Oracle® Fusion Middleware

Administrator's Guide

11*g* Release 2 (11.1.2)

E28516-02

August 2012

Describes how to manage Oracle Fusion Middleware, including how to start and stop Oracle Fusion Middleware, how to configure and reconfigure components, and how to back up and recover your environment.

ORACLE

Oracle Fusion Middleware Administrator's Guide, 11*g* Release 2 (11.1.2)

E28516-02

Copyright © 2009, 2012, Oracle and/or its affiliates. All rights reserved.

Primary Author: Helen Grembowicz

Contributing Authors: Ellen Desmond, Vinaye Misra

Contributors: Mike Blevins, Nick Fry, Greg Cook, Shalendra Goel, Harry Hsu, Christine Jacobs, Srini Indla, Pavana Jain, Gopal Kirsur, Kenneth Ma, Dan MacKinnon, Manoj Nayak, Mark Nelson, Praveen Sampath, Sachin Kapur,, Sunita Sharma

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

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

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

U.S. GOVERNMENT RIGHTS Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, duplication, disclosure, modification, and adaptation shall be subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License (December 2007). Oracle America, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information on content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services.

Contents

4 Starting and Stopping Oracle Fusion Middleware

5 Managing Ports

Part III Secure Sockets Layer

6 Configuring SSL in Oracle Fusion Middleware

7 Using the SSL Automation Tool

8 Managing Keystores, Wallets, and Certificates

9 Understanding the Deployment Process

10 Deploying Applications

13 Diagnosing Problems

15 Changing Network Configurations

Part VII Advanced Administration: Backup and Recovery

16 Introducing Backup and Recovery

17 Backing Up Your Environment

18 Recovering Your Environment

H Oracle Wallet Manager and orapki

I Troubleshooting Oracle Fusion Middleware

Index

List of Figures

List of Tables

Preface

This guide describes how to manage Oracle Fusion Middleware, including how to start and stop Oracle Fusion Middleware, how to change ports, deploy applications, and how to back up and recover Oracle Fusion Middleware.

Audience

This guide is intended for administrators of Oracle Fusion Middleware.

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

Related Documents

For more information, see the following documents in the Oracle Fusion Middleware 11*g* Release 2 (11.1.2) documentation set:

- *Oracle Fusion Middleware 2 Day Administration Guide*
- *Oracle Fusion Middleware Concepts*
- *Oracle Fusion Middleware Application Security Guide*
- *Oracle Fusion Middleware High Availability Guide*
- *Oracle Fusion Middleware Introduction to Oracle WebLogic Server*
- *Oracle Fusion Middleware Performance and Tuning Guide*
- *Oracle Fusion Middleware Administrator's Guide for Oracle SOA Suite and Oracle Business Process Management Suite*
- *Oracle Fusion Middleware Administrator's Guide for Oracle WebCenter Portal*
- *Oracle Fusion Middleware Administrator's Guide for Oracle HTTP Server*

- *Oracle Fusion Middleware Administrator's Guide for Oracle Web Cache*
- *Oracle Fusion Middleware Administrator's Guide for Oracle Access Management*
- *Oracle Fusion Middleware Security and Administrator's Guide for Web Services*
- *Oracle Fusion Middleware Administrator's Guide for Oracle Internet Directory*
- *Oracle Fusion Middleware Administrator's Guide for Oracle Virtual Directory*
- *Oracle Fusion Middleware Third-Party Application Server Guide*

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.

What's New in This Guide?

This preface introduces the new and changed administrative features of Oracle Fusion Middleware that are described in this guide, and provides pointers to additional information.

New and Changed Features for Oracle Fusion Middleware 11*g* Release 2 (11.1.2)

Oracle Fusion Middleware 11*g* Release 2 (11.1.2) includes the following new and changed administrative features:

- Support for backup and recovery of the following components:
  - Oracle Entitlements Server: See Section 16.5.2.10.
  - Oracle Privileged Account Manager: See Section 16.5.2.11.
  - Oracle Access Management Mobile and Social: See Section 16.5.2.12.
  - Oracle Access Management Secure Token Service: See Section 16.5.2.13
- Support for moving the following components from a source to a target environment:
  - Oracle Entitlements Server
  - Oracle Privileged Account Manager
  - Oracle Access Management Mobile and Social
  - Oracle Access Management Secure Token Service

  See Section 21.4.1.

Part I

Understanding Oracle Fusion Middleware

This part provides an overview to Oracle Fusion Middleware and its concepts as they relate to administering Oracle Fusion Middleware.

Part I contains the following chapters:

- Chapter 1, "Introduction to Oracle Fusion Middleware"
- Chapter 2, "Understanding Oracle Fusion Middleware Concepts"

1 Introduction to Oracle Fusion Middleware

Oracle Fusion Middleware is a comprehensive family of products ranging from application development tools and integration solutions to identity management, collaboration, and business intelligence reporting. This chapter provides an introduction to Oracle Fusion Middleware.

It includes the following topics:

1.1 What Is Oracle Fusion Middleware?

Oracle Fusion Middleware is a collection of standards-based software products that spans a range of tools and services: from Java EE and developer tools, to integration services, identity management, business intelligence, and collaboration. Oracle Fusion Middleware offers complete support for development, deployment, and management.

1.2 Oracle Fusion Middleware Components

Oracle Fusion Middleware provides the following components:

- Oracle WebLogic Server, an enterprise-ready Java application server that supports the deployment of mission-critical applications in a robust, secure, highly available, and scalable environment. Oracle WebLogic Server is an ideal foundation for building applications based on service-oriented architecture (SOA).

  See Also: *Oracle Fusion Middleware Introduction to Oracle WebLogic Server*

- Oracle SOA Suite, a complete set of service infrastructure components, in a service-oriented architecture, for designing, deploying, and managing composite applications. Oracle SOA Suite enables services to be created, managed, and orchestrated into composite applications and business processes. Composites enable you to easily assemble multiple technology components into one SOA composite application.

  See Also: *Oracle Fusion Middleware Administrator's Guide for Oracle SOA Suite and Oracle Business Process Management Suite*

- Oracle WebCenter Portal, an integrated set of components with which you can create social applications, enterprise portals, collaborative communities, and composite applications, built on a standards-based, service-oriented architecture.

Oracle WebCenter Portal combines dynamic user interface technologies with which to develop rich internet applications, the flexibility and power of an integrated, multichannel portal framework, and a set of horizontal Enterprise 2.0 capabilities delivered as services that provide content, collaboration, presence, and social networking capabilities. Based on these components, Oracle WebCenter Portal also provides an out-of-the-box, enterprise-ready customizable application, WebCenter Portal: Spaces, with a configurable work environment that enables individuals and groups to work and collaborate more effectively.

See Also: *Oracle Fusion Middleware Administrator's Guide for Oracle WebCenter Portal*

- Oracle WebCenter Content, an integrated suite of products designed for managing content. This enterprise content management platform enables you to leverage industry-leading document management, Web content management, digital asset management, and records management functionality to build your business applications. Building a strategic enterprise content management infrastructure for content and applications helps you to reduce costs, easily share content across the enterprise, minimize risk, automate expensive, time-intensive, and manual processes, and consolidate multiple Web sites onto a single platform.

  See Also: *Oracle WebCenter Content System Administrator's Guide for Content Server*

- Oracle HTTP Server, which provides a Web listener for Java EE applications and the framework for hosting static and dynamic pages and applications over the Web. Based on the proven technology of the Apache HTTP Server, Oracle HTTP Server includes significant enhancements that facilitate load balancing, administration, and configuration.

  See Also: *Oracle Fusion Middleware Administrator's Guide for Oracle HTTP Server*

- Oracle Web Cache, a content-aware server accelerator, or reverse proxy, that improves the performance, scalability, and availability of Web sites that run on Oracle Fusion Middleware.

  See Also: *Oracle Fusion Middleware Administrator's Guide for Oracle Web Cache*

- Oracle Identity Management, which provides a shared infrastructure for all Oracle applications. It also provides services and interfaces that facilitate third-party enterprise application development. These interfaces are useful for application developers who need to incorporate identity management into their applications.

  See Also: *Oracle Fusion Middleware Integration Overview for Oracle Identity Management Suite*

- Oracle Internet Directory, a general-purpose directory service that enables fast retrieval and centralized management of information about dispersed users and network resources. It combines Lightweight Directory Access Protocol (LDAP) Version 3 with the high performance, scalability, robustness, and availability of Oracle Database.

See Also: *Oracle Fusion Middleware Administrator's Guide for Oracle Internet Directory*

- Oracle Virtual Directory, an LDAP version 3 enabled service that provides virtualized abstraction of one or more enterprise data sources into a single directory view. Oracle Virtual Directory provides the ability to integrate LDAP-aware applications into diverse directory environments while minimizing or eliminating the need to change either the infrastructure or the applications. It supports a diverse set of clients, such as Web applications and portals, and it can connect to directories, databases, and Web services.

  See Also: *Oracle Fusion Middleware Administrator's Guide for Oracle Virtual Directory*

- Oracle Access Management, which provides the following:
  - Oracle Access Management Access Manager, which provides a full range of Web perimeter security functions that include Web single sign-on, authentication and authorization, policy administration, and auditing. Single sign-on (SSO) enables users, and groups of users, to access multiple applications after authentication. SSO eliminates multiple sign-on requests. Access Manager is the Oracle Fusion Middleware single sign-on solution.

    See Also: *Oracle Fusion Middleware Administrator's Guide for Oracle Access Management*

  - Oracle Access Management Identity Federation, a self-contained federation solution that provides the infrastructure that enables identities and their relevant entitlements to be propagated across security domains—this applies to domains existing within an organization as well as between organizations.

    See Also: *Oracle Fusion Middleware Administrator's Guide for Oracle Identity Federation*

  - Oracle Access Management Mobile and Social, which acts as an intermediary between a user seeking to access protected resources, and the back-end Access Management and Identity Management services that protect the resources. Mobile and Social provides simplified client libraries that allow developers to quickly add feature-rich authentication, authorization, and Identity capabilities to registered applications.

    See Also: *Oracle Fusion Middleware Administrator's Guide for Oracle Access Management*

  - Oracle Secure Token Service, which provides the foundation to the current security infrastructure to facilitate a consistent and streamlined model for token acquisition, renewal, and cancellation that is protocol and security infrastructure agnostic.

    See Also: *Oracle Fusion Middleware Administrator's Guide for Oracle Access Management*

- Oracle Web Services Manager, which provides a way to centrally define and manage policies that govern Web services operations, including access control (authentication and authorization), reliable messaging, Message Transmission

Optimization Mechanism (MTOM), WS-Addressing, and Web services management. Policies can be attached to multiple Web services, requiring no modification to the existing Web services.

See Also: *Oracle Fusion Middleware Security and Administrator's Guide for Web Services*

- Oracle Platform Security, which provides enterprise product development teams, systems integrators, and independent software vendors (ISVs) with a standards-based, portable, integrated, enterprise-grade security framework for Java Standard Edition (Java SE) and Java Enterprise Edition (Java EE) applications.

  Oracle Platform Security provides an abstraction layer in the form of standards-based application programming interfaces (APIs) that insulate developers from security and identity management implementation details. With Oracle Platform Security, developers do not need to know the details of cryptographic key management or interfaces with user repositories and other identity management infrastructures. When you use Oracle Platform Security, in-house developed applications, third-party applications, and integrated applications benefit from the same uniform security, identity management, and audit services across the enterprise.

  See Also: *Oracle Fusion Middleware Application Security Guide*

- Oracle Portal, a Web-based tool for building and deploying e-business portals. It provides a secure, manageable environment for accessing and interacting with enterprise software services and information resources. A portal page makes data from multiple sources accessible from a single location.

  See Also: *Oracle Fusion Middleware Administrator's Guide for Oracle Portal*

- Oracle Business Intelligence, a complete, integrated solution that addresses business intelligence requirements. Oracle Business Intelligence includes Oracle BI Enterprise Edition, Oracle Business Intelligence Discoverer, Oracle Business Intelligence Publisher, and Oracle Real-Time Decisions.

  See Also: *Oracle Fusion Middleware System Administrator's Guide for Oracle Business Intelligence Enterprise Edition*

Note: You can also use Oracle Fusion Middleware with IBM WebSphere. For more information, see the *Oracle Fusion Middleware Third-Party Application Server Guide*.

2

Understanding Oracle Fusion Middleware Concepts

This chapter provides information about Oracle Fusion Middleware concepts, such as the Middleware home, Oracle homes and Metadata Repository, that are related to administering Oracle Fusion Middleware.

2.1 Understanding Key Oracle Fusion Middleware Concepts

Oracle Fusion Middleware provides two types of components:

- A **Java component,** which is an Oracle Fusion Middleware component that is deployed as one or more Java EE applications and a set of resources. Java components are deployed to an Oracle WebLogic Server domain as part of a domain template. Examples of Java components are the Oracle SOA Suite and Oracle WebCenter Portal components.
- A **system component**, which is a manageable process that is not deployed as a Java application. Instead, a system component is managed by Oracle Process Manager and Notification (OPMN). The system components are:
  - Oracle HTTP Server
  - Oracle Web Cache
  - Oracle Internet Directory
  - Oracle Virtual Directory
  - Oracle Forms Services
  - Oracle Reports
  - Oracle Business Intelligence Discoverer
  - Oracle Business Intelligence

A Java component and a system component are peers.

After you install and configure Oracle Fusion Middleware, your Oracle Fusion Middleware environment contains the following:

- An Oracle WebLogic Server domain, which contains one Administration Server and one or more Managed Servers. The Administration Server contains Oracle WebLogic Server Administration Console and Fusion Middleware Control. The Managed Servers contain components, such as Oracle WebCenter Portal and Oracle SOA Suite.

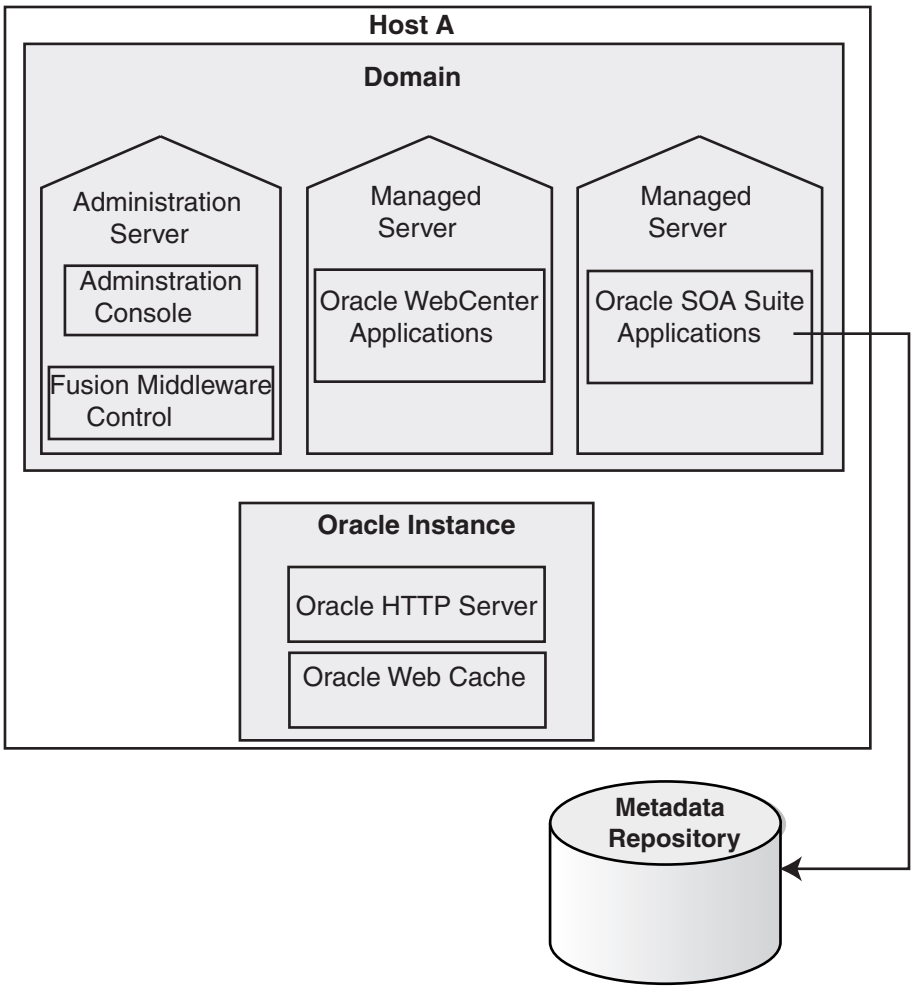
  See Section 2.2 for information about Oracle WebLogic Server domains.

- If your environment includes system components, one or more Oracle instances. See Section 2.3 for information about Oracle instances.

- A database that is used as a metadata repository, if the components you installed require one. For example, Oracle SOA Suite requires a metadata repository. See Section 2.7 for information about metadata repositories.

Figure 2–1 shows an Oracle Fusion Middleware environment with an Oracle WebLogic Server domain that contains an Administration Server, two Managed Servers, and an Oracle instance. The environment also includes a metadata repository.

Figure 2–1 Oracle Fusion Middleware Environment



Your environment also includes a Middleware home, which consists of the Oracle WebLogic Server home, and, optionally, an Oracle Common home and one or more Oracle homes. See Section 2.4 for more information.

Note: You can also use Oracle Fusion Middleware with IBM WebSphere. For more information, see the *Oracle Fusion Middleware Third-Party Application Server Guide*.

2.2 What Is an Oracle WebLogic Server Domain?

An Oracle WebLogic Server administration **domain** is a logically related group of Java components. A domain includes a special WebLogic Server instance called the **Administration Server,** which is the central point from which you configure and manage all resources in the domain. Usually, you configure a domain to include additional WebLogic Server instances called **Managed Servers.** You deploy Java components, such as Web applications, EJBs, and Web services, and other resources, to the Managed Servers and use the Administration Server for configuration and management purposes only.

