Stopping Servers and Processes
  Stop the Oracle BI instance and the system components on Node1.

- 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.

- Backing Up the mapViewerConfig.xml File
  The mapViewerConfig.xml file is overwritten by the reconfiguration templates when you run the Reconfiguration Wizard. Therefore, you must back up the mapViewerConfig.xml file before reconfiguring your existing domain.

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

- Restoring the mapViewerConfig.xml File
  The mapViewerConfig.xml file is overwritten when you run the Reconfiguration Wizard on your domain. You can know that the mapViewerConfig.xml file is overwritten by checking its file size. You must restore the original file that you backed up before upgrading your domain with the Upgrade Assistant.

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

- Verifying the Domain-Specific-Component Configurations Upgrade
  To verify that the domain-specific-component configurations upgrade was successful, sign in to the Administration console and the Oracle Enterprise Manager Fusion Middleware Control and verify that the version numbers for each component is 12.2.1.3.0.

- Packing the Upgraded Domain on Node1
  After upgrading the existing domain to 12*c* (12.2.1.3.0), pack the upgraded domain on Node1.

- Unpacking the Upgraded Domain on Node2
  You must unpack the domain that you packed on Node1 to Node2, which is the remote node.

- Synchronizing Connection Details to the Mid-Tier Database
  Synchronizing connection details to the mid-tier database ensures that the Oracle BI components can access the mid-tier database when connection details, including credentials, are changed.

- Starting the Node Manager on Node2
  After you have upgraded the Oracle BI domain on Node1, start the Node Manager on Node2.

- Starting Servers and Processes
  Start the Oracle BI instance and the system components on Node1.

- Checking the Status of the Servers in the Oracle BI Instance
  The following servers must be in the running state on both the nodes: Servers hosting Oracle BI instance (bi_server), OBIPS, OBIJH, OBICCS, OBICH, and OBIS.

- Validating the Oracle BI Deployments
  The Oracle BI Baseline Validation Tool enables you to identify differences during life cycle operations, such as migrating from the Oracle BI 11*g* release to the 12*c* release. After you complete the migration procedure, you can use this tool to compare the two deployments and verify whether the results from the 11*g* environment are the same as the results from the 12*c* environment.

About the Multi-Node Oracle Business Intelligence Upgrade Process

Review the flowchart and roadmap for an overview to upgrade a multi-node, scaled-out Oracle Business Intelligence deployment from a previous 12*c* release.

Figure 8-1   Upgrade Process Flowchart for a Multi-Node Oracle Business Intelligence Deployment from a Previous 12*c* Release
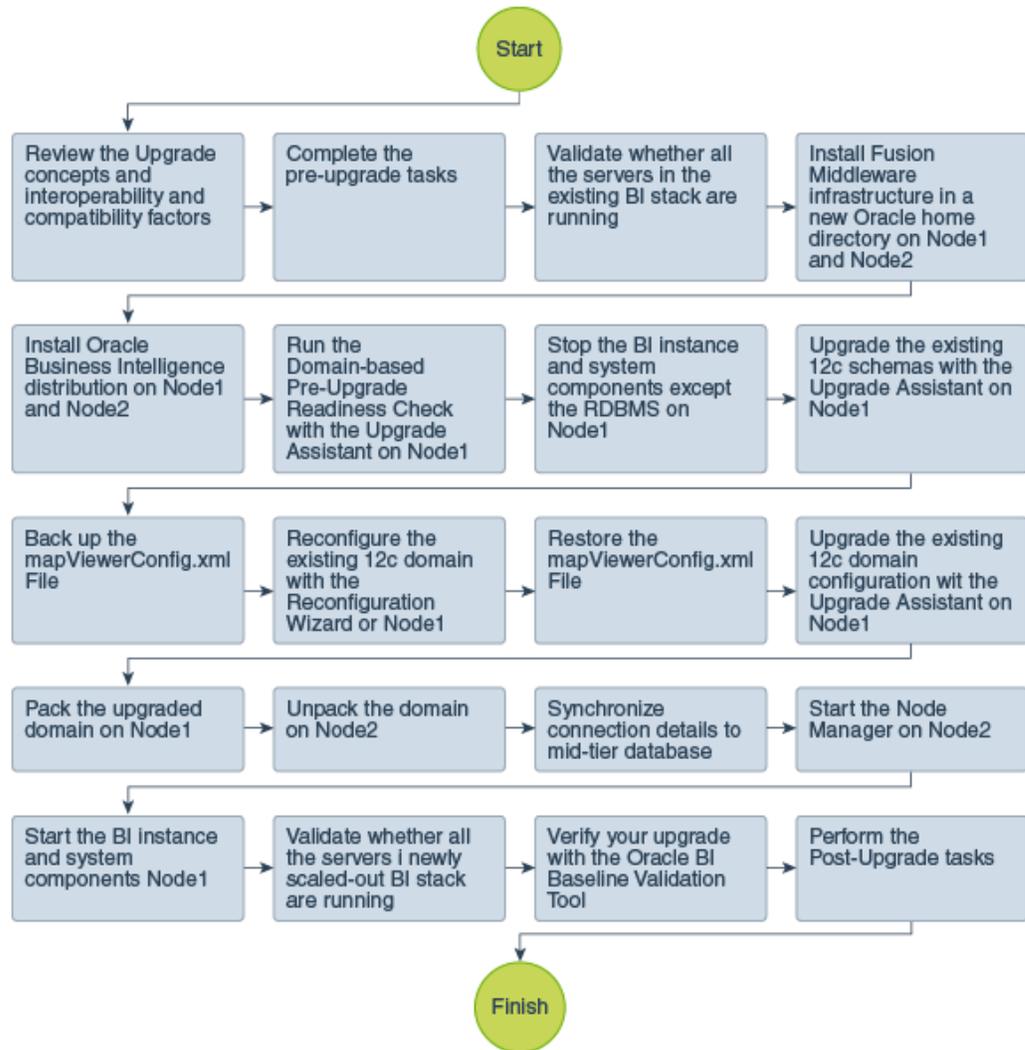
Table 8-1 lists the high-level steps that you need to perform to upgrade a multi-node Oracle BI deployment to Release 12.2.1.3.0.

Table 8-1   Tasks for Upgrading Multi-Node Oracle Business Intelligence Deployment from a Pervious 12*c* Release

Task	Description
Optional Learn about the interoperability and compatibility factors that could affect how you upgrade to BI 12.2.1.3.0.	It is important to understand how two or more Oracle Fusion Middleware products of the same version or different versions work together (interoperate) in a supported Oracle Fusion Middleware configuration. You can learn more about interoperability and compatibility in Oracle® Fusion Middleware Understanding Interoperability and Compatibility.

Table 8-1   (Cont.) Tasks for Upgrading Multi-Node Oracle Business Intelligence Deployment from a Pervious 12*c* Release

Task	Description
Required If you have not done so already, review the introductory topics in this guide and complete the required pre-upgrade tasks.	The pre-upgrade tasks include cloning your production environment, verifying system requirements and certifications, purging unused data, and creating non-SYSDBA user. For a complete list of pre-upgrade tasks, see Pre-Upgrade Requirements
Required Validate whether all the servers in the existing Oracle BI stack are running	The following servers must be in the running state on both the nodes: • Administration Server (available only on the primary node) • Servers hosting Oracle BI instance (bi_server) • OBIPS • OBIJH • OBICCS • OBICH • OBIS See Checking the Status of the Servers in the Oracle BI Instance.
Required Download and install the 12.2.1.3.0 Fusion Middleware Infrastructure and Oracle Business Intelligence distributions on Node1 and Node2.	The Infrastructure distribution combines the WebLogic Server and the Java Required Files (JRF) that are required to set up the foundation to install other Fusion Middleware products. As per the upgrade topology defined in this guide, you must install the Infrastructure in a new Oracle home. The Oracle Business Intelligence distribution packs Oracle BI EE, Publisher, and Essbase. You must install Oracle BI in the Oracle home that is created when you installed the 12.2.1.3.0 Infrastructure. To install the product distributions, follow the procedure described in Installing the Product Distributions.
Optional Run the Readiness Check on Node1.	Run the domain-based pre-upgrade readiness check on Node1 to identify potential issues before you upgrade. See Running a Pre-Upgrade Readiness Check.
Required Stop the Oracle BI instance and system components except the database on Node1.	Before starting the upgrade process, shut down the Administration Server, the Managed Servers, and your existing Oracle BI instance on Node1. However, keep the database (RDBMS) running. See Stopping Servers and Processes.
Required Upgrade the existing 12*c* schemas with the Upgrade Assistant on Node1.	The schemas you created during the 12.2.1.x installation are supported in 12.2.1.3.0. Therefore, you don't need to create the schemas again. You must upgrade all the schemas within your domain using the Upgrade Assistant. For the complete procedure, see Upgrading Product Schemas Using the Upgrade Assistant.

Table 8-1   (Cont.) Tasks for Upgrading Multi-Node Oracle Business Intelligence Deployment from a Pervious 12*c* Release

Task	Description
Required Backup the mapViewerConfig.xml File.	The mapViewerConfig.xml file is overwritten by the reconfiguration templates when you run the Reconfiguration Wizard. Therefore, you must back up the mapViewerConfig.xml file before reconfiguring your existing domain. See Backing Up the mapViewerConfig.xml File.
Required Reconfigure the existing 12*c* domain with the Reconfiguration Wizard on Node1.	When you run the Reconfiguration Wizard on your existing domain, it prepares your domain for upgrade by selecting and applying the recongifuration templates. It also tests the JDBC data sources and component schemas that are present within your domain. To reconfigure you domain, follow the procedure described in Reconfiguring the Oracle BI Domain with the Reconfiguration Wizard.
Required Restore the mapViewerConfig.xml File.	You must restore the original file that you backed up before upgrading your domain with the Upgrade Assistant. See Restoring the mapViewerConfig.xml File.
Required Upgrade the existing 12*c* domain configurations with the Upgrade Assistant on Node1.	After you have reconfigured your existing 12*c* domain, you must run the Upgrade Assistant to upgrade all configurations used by your domain. You can see all the components within your domain that will be upgraded on the Component List screen when you run the Upgrade Assistant. For the complete procedure, see Upgrading Domain Components Using the Upgrade Assistant.
Required Pack the upgraded domain on Node1.	See Packing the Upgraded Domain on Node1.
Required Unpack the domain on Node2.	See Unpacking the Upgraded Domain on Node2.
Required Synchronize connection details to the mid-tier database.	See Synchronizing Connection Details to the Mid-Tier Database.
Required Start the Node Manager on Node2.	See Starting the Node Manager on Node2.
Required Start the Oracle BI instance and system components Node1.	The upgrade process is complete. You can now restart the Administration Server, the Managed Servers, and your 12.2.1.3.0 BI instance. See Starting Servers and Processes.

Table 8-1   (Cont.) Tasks for Upgrading Multi-Node Oracle Business Intelligence Deployment from a Pervious 12*c* Release

Task	Description
Required Validate whether all the servers in the newly scaled-out Oracle BI stack are running.	The following servers must be in the running state on both the nodes: • Administration Server (available only on the primary node) • Servers hosting Oracle BI instance (bi_server) • OBIPS • OBIJH • OBICCS • OBICH • OBIS See Checking the Status of the Servers in the Oracle BI Instance.
Required Verify your upgrade.	It is important to compare your existing 12*c* and the 12.2.1.3.0 environments and verify that the data and configuration settings are consistent in the newly upgraded environment. Oracle provides the BI Baseline Validation Tool that you can use to compare and verify the upgrade.
Required Perform the post-upgrade tasks.	For a list of post-upgrade tasks, see Post-Upgrade Tasks.

Checking the Status of the Servers in the Oracle BI Instance

The following servers must be in the running state on both the nodes: Servers hosting Oracle BI instance (bi_server), OBIPS, OBIJH, OBICCS, OBICH, and OBIS.

To check the status of the servers:

1. Change to the following directory:

   (UNIX) *EXISTING_DOMAIN_HOME*/bitools/bin

   (Windows) *EXISTING_DOMAIN_HOME*\bitools\bin

2. Run the status script by entering the following command:

   (UNIX) `status.sh`

   (Windows) `status.cmd`

   Following is a sample output:

   ```
   Name            Type            Machine                     Status
   ----            ----            -------                     ------
   AdminServer     Server          machine_name
   RUNNING
   bi_server1      Server          machine_name.example.com    RUNNING
   bi_server2      Server          machine_name.example.com    RUNNING
   obips1          OBIPS           machine_name.example.com    RUNNING
   obijh1          OBIJH           machine_name.example.com    RUNNING
   obiccs1         OBICCS          machine_name.example.com    RUNNING
   ```

```
obisch1        OBISCH        machine_name.example.com    RUNNING
obis1          OBIS          machine_name.example.com    RUNNING
obips2         OBIPS         machine_name.example.com    RUNNING
obijh2         OBIJH         machine_name.example.com    RUNNING
obiccs2        OBICCS        machine_name.example.com    RUNNING
obisch2        OBISCH        machine_name.example.com    RUNNING
obis2          OBIS          machine_name.example.com    RUNNING
```

Installing the Product Distributions on Node1 and Node2

Before beginning your upgrade, download Oracle Fusion Middleware Infrastructure 12.2.1.3.0 and Oracle Business Intelligence 12*c* (12.2.1.3.0) distributions on the target system and install them on both the nodes using Oracle Universal Installer.