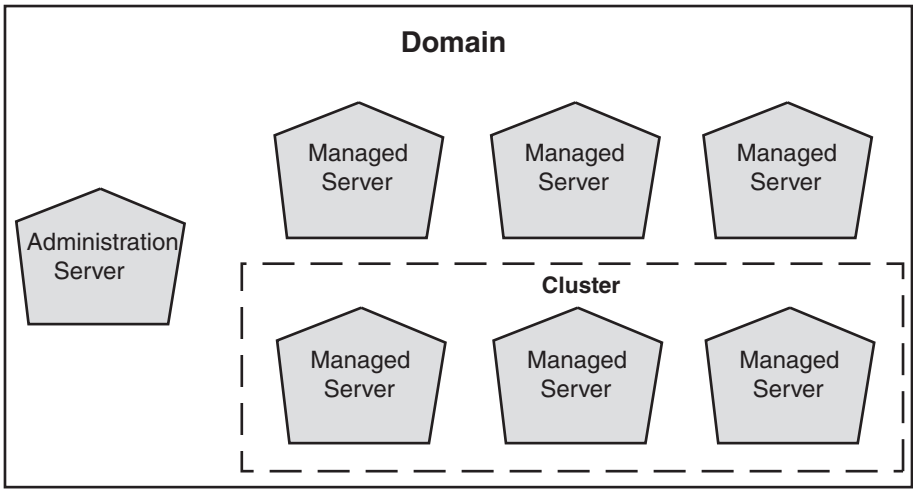
Managed Servers in a domain can be grouped together into a cluster.

The directory structure of a domain is separate from the directory structure of the WebLogic Server home. It can reside anywhere; it need not be within the Middleware home directory. The top-level directory of a domain is referred to as the **domain home**.

A domain is a peer of an Oracle instance. Both contain specific configurations outside of their Oracle homes.

Figure 2–2 shows a domain with an Administration Server, three standalone Managed Servers, and three Managed Servers in a cluster.

Figure 2–2 Oracle WebLogic Server Domain



See Also: *Oracle Fusion Middleware Understanding Domain Configuration for Oracle WebLogic Server* for more information about domain configuration

The following topics describe entities in the domain:

- What Is the Administration Server?

- Understanding Managed Servers and Managed Server Clusters
- What Is Node Manager?

2.2.1 What Is the Administration Server?

The **Administration Server** operates as the central control entity for the configuration of the entire domain. It maintains the domain's configuration documents and distributes changes in the configuration documents to Managed Servers. The Administration Server serves as a central location from which to manage and monitor all resources in a domain.

Each domain must have one server instance that acts as the Administration Server.

To interact with the Administration Server, you can use the Oracle WebLogic Server Administration Console, Oracle WebLogic Scripting Tool (WLST), or create your own JMX client. In addition, you can use Fusion Middleware Control for some tasks.

Oracle WebLogic Server Administration Console and Fusion Middleware Control run in the Administration Server. Oracle WebLogic Server Administration Console is the Web-based administration console used to manage the resources in an Oracle WebLogic Server domain, including the Administration Server and Managed Servers. Fusion Middleware Control is a Web-based administration console used to manage Oracle Fusion Middleware, including components such as Oracle HTTP Server, Oracle SOA Suite, Oracle WebCenter Portal, Oracle Portal, and Oracle Identity Management.

See Also:

- Section 3.3 for more information about Fusion Middleware Control
- Section 3.4 of this book, as well as the *Oracle Fusion Middleware Introduction to Oracle WebLogic Server* and the Oracle WebLogic Server Administration Console Online help, for more information about Oracle WebLogic Server Administration Console

2.2.2 Understanding Managed Servers and Managed Server Clusters

Managed Servers host business applications, application components, Web services, and their associated resources. To optimize performance, Managed Servers maintain a read-only copy of the domain's configuration document. When a Managed Server starts, it connects to the domain's Administration Server to synchronize its configuration document with the document that the Administration Server maintains.

When you create a domain, you create it using a particular domain template. That template supports a particular component or group of components, such as the Oracle SOA Suite. The Managed Servers in the domain are created specifically to host those particular Oracle Fusion Middleware components.

Oracle Fusion Middleware Java components (such as Oracle SOA Suite, Oracle WebCenter Portal, and some Identity Management components), as well as customer-developed applications, are deployed to Managed Servers in the domain.

If you want to add other components, such as Oracle WebCenter Portal, to a domain that was created using a template that supports another component, you can extend the domain by creating additional Managed Servers in the domain, using a domain template for the component that you want to add. See Section 19.2 for more information.

For production environments that require increased application performance, throughput, or high availability, you can configure two or more Managed Servers to

operate as a cluster. A **cluster** is a collection of multiple WebLogic Server instances running simultaneously and working together to provide increased scalability and reliability. In a cluster, most resources and services are deployed identically to each Managed Server (as opposed to a single Managed Server), enabling failover and load balancing. A single domain can contain multiple Oracle WebLogic Server clusters, as well as multiple Managed Servers that are not configured as clusters. The key difference between clustered and nonclustered Managed Servers is support for failover and load balancing. These features are available only in a cluster of Managed Servers.

> **See Also:** "Understanding WebLogic Server Clustering" in *Oracle Fusion Middleware Using Clusters for Oracle WebLogic Server*

2.2.3 What Is Node Manager?

Node Manager is a Java utility that runs as a separate process from Oracle WebLogic Server and allows you to perform common operations for a Managed Server, regardless of its location with respect to its Administration Server. While use of Node Manager is optional, it provides valuable benefits if your Oracle WebLogic Server environment hosts applications with high-availability requirements.

If you run Node Manager on a computer that hosts Managed Servers, you can start and stop the Managed Servers remotely using the Administration Console, Fusion Middleware Control, or the command line. Node Manager can also automatically restart a Managed Server after an unexpected failure.

> **See Also:** *Oracle Fusion Middleware Node Manager Administrator's Guide for Oracle WebLogic Server*

2.3 What Is an Oracle Instance?

An **Oracle instance** contains one or more system components, such as Oracle Web Cache, Oracle HTTP Server, or Oracle Internet Directory. The system components in an Oracle instance must reside on the same computer. An Oracle instance directory contains updatable files, such as configuration files, log files, and temporary files.

An Oracle instance is a peer of an Oracle WebLogic Server domain. Both contain specific configurations outside of their Oracle homes.

The directory structure of an Oracle instance is separate from the directory structure of the Oracle home. It can reside anywhere; it need not be within the Middleware home directory.

2.4 What Is a Middleware Home?

A **Middleware home** is a container for the Oracle WebLogic Server home, and, optionally, one Oracle Common home and one or more Oracle homes.

A Middleware home can reside on a local file system or on a remote shared disk that is accessible through NFS.

See Section 2.5 for information about Oracle WebLogic Server homes. See Section 2.6 for information about Oracle homes.

2.5 What Is a WebLogic Server Home?

A WebLogic Server home contains installed files necessary to host a WebLogic Server. The WebLogic Server home directory is a peer of Oracle home directories and resides within the directory structure of the Middleware home.

2.6 What Is an Oracle Home and the Oracle Common Home?

An **Oracle home** contains installed files necessary to host a specific component or software suite. For example, the SOA Oracle home contains a directory that contains binary and library files for Oracle SOA Suite.

An Oracle home resides within the directory structure of the Middleware home. Each Oracle home can be associated with multiple Oracle instances or Oracle WebLogic Server domains. There can be multiple Oracle homes within each Middleware home.

The **Oracle Common home** contains the binary and library files required for Fusion Middleware Control and Java Required Files (JRF). There can be only one Oracle Common home within each Middleware home.

2.7 What Is the Oracle Metadata Repository?

The Oracle Metadata Repository contains metadata for Oracle Fusion Middleware components, such as Oracle BPEL Process Manager, Oracle B2B, and Oracle Portal. It can also contain metadata about the configuration of Oracle Fusion Middleware and metadata for your applications.

A metadata repository can be database-based or file-based. If it is database-based, you can create it in an existing database using the Repository Creation Utility (RCU).

Oracle Fusion Middleware supports multiple repository types. A repository type represents a specific schema or set of schemas that belong to a specific Oracle Fusion Middleware component (for example, Oracle SOA Suite or Oracle Internet Directory.)

A particular type of repository, the Oracle Metadata Services (MDS) repository, contains metadata for most Oracle Fusion Middleware components, such as Oracle B2B, and for certain types of applications.

> **See Also:** Chapter 14 for more information about metadata repositories

Part II

Basic Administration

This part describes basic administration tasks for Oracle Fusion Middleware.

Part II contains the following chapters:

- Chapter 3, "Getting Started Managing Oracle Fusion Middleware"
- Chapter 4, "Starting and Stopping Oracle Fusion Middleware"
- Chapter 5, "Managing Ports"

3

Getting Started Managing Oracle Fusion Middleware

When you install Oracle Fusion Middleware, you install the binary files, such as executable files, jar files, and libraries. Then, you use configuration tools to configure the software. This chapter provides information you need to get started managing Oracle Fusion Middleware, including information about the tools you use.

This chapter includes the following topics:

3.1 Setting Up Environment Variables

When you installed Oracle Fusion Middleware, you were logged in to your operating system as a particular user. You should always log in as this user to manage your installation because this user has permission to view and modify the files in your installation's Oracle home.

To use Oracle Fusion Middleware, you must set environment variables as shown in the following tables:

- Table 3–1, " Environment Variables for Linux and UNIX"
- Table 3–2, " Environment Variables for Windows"

Table 3–1 Environment Variables for Linux and UNIX

Environment Variable	Value
DISPLAY	*hostname*:*display_number*.*screen_number* Very few tools, such as `oidadmin`, require the DISPLAY variable.

Table 3–1 (Cont.) Environment Variables for Linux and UNIX

Environment Variable	Value

LD_LIBRARY_PATH	On Solaris, ensure that the value contains the following directory:
	`$ORACLE_HOME/lib32`
	On Linux and HP-UX, ensure that the value contains the following directory:
	`$ORACLE_HOME/lib`
	On IBM AIX, ensure that this environment variable is not set.

(IBM AIX only)
LIBPATH
	If the calling application is a 32-bit application, ensure that the value contains the following directory:
	`$ORACLE_HOME/lib32`
	If the calling application is a 64-bit application, ensure that the value contains the following directory:
	`$ORACLE_HOME/lib`

(Solaris only) LD_LIBRARY_PATH_64	Ensure that the value contains the following directory: `$ORACLE_HOME/lib`
(HP-UX only) SHLIB_PATH	Ensure that the value contains the following directory: `$ORACLE_HOME/lib32`
MW_HOME	Set the value to the full path of the installation's Middleware home. Do not use a trailing slash in the definition. The following example shows the full path: `/scratch/Oracle/Middleware`
ORACLE_HOME	Setting this is useful if you are working with just one Oracle home. Set to the full path of the Oracle home. Do not use a trailing slash in the definition. The following example shows the full path: `/scratch/Oracle/Middleware/ORACLE_HOME_SOA1`
ORACLE_INSTANCE	Optional. Setting this is useful if you have only one Oracle instance in your environment or if you are working with just that one instance. Set to the full path of an Oracle instance. Do not use a trailing slash in the definition. The following example shows the full path of a Web Tier installation: `/scratch/Oracle/Middleware/WebTier/instances/instance1`

PATH	Ensure that the value contains the following directories, which contains basic commands used by all installations:
	`$ORACLE_COMMON_HOME/bin`
	`$ORACLE_COMMON_HOME/common/bin`
	When you start to work with specific components, you may want to add additional directories to your path, as recommended by the component documentation.

JAVA_HOME	Ensure that the value contains the following directory: *MW_HOME*`/jdk`*n*
CLASSPATH	Ensure that the value contains the following directories: `$ORACLE_HOME/lib:`*MW_HOME*`/jdk`*n*`/lib`

Table 3–2 shows the environment variables for Windows.

Table 3–2 Environment Variables for Windows

Environment Variable	Value
MW_HOME	Set the value to the full path of the installation's Middleware home. Do not use a trailing backslash in the definition. The following example shows the full path: `C:\oracle\Middleware`
ORACLE_HOME	Setting this is useful if you are working with just one Oracle home. Set the value to the full path of the Oracle home. Do not use a trailing backslash in the definition. The following example shows the full path: `C:\oracle\Middleware\ORACLE_SOA1`
ORACLE_INSTANCE	Optional. Setting this is useful if you have only one Oracle instance in your environment or if you are working with just that one instance. Set the value to the full path of an Oracle instance. Do not use a trailing backslash in the definition. The following example shows the full path of a Web Tier installation: `C:\oracle\Middleware\WebTier\instances\instance1`
PATH	Ensure that the value contains the following directory, which contains basic commands used by all installations: *ORACLE_COMMON_HOME*`\bin` *ORACLE_COMMON_HOME*`\common\bin`
JAVA_HOME	Ensure that the value contains the following directory: *MW_HOME*`\jdk`*n*
CLASSPATH	Ensure that the value contains the following directories: *ORACLE_HOME*`\lib:`*MW_HOME*`\jdk`*n*`\lib`
TEMP	Set the value to your temp directory, for example, C:\temp.
TMP	Set the value to your temp directory, for example, C:\temp.

Best Practices for Multiple Installations on a UNIX Host

If you have multiple installations of Oracle Fusion Middleware on a UNIX host, it is very important to completely set your environment when managing a particular installation.

Some Oracle Fusion Middleware commands use the MW_HOME and ORACLE_HOME environment variables to determine which installation to operate on, and some use the directory location of the command. It is, therefore, not sufficient to simply reset your environment variables or change directories to a different Oracle home as you move between installations. You must fully change to the new installation as follows:

1. Log in as the user who installed Oracle Fusion Middleware.

   On UNIX hosts, you may also use the `su` command to switch to the user, but be sure to use the dash (-) option so that your environment is set the same as it would have been had you actually logged in as that user. For example:

   `su - user`

2. Set the correct environment variables for the installation, as described in Table 3–1.
3. Execute commands in the Middleware home and Oracle home of the correct installation.

Multiple Installations by the Same User If you installed multiple installations as the same user, ensure that you are in the correct Middleware home and Oracle home and

have the correct environment variables set when working on a particular installation. You may want to set up some scripts to make it easy to change from one installation to another.

3.2 Overview of Oracle Fusion Middleware Administration Tools

After you install and configure Oracle Fusion Middleware, you can use the graphical user interfaces or command-line tools to manage your environment.

Oracle offers the following primary tools for managing your Oracle Fusion Middleware installations:

- Oracle Enterprise Manager Fusion Middleware Control. See Section 3.3.
- Oracle WebLogic Server Administration Console. See Section 3.4
- The Oracle Fusion Middleware command-line tools. See Section 3.5.
- The Fusion Middleware Control MBean Browser. See Section 3.6.

Note that you should use these tools, rather than directly editing configuration files, to perform all administrative tasks unless a specific procedure requires you to edit a file. Editing a file may cause the settings to be inconsistent and generate problems.

> **Note:** For information about using administration tools for IBM WebSphere, see "Summary of the Oracle Fusion Middleware Management Tools on IBM WebSphere" in the *Oracle Fusion Middleware Third-Party Application Server Guide*.

Both Fusion Middleware Control and Oracle WebLogic Server Administration Console are graphical user interfaces that you can use to monitor and administer your Oracle Fusion Middleware environment. You can install Fusion Middleware Control and the Administration Console when you install most Oracle Fusion Middleware components.

Note the following:

- If you install a standalone Oracle WebLogic Server Fusion Middleware Control is not installed, only the Administration Console is installed.
- If you install Oracle JDeveloper, neither Fusion Middleware Control or the Administration Console are installed. They can be installed if you install Oracle Fusion Middleware Application Developer.

You can perform some tasks with either tool, but for other tasks, you can only use one of the tools. Table 3–3 lists some common tasks and the recommended tool.

Table 3–3 Comparing Fusion Middleware Control and WebLogic Server Administration Console

Task	Tool to Use
Manage Oracle WebLogic Server	**Use:**
Create additional Managed Servers	WebLogic Server Administration Console
Clone Managed Servers	WebLogic Server Administration Console
Cluster Managed Servers	WebLogic Server Administration Console
Start and stop Oracle WebLogic Server	Fusion Middleware Control or WebLogic Server Administration Console

Table 3–3 (Cont.) Comparing Fusion Middleware Control and WebLogic Server Administration Console

Task	Tool to Use
Add users and groups	WebLogic Server Administration Console if using the default embedded LDAP; if using another LDAP server, use the LDAP server's tool
Manage Data Sources	**Use:**
Create data sources	WebLogic Server Administration Console
Create connection pools	WebLogic Server Administration Console
Manage JMS Resources	**Use:**
Create JMS queues	WebLogic Server Administration Console
Configure advanced queuing	WebLogic Server Administration Console
Manage SOA environment	**Use:**
Deploy SOA Composite applications	Fusion Middleware Control
Monitor SOA Composite applications	Fusion Middleware Control
Modify Oracle BPEL Process Manager MBean properties	Fusion Middleware Control
Debug applications such as Oracle BPEL Process Manager applications	Fusion Middleware Control
ADF Applications	**Use:**
Deploy ADF applications	Fusion Middleware Control
Java EE applications	**Use:**
Deploy Java EE applications	WebLogic Server Administration Console or Fusion Middleware Control
Security	**Use:**
Configure and manage auditing	Fusion Middleware Control
Configure SSL	WebLogic Server Administration Console for Oracle WebLogic Server Fusion Middleware Control for Java components and system components. See Chapter 6.
Change passwords	WebLogic Server Administration Console
Manage Components	**Use:**
View and manage log files	Fusion Middleware Control for most log files WebLogic Server Administration Console for the following logs: *DOMAIN_HOME*/servers/*server_name*/logs/access.log *DOMAIN_HOME*/servers/*server_name*/data/ldap/log/EmbeddedLDAP.log *DOMAIN_HOME*/servers/*server_name*/data/ldap/log/EmbeddedLDAPAccess.log
Change ports	WebLogic Server Administration Console for Oracle WebLogic Server and Java components For some system components, Fusion Middleware Control. See the Administration Guide for the component.

Table 3–3 (Cont.) Comparing Fusion Middleware Control and WebLogic Server Administration Console

Task	Tool to Use
Manage Oracle HTTP Server	Fusion Middleware Control
Manage Oracle Web Cache	Fusion Middleware Control
Start and stop components	Fusion Middleware Control
Start and stop applications	Fusion Middleware Control

3.3 Getting Started Using Oracle Enterprise Manager Fusion Middleware Control

Fusion Middleware Control is a Web browser-based, graphical user interface that you can use to monitor and administer a farm.

A **farm** is a collection of components managed by Fusion Middleware Control. It can contain an Oracle WebLogic Server domain, one Administration Server, one or more Managed Servers, clusters, one or more Oracle instances, and the Oracle Fusion Middleware components that are installed, configured, and running in the domain or Oracle instances.

Fusion Middleware Control organizes a wide variety of performance data and administrative functions into distinct, Web-based home pages for the farm, domain, servers, components, and applications. The Fusion Middleware Control home pages make it easy to locate the most important monitoring data and the most commonly used administrative functions—all from your Web browser.

The following topics are discussed in this section:

- Displaying Fusion Middleware Control
- Using Fusion Middleware Control Help
- Navigating Within Fusion Middleware Control
- Understanding Users and Roles for Fusion Middleware Control
- Viewing and Managing the Farm
- Viewing and Managing Components
- Viewing the Status of Applications

3.3.1 Displaying Fusion Middleware Control

To display Fusion Middleware Control, you enter the Fusion Middleware Control URL, which includes the name of the host and the administration port number assigned during the installation. The following shows the format of the URL:

```
http://hostname.domain:port/em
```

The port number is the port number of the Administration Server. By default, the port number is 7001. The port number is listed in the following file:

```
DOMAIN_HOME/config/config.xml
```

For some installation types, such as SOA or Web Tier, if you saved the installation information by clicking Save on the last installation screen, the URL for Fusion Middleware Control is included in the file that is written to disk (by default to your

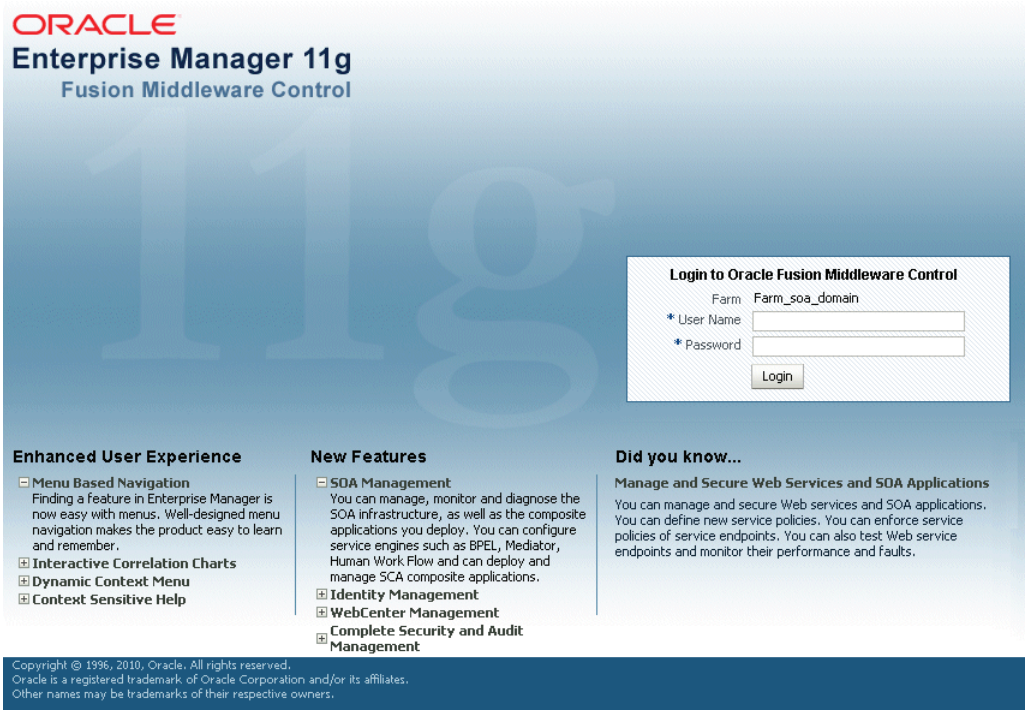
home directory). For other installation types, the information is displayed on the Create Domain screen of the Configuration Wizard when the configuration completes.

To display Fusion Middleware Control:

1. Enter the URL in your Web browser. For example:

   ```
   http://host1.example.com:7001/em
   ```

   The following shows the login page:



2. Enter the Oracle Fusion Middleware administrator user name and password and click **Login.**

   The default user name for the administrator user is `weblogic`. This is the account you can use to log in to Fusion Middleware Control for the first time. The password is the one you supplied during the installation of Oracle Fusion Middleware.

3.3.2 Using Fusion Middleware Control Help

At any time while using the Fusion Middleware Control Console, you can click **Help** at the top of the page to get more information. In most cases, the Help window displays a help topic about the current page. Click **Contents** in the Help window to browse the list of help topics, or click **Search** to search for a particular word or phrase.

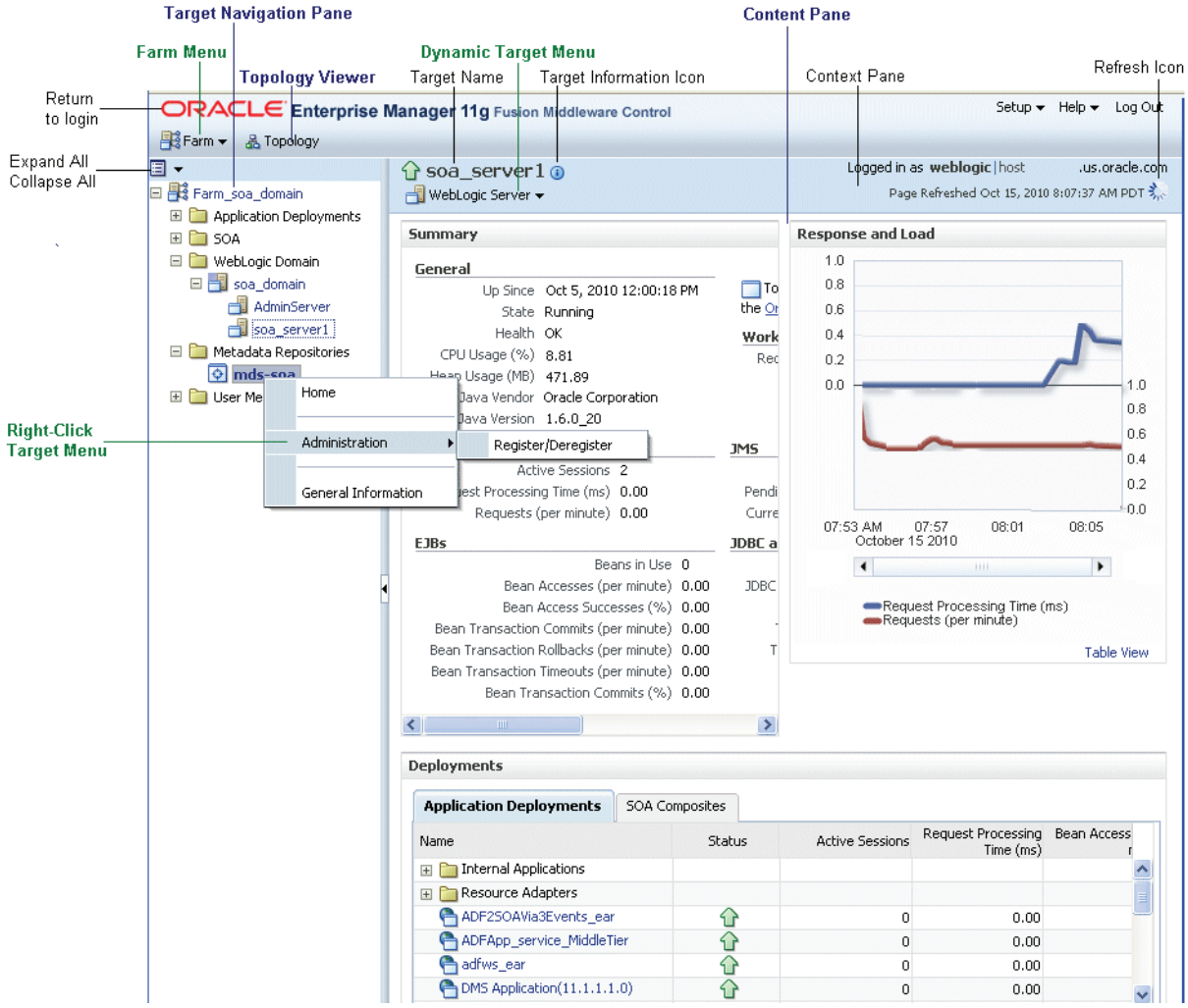
3.3.3 Navigating Within Fusion Middleware Control

Fusion Middleware Control displays the target navigation pane on the left and the content pane on the right. For example, when you first log in to Fusion Middleware Control, the farm home page is displayed on the right.

From the target navigation pane, you can expand the tree and select an Oracle WebLogic Server domain, an Oracle WebLogic Server Managed Server, a component, an application, or a Metadata Repository.

When you select a target, such as a Managed Server or a component, the target's home page is displayed in the content pane and that target's menu is displayed at the top of the page, in the context pane. For example, if you select a Managed Server, the WebLogic Server menu is displayed. You can also view the menu for a target by right-clicking the target in the navigation pane.

The following figure shows the target navigation pane and the home page of an Managed Server. Because a Managed Server was selected, the dynamic target menu listed in the context pane is the WebLogic Server menu.



In the preceding figure, the following items are called out:

- **Target Navigation Pane** lists all of the targets in the farm in a navigation tree.
- **Content Pane** shows the current page for the target. When you first select a target, that target's home page is displayed.
- **Farm Menu** provides a list of operations that you can perform on the farm. The Farm menu is always available.
- **Dynamic Target Menu** provides a list of operations that you can perform on the currently selected target. The menu that is displayed depends on the target you select. The menu for a specific target contains the same operations as those in the **Right-Click Target Menu.**

- **Right-Click Target Menu** provides a list of operations that you can perform on the currently selected target. The menu is displayed when you right-click the target name in the target navigation pane. In the figure, even though the WebLogic Server is selected and its home page is displayed, the right-click target menu displays the operations for a metadata repository because the user has right-clicked the metadata repository.

  The menu for a specific target contains the same operations as those in the **Dynamic Target Menu.**
- **Topology Viewer** displays the topology of the farm.
- **Target Name** is the name of the currently selected target.
- **Target Information Icon** provides information about the target. For example, for a domain, it displays the target name, the version, and the domain home.
- **Context Pane** provides the name of the target, the name of the current user, the host name, and the time of the last page refresh, as well as the Refresh icon.
- **Expand All/Collapse All** lets you expand or collapse the navigation tree.
- **Refresh** indicates when the page is being refreshed. Click it to refresh a page with new data. (Refreshing the browser window refreshes the page but does not retrieve new data.)
- **Return to login** takes you to the login page when you click the Oracle Enterprise Manager logo.

In addition, from Fusion Middleware Control, from the home pages of targets such as the Administration Server or Managed Servers, you can access the WebLogic Server Administration Console.

Table 3–4 describes some common ways you can navigate within Fusion Middleware Control.

Table 3–4 Navigating Within Fusion Middleware Control

To:	Take This Action:
View all of the targets in the farm	Click the **Expand All** icon at the top of the **target navigation pane**.
Navigate to the farm	Select the farm from the **target navigation pane**. The farm's home page is displayed in the content pane.
Operate on the farm	Select the **Farm menu**, which is always available at the top left of Fusion Middleware Control.
Operate on a target	Right-click the target in the **target navigation pane**. The **target menu** is displayed. Alternatively, you can select the target and use the **dynamic target menu** in the context pane.
Return to the target's home page	Click the **target name** at the top left-hand corner of the **context pane**.
Refresh a page with new data	Click the **Refresh** icon in the top right of the **context pane**.
Return to a previous page	Click the breadcrumbs, which appear below the context pane. The breadcrumbs appear when you drill down in a target. For example, choose Logs from the WebLogic Server menu, then View Log Messages. Select a log file and click View Log File. The breadcrumbs show: Log Messages > Log Files > View Log File: *logfile_name*

Table 3–4 (Cont.) Navigating Within Fusion Middleware Control

To:	Take This Action:
View the host on which the target is running	Select the target in the **target navigation pane** and view the host name in the target's **context pane**. You can also view the host name by clicking the Target Information icon.
Return to the login page	Click the **Oracle Enterprise Manager logo** at the top left of the page.
View the topology	Click **Topology.**
View a server log file	Right-click the server name in the **target navigation pane.** Choose **Logs,** and then **View Log Messages** to see a summary of log messages and to search log files.

3.3.4 Understanding Users and Roles for Fusion Middleware Control

To access Fusion Middleware Control and perform tasks, you must have the appropriate role. Fusion Middleware Control uses the Oracle WebLogic Server security realm and the roles defined in that realm. If a user is not granted one of these roles, the user cannot access Fusion Middleware Control.

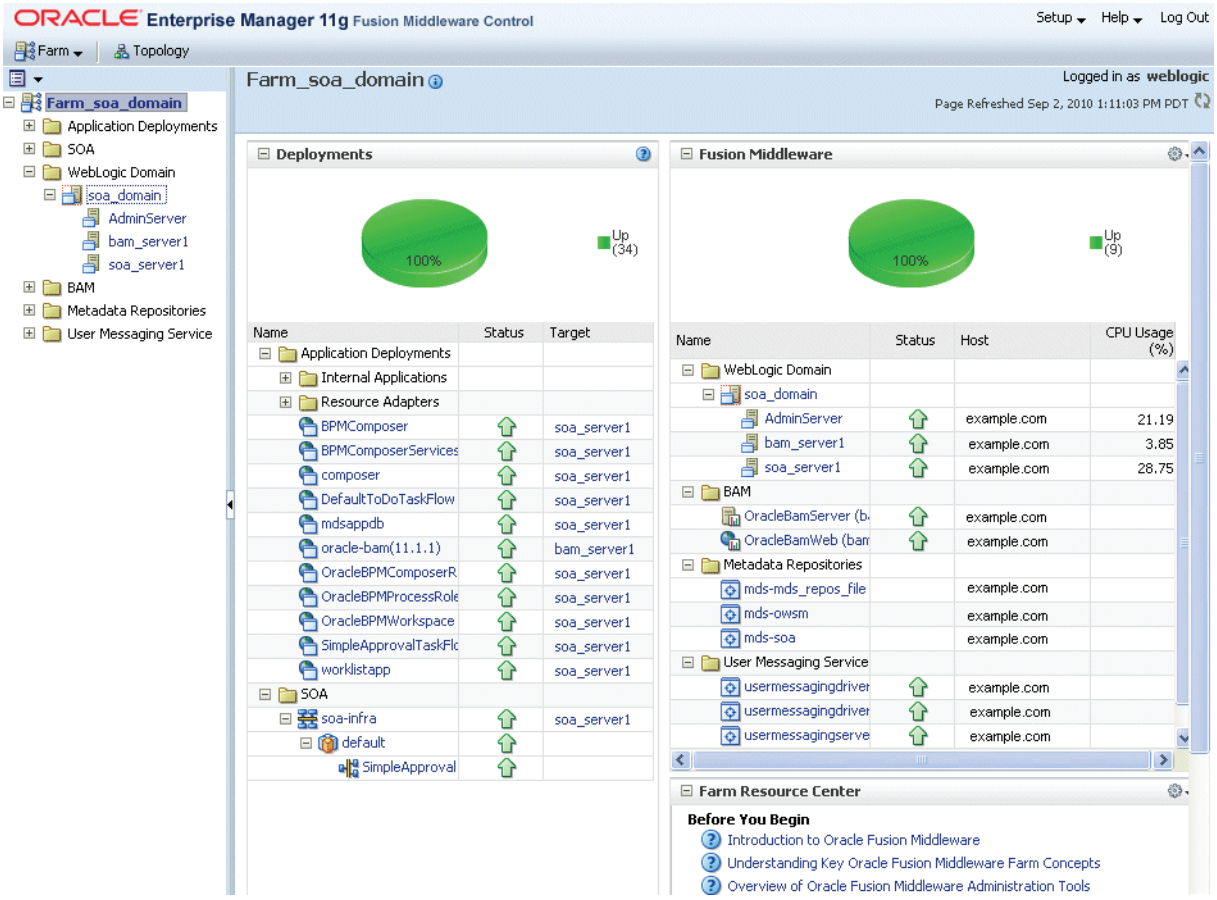
Each role defines the type of access a user has. For example, a user with the role Admin has full privileges. A user with the role Operator has privileges to perform essential day-to-day operations. A user with the role Monitor has privileges only to view the configuration.

> **See Also:** "Users, Groups, and Security Roles" in the *Oracle Fusion Middleware Securing Resources Using Roles and Policies for Oracle WebLogic Server*

3.3.5 Viewing and Managing the Farm

When you log in to Fusion Middleware Control, the first page you see is the Farm home page. You can also view this page at any time by selecting the farm in the target navigation pane.

The following figure shows the Farm home page:



The Farm menu is displayed at the top of the page. From the Farm menu, you can take the following actions:

- Create and delete components and create clusters
- View log messages.
- Specify monitoring credentials

The Farm menu is always displayed, even if you have selected other entities.

You can view the farm topology by selecting **Topology**. The Topology Viewer provides you with a high-level view of the topology, including Managed Servers, deployed applications, and the routing configuration. See Section 11.3 for information about using the Topology Viewer.

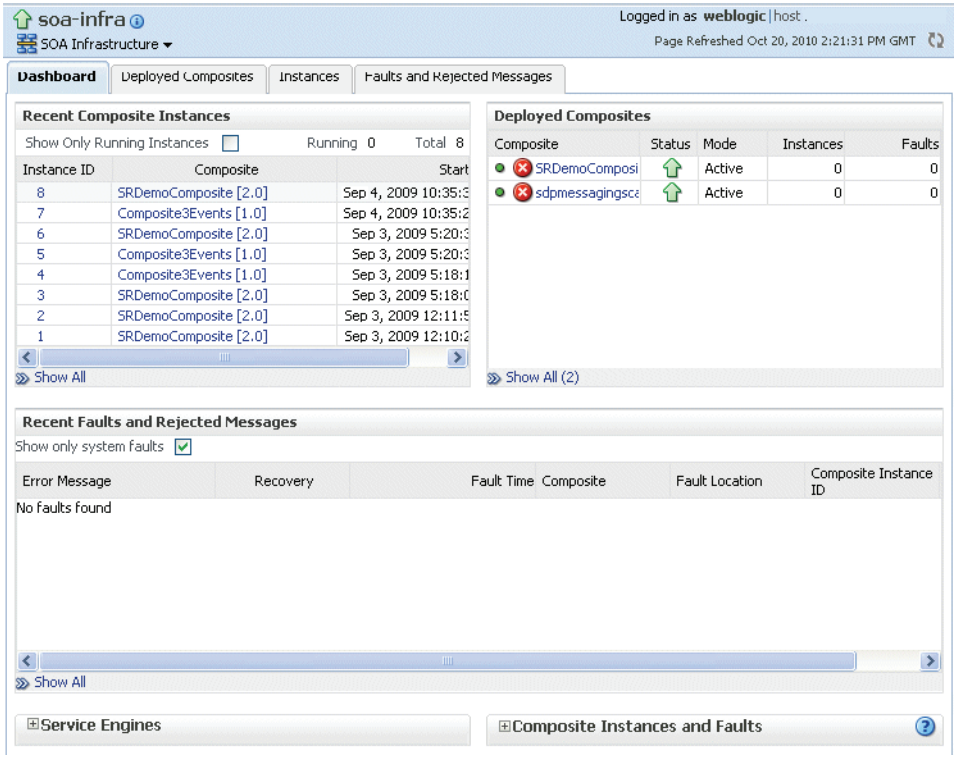
3.3.6 Viewing and Managing Components

From the target navigation pane, you can drill down to view and manage the components in your farm.