> **✎ Note:**
>
> When Infrastructure is required for the upgrade, you must install the Oracle Fusion Middleware distribution first before you install other Fusion Middleware products.

To install the 12*c* (12.2.1.3.0) distributions:

1. Sign in to the target system.

2. Download the following from Oracle Technology Network or Oracle Software Delivery Cloud to your target system:

   - Oracle Fusion Middleware Infrastructure (`fmw_12.2.1.3.0_infrastructure_generic.jar`)

   - Oracle Business Intelligence (UNIX: `fmw_12.2.1.3.0_bi_platform_linux64.bin`), (Windows: `setup_fmw_12.2.1.3.0_bi_platform_win64.exe`)

3. Change to the directory where you downloaded the 12*c* (12.2.1.3.0) product distribution.

4. Start the installation program for Oracle Fusion Middleware Infrastructure:

   - (UNIX) *JDK_HOME*/bin/java -jar fmw_12.2.1.3.0_infrastructure_generic.jar

   - (Windows) *JDK_HOME*\bin\java -jar fmw_12.2.1.3.0_infrastructure_generic.jar

5. On UNIX operating systems, the Installation Inventory Setup screen appears if this is the first time you are installing an Oracle product on this host.

   Specify the location where you want to create your central inventory. Make sure that the operating system group name selected on this screen has write permissions to the central inventory location, and click **Next**.

> **✏ Note:**
>
> The Installation Inventory Setup screen does not appear on Windows operating systems.

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

7. On the Auto Updates screen, select an option:

   - **Skip Auto Updates**: If you do not want your system to check for software updates at this time.

   - **Select patches from directory**: To navigate to a local directory if you downloaded patch files.

   - **Search My Oracle Support for Updates**: To automatically download software updates if you have a My Oracle Support account. You must enter Oracle Support credentials then click **Search**. To configure a proxy server for the installer to access My Oracle Support, click **Proxy Settings**. Click **Test Connection** to test the connection.

   Click **Next**.

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

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

9. On the Installation Type screen, select the following:

   - For Infrastructure, select **Fusion Middleware Infrastructure**

   - For Oracle Business Intelligence, select **BI Platform Distribution with Samples**

   Click **Next**.

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

    To view the list of tasks that are verified, select **View Successful Tasks**. To view log details, select **View Log**. If any prerequisite check fails, then an error message appears at the bottom of the screen. Fix the error and click **Rerun** to try again. To ignore the error or the warning message and continue with the installation, click **Skip** (not recommended).

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

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

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

13. The Installation Complete screen displays the Installation Location and the Feature Sets that are installed. Review this information and click **Finish** to close the installer.

14. After you have installed Oracle Fusion Middleware Infrastructure, enter the following command to start the installer for your product distribution and repeat the steps above to navigate through the installer screens:

    (UNIX) `./fmw_12.2.1.3.0_bi_platform_linux64.bin`

    (Windows) `setup_fmw_12.2.1.3.0_bi_platform_win64.exe`

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 8-2   Upgrade Assistant Command-Line Parameters

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

ORACLE

Table 8-2   (Cont.) Upgrade Assistant Command-Line Parameters

Parameter	Required or Optional	Description
`-logDir location`	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)
		`NEW_ORACLE_HOME/oracle_common/upgrade/logs` `NEW_ORACLE_HOME/oracle_common/upgrade/temp`
		(Windows)
		`NEW_ORACLE_HOME\oracle_common\upgrade\logs` `NEW_ORACLE_HOME\oracle_common\upgrade\temp`
`-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 8-3   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 *ORACLE_HOME*/ oracle_common/upgrade/ logs	The directory location of the generated log file.	No action required.
Readiness report location *ORACLE_HOME*/ oracle_common/upgrade/ logs	The directory location of the generated readiness report.	No action required.

Table 8-3   (Cont.) Readiness Report Elements

Report Information	Description	Required Action
Names of components that were checked	The names and versions of the components included in the check and status.	If your domain includes components that cannot be upgraded to this release, such as SOA Core Extension, do not attempt an upgrade.
Names of schemas that were checked	The names and current versions of the schemas included in the check and status.	Review the version numbers of your schemas. If your domain includes schemas that cannot be upgraded to this release, do not attempt an upgrade.
Individual Object Test Status: FAIL	The readiness check test detected an issue with a specific object.	Do not upgrade until all failed issues have been resolved.
Individual Object Test Status: PASS	The readiness check test detected no issues for the specific object.	If your readiness check report shows only the PASS status, you can upgrade your environment. Note, however, that the Readiness Check cannot detect issues with externals such as hardware or connectivity during an upgrade. You should always monitor the progress of your upgrade.
Completed Readiness Check of <Object> Status: FAILURE	The readiness check detected one or more errors that must be resolved for a particular object such as a schema, an index, or datatype.	Do not upgrade until all failed issues have been resolved.
Completed Readiness Check of <Object> Status: SUCCESS	The readiness check test detected no issues.	No action required.

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

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

This readiness check report was created on Tue March 30 11:15:52 EDT
2019
Log file is located at: ORACLE_HOME/oracle_common/upgrade/logs/
ua2016-05-30-11-14-06AM.log
Readiness Check Report File: ORACLE_HOME/oracle_common/upgrade/logs/
readiness2016-05-30-11-15-52AM.txt

Starting readiness check of components.

Oracle Metadata Services
   Starting readiness check of Oracle Metadata Services.
     Schema User Name: DEV11_MDS
     Database Type: Oracle Database
     Database Connect String: machinename@yourcompany.com
     VERSION Schema DEV11_MDS is currently at version 12.2.1.4.0.
Readiness checks will now be performed.
   Starting schema test:  TEST_REQUIRED_TABLES  Test that the schema
```

contains all the required tables
    Completed schema test: TEST_REQUIRED_TABLES --> Test that the schema
contains all the required tables +++ PASS
    Starting schema test: TEST_REQUIRED_PROCEDURES Test that the
schema contains all the required stored procedures
      **EXCEPTION   Schema is missing a required procedure:
GETREPOSITORYFEATURES**
    **Completed schema test: TEST_REQUIRED_PROCEDURES --> Test that the
schema contains all the required stored procedures +++ FAIL**
    Starting schema test: TEST_REQUIRED_VIEWS Test that the schema
contains all the required database views
    Completed schema test: TEST_REQUIRED_VIEWS --> Test that the schema
contains all the required database views +++ PASS
    Starting index test for table MDS_ATTRIBUTES: TEST_REQUIRED_INDEXES
--> Test that the table contains all the required indexes
    Completed index test for table MDS_ATTRIBUTES: TEST_REQUIRED_INDEXES
--> Test that the table contains all the required indexes +++ PASS
    Starting index test for table MDS_COMPONENTS: TEST_REQUIRED_INDEXES
--> Test that the table contains all the required indexes
    Completed index test for table MDS_TXN_LOCKS: TEST_REQUIRED_INDEXES
--> Test that the table contains all the required indexes +++ PASS
    Starting schema test: TEST_REQUIRED_TRIGGERS Test that the schema
has all the required triggers
    Completed schema test: TEST_REQUIRED_TRIGGERS --> Test that the
schema has all the required triggers +++ PASS
    Starting schema test: TEST_MISSING_COLUMNS Test that tables and
views are not missing any required columns
    Completed schema test: TEST_MISSING_COLUMNS --> Test that tables and
views are not missing any required columns +++ PASS
    Starting schema test: TEST_UNEXPECTED_TABLES Test that the schema
does not contain any unexpected tables
    Completed schema test: TEST_UNEXPECTED_TABLES --> Test that the
schema does not contain any unexpected tables +++ PASS
    Starting schema test: TEST_UNEXPECTED_PROCEDURES Test that the
schema does not contain any unexpected stored procedures
    Completed schema test: TEST_UNEXPECTED_PROCEDURES --> Test that the
schema does not contain any unexpected stored procedures +++ PASS
    Starting schema test: TEST_UNEXPECTED_VIEWS Test that the schema
does not contain any unexpected views
    Completed schema test: TEST_UNEXPECTED_VIEWS --> Test that the
schema does not contain any unexpected views +++ PASS
    Starting index test for table MDS_ATTRIBUTES:
TEST_UNEXPECTED_INDEXES --> Test that the table does not contain any
unexpected indexes
    Completed index test for table MDS_ATTRIBUTES:
TEST_UNEXPECTED_INDEXES --> Test that the table does not contain any
unexpected indexes +++ PASS
    Completed index test for table MDS_LABELS: TEST_UNEXPECTED_INDEXES
--> Test that the table does not contain any unexpected indexes +++ PASS
    Starting index test for table MDS_LARGE_ATTRIBUTES:
TEST_UNEXPECTED_INDEXES --> Test that the table does not contain any
unexpected indexes
    Starting schema test: TEST_UNEXPECTED_TRIGGERS Test that the
schema does not contain any unexpected triggers
    Completed schema test: TEST_UNEXPECTED_TRIGGERS --> Test that the