For example, to view and manage Oracle SOA Suite, take the following steps:

1. In the target navigation pane, expand the farm, then **SOA.**
2. Select the SOA instance.

   The home page for the SOA instance is displayed, as shown in the following figure:

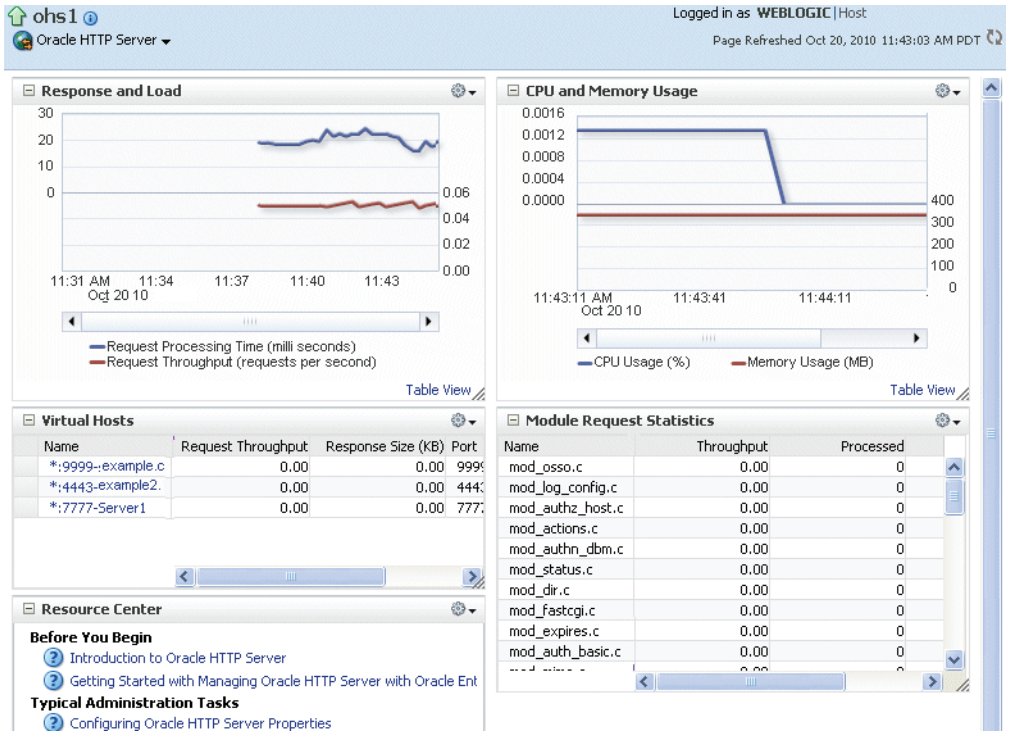


3. From the SOA Infrastructure menu, you can perform many administrative tasks, such as starting, stopping, and monitoring Oracle SOA Suite and deploying SOA composite applications.

As another example, to view and manage Oracle HTTP Server, take the following steps:

1. In the target navigation pane, expand the farm, then **Web Tier.**
2. Select the Oracle HTTP Server instance, for example, ohs1.

   The home page for the Oracle HTTP Server ohs1 is displayed, as shown in the following figure:



3. From the HTTP Server menu, you can perform many administrative tasks, such as starting, stopping, and monitoring Oracle HTTP Server.

   See Also: Section 11.1.5 for more information about monitoring components

3.3.7 Viewing the Status of Applications

From the target navigation pane, you can drill down to view and manage the applications in your farm.

To view Java EE applications:

1. From the target navigation pane, expand the farm and then **Application Deployments.**
2. Select the application that you want to view.

   The application's home page is displayed. In this page, you can view a summary of the application's status, entry points to the application, Web services and modules associated with the application, and the response and load.

To view SOA Composite Applications:

1. From the target navigation pane, expand the farm, then **SOA,** and then **soa-infra.**
2. Select the application that you want to view.

   The application's home page is displayed. It shows information about the application, such as the recent instances of the application, the faults and rejected messages and the policies.

3.4 Getting Started Using Oracle WebLogic Server Administration Console

Oracle WebLogic Server Administration Console is a Web browser-based, graphical user interface that you use to manage an Oracle WebLogic Server domain. It is accessible from any supported Web browser with network access to the Administration Server.

Use the Administration Console to:

- Configure, start, and stop WebLogic Server instances
- Configure WebLogic Server clusters
- Configure WebLogic Server services, such as database connectivity (JDBC) and messaging (JMS)
- Configure security parameters, including creating and managing users, groups, and roles
- Configure and deploy Java EE applications
- Monitor server and application performance
- View server and domain log files
- View application deployment descriptors
- Edit selected run-time application deployment descriptor elements

3.4.1 Displaying the Oracle WebLogic Server Administration Console

To display the Administration Console:

1. Enter the following URL in a browser:

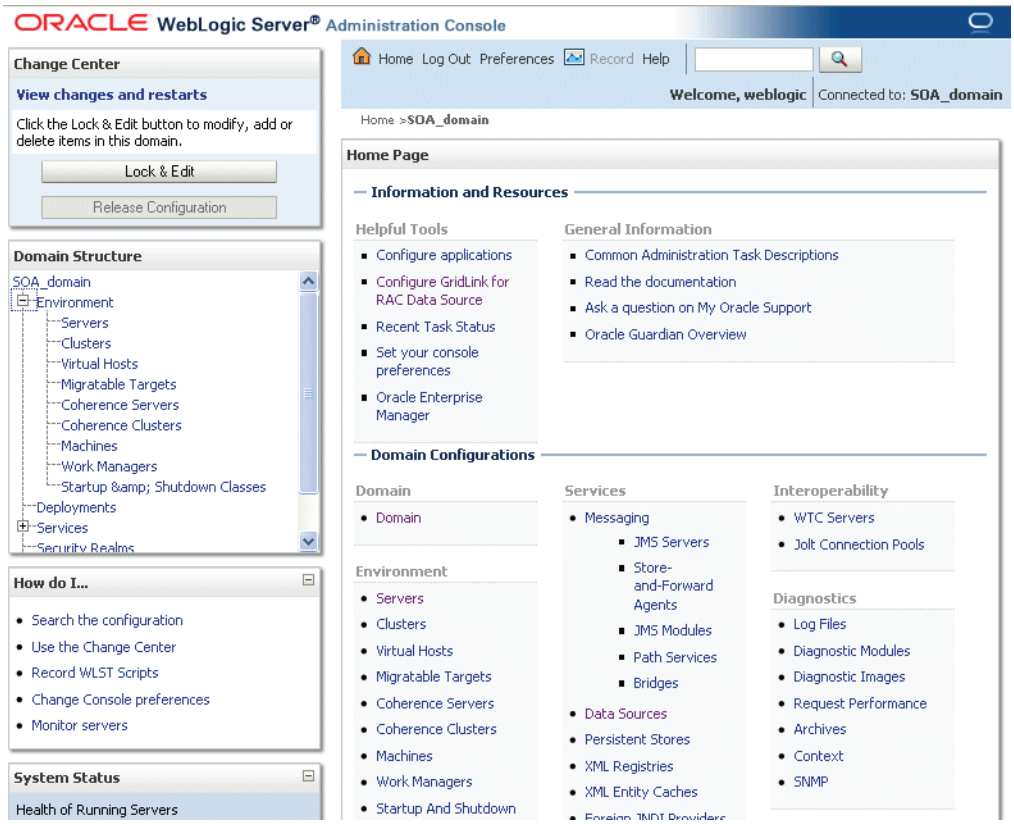
   `http://`*`hostname`*`:`*`port_number`*`/console`

   The port number is the port number of the Administration Server. By default, the port number is 7001.

   The login page is displayed.

2. Log in using the user name and password supplied during installation or another administrative user that you created.

   Oracle WebLogic Server Administration Console is displayed as shown in the following figure:



Alternatively, you can access the Administration Console from Fusion Middleware Control, from the home pages of targets such as the Administration Server or Managed Servers.

3.4.2 Locking the WebLogic Server Configuration

Before you make configuration changes, lock the domain configuration, so you can make changes to the configuration while preventing other accounts from making changes during your edit session. To lock the domain configuration:

1. Locate the Change Center in the upper left of the Administration Console screen.
2. Click **Lock & Edit** to lock the configuration edit hierarchy for the domain.

As you make configuration changes using the Administration Console, you click **Save** (or in some cases Finish) on the appropriate pages. This does not cause the changes to take effect immediately. The changes take effect when you click **Activate Changes** in the Change Center. At that point, the configuration changes are distributed to each of the servers in the domain. If the changes are acceptable to each of the servers, then they take effect. If any server cannot accept a change, then all of the changes are rolled back from all of the servers in the domain. The changes are left in a pending state; you can then either edit the pending changes to resolve the problem or revert to the previous configuration.

You can also lock the configuration by using the WLST command, startEdit:

```
startEdit()
```

For more information about the startEdit command and the stopEdit command, which releases locks, see "startEdit" and "stopEdit" in the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

3.5 Getting Started Using Command-Line Tools

The following topics describe the primary command-line tools you can use to manage most Oracle Fusion Middleware components:

- Getting Started Using the Oracle WebLogic Scripting Tool (WLST)
- Getting Started Using Oracle Process Manager and Notification Server

3.5.1 Getting Started Using the Oracle WebLogic Scripting Tool (WLST)

The Oracle WebLogic Scripting Tool (WLST) is a command-line scripting environment that you can use to create, manage, and monitor Oracle WebLogic Server domains. It is based on the Java scripting interpreter, Jython. In addition to supporting standard Jython features such as local variables, conditional variables, and flow-control statements, WLST provides a set of scripting functions (commands) that are specific to WebLogic Server. You can extend the WebLogic scripting language to suit your needs by following the Jython language syntax.

You can use WLST commands in the following ways:

- Interactively, on the command line
- In script mode, supplied in a file
- Embedded in Java code

For example, to invoke WLST interactively, and connect to the WebLogic Server, use the following commands:

```
java weblogic.WLST
connect('username', 'password', 'localhost:7001')
```

To display information about WLST commands and variables, enter the help command. For example, to display a list of categories for online commands, enter the following:

```
wls:/base_domain/serverConfig>  help('online')
    help('activate')            Activate the changes.
    help('addListener')         Add a JMX listener to the specified MBean.
    help('adminHome')           Administration MBeanHome.
    help('cancelEdit')          Cancel an edit session.
    help('cd')                  Navigate the hierarchy of beans.
    help('cmo')                 Current Management Object.
 .
 .
 .
```

To monitor the status, you use the WLST `state` command, using the following format:

```
state(name, type)
```

For example to get the status of the Managed Server soa_server1, use the following command:

```
wls:/SOA_domain/serverConfig> state('soa_server1', 'Server')
Current state of 'soa_server1' : RUNNING
```

> **See Also:** *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*

3.5.1.1 Using Custom WLST Commands

Many components, such as Oracle SOA Suite, Oracle Platform Security Services (OPSS), Oracle Fusion Middleware Audit Framework, and MDS, and services such as SSL and logging, provide custom WLST commands.

To use those custom commands, you must invoke the WLST script from the appropriate Oracle home. Do not use the WLST script in the WebLogic Server home.

- For the following components and services, invoke WLST from the Oracle Common home:
  - Oracle Application Development Framework
  - Oracle Fusion Middleware Audit Framework
  - Oracle Access Manager
  - Oracle Platform Security Services
  - Oracle Metadata Services
  - Diagnostic Framework
  - Dynamic Monitoring Service (DMS)
  - Logging
  - Secure Sockets Layer (SSL)
  - Oracle JRF
  - Oracle Web Services
  - Oracle Web Services Manager

  The script is located at:

  ```
  (UNIX) ORACLE_COMMON_HOME/common/bin/wlst.sh
  (Windows) ORACLE_COMMON_HOME\common\bin\wlst.cmd
  ```

- For other components, such as Oracle HTTP Server, Oracle SOA Suite, or Oracle WebCenter Portal, invoke WLST from the Oracle home in which the component has been installed. The script is located at:

  ```
  (UNIX) ORACLE_HOME_for_component/common/bin/wlst.sh
  (Windows) ORACLE_HOME_for_component\common\bin\wlst.cmd
  ```

  For example, to run the custom WLST commands for Oracle SOA Suite on a Linux system, use the following commands:

  ```
  cd ORACLE_HOME_for_SOA/common/bin
  ./wlst.sh
  ```

3.5.1.2 Using WLST Commands for System Components

In addition to the commands provided by WLST for Oracle WebLogic Server, WLST provides a subset of commands to manage system components. These commands are:

- startproc(*componentName* [, *componentType*] [, *componentSet*): Starts the specified component.
- stopproc(*componentName* [, *componentType*] [, *componentSet*): Stops the specified component.
- status(*componentName* [, *componentType*] [, *componentSet*): Obtains the status of the specified component.

- proclist(): Obtains the list of components.

To use these custom commands, you must invoke the WLST script from the Oracle home in which the component has been installed. Do not use the WLST script in the WebLogic Server home. The script is located at:

```
(UNIX) ORACLE_HOME_for_component/common/bin/wlst.sh
(Windows) ORACLE_HOME_for_component\common\bin\wlst.cmd
```

3.5.2 Getting Started Using Oracle Process Manager and Notification Server

Oracle Process Manager and Notification Server (OPMN) manages and monitors the following Oracle Fusion Middleware components, referred to as system components:

- Oracle HTTP Server
- Oracle Web Cache
- Oracle Internet Directory
- Oracle Virtual Directory
- Oracle Forms Services
- Oracle Reports
- Oracle Business Intelligence Discoverer
- Oracle Business Intelligence

OPMN provides the `opmnctl` command. The executable file is located in the following directories:

- *ORACLE_HOME*/opmn/bin/opmnctl: The opmnctl command from this location should be used only to create an Oracle instance or a component for an Oracle instance on the local system. Any opmnctl commands generated from this location should not be used to manage system processes or to start OPMN.

  On Windows, if you start OPMN using the opmnctl start command from this location, OPMN and its processes terminate when the Windows user has logged out.

- *ORACLE_INSTANCE*/bin/opmnctl: The opmnctl command from this location provides a per Oracle instance instantiation of opmnctl. Use opmnctl commands from this location to manage processes for this Oracle instance. You can also use this opmnctl to create components for the Oracle instance.

  On Windows, if you start OPMN using the opmnctl start command from this location, it starts OPMN as a Windows service. As a result, the OPMN parent process, and the processes which it manages, persist after the MS Windows user has logged out.

To view the status of all system components in an Oracle instance, use the following command:

```
opmnctl status
Processes in Instance: webtier_inst
---------------------------------+--------------------+---------+---------
ias-component                    | process-type       |     pid | status
---------------------------------+--------------------+---------+---------
webcache1                        | WebCache-admin     |   19556 | Alive
webcache1                        | WebCache           |   19555 | Alive
ohs1                             | OHS                |    7249 | Alive
```

To view the status of a particular component or component type, use the following command:

```
opmnctl status componentName [, componentType] [, componentSet]
```

For example, to view the status of an Oracle Virtual Directory instance named ovd1, use the following command:

```
opmnctl status ias-component=ovd1
```

You can use OPMN to start and stop system components, monitor system components, and perform many other tasks related to process management. For example, you can use the following commands to start and stop OPMN and all OPMN-managed processes, such as Oracle HTTP Server and Oracle Web Cache:

```
opmnctl startall
opmnctl stopall
```

To start a component, use the following command:

```
opmnctl startproc componentName [, componentType] [, componentSet
```

For example, to start an Oracle HTTP Server instance named ohs1, use the following command:

```
opmnctl startproc ias-component=ohs1
```

See Also:

- Chapter 4 for information about starting and stopping your Oracle Fusion Middleware environment
- Chapter 11 for more information about monitoring your Oracle Fusion Middleware environment
- *Oracle Fusion Middleware Oracle Process Manager and Notification Server Administrator's Guide*

3.6 Getting Started Using the Fusion Middleware Control MBean Browsers

A **managed bean** (MBean) is a Java object that represents a JMX manageable resource in a distributed environment, such as an application, a service, a component or a device.

MBeans are defined in the Java EE Management Specification (JSR-77), which is part of Java Management Extensions, or JMX, a set of specifications that allow standard interfaces to be created for managing applications in a Java EE environment. For information about JSR-77, see:

http://java.sun.com/j2ee/tools/management/

You can create MBeans for deployment with an application into Oracle WebLogic Server, enabling the application or its components to be managed and monitored through Fusion Middleware Control.

See Also: "Understanding WebLogic Server MBeans" in the *Oracle Fusion Middleware Developing Custom Management Utilities With JMX for Oracle WebLogic Server*

Fusion Middleware Control provides a set of MBean browsers that allow to you browse the MBeans for an Oracle WebLogic Server or for a selected application. You can also perform specific monitoring and configuration tasks from the MBean browser.

The MBeans are organized into three groups: Configuration MBeans, Runtime MBeans, and Application-Defined MBeans.

The following topics describe how to view or configure MBeans:

- Using the System MBean Browser
- Using the MBeans for a Selected Application

3.6.1 Using the System MBean Browser

You can view the System MBean Browser for many entities, including an Oracle WebLogic Server domain, an Administration Server, a Managed Server, or an application. You can search for an MBean, filter the list of MBeans, and refresh the list of MBeans in the MBean navigation tree.

To view the System MBean Browser specific to a particular Oracle WebLogic Server Managed Server and to configure and use the MBeans:

1. From the target navigation pane, expand the farm, then **WebLogic Domain,** and then the domain.
2. Select the Managed Server.
3. From the WebLogic Server menu, choose **System MBean Browser.**

   The System MBean Browser page is displayed.
4. Expand a node in the MBean navigation tree and drill down to the MBean you want to access. Select an MBean instance.

   If you do not know the location of an MBean, you can search for the MBean:

   a. Click the Find icon at the top of the MBean navigation tree.

   b. For **Find,** select **MBean Name.**

      You can also select Attributes, Operations, or JMX syntax.

   c. Enter the name of the MBean and click the arrow.
5. To view the MBean's attributes, select the Attributes tab. Some attributes allow you to change their values. To do so, enter the value in the **Value** column.
6. To view the available operations, select the Operations tab. To perform an operation, click the operation. The Operations page appears. Enter any applicable values and click **Invoke.**

See Also: The Fusion Middleware Control online help

3.6.2 Using the MBeans for a Selected Application

You can view, configure, and use the MBeans for a specific application by taking the steps described in Section 3.6.1, and drilling down to the application. As an alternative, you can navigate to an application's MBeans using the following steps:

1. From the target navigation pane, expand the farm, then **Application Deployments.**
2. Select the application.
3. From the Application Deployments menu, choose **System MBean Browser.**

The System MBean Browser page is displayed, along with the MBean information for the application.

4. To view the MBean's attributes, select the Attributes tab. Some attributes allow you to change their values. To do so, enter the value in the **Value** column.
5. To view the available operations, select the Operations tab. To perform an operation, click the operation. The Operations page appears. Enter any applicable values and click **Invoke.**

3.7 Managing Components

Oracle Fusion Middleware components include Oracle WebLogic Server, Java components that are part of Oracle SOA Suite and WebCenter Portal, such as Oracle BPEL Process Manager or Oracle Business Activity Monitoring, and system components such as Oracle Web Cache.

To manage the Oracle WebLogic Server and Java components, you can use WLST, Oracle WebLogic Server Administration Console, or Fusion Middleware Control.

To manage system components, you can use OPMN, WLST, or Fusion Middleware Control.

See:

- *Oracle Fusion Middleware Installation Planning Guide* and the individual installation guides for information about installing and configuring components
- *Oracle Fusion Middleware Installation Guide for Oracle WebLogic Server* for installing and configuring Oracle WebLogic Server
- The administration guide for each component or suite for more information about managing these components.

3.8 Changing the Administrative User Password

During the Oracle Fusion Middleware installation, you must specify a password for the administration account. Then, you can use this account to log in to Fusion Middleware Control and the Oracle WebLogic Server Administration Console for the first time. You can create additional administrative accounts using the WLST command line or the Oracle WebLogic Server Administration Console.

See Also: "Understanding Users and Roles" in the *Oracle Fusion Middleware Application Security Guide*

You can change the password of the administrative user using the Oracle WebLogic Server Administration Console or the WLST command line.

3.8.1 Changing the Administrative User Password Using the Command Line

To change the administrative user password or other user passwords using the command line, you invoke the UserPasswordEditorMBean.changeUserPassword method, which is extended by the security realm's AuthenticationProvider MBean.

For more information, see the changeUserPassword method in the *Oracle Fusion Middleware Oracle WebLogic Server MBean Reference*.

3.8.2 Changing the Administrative User Password Using the Administration Console

To change the password of an administrative user using the Oracle WebLogic Server Administration Console:

1. Navigate to the Oracle WebLogic Server Administration Console. (For example, from the home page of the domain in Fusion Middleware Control, select **To configure and managed this WebLogic Domain, use the Oracle WebLogic Server Administration Console.**)
2. From the Domain Structure pane, select **Security Realms.**

   The Summary of Security Realms page is displayed.
3. Select a realm, such as **myrealm.**

   The Settings for the realm page is displayed.
4. Select the Users and Groups tab, then the Users tab. Select the user.

   The Settings for *user* page is displayed.
5. Select the Passwords tab.
6. Enter the new password, then enter it again to confirm it.
7. Click **Save.**
8. If your environment includes components that are managed by Oracle Management Agent, you must update the targets.xml file, which is located at:

   ```
   ORACLE_INSTANCE/EMAGENT/emagent_<instanceName>/sysman/emd/targets.xml
   ```

   Take the following steps:

   a. Back up the targets.xml file.

   b. Identify all the targets that need to be updated with the new password. Then, in the target.xml file, for each target, set the WeblogicPassword property to:

   ```
   <Property NAME="WeblogicPassword" VALUE="new_password"
   ENCRYPTED="FALSE"/>
   ```

   c. Restart the Oracle Management Agent, as described in Section 4.5.

3.9 Basic Tasks for Configuring and Managing Oracle Fusion Middleware

The following provides a summary of the steps you need to take to configure and manage a basic Oracle Fusion Middleware environment after you have installed the software:

1. Configure Oracle WebLogic Server and components, such as Oracle SOA Suite, Oracle HTTP Server, or Oracle Web Cache. See *Oracle Fusion Middleware Installation Planning Guide*.
2. Configure SSL. See Chapter 6.
3. Create and manage metadata repositories, including the MDS Repository. See Section 14.2.
4. Deploy an application. See Chapter 10.
5. Configure load balancing. You can configure load balancing between different components or applications. See the *Oracle Fusion Middleware High Availability Guide*.

6. Back up your environment. See Chapter 16.
7. Monitor your environment and manage log files. See Chapter 11 and Chapter 12.
8. Expand your environment. See Chapter 19.

This guide also describes other tasks that you may need to perform, depending on your Oracle Fusion Middleware environment.

4

Starting and Stopping Oracle Fusion Middleware

This chapter describes procedures for starting and stopping Oracle Fusion Middleware, including the Administration Server, Managed Servers, and components.

It contains the following topics:

4.1 Overview of Starting and Stopping Procedures

Oracle Fusion Middleware is a flexible product that you can start and stop in different ways, depending on your requirements. In most situations, you can use Fusion Middleware Control, Oracle WebLogic Server Administration Console, or the WLST or OPMN commands to start or stop Oracle Fusion Middleware components.

These tools are completely compatible and, in most cases, can be used interchangeably. For example, you can start a J2EE component using WLST and stop it using Fusion Middleware Control.

> **Note:** For information about starting and stopping servers for IBM WebSphere, see "Starting and Stopping Servers on IBM WebSphere" in the *Oracle Fusion Middleware Third-Party Application Server Guide*.

4.2 Starting and Stopping Oracle WebLogic Server Instances

You can start Oracle WebLogic Server Administration Servers using the WLST command line. You can start and stop Managed Servers using scripts, the WLST command line, the WebLogic Server Administration Console, or Fusion Middleware Control. The following sections describe how to start and stop WebLogic Servers using the WLST command line, Fusion Middleware Control, or both:

4.2.1 Configuring Node Manager to Start Managed Servers

If a Managed Server contains other Oracle Fusion Middleware products, such as Oracle SOA Suite, Oracle WebCenter Portal, or Oracle JRF, the Managed Servers environment must be configured to set the correct classpath and parameters. This environment information is provided through the start scripts, such as startWebLogic and setDomainEnv, which are located in the following directory:

DOMAIN_HOME/bin

If the Managed Servers are started by Node Manager (as is the case when the servers are started by the Oracle WebLogic Server Administration Console or Fusion Middleware Control), Node Manager must be instructed to use these start scripts so that the server environments are correctly configured. Specifically, Node Manager must be started with the property `StartScriptEnabled=true`.

There are several ways to ensure that Node Manager starts with this property enabled. As a convenience, Oracle Fusion Middleware provides the following script, which adds the property `StartScriptEnabled=true` to the nodemanager.properties file:

(UNIX) *ORACLE_COMMON_HOME*/common/bin/setNMProps.sh.
(Windows) *ORACLE_COMMON_HOME*\common\bin\setNMProps.cmd

For example, on Linux, execute the setNMProps script and start Node Manager:

ORACLE_COMMON_HOME/common/bin/setNMProps.sh
MW_HOME/wlserver_*n*/server/bin/startNodeManager.sh

When you start Node Manager, it reads the nodemanager.properties file with the `StartScriptEnabled=true` property, and uses the start scripts when it subsequently starts Managed Servers. Note that you need to run the setNMProps script only once.

Also note that when the StartScriptEnable property is set to true, the Node Manager reads the startWebLogic script, which in turns reads the setDomainEnv script. As a result, you must make any tuning changes by editing the setDomainEnv script. Any changes that are performed using the command line or Administration Console will not be implemented when Node Manager starts the servers. For example, if you use the Administration Console to change the server start arguments, those changes are written to config.xml, but the Node Manager ignores these settings and uses those in setDomainEnv.

> **See Also:** "Using Node Manager" in the *Oracle Fusion Middleware Node Manager Administrator's Guide for Oracle WebLogic Server* for other methods of configuring and starting Node Manager

4.2.2 Starting and Stopping Administration Servers

You can start and stop Oracle WebLogic Server Administration Servers using the WLST command line or a script. When you start or stop the Administration Server,

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

For example, to start an Administration Server, use the following script:

```
MW_HOME/user_projects/domains/domain_name/bin/startWebLogic.sh
         -Dweblogic.management.username=weblogic
         -Dweblogic.management.password=password
         -Dweblogic.system.StoreBootIdentity=true
```

To stop an Administration Server, use the following script:

```
MW_HOME/user_projects/domains/domain_name/bin/stopWebLogic.sh
       username password [admin_url]
```

4.2.3 Starting and Stopping Managed Servers

You can start and stop Managed Servers using Fusion Middleware Control or WLST commands and scripts, as described in the following topics:

- Starting and Stopping Managed Servers Using Fusion Middleware Control
- Starting and Stopping Managed Servers Using WLST

4.2.3.1 Starting and Stopping Managed Servers Using Fusion Middleware Control

Fusion Middleware Control and the Oracle WebLogic Server Administration Console use Node Manager to start Managed Servers. If you are starting a Managed Server that does not contain Oracle Fusion Middleware products other than Oracle WebLogic Server, you can start the servers using the procedure in this section.

However, if the Managed Server contains other Oracle Fusion Middleware products, such as Oracle SOA Suite, Oracle WebCenter Portal, or Oracle JRF, you must first configure Node Manager, as described in Section 4.2.1.

To start or stop a WebLogic Server Managed Server using Fusion Middleware Control:

1. From the navigation pane, expand the farm, then **WebLogic Domain,** and then the domain.
2. Select the Managed Server.
3. From the WebLogic Server menu, choose **Control,** then **Start Up** or **Shut Down.**

Alternatively, you can right-click the server, then choose **Control,** then **Start Up** or **Shut Down.**

4.2.3.2 Starting and Stopping Managed Servers Using WLST

You can use a script or WLST to start and stop a WebLogic Server Managed Server.

For example, to start a WebLogic Server Managed Server, use the following script:

```
(UNIX) MW_HOME/user_projects/domains/domain_name/bin/startManagedWebLogic.sh
          managed_server_name admin_url
(Windows) MW_HOME\user_projects\domains\domain_name\bin\startManagedWebLogic.cmd
          managed_server_name admin_url
```

When prompted, enter your user name and password.

To stop a WebLogic Server Managed Server, use the following script:

```
(UNIX) MW_HOME/user_projects/domains/domain_name/bin/stopManagedWebLogic.sh
           managed_server_name admin_url username password
(Windows) MW_HOME\user_projects\domains\domain_name\bin\stopManagedWebLogic.cmd
```

```
managed_server_name admin_url username password
```

4.2.4 Enabling Servers to Start Without Supplying Credentials

You can enable the Administration Server and Managed Servers to start without prompting you for the administrator user name and password.

1. For the Administration Server, create a boot.properties file:
   a. Create the following directory:

      ```
      MW_HOME/user_
      projects/domains/domain_name/servers/AdminServer/security
      ```

   b. Use a text editor to create a file called boot.properties in the security directory created in the previous step, and enter the following lines in the file:

      ```
      username=adminuser
      password=password
      ```

2. For each Managed Server:
   a. Create the following directory:

      ```
      MW_HOME/user_
      projects/domains/domain_name/servers/server_name/security
      ```

   b. Copy the boot.properties file you created for the Administration Server to the security directory you created in the previous step.
3. Restart the Administration Server and Managed Servers, as described in Section 4.2.2 and Section 4.2.3.

Note: When you start the Administration Server or Managed Server, the user name and password entries in the file are encrypted.

For security reasons, minimize the time the entries in the file are left unencrypted. After you edit the file, start the server as soon as possible in order for the entries to be encrypted.

See Also: "Boot Identity Files" in the *Oracle Fusion Middleware Managing Server Startup and Shutdown for Oracle WebLogic Server* for more information

4.2.5 Setting Up an Oracle WebLogic Server as a Windows Service

If you want a WebLogic Server instance to start automatically when you boot a Windows host computer, you can set up the server as a Windows service. For complete information, see "Setting Up a WebLogic Server Instance as a Windows Service" in the *Oracle Fusion Middleware Managing Server Startup and Shutdown for Oracle WebLogic Server*.

However, that chapter describes the process for a standalone Oracle WebLogic Server installation. When Oracle WebLogic Server is part of an Oracle Fusion Middleware environment, you must set the environment to include references to *ORACLE_COMMON*. To do that, the script that you create is slightly different from that in "Example Script for Setting Up a Server as a Windows Service". The following shows the correct script:

```
echo off
```

```
SETLOCAL
set DOMAIN_NAME=myWLSdomain
set USERDOMAIN_HOME=d:\Oracle\Middleware\user_projects\domains\myWLSdomain
set SERVER_NAME=myWLSserver
set PRODUCTION_MODE=true
set
JAVA_OPTIONS=-Dweblogic.Stdout="d:\Oracle\Middleware\user_
projects\domains\myWLSdomain\
stdout.txt" -Dweblogic.Stderr="d:\Oracle\Middleware\user_
projects\domains\myWLSdomain\
stderr.txt"
set ADMIN_URL=http://adminserver:7501
set MEM_ARGS=-Xms40m -Xmx250m
call %USERDOMAIN_HOME%\bin\setDomainEnv.cmd
call "d:\Oracle\Middleware\wlserver_10.3\server\bin\installSvc.cmd"
ENDLOCAL
```

4.3 Starting and Stopping Components

You can start and stop components using the command line, the WebLogic Server Administration Console, or Fusion Middleware Control, depending upon the component. The following topics describe how to start and stop components using Fusion Middleware Control and the command line:

- Starting and Stopping Components Using Fusion Middleware Control
- Starting and Stopping Components Using the Command Line

4.3.1 Starting and Stopping Components Using Fusion Middleware Control

To start or stop a component:

1. From the navigation pane, expand the farm, then navigate to the component.
2. Select the component, such as **SoaInfra.**
3. From the dynamic target menu, choose **Control,** then **Start Up** or **Shut Down.**

Note: If OPMN is not started, you cannot start system components such as Oracle HTTP Server or Oracle Internet Directory using Fusion Middleware Control. To start OPMN, use the following command:

```
opmnctl start
```

4.3.2 Starting and Stopping Components Using the Command Line

If a component is a Java component, you use WLST commands to start and stop the component. If a component is a system component, you use `opmnctl` commands to start and stop the components.

- To start and stop Java components, use the WLST `startApplication` and `stopApplication` commands:

  ```
  startApplication(appName, [options])
  stopApplication(appName, [options])
  ```

  For example, to start Oracle Directory Integration Platform, use the following command:

  ```
  startApplication("DIP")
  ```

- To start and stop system components, use the `opmnctl` command-line tool. It is located in the following directory:

  ```
  (UNIX) ORACLE_INSTANCE/bin
  (Windows) ORACLE_INSTANCE\bin
  ```

  To start or stop OPMN and all system processes, such as Oracle HTTP Server:

  ```
  opmnctl startall
  opmnctl stopall
  ```

  To start, stop, or restart a component using opmnctl:

  ```
  opmnctl startproc ias-component=component_name
  opmnctl stopproc ias-component=component_name
  opmnctl restartproc ias-component=component_name
  ```

  For example, to start Oracle HTTP Server, ohs1:

  ```
  opmnctl startproc ias-component=ohs1
  ```

  To start, stop, or restart the subprocess of a component:

  ```
  opmnctl stopproc process-type=process
  opmnctl startproc process-type=process
  opmnctl restartproc process-type=process
  ```

4.4 Starting and Stopping Fusion Middleware Control

If Fusion Middleware Control is configured for a domain, it is automatically started or stopped when you start or stop an Oracle WebLogic Server Administration Server, as described in Section 4.2.2.

4.5 Starting and Stopping Oracle Management Agent

Oracle Management Agent is designed specifically for monitoring particular Oracle Fusion Middleware components.

To start Oracle Management Agent, use the following command:

```
opmnctl startproc ias-component=EMAGENT
```

To stop Oracle Management Agent, use the following command:

```
opmnctl stopproc ias-component=EMAGENT
```

4.6 Starting and Stopping Applications

You can start and stop applications using Fusion Middleware Control, the WebLogic Server Administration Console, or the WLST command line. The following topics describe how to start and stop applications using Fusion Middleware Control and the command line:

- Starting and Stopping Java EE Applications Using Fusion Middleware Control
- Starting and Stopping Java EE Applications Using WLST

4.6.1 Starting and Stopping Java EE Applications Using Fusion Middleware Control

To start or stop a Java EE application using Fusion Middleware Control:

1. From the navigation pane, expand **Application Deployments.**
2. Select the application.
3. From the Application Deployment menu, choose **Control,** then **Start Up** or **Shut Down.**

To start or stop a SOA Composite application using Fusion Middleware Control:

1. From the navigation pane, expand the farm, then **SOA,** and then **soa-infra.**
2. Select the application.
3. On the SOA Composite page, click **Start Up** or **Shut Down.**

4.6.2 Starting and Stopping Java EE Applications Using WLST

To start or stop a Java EE application with the WLST command line, use the following commands:

```
startApplication(appName, [options])
stopApplication(appName, [options])
```

The application must be fully configured and available in the domain. The `startApplication` command returns a WLSTProgress object that you can access to check the status of the command. In the event of an error, the command returns a WLSTException. For more information about the WLSTProgress object, see "WLSTProgress Object" in the *Oracle Fusion Middleware Oracle WebLogic Scripting Tool*.