```
schema does not contain any unexpected triggers +++ PASS
    Starting schema test:  TEST_UNEXPECTED_COLUMNS  Test that tables and
views do not contain any unexpected columns
    Completed schema test: TEST_UNEXPECTED_COLUMNS --> Test that tables
and views do not contain any unexpected columns +++ PASS
    Starting datatype test for table MDS_ATTRIBUTES:
TEST_COLUMN_DATATYPES_V2 --> Test that all table columns have the
proper datatypes
    Completed datatype test for table MDS_ATTRIBUTES:
TEST_COLUMN_DATATYPES_V2 --> Test that all table columns have the
proper datatypes +++ PASS
    Starting datatype test for table MDS_COMPONENTS:
TEST_COLUMN_DATATYPES_V2 --> Test that all table columns have the
proper datatypes
    Starting permissions test:  TEST_DBA_TABLE_GRANTS  Test that DBA
user has privilege to view all user tables
    Completed permissions test: TEST_DBA_TABLE_GRANTS --> Test that DBA
user has privilege to view all user tables +++ PASS
    Starting schema test:  TEST_ENOUGH_TABLESPACE  Test that the schema
tablespaces automatically extend if full
    Completed schema test: TEST_ENOUGH_TABLESPACE --> Test that the
schema tablespaces automatically extend if full +++ PASS
    Starting schema test:  TEST_USER_TABLESPACE_QUOTA  Test that
tablespace quota for this user is sufficient to perform the upgrade
    Completed schema test: TEST_USER_TABLESPACE_QUOTA --> Test that
tablespace quota for this user is sufficient to perform the upgrade
+++ PASS
    Starting schema test:  TEST_ONLINE_TABLESPACE  Test that schema
tablespaces are online
    Completed schema test: TEST_ONLINE_TABLESPACE --> Test that schema
tablespaces are online +++ PASS
    Starting schema test:  TEST_DATABASE_VERSION  Test that the database
server version number is supported for upgrade
     INFO   Database product version: Oracle Database 12c Enterprise
Edition Release 12.2.1.4.0 - 64bit Production
With the Partitioning, OLAP, Data Mining and Real Application Testing
options
    Completed schema test: TEST_DATABASE_VERSION --> Test that the
database server version number is supported for upgrade +++ PASS
    Finished readiness check of Oracle Metadata Services with status:
FAILURE.
```

If you are running the 12.1.3.0 version of Oracle Fusion Middleware IAU Schemas, and those schemas were upgraded from 11*g* (11.1.1.7 and later) or 12*c* (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 12*c* (12.2.1.3.0) by using the RCU.

Stopping Servers and Processes

Stop the Oracle BI instance and the system components on Node1.

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

To stop your Oracle BI instance:

1. Change to the following directory:

   (UNIX) *NEW_DOMAIN_HOME*/bitools/bin

   (Windows) *NEW_DOMAIN_HOME*\bitools\bin

2. To stop the Oracle BI instance and servers, enter the following command:

   (UNIX) stop.sh -r

   (Windows) stop.cmd

   > **Note:**
   >
   > When prompted to enter the password, specify the Node Manager
   > password that you entered while configuring the Oracle BI domain.

Upgrading Product Schemas

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

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

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

- Upgrading Product Schemas Using the Upgrade Assistant
  Navigate through the screens in the Upgrade Assistant to upgrade the product schemas.

- Verifying the Schema Upgrade
  After completing all the upgrade steps, verify that the upgrade was successful by checking that the schema version in `schema_version_registry` has been properly updated.

Starting the Upgrade Assistant

Run the Upgrade Assistant to upgrade product schemas, domain component configurations, or standalone system components to 12*c* (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 8-4   Upgrade Assistant Command-Line Parameters

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

Table 8-4   (Cont.) Upgrade Assistant Command-Line Parameters

Parameter	Required or Optional	Description
`-logLevel attribute`	Optional	Sets the logging level, specifying one of the following attributes: • `TRACE` • `NOTIFICATION` • `WARNING` • `ERROR` • `INCIDENT_ERROR` The default logging level is `NOTIFICATION`. Consider setting the `-logLevel TRACE` attribute to so that more information is logged. This is useful when troubleshooting a failed upgrade. The Upgrade Assistant's log files can become very large if `-logLevel TRACE` is used.

`-logDir location`	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)
		
		`NEW_ORACLE_HOME/oracle_common/upgrade/logs`
		`NEW_ORACLE_HOME/oracle_common/upgrade/temp`
		
		(Windows)
		
		`NEW_ORACLE_HOME\oracle_common\upgrade\logs`
		`NEW_ORACLE_HOME\oracle_common\upgrade\temp`

`-help`	Optional	Displays all of the command-line options.

Upgrading Product Schemas Using the Upgrade Assistant

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

If you are using DB2 in your existing BI instance, then perform the following before starting the upgrade process:

1. Sign in to the WebLogic Console.

2. Go to **Datasources** under **Services** and select **wlsservices_datasource**.

3. Change the user for data source 'wlsservices_datasource' from
   *<PREFIX>*_WLS_RUNTIME to *<PREFIX>*_WLS.

   For example: Change the user from L1_WLS_RUNTIME to L1_WLS.

You can now proceed with the upgrade process.

To upgrade product schemas with the Upgrade Assistant:

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

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

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

   • **All Schemas Used by a Domain** to allow the Upgrade Assistant to discover
     and select all components that have a schema available to upgrade in the
     domain specified in the **Domain Directory** field. This is also known as a
     domain assisted schema upgrade. Additionally, the Upgrade Assistant pre-
     populates connection information on the schema input screens.

     > **Note:**
     >
     > Oracle recommends that you select **All Schemas Used by a
     > Domain** for most upgrades to ensure all of the required schemas
     > are included in the upgrade.

   Click **Next**.

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

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

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

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

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

5. On the BIPLATFORM Schema [BIEE_PLATFORM] screen, the system automatically populates data in the other fields from the existing 12*c* schema.

    Enter the database administrator account: DBA username and password for the database containing the BIEE_PLATFORM schema and click **Connect**.

    When you see the message "Connection to database successfully completed", click **Next**.

6. On the IAU Schema [BIEE_IAU] screen, the system automatically populates data in the other fields from the existing 12*c* schema.

    Enter the database administrator account: DBA username and password for the database containing the BIEE_IAU schema and click **Connect**.

    When you see the message "Connection to database successfully completed", click **Next**.

7. On the OPSS Schema [BIEE_OPSS] screen, the system automatically populates data in the other fields from the existing 12*c* schema.

    Enter the database administrator account: DBA username and password for the database containing the BIEE_OPSS schema and click **Connect**.

    When you see the message "Connection to database successfully completed", click **Next**.

8. On the MDS Schema [BIEE_MDS] screen, the system automatically populates data in the other fields from the existing 12*c* schema.

    Enter the database administrator account: DBA username and password for the database containing the BIEE_MDS schema and click **Connect**.

    When you see the message "Connection to database successfully completed", click **Next**.

9. On the STB Schema [BIEE_STB] screen, the system automatically populates data in the other fields from the existing 12*c* schema.

    Enter the database administrator account: DBA username and password for the database containing the BIEE_STB schema and click **Connect**.

    When you see the message "Connection to database successfully completed", click **Next**.

10. On the WLS Schema [BIEE_WLS_RUNTIME] screen, the system automatically populates data in the other fields from the existing 12*c* schema.

    Enter the database administrator account: DBA username and password for the database containing the BIEE_WLS_RUNTIME schema and click **Connect**.

    When you see the message "Connection to database successfully completed", click **Next**.

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

ORACLE

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

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

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

    Verify that the correct Source and Target Versions are listed for each schema you intend to upgrade.

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

    Click **Next** .

13. On the Upgrade Progress screen, monitor the status of the upgrade.

    > **Caution:**
    >
    > Allow the Upgrade Assistant enough time to perform the upgrade. Do not cancel the upgrade operation unless absolutely necessary. Doing so may result in an unstable environment.

    If any schemas are not upgraded successfully, refer to the Upgrade Assistant log files for more information.

    > **Note:**
    >
    > The progress bar on this screen displays the progress of the current upgrade procedure. It does not indicate the time remaining for the upgrade.

    Click **Next**.

14. If the upgrade is successful: On the Upgrade Success screen, click **Close** to complete the upgrade and close the wizard.

    If the upgrade fails: On the Upgrade Failure screen, click **View Log** to view and troubleshoot the errors. The logs are available at *NEW_ORACLE_HOME/* `oracle_common/upgrade/logs`.

    > **Note:**
    >
    > If the upgrade fails, you must restore your pre-upgrade environment from backup, fix the issues, then restart the Upgrade Assistant.

Verifying the Schema Upgrade

After completing all the upgrade steps, verify that the upgrade was successful by checking that the schema version in `schema_version_registry` has been properly updated.

If you are using an Oracle database, connect to the database as a user having Oracle DBA privileges, and run the following from SQL*Plus to get the current version numbers:

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

In the query result:

- Check that the number in the `VERSION` column matches the latest version number for that schema. For example, verify that the schema version number is 12.2.1.3.0.

    > **Note:**
    >
    > However, that not all schema versions will be updated. Some schemas do not require an upgrade to this release and will retain their pre-upgrade version number.

- The `STATUS` field will be either `UPGRADING` or `UPGRADED` during the schema patching operation, and will become `VALID` when the operation is completed.

- If the status appears as `INVALID`, the schema update failed. You should examine the logs files to determine the reason for the failure.

- Synonym objects owned by `IAU_APPEND` and `IAU_VIEWER` will appear as `INVALID`, but that does not indicate a failure.

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

Backing Up the mapViewerConfig.xml File

The mapViewerConfig.xml file is overwritten by the reconfiguration templates when you run the Reconfiguration Wizard. Therefore, you must back up the mapViewerConfig.xml file before reconfiguring your existing domain.

Backing Up the mapViewerConfig.xml File on UNIX Operating Systems

To back up the mapViewerConfig.xml file on UNIX systems:

1. The mapViewerConfig.xml file is present in the following location on your system. Use the list command: `ls -ltr` to check its file size.

   EXISTING_DOMAIN_HOME`/config/fmwconfig/mapviewer/conf/`
   `mapViewerConfig.xml`

   `-rw-r----- 1 mwport svrtech `**`2xxx7`**` May  5 18:45 `*EXISTING_DOMAIN_HOME*`/config/`
   `fmwconfig/mapviewer/conf/mapViewerConfig.xml`

2. Copy the mapViewerConfig.xml file as shown in the following example:

   `cp `*EXISTING_DOMAIN_HOME*`/config/fmwconfig/mapviewer/conf/`
   `mapViewerConfig.xml `*EXISTING_DOMAIN_HOME*`/config/fmwconfig/`
   `mapviewer/conf/mapViewerConfig_orig.xml`

   A copy of the mapViewerConfig.xml file is created with the filename 'mapViewerConfig_orig.xml'.

Backing Up the mapViewerConfig.xml File on Windows Operating Systems

To back up the mapViewerConfig.xml file on Windows systems:

1. Open a command prompt window and change to the following directory:

   EXISTING_DOMAIN_HOME`\config\fmwconfig\mapviewer\conf\mapViewe`
   `rConfig.xml`

2. Enter the following command to check the file size of the mapViewerConfig.xml file:

   `dir`

3. Copy the mapViewerConfig.xml file as shown in the following example:

   copy "mapViewerConfig.xml" "mapViewerConfig_original.xml"

   A copy of the mapViewerConfig.xml file is created with the filename 'mapViewerConfig_original.xml'.

About Reconfiguring the Domain

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

When you reconfigure a WebLogic Server domain, the following items are automatically updated, depending on the applications in the domain:

- WebLogic Server core infrastructure
- Domain version

> ✎ **Note:**
>
> Before you begin the domain reconfiguration, note the following limitations:
>
> - The Reconfiguration Wizard does not update any of your own applications that are included in the domain.
>
> - Transforming a non-dynamic cluster domain to a dynamic cluster domain during the upgrade process is not supported.
>
>   The dynamic cluster feature is available when running the Reconfiguration Wizard, but Oracle only supports upgrading a non-dynamic cluster upgrade and then adding dynamic clusters. You cannot add dynamic cluster during the upgrade process.
>
> - If the installation that you're upgrading does not use Oracle Access Management (OAM), then you must edit two files to prevent the Reconfiguration Wizard from attempting to update the nonexistent OAM Infrastructure schema, which causes the upgrade to fail.
>
>   Comment out the lines in your `$DOMAIN/init-info/domain-info.xml` that are similar to this example:
>
> ```
> <!--extention-template-ref name="Oracle Identity Navigator"
>   version="11.1.1.3.0"
>   location="/u01/app/
> oracle/product/fmw/iam111130/common/templates/applications/
> oracle.oinav_11.1.1.3.0_template.jar"
>   symbol=""/-->
>
> <!--install-comp-ref name="oracle.idm.oinav"
> version="11.1.1.3.0"
>
> symbol="oracle.idm.oinav_11.1.1.3.0_iam111130_ORACLE_HOME"
>   product_home="/u01/app/oracle/product/fmw/iam111130"/-->
> ```
>
>   and similarly comment out the lines in `$DOMAIN/config/config.xml` that are similar to this example:
>
> ```
> <!--app-deployment>
>   <name>oinav#11.1.1.3.0</name>
>   <target>AdminServer</target>
>   <module-type>ear</module-type>
>
>   <source-path>/u01/app/oracle/product/fmw/iam111130/oinav/
> modules/oinav.ear_11.1.1.3.0/oinav.ear</source-path>
>   <deployment-order>500</deployment-order>
>   <security-dd-model>DDOnly</security-dd-model>
>   <staging-mode>nostage</staging-mode>
> </app-deployment-->
> ```

Specifically, when you reconfigure a domain, the following occurs:

ORACLE

- The domain version number in the `config.xml` file for the domain is updated to the Administration Server's installed WebLogic Server version.

- Reconfiguration templates for all installed Oracle products are automatically selected and applied to the domain. These templates define any reconfiguration tasks that are required to make the WebLogic domain compatible with the current WebLogic Server version.

- Start scripts are updated.

  If you want to preserve your modified start scripts, be sure to back them up before starting the Reconfiguration Wizard.

> **Note:**
>
> When the domain reconfiguration process starts, you can't undo the changes that it makes. Before running the Reconfiguration Wizard, ensure that you have backed up the domain as covered in the pre-upgrade checklist. If an error or other interruption occurs while running the Reconfiguration Wizard, you must restore the domain by copying the files and directories from the backup location to the original domain directory. This is the only way to ensure that the domain has been returned to its original state before reconfiguration.

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

- Backing Up the Domain
- Starting the Reconfiguration Wizard
- Reconfiguring the Oracle BI Domain with the Reconfiguration Wizard
  Navigate through the screens in the Reconfiguration Wizard to reconfigure your existing domain.

Backing Up the Domain

Before running the Reconfiguration Wizard, create a backup copy of the domain directory.

To create a backup of the domain directory:

1. Copy the source domain to a separate location to preserve the contents.

   (Windows) `copy C:\domains\mydomain to C:\domains\mydomain_backup`.

   (UNIX) `cp mydomain /domains/mydomain_backup`

2. Before updating the domain on each remote Managed Server, create a backup copy of the domain directory on each remote machine.

3. Verify that the backed up versions of the domain are complete.

If domain reconfiguration fails for any reason, you must copy all files and directories from the backup directory into the original domain directory to ensure that the domain is returned entirely to its original state before reconfiguration.

ORACLE

Starting the Reconfiguration Wizard

> **Note:**
>
> Shut down the administration server and all collocated managed servers before starting the reconfiguration process. See Stopping Servers and Processes .

To start the Reconfiguration Wizard in graphical mode:

1. Sign in to the system on which the domain resides.

2. Open the command shell (on UNIX operating systems) or open a command prompt window (on Windows operating systems).

3. **Edition Based Database Users Only**: If your schemas are configured with EBR database, a default edition name must be manually supplied before you run the Reconfiguration Wizard.

   Run the following SQL command to set the default edition:
   ```
   ALTER DATABASE DEFAULT EDITION = edition_name;
   ```
   where `edition_name` is the child edition name.

4. Go to the `oracle_common/common/bin` directory:

   - (UNIX) `NEW_ORACLE_HOME/oracle_common/common/bin`

   - (Windows) `NEW_ORACLE_HOME\oracle_common\commom\bin`

5. Start the Reconfiguration Wizard with the following logging options:

   - (UNIX) `./reconfig.sh -log=log_file -log_priority=ALL`

   - (Windows) `reconfig.cmd -log=log_file -log_priority=ALL`

   where `log_file` is the absolute path of the log file you'd like to create for the domain reconfiguration session. This can be helpful if you need to troubleshoot the reconfiguration process.

   The parameter `-log_priority=ALL` ensures that logs are logged in fine mode.

   > **Note:**
   >
   > When you run this command, the following error message might appear to indicate that the default cache directory is not valid:
   >
   > ```
   > *sys-package-mgr*: can't create package cache dir
   > ```
   >
   > You can change the cache directory by setting the environment variable `CONFIG_JVM_ARGS`. For example:
   >
   > ```
   > CONFIG_JVM_ARGS=-Dpython.cachedir=valid_directory
   > ```

Reconfiguring the Oracle BI Domain with the Reconfiguration Wizard

Navigate through the screens in the Reconfiguration Wizard to reconfigure your existing domain.

To reconfigure the domain:

1. On the Select Domain screen, specify the location of the domain you want to upgrade or click **Browse** to navigate and select the domain directory. Click **Next**.

2. On the Reconfiguration Setup Progress screen, view the progress of the setup process. When complete, click **Next**.

   During this process:

   • The reconfiguration templates for your installed products, including Fusion Middleware products, are automatically applied. This updates various domain configuration files such as `config.xml`, `config-groups.xml`, and `security.xml` (among others).

   • Schemas, scripts, and other such files that support your Fusion Middleware products are updated.

   • The domain upgrade is validated.

3. On the Domain Mode and JDK screen, select the JDK to use in the domain or click **Browse** to navigate to the JDK you want to use. The supported JDK version for 12*c* (12.2.1.3.0) is 1.8.0_131 and later. Click **Next**.

   > ✎ **Note:**
   >
   > You cannot change the **Domain Mode** at this stage.

   For a list of JDKs that are supported for a specific platform, see Oracle Fusion Middleware Supported System Configurations.

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

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

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

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

   • Convert to GridLink

   • Convert to RAC multi data source

   • Don't convert

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

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

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