4.7 Starting and Stopping Your Oracle Fusion Middleware Environment

This section provides procedures for starting and stopping an Oracle Fusion Middleware environment. An environment can consist of an Oracle WebLogic Server domain, an Administration Server, multiple Managed Servers, Java components, system components, including Identity Management components, and a database used as a repository for metadata. The components may be dependent on each other. Therefore, it is important to start and stop them in the proper order.

You can follow these procedures when you need to completely shut down your Oracle Fusion Middleware environment. For example, when preparing to perform a complete backup of your environment, or apply a patch.

4.7.1 Starting an Oracle Fusion Middleware Environment

To start an Oracle Fusion Middleware environment:

1. Start any database-based repository:
   a. Set the ORACLE_HOME environment variable to the Oracle home for the database.
   b. Set the ORACLE_SID environment variable to the SID for the database (default is `orcl`.)
   c. Start the Net Listener:

      ```
      ORACLE_HOME/bin/lsnrctl start
      ```

   d. Start the database instance:

      ```
      ORACLE_HOME/bin/sqlplus /nolog
      SQL> connect SYS as SYSDBA
      SQL> startup
      ```

```
SQL> quit
```

2. Start the Oracle WebLogic Server Administration Server as described in Section 4.2.2.
3. If you have not already done so, configure Node Manager, as described in Section 4.2.1.
4. Ensure Node Manager is running. If Node Manager is not running, start it by executing the following command:

   ```
   MW_HOME/user_projects/domains/DOMAIN_NAME/bin/startNodeManager.sh
   ```

5. Start Oracle Identity Management system components:
   a. Set the ORACLE_HOME environment variable to the Oracle home and ORACLE_INSTANCE environment variables for the Identity Management components.
   b. Start OPMN and all system components:

      ```
      (UNIX) ORACLE_INSTANCE/bin/opmnctl startall
      (Windows) ORACLE_INSTANCE\bin\opmnctl startall
      ```

6. Start the Oracle WebLogic Server Managed Servers as described in Section 4.2.3.2. Any applications deployed to the server are also started.
7. Start OPMN and all other system components, such as Oracle HTTP Server.
   a. Set the ORACLE_HOME and ORACLE_INSTANCE environment variables to the Oracle home and Oracle instance for the system components.
   b. Start OPMN and all system components in that Oracle instance:

      ```
      (UNIX) ORACLE_INSTANCE/bin/opmnctl startall
      (Windows) ORACLE_INSTANCE\bin\opmnctl startall
      ```

8. If your environment includes components that are targets monitored by Oracle Management Agent, start Oracle Management Agent, as described in Section 4.5.

4.7.2 Stopping an Oracle Fusion Middleware Environment

To stop an Oracle Fusion Middleware environment:

1. Stop system components, such as Oracle HTTP Server. You can stop them in any order.
   a. Set the ORACLE_HOME and ORACLE_INSTANCE environment variables to the Oracle home and Oracle instance for the system components.
   b. Stop OPMN and all system components in that Oracle instance:

      ```
      (UNIX) ORACLE_INSTANCE/bin/opmnctl stopall
      (Windows) ORACLE_INSTANCE\bin\opmnctl stoptall
      ```

2. If your environment includes components that are targets monitored by Oracle Management Agent, stop Oracle Management Agent, as described in Section 4.5.
3. Stop the Oracle WebLogic Server Managed Servers, as described in Section 4.2. Any applications deployed to the server are also stopped.
4. Stop Oracle Identity Management components:
   a. Set the ORACLE_HOME environment variable to the Oracle home for the Identity Management components.

b. Stop OPMN and all system components:

```
(UNIX) ORACLE_INSTANCE/bin/opmnctl stopall
(Windows) ORACLE_INSTANCE\bin\opmnctl stoptall
```

5. Stop the Administration Server as described in Section 4.2.2.
6. If you want to stop Node Manager, you can use the kill command:

```
kill -9 PID
```

7. Stop any database-based repository:
   a. Set the ORACLE_HOME environment variable to the Oracle home for the database.
   b. Set the ORACLE_SID environment variable to the SID for the database (default is `orcl`).
   c. Stop the database instance:

```
ORACLE_HOME/bin/sqlplus /nolog
SQL> connect SYS as SYSDBA
SQL> shutdown
SQL> quit
```

   d. Stop the Net Listener:

```
ORACLE_HOME/bin/lsnrctl stop
```

4.8 Starting and Stopping: Special Topics

This section contains the following special topics about starting and stopping Oracle Fusion Middleware:

- Starting and Stopping in High Availability Environments
- Forcing a Shutdown of Oracle Database

4.8.1 Starting and Stopping in High Availability Environments

There are special considerations and procedures for starting and stopping High Availability environments, such as:

- Oracle Fusion Middleware Cold Failover Cluster
- Oracle Application Server Disaster Recovery

> **See:** *Oracle Fusion Middleware High Availability Guide* for information about starting and stopping in high-availability environments

4.8.2 Forcing a Shutdown of Oracle Database

If you find that the Oracle Database instance is taking a long time to shut down, you can use the following commands to force an immediate shutdown:

```
ORACLE_HOME/bin/sqlplus /nolog
SQL> connect SYS as SYSDBA
SQL> SHUTDOWN IMMEDIATE;
```

An immediate database shutdown proceeds with the following conditions:

- No new connections are allowed, nor are new transactions allowed to be started, after the statement is issued.
- Any uncommitted transactions are rolled back. (If long uncommitted transactions exist, this method of shutdown might not complete quickly, despite its name.)
- Oracle does not wait for users currently connected to the database to disconnect. Oracle implicitly rolls back active transactions and disconnects all connected users.

The next startup of the database will not require any instance recovery procedures.

> **See Also:** *Oracle Database Administrator's Guide* in the Oracle Database 11*g* documentation library

5 Managing Ports

This chapter describes how to view and change Oracle Fusion Middleware port numbers, such as port numbers used by Oracle WebLogic Server or Oracle HTTP Server.

It contains the following topics:

- About Managing Ports
- Viewing Port Numbers
- Changing the Port Numbers Used by Oracle Fusion Middleware

5.1 About Managing Ports

Many Oracle Fusion Middleware components and services use ports. Most port numbers are assigned during installation. As an administrator, it is important to know the port numbers used by these services, and to ensure that the same port number is not used by two services on your host.

For some ports, you can specify a port number assignment during installation.

> **See Also:** Appendix C for a list of port numbers. Refer to the installation guide for directions on overriding port assignments during installation.

5.2 Viewing Port Numbers

You can view the port numbers currently in use with the command line or Fusion Middleware Control, as described in the following topics:

- Viewing Port Numbers Using the Command Line
- Viewing Port Numbers Using Fusion Middleware Control

5.2.1 Viewing Port Numbers Using the Command Line

To view the current port numbers for system components, use the following command:

```
(UNIX) ORACLE_INSTANCE/bin/opmnctl status -l
(Windows) ORACLE_INSTANCE\bin\opmnctl status -l
```

To view the port numbers for Oracle WebLogic Server, you can use the WLST `get` command, with an attribute. For example, to get the Administration Port, use the following command:

```
wls:/SOA_domain/serverConfig> get('AdministrationPort')
9002
```

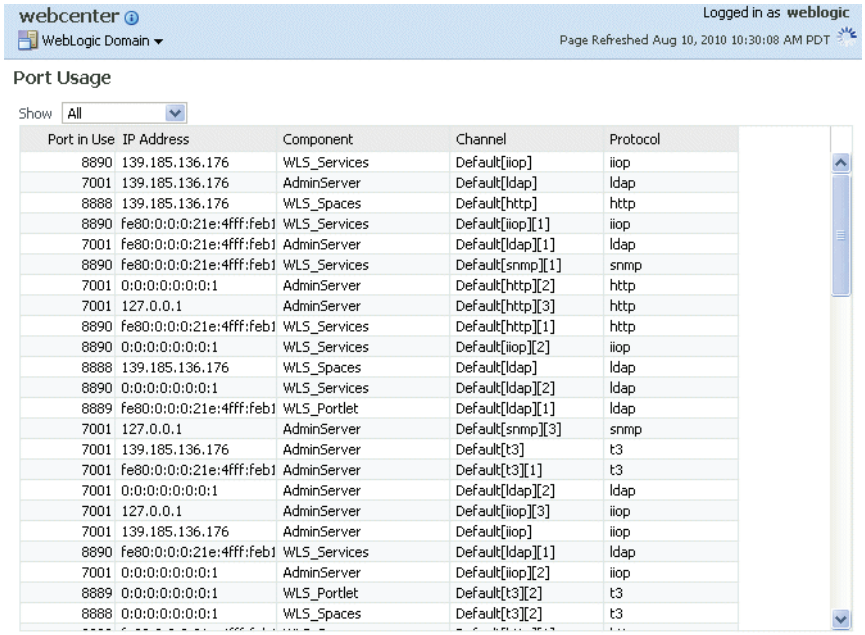
5.2.2 Viewing Port Numbers Using Fusion Middleware Control

You can view the port numbers of the domain, the Administration Server, Managed Servers, or components, such as the SOA Infrastructure and Oracle Web Cache, using Fusion Middleware Control.

For example, to view the ports of a domain:

1. From the navigation pane, expand the farm and then **WebLogic Domain.**
2. Select the domain.
3. From the WebLogic Domain menu, choose **Port Usage.**

   The Port Usage page is displayed, as shown in the following figure:



Port in Use	IP Address	Component	Channel	Protocol
8890	139.185.136.176	WLS_Services	Default[iiop]	iiop
7001	139.185.136.176	AdminServer	Default[ldap]	ldap
8888	139.185.136.176	WLS_Spaces	Default[http]	http
8890	fe80:0:0:0:21e:4fff:feb1	WLS_Services	Default[iiop][1]	iiop
7001	fe80:0:0:0:21e:4fff:feb1	AdminServer	Default[ldap][1]	ldap
8890	fe80:0:0:0:21e:4fff:feb1	WLS_Services	Default[snmp][1]	snmp
7001	0:0:0:0:0:0:0:1	AdminServer	Default[http][2]	http
7001	127.0.0.1	AdminServer	Default[http][3]	http
8890	fe80:0:0:0:21e:4fff:feb1	WLS_Services	Default[http][1]	http
8890	0:0:0:0:0:0:0:1	WLS_Services	Default[iiop][2]	iiop
8888	139.185.136.176	WLS_Spaces	Default[ldap]	ldap
8890	0:0:0:0:0:0:0:1	WLS_Services	Default[ldap][2]	ldap
8889	fe80:0:0:0:21e:4fff:feb1	WLS_Portlet	Default[ldap][1]	ldap
7001	127.0.0.1	AdminServer	Default[snmp][3]	snmp
7001	139.185.136.176	AdminServer	Default[t3]	t3
7001	fe80:0:0:0:21e:4fff:feb1	AdminServer	Default[t3][1]	t3
7001	0:0:0:0:0:0:0:1	AdminServer	Default[ldap][2]	ldap
7001	127.0.0.1	AdminServer	Default[iiop][3]	iiop
7001	139.185.136.176	AdminServer	Default[iiop]	iiop
8890	fe80:0:0:0:21e:4fff:feb1	WLS_Services	Default[ldap][1]	ldap
7001	0:0:0:0:0:0:0:1	AdminServer	Default[iiop][2]	iiop
8889	0:0:0:0:0:0:0:1	WLS_Portlet	Default[t3][2]	t3
8888	0:0:0:0:0:0:0:1	WLS_Spaces	Default[t3][2]	t3

   Optionally, you can filter the ports shown by selecting a Managed Server from **Show.**

The Port Usage detail table shows the ports that are in use, the IP Address, the component, the channel, and the protocol.

You can also view similar pages for the Administration Server, Managed Servers, and components, such as the SOA Infrastructure and Oracle Web Cache, by navigating to the target and choosing **Port Usage** from the target's menu.

5.3 Changing the Port Numbers Used by Oracle Fusion Middleware

You can change the port numbers for some Oracle Fusion Middleware components, using Fusion Middleware Control, Oracle WebLogic Server Administration Console, or the command line.

Note: You can change a port number to any number you want, if it is an unused port. You do not have to use a port in the allotted port range for the component. See Appendix C for information on allotted port ranges.

This section provides the following topics:

- Changing the Oracle WebLogic Server Listen Ports
- Changing the Oracle HTTP Server Listen Ports
- Changing Oracle Web Cache Ports
- Changing OPMN Ports (ONS Local, Request, and Remote)
- Changing Oracle Portal Ports
- Changing the Oracle Database Net Listener Port

For information about changing other ports, see:

- "Configuring Server Properties" or "Setting System Configuration Attributes by Using ldapmodify" in the *Oracle Fusion Middleware Administrator's Guide for Oracle Internet Directory* for information about changing Oracle Internet Directory ports
- "Overview of Node Manager Configuration" in the *Oracle Fusion Middleware Node Manager Administrator's Guide for Oracle WebLogic Server* for information about changing the Node Manager port.
- "Configuring Oracle Virtual Directory to Listen on Privileged Ports" in the *Oracle Fusion Middleware Administrator's Guide for Oracle Virtual Directory*

5.3.1 Changing the Oracle WebLogic Server Listen Ports

You can change the non-SSL (HTTP) listen port and the SSL (HTTPS) listen port for a WebLogic Server Administration Server or a Managed Server using the Oracle WebLogic Server Administration Console or WLST, as described in the following topics:

- Changing the Oracle WebLogic Server Listen Ports Using the Administration Console
- Changing the Oracle WebLogic Server Listen Ports Using WLST

See Also: *Oracle Fusion Middleware Configuring Server Environments for Oracle WebLogic Server* for more information about changing Oracle WebLogic Server ports

5.3.1.1 Changing the Oracle WebLogic Server Listen Ports Using the Administration Console

To change the non-SSL (HTTP) listen port and the SSL (HTTPS) listen port for a WebLogic Server Administration Server or a Managed Server using the Oracle WebLogic Server Administration Console:

1. Navigate to the server.

   The Settings for *server_name* page is displayed.

2. On the General tab, change the number of the **Listen Port** or **SSL Listen Port.**
3. If the server is running, restart the server.

4. If other components rely on the Oracle WebLogic Server listen ports, you must reconfigure those components. For example for Oracle Portal, if the listen port for the Oracle WebLogic Server configured as WLS_PORTAL is changed, then you must make a corresponding change to the configuration in Oracle HTTP Server, which is pointing to the older port. Change the port number in the following file:

   ORACLE_INSTANCE/OHS/*ohs_name*/moduleconf/portal.conf

5.3.1.2 Changing the Oracle WebLogic Server Listen Ports Using WLST

To change the non-SSL (HTTP) listen port and the SSL (HTTPS) listen port for a WebLogic Server Administration Server or a Managed Server using the WLST command line. You must run the commands in offline mode; that is, you must not be connected to a server.

For example to change the Administration Server HTTP listen port to port 8001, use the following WLST commands:

```
readDomain("MW_HOME/user_projects/domains/domain_name")
cd("servers/AdminServer")
cmo.setListenPort(8001)
updateDomain()
```

5.3.2 Changing the Oracle HTTP Server Listen Ports

To change the Oracle HTTP Server Listen ports (non-SSL or SSL), there are often dependencies that must also be set. For example, if you are using Oracle Web Cache to improve the performance of your Oracle Fusion Middleware environment, you must modify the Oracle Web Cache origin server settings whenever you modify the Oracle HTTP Server Listen ports.

The following topics describe how to modify the Oracle HTTP Server HTTP or HTTPS Listen port:

- Enabling Oracle HTTP Server to Run as Root for Ports Set to Less Than 1024 (UNIX Only)
- Changing the Oracle HTTP Server Non-SSL Listen Port
- Changing the Oracle HTTP Server SSL Listen Port

5.3.2.1 Enabling Oracle HTTP Server to Run as Root for Ports Set to Less Than 1024 (UNIX Only)

On a UNIX system, if you are changing the Listen port to a number less than 1024, perform these steps before you change the Oracle HTTP Server Listen port.

By default, Oracle HTTP Server runs as a non-root user (the user that installed Oracle Fusion Middleware). On UNIX systems, if you change the Oracle HTTP Server Listen port number to a value less than 1024, you must enable Oracle HTTP Server to run as root, as follows:

1. Log in as root.
2. Run the following commands in the Oracle home:

   ```
   cd ORACLE_HOME/ohs/bin
   chown root .apachectl
   chmod 6750 .apachectl
   ```

5.3.2.2 Changing the Oracle HTTP Server Non-SSL Listen Port

To change the Oracle HTTP Server non-SSL (HTTP) Listen port, follow the procedures in the following tasks. Note that, on a UNIX system, if you are changing the Listen port to a number less than 1024, you must first perform the steps in Section 5.3.2.1.

- Task 1, "Modify the Oracle HTTP Server Listen Port"
- Task 2, "Update Oracle Web Cache"
- Task 3, "Restart the System Components"

Task 1 Modify the Oracle HTTP Server Listen Port

To change the Oracle HTTP Server Listen port using Fusion Middleware Control:

1. From the navigation pane, expand the farm, then **Web Tier,** then select the Oracle HTTP Server instance.
2. From the Oracle HTTP Server menu, choose **Administration,** then **Ports Configuration.**
3. Select the Listen port that uses the HTTP protocol, then click **Edit.**
4. Change the port number, then click **OK.**
5. Restart Oracle HTTP Server. (From the Oracle HTTP Server menu, choose **Control,** then **Restart.**)

Task 2 Update Oracle Web Cache

If you are using Oracle Web Cache as a reverse proxy, you must update Oracle Web Cache:

1. From the Fusion Middleware Control navigation pane, expand the farm, then **Web Tier.** Select the Oracle Web Cache instance.
2. From the Web Cache menu, choose **Administration,** then **Origin Servers.**
3. Select the origin server for which you have changed the port, and click **Edit.**

   The Edit Origin Server page is displayed.
4. In the **Port** field, change the port number.
5. Click **OK.**
6. Restart Oracle Web Cache. (From the Web Cache menu, choose **Control,** then **Restart.**)

Task 3 Restart the System Components

Restart OPMN and all system components in that Oracle instance:

```
opmnctl stopall
opmnctl startall
```

5.3.2.3 Changing the Oracle HTTP Server SSL Listen Port

To change the Oracle HTTP Server SSL (HTTPS) Listen port, follow the procedures in the following tasks. Note that, on a UNIX system, if you are changing the Listen port to a number less than 1024, you must perform the steps in Section 5.3.2.1.

- Task 1, "Modify the Oracle HTTP Server SSL Listen Port"
- Task 2, "Update Oracle Web Cache"
- Task 3, "Re-register mod_osso"

- Task 4, "Restart System Components"

Task 1 Modify the Oracle HTTP Server SSL Listen Port

To change the Oracle HTTP Server SSL Listen port using Fusion Middleware Control:

1. From the navigation pane, expand the farm, then **Web Tier,** then select the Oracle HTTP Server instance.
2. From the Oracle HTTP Server menu, choose **Administration,** then **Ports Configuration.**
3. Select the Listen port that uses the HTTPS protocol, then click **Edit.**
4. Change the port number, then click **OK.**
5. Restart Oracle HTTP Server. (From the Oracle HTTP Server menu, choose **Control,** then **Restart.**)

Task 2 Update Oracle Web Cache

If you are using Oracle Web Cache as a reverse proxy, you must update Oracle Web Cache:

1. From the Fusion Middleware Control navigation pane, expand the farm, then **Web Tier.** Select the Oracle Web Cache instance.
2. From the Web Cache menu, choose **Administration,** then **Origin Servers.**
3. Select the origin server for which you have changed the port, and click **Edit.**

   The Edit Origin Server page is displayed.
4. In the **Port** field, change the port number.
5. Click **OK.**
6. Restart Oracle Web Cache. (From the Web Cache menu, choose **Control,** then **Restart.**)

Task 3 Re-register mod_osso

If you are using Oracle Single Sign-On, you must use Release 10.1.4.3. If you have enabled Oracle Single Sign-On authentication (that is, you registered mod_osso), follow these steps to re-register mod_osso:

1. On the Oracle Single Sign-On host, set the environment variables ORACLE_HOME and ORACLE_SID.
2. On the Oracle Single Sign-On host, run the `ssoreg` script, using the `-remote_midtier` option. The script is located at:

   ```
   (UNIX) ORACLE_HOME/sso/bin/ssoreg.sh
   (Windows)ORACLE_HOME\sso\bin\ssoreg.bat
   ```

   For example, on LINUX:

   ```
   $ORACLE_HOME/sso/bin/ssoreg.sh -oracle_home_path $ORACLE_HOME
    -config_mod_osso TRUE
    -site_name example.com:7778
    -remote_midtier
    -config_file $ORACLE_HOME/Apache/Apache/conf/osso/myosso.conf
    -mod_osso_url http://example.com:7778
   ```

   The resulting configuration file (`myosso.conf` in the example) is an obfuscated osso configuration file.

3. Copy the obfuscated osso configuration file to the Oracle HTTP Server host moduleconf directory for editing:

   ORACLE_INSTANCE/config/OHS/*ohs_name*/moduleconf

Task 4 Restart System Components

Restart OPMN and the system components in that Oracle instance:

```
opmnctl stopall
opmnctl startall
```

5.3.3 Changing Oracle Web Cache Ports

You can change the HTTP and HTTPS listen ports, the administration port, the statistics port and the invalidation port for Oracle Web Cache using Fusion Middleware Control.

To change the port number:

1. From the navigation pane, expand the farm, then **Web Tier,** then select the Oracle Web Cache instance.
2. From the Web Cache menu, choose **Administration,** then **Ports Configuration.**
3. Select a port, then click **Edit.**
4. Change the port number, then click **OK.**
5. Restart Oracle Web Cache. (From the Web Cache menu, choose **Control,** then **Restart.**)
6. If you reconfigure the Web Cache invalidation port and you use Oracle Portal, you must update the port information maintained by Oracle Portal, as described in Section 5.3.5.2.

Note: To configure Oracle Web Cache to start as root, see "Running webcached with Root Privilege" in the *Oracle Fusion Middleware Administrator's Guide for Oracle Web Cache*.

5.3.4 Changing OPMN Ports (ONS Local, Request, and Remote)

This section describes how to change any of the following port numbers:

- ONS Local port
- ONS Request port
- ONS Remote port

To change these ports:

1. Stop OPMN, and all OPMN-managed processes:

   (UNIX) *ORACLE_INSTANCE*/bin/opmnctl stopall
   (Windows)*ORACLE_INSTANCE*\bin\opmnctl stopall

2. Open the `opmn.xml` file:

   (UNIX) *ORACLE_INSTANCE*/config/OPMN/opmn
   (Windows) *ORACLE_INSTANCE*\config\OPMN\opmn

3. Under the `<notification-server>` element, modify the `local`, `remote`, or `request` parameter, depending on the port you are changing, in the `<port>` element, and then save the file.

   For example:

   ```
   <port local="6101" remote="6201" request="6004"/>
   ```

4. Start OPMN, and all OPMN-managed processes:

   ```
   (UNIX) ORACLE_INSTANCE/bin/opmnctl startall
   (Windows) ORACLE_INSTANCE\bin\opmnctl startall
   ```

5. If this instance is registered with the Oracle WebLogic Server Administration Server to be administered using Fusion Middleware Control, you will not be able to start or stop the instance using Fusion Middleware Control after you changed the ONS port. You must unregister, then reregister the instance with the Administration Server:

   a. Unregister the instance:

      ```
      ORACLE_INSTANCE/bin/opmnctl unregisterinstance
         -adminHost hostname
         -adminPort weblogic_port
         -adminUsername weblogic_admin
      ```

   b. Register the instance:

      ```
      ORACLE_INSTANCE/bin/opmnctl registerinstance
         -adminHost hostname
         -adminPort weblogic_port
         -adminUsername weblogic_admin
         -adminPasswordFile 'file_with_weblogic_admin_password'
      ```

   Now, you can refresh the Fusion Middleware Control page and start or stop the instance.

5.3.5 Changing Oracle Portal Ports

Oracle Portal maintains information about some of the ports used by its underlying components. This section describes how to manage Oracle Portal ports. It includes the following topics:

- Changing the Oracle Portal Midtier Port
- Changing the Oracle Web Cache Invalidation Port for Oracle Portal
- Changing the Oracle Internet Directory Port for Oracle Portal
- Changing the PPE Loopback Port
- Changing Oracle Portal SQL*Net Listener Port
- Restarting WLS_PORTAL Managed Server

Note: When you change these ports as described in this section, only the Oracle Portal configuration is updated. To update or change the port numbers of an underlying component, such as Oracle Web Cache or Oracle Internet Directory, see the component-specific documentation for information about managing ports.

The configuration procedures described in this section require you to restart the WLS_PORTAL Managed Server.

5.3.5.1 Changing the Oracle Portal Midtier Port

In a default installation, you can access Oracle Portal through the Oracle Web Cache port, such as 8090. This port is referred to as the Oracle Portal midtier port. You must update this port if Oracle Web Cache is configured to listen on a different port or Oracle Web Cache is front-ended by a Proxy or Load Balancing Router (LBR).

To change the Oracle Portal midtier port using Fusion Middleware Control:

1. From the navigation pane, expand the farm, choose **Portal,** and select the Portal instance.
2. From the Portal menu, choose **Settings,** and then **Wire Configuration,**
3. Select the Database Access Descriptor, such as `portal`.
4. Expand the Portal Midtier section.
5. Change the port number, and click **Apply.**
6. Restart the WLS_PORTAL Managed Server. For more information, see Section 5.3.5.6.

5.3.5.2 Changing the Oracle Web Cache Invalidation Port for Oracle Portal

Oracle Portal caches content in Oracle Web Cache. When content changes, Oracle Portal invalidates such cached content and maintains the Oracle Web Cache invalidation port. If you reconfigure the Web Cache invalidation port, you must update the port information maintained by Oracle Portal.

To change the Oracle Portal Invalidation port using Fusion Middleware Control:

1. From the navigation pane, expand the farm, choose **Portal**, and select the Portal instance.
2. From the Portal menu, choose **Settings**, and then **Wire Configuration**.
3. Select the Database Access Descriptor, such as `portal`.
4. Expand the Web Cache section.
5. Change the Invalidation Port number. If the Invalidation user name and the password are blank, enter the user name and the password.

Note: The Port number, Invalidation user name, and Invalidation password entered here must match the corresponding values of the Oracle Web Cache instance used by Oracle Portal. For more information about resetting these values, see the *Oracle Fusion Middleware Administrator's Guide for Oracle Web Cache*.

---

6. Click **Apply**.

7. Restart the WLS_PORTAL Managed Server. For more information, see Section 5.3.5.6.

5.3.5.3 Changing the Oracle Internet Directory Port for Oracle Portal

Oracle Portal maintains information about Oracle Internet Directory ports.

To change the Oracle Portal Oracle Internet Directory (OID) port using Fusion Middleware Control:

1. From the navigation pane, expand the farm, choose **Portal**. and select the Portal instance.
2. From the Portal menu, choose **Settings,** and then **Wire Configuration.**
3. Select the Database Access Descriptor, such as `portal`.
4. Expand the OID section.
5. Change the port number.
6. Enter the Oracle Internet Directory user name and the password.
7. Click **Apply.**
8. Restart the WLS_PORTAL Managed Server. For more information, see Section 5.3.5.6.

5.3.5.4 Changing the PPE Loopback Port

While servicing Portal pages, Oracle Portal makes loopback calls using the default site port. In some configurations, such as external SSL, you must configure the loopback call to a port other than the default site port.

To change the PPE Loopback port using Fusion Middleware Control:

1. From the navigation pane, expand the farm, choose **Portal,** and select the Portal instance.
2. From the Portal menu, choose **Settings**, and then **Page Engine.**
3. Expand the Advanced Properties section.
4. Change the port number in the **Use Port.**
5. Specify the protocol in the **Use Protocol** field.
6. Click **Apply.**
7. Restart the WLS_PORTAL Managed Server. For more information, see Section 5.3.5.6.

5.3.5.5 Changing Oracle Portal SQL*Net Listener Port

Oracle Portal maintains information about the repository connection in the `host:port:servicename` format inside a Database Access Descriptor (in a file named `portal_dads.conf`). If the SQL*Net listener is reconfigured to listen on a different port, you must reconfigure this port value in Oracle Portal.

To change the Oracle Portal SQL*Net Listener port in Fusion Middleware Control:

1. From the navigation pane, expand the farm, choose **Portal,** and select the Portal instance.
2. From the Portal menu, choose **Settings**, and then **Database Access Descriptor.**
3. Select the Database Access Descriptor, such as `/pls/portal`.

4. Click **Edit.**
5. Expand the Portal Database Access Details section.
6. Update the **Database Connect String** field to reflect the new port.
7. Click **OK.**
8. Restart the WLS_PORTAL Managed Server. For more information, see Section 5.3.5.6.

5.3.5.6 Restarting WLS_PORTAL Managed Server

To restart WLS_PORTAL Managed Server in Fusion Middleware Control:

1. Expand the Farm domain, such as `Farm_ClassicDomain`.
2. Expand WebLogic Domain.
3. Expand the domain, such as `Classic Domain`.
4. Expand cluster_portal, when applicable.
5. Choose WLS_PORTAL.
6. From the WLS_PORTAL WebLogic Server menu, choose **Control,** then **Shut Down.** Ensure that the status of WLS_PORTAL shows Down.
7. From the WLS_PORTAL WebLogic Server menu, choose **Control,** then **Start Up.** Ensure that the status of WLS_PORTAL shows Up.

5.3.6 Changing the Oracle Database Net Listener Port

If your environment includes an Oracle Database that functions as a metadata repository, and you want to change the listener port number for that database, perform the procedure in this section.

First, determine if it is necessary to change the listener port number. If you are concerned that you have another database on your host using the same port, both databases can possibly use the same port.

Note that multiple Oracle Database 10*g* and Oracle Database 11*g* databases can share the same Oracle Net listener port. If you are using an Oracle Database as a metadata repository on the same host that contains another Oracle Database 10*g* or Oracle Database 11*g* database, they can all use port 1521. There is no need to change the listener port number.

Note: To run two listeners that use the same key value on one host, refer to Section 5.3.6.1, "Changing the KEY Value for an IPC Listener"

A metadata repository may be used in several different ways. Use the following table to determine the steps that are required for changing your type of metadata repository:

If the Metadata Repository is used as follows:	Follow these tasks to change its Oracle Net listener port:
Identity Management repository and product metadata repository	Task 1, "Stop Components" Task 2, "Change the Metadata Repository for Oracle Net Listener Port" Task 3, "Change the System Data Source" Task 4, "Update Oracle Internet Directory" Task 5, "Update Oracle Single Sign-On 10g" Task 6, "Update Oracle Portal" Task 7, "Update Other Components"
Identity Management repository only	Task 1, "Stop Components" Task 2, "Change the Metadata Repository for Oracle Net Listener Port" Task 4, "Update Oracle Internet Directory" Task 5, "Update Oracle Single Sign-On 10g"
Product metadata repository	Task 1, "Stop Components" Task 2, "Change the Metadata Repository for Oracle Net Listener Port" Task 3, "Change the System Data Source" Task 4, "Update Oracle Internet Directory" Task 6, "Update Oracle Portal" Task 7, "Update Other Components"

The procedure consists of the following tasks:

- Task 1, "Stop Components"
- Task 2, "Change the Metadata Repository for Oracle Net Listener Port"
- Task 3, "Change the System Data Source"
- Task 4, "Update Oracle Internet Directory"
- Task 5, "Update Oracle Single Sign-On 10g"
- Task 6, "Update Oracle Portal"
- Task 7, "Update Other Components"

Task 1 Stop Components

Stop all components that use the Metadata Repository. See Chapter 4 for instructions.

Task 2 Change the Metadata Repository for Oracle Net Listener Port

On the metadata repository host:

1. Ensure that the ORACLE_HOME and ORACLE_SID environment variables are set.
2. Stop the metadata repository listener:

   ```
   lsnrctl stop
   ```

3. Edit the `listener.ora` file, which is located at:

   ```
   (UNIX) ORACLE_HOME/network/admin/listener.ora
   (Windows) ORACLE_HOME\network\admin\listener.ora
   ```

   Under the `LISTENER` entry, update the value for `PORT`. Save the file.

4. Edit the `tnsnames.ora` file. The default location is:

   ```
   (UNIX) ORACLE_HOME/network/admin/tnsnames.ora
   (Windows) ORACLE_HOME\network\admin\tnsnames.ora
   ```

   Make the following changes to the file:

   a. Update the `PORT` value in each entry that applies to MDS Repository.

   b. Add an entry similar to the following:

      ```
      newnetport =
        (DESCRIPTION =
          (ADDRESS = (PROTOCOL = tcp) (HOST = hostname) (PORT = port)))
      ```

      In the example, *`hostname`* is the fully qualified host name and *`port`* is the new port number.

5. Start the metadata repository listener:

   ```
   lsnrctl start
   ```

6. Using SQL*Plus, log in to the metadata repository as the SYSTEM user with SYSDBA privileges and run the following command:

   ```
   SQL> ALTER SYSTEM SET local_listener='newnetport' scope=spfile;
   ```

7. Using SQL*Plus, restart the metadata repository:

   ```
   SQL> SHUTDOWN
   SQL> STARTUP
   ```

8. Start Oracle Internet Directory:

   ```
   opmnctl start
   opmnctl startproc ias-component=OID
   ```

Task 3 Change the System Data Source

Change the system data source to use the new port number for the metadata repository. To do so, you use Oracle WebLogic Server Administration Console:

1. In the Change Center, click **Lock & Edit.**
2. In the Domain Structure section, expand **Services** and select **Data Sources.**

   The Summary of JDBC Data Sources page is displayed.
3. Select the data source you want to change.

   The Settings page is displayed.
4. Select the Connection Pool tab.
5. To change the database port, modify the **URL** field. For example:

   ```
   jdbc:oracle:thin:@hostname.domainname.com:1522/orcl
   ```

6. Click **Save.**
7. Restart the servers that use this data source. (Click the Target tab to see the servers that use this data source.)

Task 4 Update Oracle Internet Directory

On the Identity Management host, update Oracle Internet Directory with the new Oracle Net listener port number:

1. Update the port number in tnsnames.ora file, which is located in the following directory:

   ```
   (UNIX) ORACLE_INSTANCE/config
   (Windows) ORACLE_INSTANCE\config
   ```

2. Update the registration of the component with the Administration Server, using the opmnctl `updatecomponentregistration` command with the new port number, as shown in the following example:

   ```
   opmnctl updatecomponentregistration -Db_info DBHostName:TNSPORT:DBSERVICENAME
      -componentName oid1 -componentType OID
   ```

3. Start OPMN and all processes in the Oracle instance in the Oracle Internet Directory Oracle home:

   ```
   opmnctl startall
   ```

Task 5 Update Oracle Single Sign-On 10*g*

If you are using Oracle Single Sign-On 10*g*, from the Oracle Single Sign-On Oracle home:

1. On UNIX systems, set the LD_LIBRARY_PATH, LD_LIBRARY_PATH_64, LIB_PATH, or SHLIB_PATH environment variables to the proper values, as shown in Table 3–1. The actual environment variables and values that you must set depend on the type of your UNIX operating system.
2. Update Oracle Single Sign-On with the new repository port number by executing the following command:
   - On UNIX systems:

     ```
     $ORACLE_HOME/jdk/bin/java -jar $ORACLE_HOME/sso/lib/ossoca.jar reassoc
     -repos $ORACLE_HOME
     ```

   - On Windows systems:

     ```
     %ORACLE_HOME%\jdk\bin\java -jar %ORACLE_HOME%\sso\lib\ossoca.jar reassoc
     -repos %ORACLE_HOME%
     ```

Task 6 Update Oracle Portal

To update Oracle Portal, follow the steps in Section 5.3.5.5.

Task 7 Update Other Components

In each Oracle instance that uses the metadata repository:

1. Update the following file with the new Oracle Net listener port number:

   ```
   (UNIX) ORACLE_INSTANCE/config/tnsnames.ora
   (Windows) ORACLE_INSTANCE\config\tnsnames.ora
   ```

2. Check the following file:

   ```
   (UNIX) ORACLE_HOME/ohs/conf/dads.conf
   (Windows) ORACLE_HOME\ohs\modplsql\conf\dads.conf
   ```

   Locate the line that begins with `PlsqlDatabaseConnectString`.