```
Missing Driver
```

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

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

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

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

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

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

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

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

> **Note:**
>
> For any existing 11*g* datasource, the reconfiguration will preserve the existing values. For new datasources where the schema was created for 12*c* by the RCU, the default connection data will be retrieved from the _STB schema. If no connection data for a given schema is found in the _STB schema, then the default connection data is used.

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

> **Note:**
>
> If you are upgrading from 11*g*, and your database has _OPSS or
> _IAU 11*g* database schemas, you must manually enter database
> connection details for those schemas. These schemas were not required
> in 11*g* and had to be created manually. Users could assign any
> name to these schemas, therefore the Reconfiguration Wizard does not
> recognize them. When providing connection information for _IAU, use
> the IAU_APPEND user information.

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

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

8. On the Advanced Configuration screen, you can select all categories for which
    you want to perform advanced configuration. For each category you select, the
    appropriate configuration screen is displayed to allow you to perform advanced
    configuration.

    > **Note:**
    >
    > The categories that are listed on the Advanced Configuration screen
    > depend on the resources defined in the templates you selected for the
    > domain.

    For this upgrade, select none of the options and click **Next**.

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

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

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

    > **Note:**
    >
    > The location of the domain does not change when you reconfigure it.

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

    During this process:

    •   Domain information is extracted, saved, and updated.

    •   Schemas, scripts, and other such files that support your Fusion Middleware
        products are updated.

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

11. The End of Configuration screen indicates whether the reconfiguration process
    completed successfully or failed. It also displays the location of the domain that

ORACLE

was reconfigured as well as the Administration Server URL (including the listen port). If the reconfiguration is successful, it displays **Oracle WebLogic Server Reconfiguration Succeeded**.

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

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

Restoring the mapViewerConfig.xml File

The mapViewerConfig.xml file is overwritten when you run the Reconfiguration Wizard on your domain. You can know that the mapViewerConfig.xml file is overwritten by checking its file size. You must restore the original file that you backed up before upgrading your domain with the Upgrade Assistant.

Restoring the mapViewerConfig.xml File on UNIX Operating Systems

To restore the original mapViewerConfig.xml file:

1. Change to the following directory:

   EXISTING_DOMAIN_HOME`/config/fmwconfig/mapviewer/conf`

2. Enter the following command to list the files within this directory and check the file size of the mapViewerConfig.xml file:

   `ls –ltr`

   Following is a sample output:

   `-rw-r----- 1 mwport svrtech `**`2xxx2`**` May 5 19:07 `*EXISTING_DOMAIN_HOME*`/config/`
   `fmwconfig/mapviewer/conf/mapViewerConfig.xml`

   > **✎ Note:**
   >
   > The change in the file size implies that the mapViewerConfig.xml file is overwritten by the reconfiguration template.

3. Restore the original file by renaming the mapViewerConfig_original.xml to mapViewerConfig.xml.

4. Enter the following command to list the files again and check the file size of the mapViewerConfig.xml file:

   `ls –ltr`

   The file size of the mapViewerConfig.xml file should now match with the file size you obtained while backing up the mapViewerConfig.xml file.

   In this example, **2xxx7**.

Restoring the mapViewerConfig.xml File on Windows Operating Systems

To restore the original mapViewerConfig.xml file:

1. Change to the following directory:

   EXISTING_DOMAIN_HOME\config\fmwconfig\mapviewer\conf

2. Enter the following command to list the files within this directory and check the file size of the mapViewerConfig.xml file:

   ```
   dir
   ```

   > ✎ **Note:**
   >
   > The change in the file size implies that the mapViewerConfig.xml file is overwritten by the reconfiguration template.

3. Restore the original file by renaming the mapViewerConfig_original.xml to mapViewerConfig.xml.

4. Enter the following command to list the files again and check the file size of the mapViewerConfig.xml file:

   ```
   dir
   ```

   The file size of the mapViewerConfig.xml file should now match with the file size you obtained while backing up the mapViewerConfig.xml file.

Upgrading Domain Component Configurations

After reconfiguring the domain, use the Upgrade Assistant to upgrade the domain *component* configurations inside the domain to match the updated domain configuration.

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

- Upgrading Domain Components Using the Upgrade Assistant
  Navigate through the screens in the Upgrade Assistant to upgrade component configurations in the WebLogic domain.

Starting the Upgrade Assistant

Run the Upgrade Assistant to upgrade product schemas, domain component configurations, or standalone system components to 12*c* (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 8-5   Upgrade Assistant Command-Line Parameters

Parameter	Required or Optional	Description
`-readiness`	Required for readiness checks **Note**: Readiness checks cannot be performed on standalone installations (those not managed by the WebLogic Server).	Performs the upgrade readiness check without performing an actual upgrade. Schemas and configurations are checked. Do not use this parameter if you have specified the `-examine` parameter.
`-threads`	Optional	Identifies the number of threads available for concurrent schema upgrades or readiness checks of the schemas. The value must be a positive integer in the range 1 to 8. The default is 4.

Table 8-5   (Cont.) Upgrade Assistant Command-Line Parameters

Parameter	Required or Optional	Description
-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 *silent mode* (without displaying Upgrade Assistant screens).
-examine	Optional	Performs the examine phase but does not perform an actual upgrade. Do not specify this parameter if you have specified the -readiness parameter.
-logLevel *attribute*	Optional	Sets the logging level, specifying one of the following attributes: • TRACE • NOTIFICATION • WARNING • ERROR • INCIDENT_ERROR The default logging level is NOTIFICATION. Consider setting the -logLevel TRACE attribute to so that more information is logged. This is useful when troubleshooting a failed upgrade. The Upgrade Assistant's log files can become very large if -logLevel TRACE is used.

Table 8-5   (Cont.) Upgrade Assistant Command-Line Parameters

Parameter	Required or Optional	Description
`-logDir` *location*	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) *NEW_ORACLE_HOME*`/oracle_common/upgrade/logs` *NEW_ORACLE_HOME*`/oracle_common/upgrade/temp` (Windows) *NEW_ORACLE_HOME*`\oracle_common\upgrade\logs` *NEW_ORACLE_HOME*`\oracle_common\upgrade\temp`
`-help`	Optional	Displays all of the command-line options.

Upgrading Domain Components Using the Upgrade Assistant

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

After running the Reconfiguration Wizard to reconfigure the WebLogic domain to 12*c* (12.2.1.3.0), you must run the Upgrade Assistant to upgrade the domain *component* configurations to match the updated domain configuration.

To upgrade the 12.2.1.3.0 domain:

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

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

2. On the next screen:

   - Select **All Configurations Used By a Domain**. The screen name changes to WebLogic Components.

   - In the **Domain Directory** field, enter the WebLogic domain directory path.

Click **Next**.

3. On the Component List screen, verify that the list includes all the components for which you want to upgrade configurations and click **Next**.

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

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

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

5. On the MapViewer Configuration screen, select **12c** and click **Next**.

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

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

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

7. On the Upgrade Summary screen, review the summary of the options you have selected for component configuration upgrade.

   The response file collects and stores all the information that you have entered, and enables you to perform a silent upgrade at a later time. The silent upgrade performs exactly the same function that the Upgrade Assistant performs, but you do not have to manually enter the data again. If you want to save these options to a response file, click **Save Response File** and provide the location and name of the response file.

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

8. On the Upgrade Progress screen, monitor the status of the upgrade.

> **⚠ Caution:**
>
> Allow the Upgrade Assistant enough time to perform the upgrade. Do not cancel the upgrade operation unless absolutely necessary. Doing so may result in an unstable environment.

If any components are not upgraded successfully, refer to the Upgrade Assistant log files for more information.

> **✎ Note:**
>
> The progress bar on this screen displays the progress of the current upgrade procedure. It does not indicate the time remaining for the upgrade.

Click **Next**.

9. If the upgrade is successful: On the Upgrade Success screen, click **Close** to complete the upgrade and close the wizard. The Post-Upgrade Actions window describes the manual tasks you must perform to make components functional in the new installation. This window appears only if a component has post-upgrade steps.

   If the upgrade fails: On the Upgrade Failure screen, click **View Log** to view and troubleshoot the errors. The logs are available at `NEW_ORACLE_HOME/oracle_common/upgrade/logs`.

> **✎ Note:**
>
> If the upgrade fails you must restore your pre-upgrade environment from backup, fix the issues, then restart the Upgrade Assistant.

Verifying the Domain-Specific-Component Configurations Upgrade

To verify that the domain-specific-component configurations upgrade was successful, sign in to the Administration console and the Oracle Enterprise Manager Fusion Middleware Control and verify that the version numbers for each component is 12.2.1.3.0.

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

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

> **✎ Note:**
>
> After upgrade, make sure you run the administration tools from the new 12*c*
> Oracle home directory and not from the previous Oracle home directory.
>
> During the upgrade process, some OWSM documents, including policy sets
> and predefined documents such as policies and assertion templates, may
> need to be upgraded. If a policy set or a predefined document is upgraded,
> its version number is incremented by 1.
>
> If you created the FMW user to run the Upgrade Assistant, ensure that you
> delete the account after verifying your upgrade was successful.

Packing the Upgraded Domain on Node1

After upgrading the existing domain to 12*c* (12.2.1.3.0), pack the upgraded domain on
Node1.

To pack the domain:

1. Sign in to the Node1.

2. Change to the following directory:

    (UNIX) *NEW_ORACLE_HOME*/oracle_common/common/bin

    (Windows) *NEW_ORACLE_HOME*\oracle_common\common\bin

3. Enter the following command to pack the domain:

    (UNIX) ./pack.sh -domain=*path to domain* -template=*path to
    template* -template_name=*name* -managed=true

    (Windows) pack.cmd -domain=*path to domain* -template=*path to
    template* -template_name=*name* -managed=true

    Example:

    ```
    pack.cmd -
    domain=C:\Oracle\Middleware\Oracle_Home\user_projects\domains
    \bi\ -template=C:\aaa\12.2.1.3.0_template_managed.jar -
    template_name=Node1_Upgrade -managed=true
    ```

    Sample output:

    ```
    << read domain from
    "C:\Oracle\Middleware\Oracle_Home\user_projects\domains\bi"
    >>  succeed: read domain from
    "C:\Oracle\Middleware\Oracle_Home\user_projects\domains\bi"
    << set config option Managed to "true"
    >>  succeed: set config option Managed to "true"
    << write template to "C:\aaa\12.2.1.3.0_template_managed.jar"
    ............................
    >>  succeed: write template to
    "C:\aaa\12.2.1.3.0_template_managed.jar"
    << close template
    >>  succeed: close template
    ```

ORACLE®

Unpacking the Upgraded Domain on Node2

You must unpack the domain that you packed on Node1 to Node2, which is the remote node.

To unpack the domain:

1. Sign in to the Node2.

2. Copy the template file that you created from Node1 to Node2 using the following command.

   ```
   scp template_filename username@remote_host:/directory
   ```

   Example:

   ```
   scp 12.2.1.3.0_template_managed.jar
   company@Node2:$12c_ORACLE_HOME/oracle_common/common/bin
   ```

3. Change to the following directory:

   (UNIX) *NEW_ORACLE_HOME*/oracle_common/common/bin

   (Windows) *NEW_ORACLE_HOME*\oracle_common\common\bin

4. Enter the following command to unpack the domain:

   (UNIX) ./unpack.sh -domain=*path to domain* -template=*path to template* -overwrite_domain=true

   (Windows) *DOMAIN_HOME*\common\bin\pack.cmd -domain=*path to domain* -overwrite_domain=true

   Example:

   ```
   unpack.cmd -
   domain=C:\Oracle\Middleware\Oracle_Home\user_projects\domains
   \bi -template=c:\aaa\12.2.1.3.0_template_managed.jar -
   overwrite_domain=true
   ```

   Sample output:

   ```
   << read template from "C:\aaa\12.2.1.3.0_template_managed.jar"
   >>  succeed: read template from
   "C:\aaa\12.2.1.3.0_template_managed.jar"
   << set config option OverwriteDomain to "true"
   >>  succeed: set config option OverwriteDomain to "true"
   << set config option DomainName to "bi"
   >>  succeed: set config option DomainName to "bi"
   << write Domain to
   "C:\Oracle\Middleware\Oracle_Home\user_projects\domains\bi"
   ...............................................................
   ..........................
   >>  succeed: write Domain to
   "C:\Oracle\Middleware\Oracle_Home\user_projects\domains\bi"
   << close template
   >>  succeed: close template
   ```

Synchronizing Connection Details to the Mid-Tier Database

Synchronizing connection details to the mid-tier database ensures that the Oracle BI components can access the mid-tier database when connection details, including credentials, are changed.

Supported mid-tier database types are DB2, SQLServer, MSSQL, and Oracle.

On UNIX, perform this task on the primary host. On Windows, perform this task on every host.

To synchronize connection details:

1. Change to the following directory:

   (UNIX) `NEW_DOMAIN_HOME`/bitools/bin

   (Windows) `NEW_DOMAIN_HOME`\bitools\bin

2. Run the synchronisation script:

   (UNIX) `sync_midtier_db.sh`

   (Windows) `sync_midtier_db.cmd`

Starting the Node Manager on Node2

After you have upgraded the Oracle BI domain on Node1, start the Node Manager on Node2.

To start the Node Manager:

1. Change to the following directory on Node2:

   (UNIX) `NEW_DOMAIN_HOME`/bin

   (Windows) `NEW_DOMAIN_HOME`\bin

2. Enter the following command:.

   `startNodeManager.sh`

   `startNodeManager.cmd`

Starting Servers and Processes

Start the Oracle BI instance and the system components on Node1.

> **✎ Note:**
>
> The procedure in this section describe how to start 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 start the Oracle BI instance:

1. Change directory to the following:

   (UNIX) *NEW_DOMAIN_HOME*/bitools/bin

   (Windows) *NEW_DOMAIN_HOME*\bitools\bin

2. To start the Oracle BI instance and servers, enter the following command:

   (UNIX) `start.sh`

   (Windows) `start.cmd`

> **✎ Note:**
>
> When prompted to enter the password, specify the Node Manager
> password that you entered while configuring the Oracle BI domain.

Checking the Status of the Servers in the Oracle BI Instance

The following servers must be in the running state on both the nodes: Servers hosting
Oracle BI instance (bi_server), OBIPS, OBIJH, OBICCS, OBICH, and OBIS.

To check the status of the servers:

1. Change to the following directory:

   (UNIX) *NEW_DOMAIN_HOME*/bitools/bin

   (Windows) *NEW_DOMAIN_HOME*\bitools\bin

2. Run the status script by entering the following command:

   (UNIX) `status.sh`

   (Windows) `status.cmd`

   Following is a sample output:

```
Name            Type            Machine                     Status
----            ----            -------                     ------
AdminServer     Server          machine_name
RUNNING
bi_server1      Server          machine_name.example.com    RUNNING
bi_server2      Server          machine_name.example.com    RUNNING
obips1          OBIPS           machine_name.example.com    RUNNING
obijh1          OBIJH           machine_name.example.com    RUNNING
obiccs1         OBICCS          machine_name.example.com    RUNNING
obisch1         OBISCH          machine_name.example.com    RUNNING
obis1           OBIS            machine_name.example.com    RUNNING
obips2          OBIPS           machine_name.example.com    RUNNING
obijh2          OBIJH           machine_name.example.com    RUNNING
obiccs2         OBICCS          machine_name.example.com    RUNNING
obisch2         OBISCH          machine_name.example.com    RUNNING
obis2           OBIS            machine_name.example.com    RUNNING
```

ORACLE

Validating the Oracle BI Deployments

The Oracle BI Baseline Validation Tool enables you to identify differences during life cycle operations, such as migrating from the Oracle BI 11*g* release to the 12*c* release. After you complete the migration procedure, you can use this tool to compare the two deployments and verify whether the results from the 11*g* environment are the same as the results from the 12*c* environment.

You can download the Oracle BI Validation Tool from Oracle BI Baseline Validation Tool Downloads.

For more information about using the Oracle BI Validation Tool, see Comparing Oracle Business Intelligence Deployments Using the Oracle Business Intelligence Baseline Validation Tool.

> ✏ **Note:**
>
> You can download the Oracle BI Validation Tool along with the other Oracle Business Intelligence download on the Oracle Technology Network. See the document that is included in the Oracle BI Validation Tool download for more information. For specific information on the distributions you want to download for each product, see Oracle Fusion Middleware Download, Installation, and Configuration Readme Files page.

9
Post-Upgrade Tasks

After completing the upgrade process, perform the tasks listed in this chapter to start the Oracle BI instance and validate the upgrade.

- **Enabling Internal SSL**
  Enable SSL on internal communication links after you complete the upgrade process.

- **Migrating Catalog Groups**
  Catalog groups were a feature of Oracle BI EE that allowed administrators to organize users and application roles for security administration purposes.

- **Allowing Content Developers to Include HTML Markup**
  Oracle BI EE features an option that allows content developers to include HTML markup such as Javascript in various catalog objects. Various properties dialogs in BI EE include the **Contains HTML Markup** option. By default, when you install or upgrade to BI 12*c* (12.2.1.3.0), this option is turned off for security reasons. Having the setting off by default is a change from earlier versions of BI EE.

- **Ensuring the Proper Version of Oracle R**
  If you use Oracle R for advanced analytics, then you must ensure that the proper version is installed or error messages might be displayed when using Oracle R functions after upgrading the system. For using advanced analytics, install Oracle R that is distributed with Oracle BI.

Enabling Internal SSL

Enable SSL on internal communication links after you complete the upgrade process.

To enable internal SSL:

1. Stop the system by entering the following command:

   (UNIX) *EXISTING_DOMAIN_HOME*/bitools/bin/stop.sh

   (Windows) *EXISTING_DOMAIN_HOME*\bitools\bin\stop.cmd

2. Enter the following command to ensable SSL on WebLogic internal channels and internal components:

   (UNIX) *EXISTING_DOMAIN_HOME*/bitools/bin/ssl.sh internalssl true

   (Windows) *EXISTING_DOMAIN_HOME*\bitools\bin\ssl.cmd internalssl true

3. Restart the system by entering the following command:

   (UNIX) *EXISTING_DOMAIN_HOME*/bitools/bin/start.sh

   (Windows) *EXISTING_DOMAIN_HOME*\bitools\bin\start.cmd

Migrating Catalog Groups

Catalog groups were a feature of Oracle BI EE that allowed administrators to organize users and application roles for security administration purposes.

In Oracle BI Release 12*c* (12.2.1.1 and higher), catalog groups are obsolete. The catalog groups are migrated by the upgrade process to application roles.

Allowing Content Developers to Include HTML Markup

Oracle BI EE features an option that allows content developers to include HTML markup such as Javascript in various catalog objects. Various properties dialogs in BI EE include the **Contains HTML Markup** option. By default, when you install or upgrade to BI 12*c* (12.2.1.3.0), this option is turned off for security reasons. Having the setting off by default is a change from earlier versions of BI EE.

To allow content developers to include HTML markup, you must edit the `instanceconfig.xml` file to include the EnableSavingContentWithHTML element set to true, and you must specify the appropriate privileges for content developers. See EnableSavingContentWithHTML in *Security Guide for Oracle Business Intelligence Enterprise Edition* for details on enabling this functionality.

Ensuring the Proper Version of Oracle R

If you use Oracle R for advanced analytics, then you must ensure that the proper version is installed or error messages might be displayed when using Oracle R functions after upgrading the system. For using advanced analytics, install Oracle R that is distributed with Oracle BI.

You must install Oracle R using the R installer provided with Data Visualization. This installation enables Oracle R for the corresponding Data Visualization installation, only. No other installation of Oracle R works with Data Visualization.

The r-installer.zip file contains the installer for Oracle R on Linux servers and Windows client machines. The file is located in the following directory:

`NEW_ORACLE_HOME`/bi/bifoundation/advanced_analytics.

To ensure the proper version of Oracle R:

1. Manually copy the r-installer.zip from the `NEW_ORACLE_HOME`/bi/bifoundation/advanced_analytics directory to the `EXISTING_ORACLE_HOME`/bi/bifoundation/advanced_analytics directory.

2. Unzip r-installer.zip.

3. Run the installer for Oracle R.

   For example, `r-package-install-linux320.R` for Linux or `r-package-install-win320.R` for Windows.

4. Run an Oracle R function to ensure that it works properly.

A

Known Differences: Oracle Business Intelligence 12*c*

This appendix lists the differences in appearance and functionality of the Oracle BI 12*c* system after migration.

Oracle Business Intelligence Release 12*c* includes many new and enhanced features. To learn about the features and enhancements, see the "What's New" sections described in those guides.

- SCM: DSO and AR Balance Trend Are Displayed Differently in 12c
  The DSO and AR Balance Trend graph in the Supply Chain and Order Management module can be displayed differently in Oracle BI 12*c* even if the underlying data used to generate this graph is the same. The display difference occurs due to the manner in which the BI Server handles the order of the time period data that is fetched from the database.

- Visual Cues in BI Composer are Displayed the Same Accessibility Mode
  Visuals cues, such as icons and check boxes, which are displayed on pages in BI Composer, are displayed the same whether you have accessibility mode turned on or not. Even though the list of text-based wizard page links are not displayed on a wizard page in accessibility mode, screen reader applications such as JAWS can read them.

- Enabling Embedded Content in Dashboards
  For security reasons, you can no longer embed content from external domains in dashboards. To embed external content in dashboards, you must edit the instanceconfig.xml file.

- Trellis Views Display Without Grid Lines in 12c
  A trellis is a view type that presents multidimensional data laid out in a set of cells in a grid, with each cell displaying a subset of data shown as numbers or as graphs. The trellis view allows users to display multiple views all at once for quick comparison, and to display data that reveals trends. In Oracle BI 11*g*, the trellis view contained grid lines, by default. With Oracle BI 12*c*, you can select horizontal or vertical lines, or you can select the default option.

- Domain Home and bipublisher Folder Path Different in 12c
  The Domain home and bipublisher file paths are different in 12*c* as compared to the 11*g* system.

- Oracle BI EE Content Changes for 12c
  This topic describes the difference in appearance for analyses and views after you migrate to 12*c*.

- Changes to Catalog Structure
  In releases prior to Release 11.1.1.9.0, when you first sign in to Oracle Business Intelligence, folders are automatically created for you to hold content that you eventually create. Examples of these folders are the _portal folder to hold personal dashboards and the _delivers folder to hold alerts. Starting with Release 11.1.1.9.0, these folders are not automatically created when you sign in for the first time. Instead, the folders are created when they are needed.

- Installing Oracle BI with Oracle Database Release 12.2.0.1.0
  If you are installing Oracle BI 12*c* (12.2.1.3.0) on Oracle Database Release
  12.2.0.1.0, you might see a "BI Startup failure" error message on the Configuration
  Progress screen.

- Specifying Widths of Sections on Dashboard Pages
  HTML style properties, such as those for specifying the widths of sections on
  pages in dashboards, behave differently in Oracle BI 11*g* than 12*c*.

SCM: DSO and AR Balance Trend Are Displayed Differently in 12*c*

The DSO and AR Balance Trend graph in the Supply Chain and Order Management
module can be displayed differently in Oracle BI 12*c* even if the underlying data used
to generate this graph is the same. The display difference occurs due to the manner in
which the BI Server handles the order of the time period data that is fetched from the
database.

Visual Cues in BI Composer are Displayed the Same Accessibility Mode

Visuals cues, such as icons and check boxes, which are displayed on pages in BI
Composer, are displayed the same whether you have accessibility mode turned on or
not. Even though the list of text-based wizard page links are not displayed on a wizard
page in accessibility mode, screen reader applications such as JAWS can read them.

Enabling Embedded Content in Dashboards

For security reasons, you can no longer embed content from external domains
in dashboards. To embed external content in dashboards, you must edit the
instanceconfig.xml file.

For a complete procedure, see Embedding External Content in Dashboards in *System
Administrator's Guide for Oracle Business Intelligence Enterprise Edition*.

Trellis Views Display Without Grid Lines in 12*c*

A trellis is a view type that presents multidimensional data laid out in a set of cells
in a grid, with each cell displaying a subset of data shown as numbers or as graphs.
The trellis view allows users to display multiple views all at once for quick comparison,
and to display data that reveals trends. In Oracle BI 11*g*, the trellis view contained grid
lines, by default. With Oracle BI 12*c*, you can select horizontal or vertical lines, or you
can select the default option.

To specify the grid options:

1. Open the trellis view for editing.

2. Click the properties dialog.

3. Based on your choice, select:

   - Horizontal

- Vertical
- Default option

Domain Home and bipublisher Folder Path Different in 12*c*

The Domain home and bipublisher file paths are different in 12*c* as compared to the 11*g* system.

For Domain Home:

In 11*g*: `mwhome/user_projects/domains/bifoundation_domain/`

In 12*c*: `ORACLE_HOME/user_projects/domains/bi/`

For bipublisher folder:

In 11*g*: `DOMAIN_HOME/config/bipublisher/`

In 12*c*: `DOMAIN_HOME/bidata/components/bipublisher/`

Oracle BI EE Content Changes for 12*c*

This topic describes the difference in appearance for analyses and views after you migrate to 12*c*.

The following describes changes to analyses and views between 11*g* and 12*c*:

Change	Description
The Gauge view shows a data column	In 11*g*, the gauge view did not display unit data. In 12*c*, the gauge view displays the value of the gauge chart in the form of a data column below the chart.
Prompts are left-aligned in PDF files	In 11*g*, when you export an analysis to PDF, prompts are center-aligned. In 12*c*, when you export the same analysis to PDF, prompts are left-aligned.
The bottom border of the title view does not span the page width	In 11*g*, when you export an analysis to a PDF file, the bottom border of the title view spans the width of the page. In 12*c*, when you export the same analysis to PDF, the bottom border spans only the width of the title view.
Measures hidden in views	In previous releases, if you marked a measure as hidden in the Column Properties dialog, then the measure was still displayed in views if the measure was part of the view's measure list (that is, not added to an edge of the view but used as intended as a metric). In 12*c*, a measure that you mark as hidden isn't displayed in views.

Changes to Catalog Structure

In releases prior to Release 11.1.1.9.0, when you first sign in to Oracle Business Intelligence, folders are automatically created for you to hold content that you eventually create. Examples of these folders are the _portal folder to hold personal dashboards and the _delivers folder to hold alerts. Starting with Release 11.1.1.9.0,

these folders are not automatically created when you sign in for the first time. Instead, the folders are created when they are needed.

The following list shows the structure of the Oracle BI Presentation Catalog for a user before Release 11.1.1.9.0:

```
/users/<user>
/users/<user>/_delivers
    /users/<user>/_delivers/_deliveries
/users/<user>/_filters
/users/<user>/_portal
/users/<user>/_selections
/users/<user>/_savedcolumns
/users/<user>/_subscriptions
/users/<user>/_thumbnails
/users/<user>/_prefs
    /users/<user>/_prefs/deliveryprofiles
/users/<user>/_prefs/devices
```

The following list shows the structure of the Oracle BI Presentation Catalog for a user in Release 11.1.1.9.0 and later. Objects previously stored in the "_prefs" and "_subscriptions" folders are moved to the "_internals" folder. The "_deliveries" folder is merged with its parent folder "_delivers".

```
/users/<user>/_delivers
/users/<user>/_portal
/users/<user>/_filters
/users/<user>/_selections
/users/<user>/_savedcolumns
/users/<user>/_thumbnails
/users/<user>/_internals
    /users/<user>/_internals/subscriptions
        /users/<user>/_internals/subscriptions.atr
    /users/<user>/_internals/_deliveryprofile_profile1
        /users/<user>/_internals/_deliveryprofile_profile1.atr
    /users/<user>/_internals/_device_device1
        /users/<user>/_internals/_device_device1.atr
    /users/<user>/_internals/_favorites
        /users/<user>/_internals/_favorites.atr
    /users/<user>/_internals/cacheduserinfo
        /users/<user>/_internals/cacheduserinfo.atr
    /users/<user>/_internals/volatileuserdata
        /users/<user>/_internals/volatileuserdata.atr
    /users/<user>/_internals/defaultdevices
        /users/<user>/_internals/defaultdevices.atr
    /users/<user>/_internals/defaultdeliveryprofile
        /users/<user>/_internals/defaultdeliveryprofile.atr
    /users/<user>/_internals/userprefsxmlstore
        /users/<user>/_internals/userprefsxmlstore.atr
```

During upgrade, you can optionally remove empty folders within each existing user's home directory or relocate existing folders to their new locations. This change has no effect on folders for existing users.

To remove empty folders within each existing user's home directory or relocate existing folders to their new locations, specify the following elements within the <Catalog></Catalog> nodes in the instanceconfig.xml file:

- To remove all empty folders: `<CleanEmptyFolderInHome>true</CleanEmptyFolderInHome>`

- To relocate existing folders to their new locations: `<CoalesceHomeDirectoryFolders>true</CoalesceHomeDirectoryFolders><`

> **Note:**
>
> Revert the changes after completing the task.

Installing Oracle BI with Oracle Database Release 12.2.0.1.0

If you are installing Oracle BI 12*c* (12.2.1.3.0) on Oracle Database Release 12.2.0.1.0, you might see a "BI Startup failure" error message on the Configuration Progress screen.

To work around this issue:

1. Stop the database listener by entering the following command:

   `lsnrctl stop`

2. Open the `sqlnet.ora` file in an editor.

   By default, this file is located in the `NEW_ORACLE_HOME/network/admin` directory. You might also find the file in the directory that is specified by the `TNS_ADMIN` environment variable.

3. Add the following line in the `sqlnet.ora` file:

   `SQLNET.ALLOWED_LOGON_VERSION_SERVER=8`

4. Save your changes and close the file.

5. Start the database listener by entering the following command:

   `lsnrctl start`

6. Try to install Oracle BI 12*c* (12.2.1.3.0) again.

Specifying Widths of Sections on Dashboard Pages

HTML style properties, such as those for specifying the widths of sections on pages in dashboards, behave differently in Oracle BI 11*g* than 12*c*.

You specify a size by setting the **Size** option in the **Additional Formatting Options** area of the **Section Properties** dialog when editing a dashboard page.

In Oracle BI 12*c*, if you want a section to have a certain width, don't select the **Minimum Size** option in the Size dialog.

B

Troubleshooting a Migrated Oracle BI Instance

This appendix provides procedures to troubleshoot a migrated BI instance from 11*g* to 12*c*.

- Configuration Failure with TRANSFORMBAREXCEPTION
  The 12*c* configuration might fail with a TRANSFORMBAREEXCEPTION error, if any objects in the catalog are older than Release 11.1.1.7. You can confirm the need to update the catalog items by viewing the metrics in Oracle Enterprise Manager Fusion Middleware Control. In the Catalog folder, find a metric called `Reads Needing Upgrade` with the description "The number of objects read that required upgrading." If the number is large, then you can resolve this issue by updating objects in the catalog using the Administration page in Presentation Services.

- Multiple Y-Axis Highcharts Graphs Not Displayed
  Multiple Y-axis highchart graphs likely don't render after migration, because the Oracle BI 12*c* instance does not include the required JavaScript (.js) files. You must copy the files to the instance manually.

- Setting the Evaluate Parameter in the NQSConfig.INI File
  The EVALUATE function in your analyses is not enabled by default in Logical SQL queries because it exposes the database to SQL injection attacks. If you encounter any error while generating any views after migrating to Oracle BI 12*c*, then you must set the value of the EVALUATE_SUPPORT_LEVEL parameter appropriately in the NQSConfig.INI file.

- Error: java.lang.NoClassDefFoundError: Could not initialize class sun.awt.X11GraphicsEnvironment
  If you have set the value of the DISPLAY variable manually or the system has manipulated it automatically, you can encounter the *java.lang.NoClassDefFoundError: Could not initialize class sun.awt.X11GraphicsEnvironment* error . In case you change the value of the DISPLAY variable to a Windows-client system IP, to run the 64-bit Linux installer through a Windows client, reset it to its original value after the installation is complete. If you generate an analysis that contains graphs without resetting the value of the DISPLAY variable, you can get this error.

- Verifying File Data Sources in Oracle BI Publisher
  While generating reports, if you encounter an error stating `oracle.xdo.XDOException: File does not exist or is not readable:...`, you might not have copied the supporting XML files to the 12*c* system. To solve this issue, manually validate that the path for each data source points to the appropriate directory on the 12*c* system.

- Error on checkSoftLimit Failed
  You see the `[nQSError: 43119] Query Failed: [nQSError: 46168] Operation RawFile::checkSoftLimit failed, it'll exceed the current limitation. 1073758528` error if the query requires more temporary space to complete than the BI Administrator has allocated.

- Act As Configuration Failing in 12c
  The Act As functionality allows you to act for another user in Oracle BI EE. This functionality is useful, for example, when you need to work on another user's dashboard or content, or when IT support staff wants to troubleshoot another user's account. If you have migrated from Oracle BI 11*g* to 12*c* and cannot configure the Act As functionality on the 12*c* instance, you must first copy a file from the 11*g* instance.

- Replacing Custom Links in the Global Header
  After migrating to Oracle BI 12*c*, you might notice that links that you had added to the global header in the 11*g* instance are no longer available. The links are missing because the customlinks.xml file isn't migrated from the 11*g* to the 12*c* instance. You must manually copy the file to the 12*c* instance and enable the Custom Links functionality in the instanceconfig.xml file.

- Migrating 11g WebLogic Users to 12c
  After migrating from 11*g* to 12*c*, you must manually migrate the WebLogic users in to your 12*c* domain. You can do this by exporting the security data from a security realm and importing it in to another security realm. Security data includes authentication, authorization, credential map, and role data.

- View Display Error: Invalid Column Encountered in Orders Outstanding Report
  You might see a View Display Error message that states that an invalid column was encountered when loading the Orders Outstanding Report in 12*c*. This error is displayed because the ID of the column that causes the error exists in the XML definition of the report, but not in the subject area table. You must manually edit the XML definition and remove the invalid column.

- View Display Error While Generating Oracle R-based Advanced Analytic Reports
  If you see the following error message while generating Oracle R-based Advanced Analytic reports after you have upgraded to Oracle BI 12*c*, see the procedure in this topic to fix the issue. View Display Error: Odbc driver returned an error (SQLExecDirectW). Error codes: OPR4ONWY, U9IM8TAC, U9IM8TAC, U9IM8TAC, OI2DL65P.

Configuration Failure with TRANSFORMBAREXCEPTION

The 12*c* configuration might fail with a TRANSFORMBAREEXCEPTION error, if any objects in the catalog are older than Release 11.1.1.7. You can confirm the need to update the catalog items by viewing the metrics in Oracle Enterprise Manager Fusion Middleware Control. In the Catalog folder, find a metric called `Reads Needing Upgrade` with the description "The number of objects read that required upgrading." If the number is large, then you can resolve this issue by updating objects in the catalog using the Administration page in Presentation Services.

To update the catalog:

1. Sign in to Oracle BI EE with administration credentials.

2. In the global header, click **Administration**.

3. Click the **Scan and Update Catalog Objects That Require Updates** link.

4. Click **Update Catalog Objects** to begin the update process.

   Click the other links on the page to see which objects were updated and which were not. You can view the log files for details on objects that were not updated.

Restart Oracle BI Presentation Services using Oracle Enterprise Manager Fusion Middleware Control.

- Regenerating User GUIDs
  This task requires that you manually edit the configuration files to instruct Oracle BI Server and Oracle BI Presentation Server to regenerate the GUIDs on restart. Once completed, you edit these files again to remove the modification.

- Updating Oracle BI Presentation Services Catalog
  When migrating from 11*g*, you must manually update your Oracle BI Presentation Services Catalog.

Regenerating User GUIDs

This task requires that you manually edit the configuration files to instruct Oracle BI Server and Oracle BI Presentation Server to regenerate the GUIDs on restart. Once completed, you edit these files again to remove the modification.

For information about locating the Oracle Business Intelligence configuration files, see Configuration Files in *System Administrator's Guide for Oracle Business Intelligence Enterprise Edition*.

To regenerate user GUIDs:

1. Update the `FMW_UPDATE_ROLE_AND_USER_REF_GUIDS` parameter in the 11*g* NQSConfig.INI file.

   a. Open the 11*g* NQSConfig.INI file for editing present at these locations:

      (UNIX) `11g_DOMAIN_HOME/config/OracleBIServerComponent/coreapplication_obisn`

      (Windows)
      `11g_DOMAIN_HOME\config\OracleBIServerComponent\coreapplication_obisn`

   b. Locate the `FMW_UPDATE_ROLE_AND_USER_REF_GUIDS` parameter and set it to `YES`, as follows

      `FMW_UPDATE_ROLE_AND_USER_REF_GUIDS = YES;`

   c. Save and close the file.

2. Update the Catalog element in the 11*g* instanceconfig.xml file present at these locations:

   (UNIX) `11g_DOMAIN_HOME/config/OracleBIPresentationServicesComponent/coreapplication_obipsn`

   (Windows)
   `11g_DOMAIN_HOME\config\OracleBIPresentationServicesComponent\coreapplication_obipsn`

   a. Open the instanceconfig.xml file for editing:

   b. Locate the Catalog element and update it as follows:

      ```
      <Catalog>
      <UpgradeAndExit>false</UpgradeAndExit>
      <UpdateAccountGUIDs>UpdateAndExit</UpdateAccountGUIDs>
      </Catalog>
      ```

   c. Save and close the file.

3. Restart the Oracle BI system components using opmnctl:

```
cd ORACLE_HOME/admin/instancen/bin
./opmnctl stopall
./opmnctl startall
```

4. Set the `FMW_UPDATE_ROLE_AND_USER_REF_GUIDS` parameter in the NQSConfig.INI file back to `NO`.

> **❶ Important:**
>
> You must perform this step to ensure that your system is secure.

5. Update the Catalog element in the 11*g* instanceconfig.xml file to remove the UpdateAccount GUIDs entry.

6. Restart the Oracle BI system components using opmnctl:

```
cd ORACLE_HOME/admin/instancen/bin
./opmnctl stopall
./opmnctl startall
```

Updating Oracle BI Presentation Services Catalog

When migrating from 11*g*, you must manually update your Oracle BI Presentation Services Catalog.

To update the Oracle BI Presentation Services Catalog:

1. Shut down Oracle BI Presentation Services using Oracle Enterprise Manager Fusion Middleware Control.

   See Using Fusion Middleware Control to Start and Stop BI System Component Processes in *System Administrator's Guide for Oracle Business Intelligence Enterprise Edition*.

2. Back up your existing Oracle BI Presentation Services Catalog by using the 7-Zip utility to create a compressed file.

3. Create a backup copy of the instanceconfig.xml file present at these locations:

   (UNIX) *11g_DOMAIN_HOME*/config/
   `OracleBIPresentationServicesComponent/coreapplication_obips`*n*

   (Windows)
   11g_DOMAIN_HOME\config\OracleBIPresentationServicesComponent\
   `coreapplication_obips`*n*

4. Change the `UpgradeAndExit` option to `true` in the instanceconfig.xml file.

   Find the following code:

```
<ps:Catalog xmlns:ps="oracle.bi.presentation.services/config/v1.1">
   <ps:UpgradeAndExit>false</ps:UpgradeAndExit>
</ps:Catalog>
```

Change it to the following:

```
<ps:Catalog xmlns:ps="oracle.bi.presentation.services/config/v1.1">
    <ps:UpgradeAndExit>true</ps:UpgradeAndExit>
</ps:Catalog>
```

5. Start Oracle BI Presentation Services using the following OPMN command to update the Oracle BI Presentation Services Catalog:

    `opmnctl startproc ias-component=coreapplication_obipsn`

    Where, *n* is the instance.

    For example:

    (UNIX) `cd ORACLE_INSTANCE/bin`

    (UNIX) `./opmnctl startproc ias-component=coreapplication_obips1`

    (Windows) `cd ORACLE_INSTANCE\bin`

    (Windows) `opmnctl startproc ias-component=coreapplication_obips1`

6. After catalog is updated, edit the instanceconfig.xml file again and change the `UpgradeAndExit` option back to `false`.

7. Restart Oracle BI Presentation Services using Oracle Enterprise Manager Fusion Middleware Control.

Multiple Y-Axis Highcharts Graphs Not Displayed

Multiple Y-axis highchart graphs likely don't render after migration, because the Oracle BI 12*c* instance does not include the required JavaScript (.js) files. You must copy the files to the instance manually.

To copy the JavaScript files to the 12*c* instance:

1. Change to the `\AnalyticsRes` directory on the 11*g* system.

2. Copy the following files from the 11*g* `\AnalyticsRes` directory to the 12*c* `\AnalyticsRes` directory:

    *   highcharts.js
    *   exporting.js
    *   jquery-1.8.2.min.js

Setting the Evaluate Parameter in the NQSConfig.INI File

The EVALUATE function in your analyses is not enabled by default in Logical SQL queries because it exposes the database to SQL injection attacks. If you encounter any error while generating any views after migrating to Oracle BI 12*c*, then you must set the value of the EVALUATE_SUPPORT_LEVEL parameter appropriately in the NQSConfig.INI file.

The default value of the EVALUATE_SUPPORT_LEVEL parameter is "0", which means that evaluate is not supported.

```
# EVALUATE_SUPPORT_LEVEL:
# 1: evaluate is supported for users with manageRepositories permission
# 2: evaluate is supported for any user.
# other: evaluate is not supported if the value is anything else.
EVALUATE_SUPPORT_LEVEL = 0;
```

Manually set the value of the EVALUATE_SUPPORT_LEVEL parameter same as it is in the 11*g* NQSConfig.INI file or as required by your organization.

The NQSConfig.INI file is located at the following location:

(UNIX) *12c_DOMAIN_HOME*/config/fmwconfig/biconfig/OBIS*n*

(Windows) *12c_DOMAIN_HOME*\config\fmwconfig\biconfig\OBIS*n*

Error: java.lang.NoClassDefFoundError: Could not initialize class sun.awt.X11GraphicsEnvironment

If you have set the value of the DISPLAY variable manually or the system has manipulated it automatically, you can encounter the *java.lang.NoClassDefFoundError: Could not initialize class sun.awt.X11GraphicsEnvironment* error . In case you change the value of the DISPLAY variable to a Windows-client system IP, to run the 64-bit Linux installer through a Windows client, reset it to its original value after the installation is complete. If you generate an analysis that contains graphs without resetting the value of the DISPLAY variable, you can get this error.

To fix this error:

1. Reset the value of the DISPLAY variable.

2. Restart the JavaHost process.

Verifying File Data Sources in Oracle BI Publisher

While generating reports, if you encounter an error stating `oracle.xdo.XDOException: File does not exist or is not readable:...`, you might not have copied the supporting XML files to the 12*c* system. To solve this issue, manually validate that the path for each data source points to the appropriate directory on the 12*c* system.

To validate paths for the data source names:

1. Sign in to Oracle BI Publisher.

2. Go to **File** under **Data Source** on the Administration page.

   On the File page, you can see the directory path for each DSN.

3. Check whether the file is pointing to the correct 12*c* directory.

4. To change the directory path, click the **DSN**.

   You see the Update Data Source page.

5. Enter the correct and the complete path of the top-level directory and click **Apply**.

Error on checkSoftLimit Failed

You see the `[nQSError: 43119] Query Failed: [nQSError: 46168] Operation RawFile::checkSoftLimit failed, it'll exceed the current limitation. 1073758528` error if the query requires more temporary space to complete than the BI Administrator has allocated.

To resolve this issue:

1. Modify the query to reduce the volume of data that is processed by the BI Server by:

    a. Adding additional filters to the query.

    b. Replacing the most detailed attribute columns with less detailed columns to reduce the overall level of detail of the query.

    c. Removing attribute columns containing long text values.

    d. Avoiding the use of SQL functions that are not supported by the backend data source.

    e. Removing certain attribute or measure columns that cause more complex internal processing in the BI Server.

2. If you cannot resolve the issue by modifying or restructuring the query and if the system is capable of handling more load, request the BI Administrator to increase the maximum size allowed for temporary files by modifying the following parameters in the NQSConfig.INI file:

    *   `WORK_DIRECTORY_SIZE_GLOBAL_LIMIT` or

    *   `MAX_WORK_FILE_SIZE_PERCENT`

    For example:

    ```
    [General]
    WORK_DIRECTORY_SIZE_GLOBAL_LIMIT = 100 GB;
    # No temporary file will be allowed to exceed this percentage of
    the global work
    # directory limit.
    MAX_WORK_FILE_SIZE_PERCENT = 5;
    ```

> **Note:**
>
> - The maximum size allowed for any single temporary file is determined by multiplying the total size limit of the temporary directory set in the `WORK_DIRECTORY_SIZE_GLOBAL_LIMIT` parameter by the percentage specified in `MAX_WORK_FILE_SIZE_PERCENT`. By default the temporary file size limit is 100 GB * 5% = 5 GB. This per-file limit applies to each temporary file individually, while the total `WORK_DIRECTORY_SIZE_GLOBAL_LIMIT` applies collectively to all the temporary files created by the BI Server.
>
> - You must increase the `MAX_WORK_FILE_SIZE_PERCENT` parameter cautiously. Setting a large percentage can allow a single request to consume a large portion of the total temporary space, possibly causing the system and other BI requests to run out of temporary space and fail. Increasing the `WORK_DIRECTORY_SIZE_GLOBAL_LIMIT` is safer as long as there is sufficient disk space. However, you must observe caution while increasing the value of the `WORK_DIRECTORY_SIZE_GLOBAL_LIMIT` parameter. Setting a large value can cause the system to run out of disk space or allow BI requests to consume too many system resources.

Act As Configuration Failing in 12*c*

The Act As functionality allows you to act for another user in Oracle BI EE. This functionality is useful, for example, when you need to work on another user's dashboard or content, or when IT support staff wants to troubleshoot another user's account. If you have migrated from Oracle BI 11*g* to 12*c* and cannot configure the Act As functionality on the 12*c* instance, you must first copy a file from the 11*g* instance.

Before configuring the Act As functionality in 12*c*:

1. Create the "customMessages" folder under the following directory:

    (UNIX) `NEW_ORACLE_HOME/bi/bifoundation/web/msgdb/l_en`

    (Windows) `NEW_ORACLE_HOME\bi\bifoundation\web\msgdb\l_en`

2. Copy the 11*g* LogonParamSQLTemplate.xml file to the "customMessages" folder created in step 1.

For more information about creating a custom message template, see Creating a Custom Message Template for Proxy Functionality .

Replacing Custom Links in the Global Header

After migrating to Oracle BI 12*c*, you might notice that links that you had added to the global header in the 11*g* instance are no longer available. The links are missing because the customlinks.xml file isn't migrated from the 11*g* to the 12*c* instance.

You must manually copy the file to the 12*c* instance and enable the Custom Links functionality in the instanceconfig.xml file.

See Providing Custom Links in Presentation Services in *System Administrator's Guide for Oracle Business Intelligence Enterprise Edition*.

To enable custom links functionality:

1. Copy the customlinks.xml file to the following directory on the 12*c* instance:

   (UNIX) `12c_DOMAIN_HOME`/bidata/components/OBIPS/customMessages

   (Windows)
   `12c_DOMAIN_HOME`\bidata\components\OBIPS\customMessages

2. Open the instanceconfig.xml file for editing from the following location:

   (UNIX) `12c_DOMAIN_HOME`/config/fmwconfig/biconfig/OBIPS

   (Windows) `12c_DOMAIN_HOME`\config\fmwconfig\biconfig\OBIPS

3. Enable the custom links as shown in the following example:

   ```
   <CustomLinks>
   <Enabled>true</Enabled>
   </CustomLinks>
   ```

4. Save your changes and close the file.

5. Restart Presentation Services.

Migrating 11*g* WebLogic Users to 12*c*

After migrating from 11*g* to 12*c*, you must manually migrate the WebLogic users in to your 12*c* domain. You can do this by exporting the security data from a security realm and importing it in to another security realm. Security data includes authentication, authorization, credential map, and role data.

Understand when you would want to migrate security data. See Overview of Security Data Migration.

Understand the concepts associated with importing and exporting security data. See Migration Concepts.

Understand which formats and constraints are supported by the WebLogic Security providers. See Formats and Constraints Supported by WebLogic Security Providers.

> **✎ Note:**
>
> You can only export and import security data between security realms in the same WebLogic Server release.

Exporting the Security Data from a Security Provider

To export security data from a security provider to a file:

1. In the left pane, select **Security Realms** and then select the name of the realm you are configuring (for example, myrealm).

2. Select the type of provider from which you want to export security data (for example, Authentication).

3. Select the security provider from which you want to export security data.

4. Select **Export** under **Migration**.

5. Specify the directory and filename in which to export the security data in the **Export File on Server** field. The directory must exist.

> **Note:**
>
> The directory and file into which you export the security data should be carefully protected with operating system security as they contain secure information about your deployment.

6. (Optional) Define a specific set of security data to be exported in the **Export Constraints** box.

7. Click **Save**.

   After you export the data from the security provider, you can import it any time.

Importing the Security Data into a Security Provider

To import security data into a security provider:

1. In the left pane of the Administration Console, select **Security Realms**.

2. Select the name of the security realm into which the security data is to be imported (for example, myrealm).

3. Select **Providers** and then the type of provider into which security data is to be imported (for example, **Authentication** under **Providers**).

4. Select the security provider in which the security data is to be imported and select **Import** under **Migration**.

5. Specify the directory and file name of the file that contains the exported security data in the **Import File on Server** field.

6. Click **Save**.

View Display Error: Invalid Column Encountered in Orders Outstanding Report

You might see a View Display Error message that states that an invalid column was encountered when loading the Orders Outstanding Report in 12*c*. This error is displayed because the ID of the column that causes the error exists in the XML definition of the report, but not in the subject area table. You must manually edit the XML definition and remove the invalid column.

View Display Error While Generating Oracle R-based Advanced Analytic Reports

If you see the following error message while generating Oracle R-based Advanced Analytic reports after you have upgraded to Oracle BI 12*c*, see the procedure in this topic to fix the issue. View Display Error: Odbc driver returned an error (SQLExecDirectW). Error codes: OPR4ONWY, U9IM8TAC, U9IM8TAC, U9IM8TAC, OI2DL65P.

To resolve the view display error:

1. Verify that you are able to successfully generate R-based Advanced Analytic reports in 12.2.1.x environment.

2. Back up the 12.2.1.x R script that you have customized.

3. After you have upgraded to 12.2.1.3.0, rerun the R installation packages (r-installer.zip). The R installation packages are bundled with the Oracle BI distribution and are available at the following location:

   (UNIX) *NEW_ORACLE_HOME*/bi/bifoundation/advanced_analytics/

   (Windows) *NEW_ORACLE_HOME*\bi\bifoundation\advanced_analytics\

   For detailed information about installing Oracle Enterprise R, see Installing R and Oracle R Enterprise for External Logical SQL Functions in *Installing and Configuring Oracle Business Intelligence*.

4. After installing the 12.2.1.3.0 R script, copy the custom settings from the 12.2.1.x R script to the 12.2.1.3.0 R script.

C

Migrating Security Certificates when Using Secure LDAP

From 12*c* (12.2.1.3.0) onwards, the IBM LDAP client libraries used in previous versions of the product have been replaced with Oracle LDAP libraries. If you are using secure LDAP in your Oracle BI deployment, you need to create an Oracle wallet to manage your security credentials, and migrate your existing CA certificate to the wallet.

To check whether you are using secure LDAP:

1. On the Oracle BI Administration Tool, click **Manage** and then **Identity** to open the Identity Manager.

2. On the Identity Manager window, click **Directory Servers** on the left pane.

   The right pane displays the available LDAP objects.

3. Select and open the LDAP object to open the LDAP Server dialog.

4. On the LDAP Server dialog, click **Advanced**.

   If the **SSL** option is selected, then secure LDAP is in use.

If you are using secure LDAP, then you must first create an Oracle wallet and then specify the Oracle wallet directory on the Repository tab using the Oracle BI Administration Tool.

- Creating an Oracle Wallet
- Specifying the Wallet Directory

Creating an Oracle Wallet

To create an Oracle Wallet:

1. Export the CA certificate from the key database file using the IBM key management utility (iKeyman) and save it in a `.kdb` file.

2. Run the Oracle Wallet Manager.

3. Create a new wallet and specify a password.

4. From the Operations window, click **Import Trusted Certificate**.

5. On the Import Trusted Certificate dialog, select the **Select a file that contains the certificate** option and click **OK**.

6. Browse and select the file that you exported in step 1.

7. Save the wallet and note its location.

Specifying the Wallet Directory

To specify the Oracle wallet directory:

---

1. In the Oracle BI Administration Tool, click **Tools**.

2. In the Options dialog, select the Repository tab.

3. In the **Wallet directory** field, specify the location of the Oracle wallet that you created.

4. Specify and confirm the password for this wallet and click **OK**.

D

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

Consider that you have a JDK version jdk1.8.0_121 installed on your machine. When you install and configure an Oracle Fusion Middleware product, the utilities, such as Configuration Wizard (`config.sh|exe`), OPatch, or RCU point to a default JDK, for example, jdk1.8.0_121. After some time, Oracle releases a new version of the JDK, say jdk1.8.0_131 that carries security enhancements and bug fixes. From 12*c* (12.2.1.3.0) onwards, you can upgrade the existing JDK to a newer version, and can have the complete product stack point to the newer version of the JDK.

You can maintain multiple versions of JDK and switch to the required version on need basis.

- About Updating the JDK Location After Installing an Oracle Fusion Middleware Product
  The binaries and other metadata and utility scripts in the Oracle home and Domain home, such as RCU or Configuration Wizard, use a JDK version that was used while installing the software and continue to refer to the same version of the JDK. The JDK path is stored in a variable called JAVA_HOME which is centrally located in .globalEnv.properties file inside the *ORACLE_HOME*/oui directory.

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

The binaries and other metadata and utility scripts in the Oracle home and Domain home, such as RCU or Configuration Wizard, use a JDK version that was used while installing the software and continue to refer to the same version of the JDK. The JDK path is stored in a variable called JAVA_HOME which is centrally located in .globalEnv.properties file inside the *ORACLE_HOME*/oui directory.

The utility scripts such as config.sh|cmd, launch.sh, or opatch reside in the *ORACLE_HOME*, and when you invoke them, they refer to the JAVA_HOME variable located in .globalEnv.properties file. To point these scripts and utilities to the newer version of JDK, you must update the value of the JAVA_HOME variable in the .globalEnv.properties file by following the directions listed in Updating the JDK Location in an Existing Oracle Home .

To make the scripts and files in your Domain home directory point to the newer version of the JDK, you can follow one of the following approaches:

- Specify the path to the newer JDK on the Domain Mode and JDK screen while running the Configuration Wizard.

  For example, consider that you installed Oracle Fusion Middleware Infrastructure with the JDK version 8u121. So while configuring the WebLogic domain with the Configuration Assistant, you can select the path to the newer JDK on the Domain

---

Mode and JDK screen of the Configuration Wizard. Example: `/scratch/jdk/jdk1.8.0_131`.

- Manually locate the files that have references to the JDK using `grep` (UNIX) or `findstr` (Windows) commands and update each reference. See Updating the JDK Location in an Existing Domain Home.

> **Note:**
>
> If you install the newer version of the JDK in the same location as the existing JDK by overwriting the files, then you don't need to take any action.

- Updating the JDK Location in an Existing Oracle Home
  The `getProperty.sh|cmd` script displays the value of a variable, such as JAVA_HOME, from the .globalEnv.properties file. The `setProperty.sh|cmd` script is used to set the value of variables, such as OLD_JAVA_HOME or JAVA_HOME that contain the locations of old and new JDKs in the .globalEnv.properties file.

- Updating the JDK Location in an Existing Domain Home
  You must search the references to the current JDK, for example jdk1.8.0_121 manually, and replace those instances with the location of the new JDK.

Updating the JDK Location in an Existing Oracle Home

The `getProperty.sh|cmd` script displays the value of a variable, such as JAVA_HOME, from the .globalEnv.properties file. The `setProperty.sh|cmd` script is used to set the value of variables, such as OLD_JAVA_HOME or JAVA_HOME that contain the locations of old and new JDKs in the .globalEnv.properties file.

The `getProperty.sh|cmd` and `setProperty.sh|cmd` scripts are located in the following location:

(UNIX) *ORACLE_HOME*`/oui/bin`
(Windows) *ORACLE_HOME*`\oui\bin`
Where, *ORACLE_HOME* is the directory that contains the products using the current version of the JDK, such as jdk1.8.0_121.

To update the JDK location in the .globalEnv.properties file:

1. Use the `getProperty.sh|cmd` script to display the path of the current JDK from the JAVA_HOME variable. For example:

   (UNIX) *ORACLE_HOME*`/oui/bin/getProperty.sh JAVA_HOME`
   (Windows) *ORACLE_HOME*`\oui\bin\getProperty.cmd JAVA_HOME`
   `echo JAVA_HOME`
   Where JAVA_HOME is the variable in the .globalEnv.properties file that contains the location of the JDK.

2. Back up the path of the current JDK to another variable such as OLD_JAVA_HOME in the .globalEnv.properties file by entering the following commands:

   (UNIX) *ORACLE_HOME*`/oui/bin/setProperty.sh -name OLD_JAVA_HOME -value `*specify_the_path_of_current_JDK*
   (Windows) *ORACLE_HOME*`\oui\bin\setProperty.cmd -name OLD_JAVA_HOME -value `*specify_the_path_of_current_JDK*

ORACLE

This command creates a new variable called OLD_JAVA_HOME in the .globalEnv.properties file, with a value that you have specified.

3. Set the new location of the JDK in the JAVA_HOME variable of the .globalEnv.properties file, by entering the following commands:

(UNIX) *ORACLE_HOME*`/oui/bin/setProperty.sh -name JAVA_HOME -value` *specify_the_location_of_new_JDK*
(Windows) *ORACLE_HOME*`\oui\bin\setProperty.cmd -name JAVA_HOME -value` *specify_the_location_of_new_JDK*

After you run this command, the JAVA_HOME variable in the .globalEnv.properties file now contains the path to the new JDK, such as jdk1.8.0_131.

Updating the JDK Location in an Existing Domain Home

You must search the references to the current JDK, for example jdk1.8.0_121 manually, and replace those instances with the location of the new JDK.

You can use the `grep` (UNIX) or `findstr` (Windows) commands to search for the jdk-related references.

You'll likely be required to update the location of JDK in the following three files:

(UNIX) *DOMAIN_HOME*`/bin/setNMJavaHome.sh`
(Windows) *DOMAIN_HOME*`\bin\setNMJavaHome.cmd`

(UNIX) *DOMAIN_HOME*`/nodemanager/nodemanager.properties`
(Windows) *DOMAIN_HOME*`\nodemanager\nodemanager.properties`

(UNIX) *DOMAIN_HOME*`/bin/setDomainEnv.sh`
(Windows) *DOMAIN_HOME*`\bin\setDomainEnv.cmd`

E

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*.

F
Starting Servers and Processes

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

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

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

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

Step 1: Start the Administration Server

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

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

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

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

Step 2: Start Node Manager

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

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

Step 3: Start Oracle Identity Management Components

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

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

Step 4: Start the Managed Servers

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

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

When prompted, enter your user name and password.

> **Note:**
>
> The startup of a Managed Server will typically start the applications that are deployed to it. Therefore, it should not be necessary to manually start applications after the Managed Server startup.

Step 5: Start System Components

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

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

You can start system components in any order.