   - If the line ends with `ServiceNameFormat` or `SIDFormat`, update the line with the new MDS Repository port number, save the file, and restart Oracle HTTP Server.

- If the line ends with `NetServiceNameFormat`, you do not need to do anything.

3. Start the components that use the metadata repository, as described in Section 4.3.

5.3.6.1 Changing the KEY Value for an IPC Listener

It is not possible to run two listeners at the same time that are configured to use the same KEY value in their IPC protocol address. By default, the metadata repository listener has its IPC KEY value set to EXTPROC. Hence, if your computer has another IPC listener that uses the EXTPROC key, you should configure the metadata repository listener to use some other key value such as EXTPROC1.

To change the KEY value of an IPC listener:

1. Stop the listener (ensure that your ORACLE_HOME environment variable is set first):

   ```
   lsnrctl stop
   ```

2. Edit the `listener.ora` and `tnsnames.ora` files. In each file, find the following line:

   ```
   (ADDRESS = (PROTOCOL = IPC)(KEY = EXTPROC))
   ```

   Change it to the following:

   ```
   (ADDRESS = (PROTOCOL = IPC)(KEY = EXTPROC1))
   ```

3. Restart the listener:

   ```
   lsnrctl start
   ```

Part III

Secure Sockets Layer

This part describes how to secure communications between Oracle Fusion Middleware components using the Secure Sockets Layer (SSL) and how to use Oracle Fusion Middleware security features to administer keystores, wallets, and certificates.

Part III contains the following chapters:

6

Configuring SSL in Oracle Fusion Middleware

You can configure Oracle Fusion Middleware to secure communications between Oracle Fusion Middleware components using SSL, which is an industry standard for securing communications. Oracle Fusion Middleware supports SSL version 3, as well as TLS version 1.

> **Note:** SSL version 2 has been desupported in 11*g* Release 2 (11.1.2) due to security concerns; components or applications that used SSL version 2 in pre-11*g* Release 2 (11.1.2) will automatically be upgraded to use other SSL versions, that is, SSL version 3 and TLS version 1.

> **See Also :** Chapter 7, "Using the SSL Automation Tool." The SSL Automation Tool enables you to configure SSL for multiple components using a domain-specific CA.

This chapter provides an overview of SSL and how you can use it with Oracle Fusion Middleware components and applications. It contains these topics:

- How SSL Works
- About SSL in Oracle Fusion Middleware
- Configuring SSL for Configuration Tools
- Configuring SSL for the Web Tier
- Configuring SSL for the Middle Tier
- Configuring SSL for the Data Tier
- Advanced SSL Scenarios
- Best Practices for SSL
- WLST Reference for SSL

> **Note:** Where SSL connections are configured within Oracle WebLogic Server, this chapter provides references to the relevant Oracle WebLogic Server documentation rather than duplicating the instructions here.

6.1 How SSL Works

This section introduces basic SSL concepts. It contains these topics:

- What SSL Provides
- About Private and Public Key Cryptography
- Keystores and Wallets
- How SSL Sessions Are Conducted

6.1.1 What SSL Provides

SSL secures communication by providing message encryption, integrity, and authentication. The SSL standard allows the involved components (such as browsers and HTTP servers) to negotiate which encryption, authentication, and integrity mechanisms to use.

- Encryption provides confidentiality by allowing only the intended recipient to read the message. SSL can use different encryption algorithms to encrypt messages. During the SSL handshake that occurs at the start of each SSL session, the client and the server negotiate which algorithm to use. Examples of encryption algorithms supported by SSL include AES, RC4, and 3DES.
- Integrity ensures that a message sent by a client is received intact by the server, untampered. To ensure message integrity, the client hashes the message into a digest using a hash function and sends this message digest to the server. The server also hashes the message into a digest and compares the digests. Because SSL uses hash functions that make it computationally infeasible to produce the same digest from two different messages, the server can tell that if the digests do not match, then someone had tampered with the message. An example of a hash function supported by SSL is SHA1.
- Authentication enables the server and client to check that the other party is who it claims to be. When a client initiates an SSL session, the server typically sends its certificate to the client. Certificates are digital identities that are issued by trusted certificate authorities, such as Verisign. Chapter 8, "Managing Keystores, Wallets, and Certificates" describes certificates in more detail.

  The client verifies that the server is authentic and not an imposter by validating the certificate chain in the server certificate. The server certificate is guaranteed by the certificate authority (CA) who signed the server certificate.

  The server can also require the client to have a certificate, if the server needs to authenticate the identity of the client.

6.1.2 About Private and Public Key Cryptography

To provide message integrity, authentication, and encryption, SSL uses both private and public key cryptography.

Secret Key Cryptography

Symmetric key cryptography requires a single, secret key shared by two or more parties to secure communication. This key is used to encrypt and decrypt secure messages sent between the parties. This requires prior and secure distribution of the key to each party. The problem with this method is that it is difficult to securely transmit and store the key.

In SSL, each party calculates the secret key individually using random values known to each side. The parties then send messages encrypted using the secret key.

Public Key Cryptography

Public key cryptography solves this problem by employing public and private key pairs and a secure method for key distribution. The freely available public key is used to encrypt messages that can *only* be decrypted by the holder of the associated private key. The private key is securely stored, together with other security credentials, in an encrypted container such as an Oracle wallet.

Public key algorithms can guarantee the secrecy of a message, but they do not necessarily guarantee secure communication because they do not verify the identities of the communicating parties. To establish secure communication, it is important to verify that the public key used to encrypt a message does in fact belong to the target recipient. Otherwise, a third party can potentially eavesdrop on the communication and intercept public key requests, substituting its own public key for a legitimate key (the man-in-the-middle attack).

To avoid such an attack, it is necessary to verify the owner of the public key, a process called authentication. Authentication can be accomplished through a certificate authority (CA), which is a third party trusted by both of the communicating parties.

The CA issues public key certificates that contain an entity's name, public key, and certain other security credentials. Such credentials typically include the CA name, the CA signature, and the certificate effective dates (From Date, To Date).

The CA uses its private key to encrypt a message, while the public key is used to decrypt it, thus verifying that the message was encrypted by the CA. The CA public key is well known, and does not have to be authenticated each time it is accessed. Such CA public keys are stored in wallets.

6.1.3 Keystores and Wallets

In Oracle Fusion Middleware, most components use the Oracle wallet as their storage mechanism. An Oracle wallet is a container that stores your credentials, such as certificates, trusted certificates, certificate requests, and private keys. You can store Oracle wallets on the file system or in LDAP directories such as Oracle Internet Directory. Oracle wallets can be auto-login or password-protected wallets.

Components that use Oracle wallet include:

- Oracle HTTP Server
- Oracle Web Cache
- Oracle Internet Directory

Configuring SSL for these components thus requires setting up and using Oracle wallets.

A component such as Oracle Virtual Directory uses a JKS keystore to store keys and certificates. Configuring SSL for Oracle Virtual Directory thus requires setting up and using JKS keystores.

For more information about configuring keystores and wallets, see:

- Section 6.2, "About SSL in Oracle Fusion Middleware" for a fuller description of keystore and wallet usage in Oracle Fusion Middleware
- Chapter 8, "Managing Keystores, Wallets, and Certificates" for a discussion of these terms, and administration details

6.1.4 How SSL Sessions Are Conducted

The SSL protocol has two phases: the handshake phase and the data transfer phase. The handshake phase authenticates the server and optionally the client, and establishes the cryptographic keys that will be used to protect the data to be transmitted in the data transfer phase.

When a client requests an SSL connection to a server, the client and server first exchange messages in the handshake phase. (A common scenario is a browser requesting a page using the `https://` instead of `http://` protocol from a server. The HTTPS protocol indicates the usage of SSL with HTTP.)

Figure 6–1 shows the handshake messages for a typical SSL connection between a Web server and a browser. The following steps are shown in the figure:

1. The client sends a Hello message to the server.

   The message includes a list of algorithms supported by the client and a random number that will be used to generate the keys.

2. The server responds by sending a Hello message to the client. This message includes:

   - The algorithm to use. The server selected this from the list sent by the client.
   - A random number, which will be used to generate the keys.

3. The server sends its certificate to the client.

4. The client authenticates the server by checking the validity of the server's certificate, the issuer CA, and optionally, by checking that the host name of the server matches the subject DN. The client sends a Session ID for session caching.

5. The client generates a random value ("pre-master secret"), encrypts it using the server's public key, and sends it to the server.

6. The server uses its private key to decrypt the message to retrieve the pre-master secret.

7. The client and server separately calculate the keys that will be used in the SSL session.

   These keys are not sent to each other because the keys are calculated based on the pre-master secret and the random numbers, which are known to each side. The keys include:

   - Encryption key that the client uses to encrypt data before sending it to the server
   - Encryption key that the server uses to encrypt data before sending it to the client
   - Key that the client uses to create a message digest of the data
   - Key that the server uses to create a message digest of the data

   The encryption keys are symmetric, that is, the same key is used to encrypt and decrypt the data.

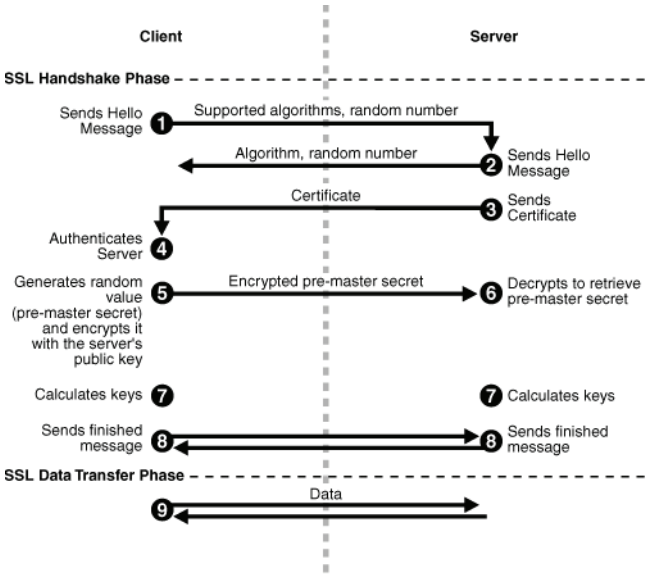
8. The client and server send a `Finished` message to each other. These are the first messages that are sent using the keys generated in the previous step (the first "secure" messages).

   The `Finished` message includes all the previous handshake messages that each side sent. Each side verifies that the previous messages that it received match the

messages included in the `Finished` message. This checks that the handshake messages were not tampered with.

9. The client and server now transfer data using the encryption and hashing keys and algorithms.

Figure 6–1 SSL Handshake



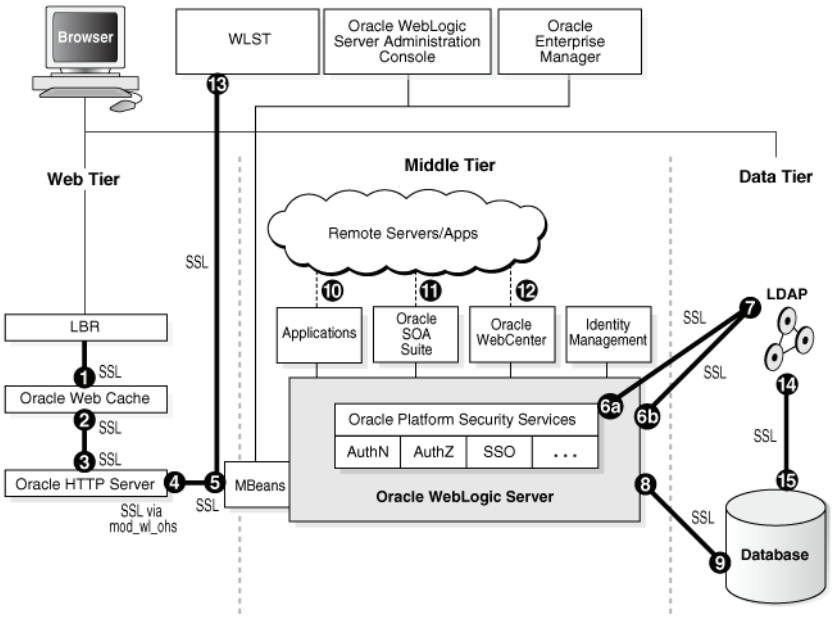
6.2 About SSL in Oracle Fusion Middleware

This section introduces SSL in Oracle Fusion Middleware. It contains these topics:

- SSL in the Oracle Fusion Middleware Architecture
- Keystores and Oracle Wallets
- Authentication Modes
- Tools for SSL Configuration

6.2.1 SSL in the Oracle Fusion Middleware Architecture

Figure 6–2 SSL in Oracle Fusion Middleware



Notes:

- In Figure 6–2, the label "Oracle Enterprise Manager" refers to the Fusion Middleware Control user interface.
- Other administrative tools, such as `opmn`, are available for specific tasks.

In the Oracle Fusion Middleware architecture shown in Figure 6–2, the numbered circles represent the endpoints that can be SSL-enabled. For configuration details about each endpoint, see:

1. Section 6.4.2.1, "Enable Inbound SSL for Oracle Web Cache Using Fusion Middleware Control" and Section 6.4.2.2, "Enable Inbound SSL for Oracle Web Cache Using WLST"
2. Section 6.4.2.3, "Enable Outbound SSL for Oracle Web Cache Using Fusion Middleware Control" and Section 6.4.2.4, "Specify the Wallet for Outbound SSL from Oracle Web Cache Using WLST"
3. Section 6.4.3.1, "Enable SSL for Inbound Requests to Oracle HTTP Server Virtual Hosts Using Fusion Middleware Control" and Section 6.4.3.2, "Enable SSL for Inbound Requests to Oracle HTTP Server Virtual Hosts Using WLST"
4. Section 6.4.3.3, "Enable SSL for Outbound Requests from Oracle HTTP Server"
5. Section 6.5.1.1, "Inbound SSL to Oracle WebLogic Server"
6. Outbound connections to the LDAP server can originate from Oracle Platform Security Services or from Oracle WebLogic Server:
   a. Section 6.5.1.2.1, "Outbound SSL from Oracle Platform Security Services to LDAP"

   b. Section 6.5.1.2.3, "Outbound SSL from LDAP Authenticator to LDAP"

7. Section 6.6.1.1, "Enable Inbound SSL on an Oracle Internet Directory Listener Using Fusion Middleware Control" and Section 6.6.1.2, "Enabling Inbound SSL on an Oracle Internet Directory Listener Using WLST"
8. Section 6.6.3.2, "SSL-Enable a Data Source"
9. Section 6.6.3.1, "SSL-Enable Oracle Database"
10. Section 6.5.6, "Client-Side SSL for Applications"
11. Section 6.5.2, "Configuring SSL for Oracle SOA Suite"
12. Section 6.5.3, "Configuring SSL for Oracle WebCenter Portal"
13. Section 6.3.3, "WLST Command-Line Tool"
14. Section 6.6.1.3, "Enabling Outbound SSL from Oracle Internet Directory to Oracle Database"
15. Section 6.6.3.1, "SSL-Enable Oracle Database"

In addition, you can configure SSL for identity management components. For details, see:

- Section 6.5.4.1, "Configuring SSL for Oracle Directory Integration Platform"
- Section 6.5.4.2, "Configuring SSL for Oracle Access Management Identity Federation"
- Section 6.5.4.3, "Configuring SSL for Oracle Directory Services Manager"

Keystores and Wallets

Keystores and wallets are central to SSL configuration and are used to store certificates and keys.

For details, see Section 6.2.2, "Keystores and Oracle Wallets."

6.2.2 Keystores and Oracle Wallets

Oracle Fusion Middleware supports two types of keystores for keys and certificates:

- JKS-based keystore and truststore
- Oracle wallet

In 11*g* Release 2 (11.1.2), all Java components and applications use the JKS keystore. Thus all Java components and applications running on Oracle WebLogic Server use the JKS-based KeyStore and TrustStore.

The following system components continue to use the Oracle wallet:

- Oracle HTTP Server
- Oracle Web Cache
- Oracle Internet Directory

You can use Fusion Middleware Control or the command-line WLST and `orapki` interfaces, to manage wallets and their certificates for these system components. You can use either the Fusion Middleware Control or WLST to SSL-enable the listeners for these components.

Oracle Virtual Directory uses a JKS-based keystore. You can use Fusion Middleware Control or WLST to manage JKS keystores and their certificates for Oracle Virtual

Directory. You can use either the Fusion Middleware Control or WLST to SSL-enable the listeners for Oracle Virtual Directory.

JDK's `keytool` utility manages the keystore used by Oracle WebLogic Server listeners for Java EE applications. This is the only keystore tool to manage these keystores; no graphical user interface is available for this purpose.

For more information about these types of stores, and when to use which type of store, see Section 6.1.3.

See Also: Section 8.1 for keystore management

6.2.3 Authentication Modes

The following authentication modes are supported:

- In *no-authentication mode*, neither server nor client are required to authenticate.

  Other names for this mode include Anonymous SSL/No Authentication/Diffie-Hellman. This mode is considered unsecured.

- In *server authentication mode*, a server authenticates itself to a client.

  This mode is also referred to as One-way SSL/Server Authentication.

- In *mutual authentication mode*, a client authenticates itself to a server and that server authenticates itself to the client.

  This mode is also known as Two-way SSL/Client Authentication.

- In *optional client authentication mode*, the server authenticates itself to the client, but the client may or may not authenticate itself to the server. Even if the client does not authenticate itself, the SSL session still goes through.

6.2.4 Tools for SSL Configuration

Oracle Fusion Middleware uses two kinds of configuration tools, common and advanced.

Common Tools

- Fusion Middleware Control
- WLST command-line interface
- Oracle WebLogic Server Administration Console
- `keytool` command-line tool

These tools allow you to configure SSL and manage Oracle Wallet/JKS keystore for any listener or component in Oracle Fusion Middleware.

The first three tools on this list are usable when the component is associated with the application server domain (when the component is not a stand-alone installation).

Advanced Tools

- Oracle Wallet Manager graphical user interface
- `orapki` command-line interface

These tools are needed to manage wallets for stand-alone Web tier and stand-alone Oracle Internet Directory installations.

In addition, these tools allow you to configure advanced features like managing file-based CRLs, PKCS11-based wallets, and so on.

See Also: Section 8.1 for keystore management

6.3 Configuring SSL for Configuration Tools

Several tools are available for Oracle Fusion Middleware configuration. This section describes how to configure SSL for these tools to enable them to connect to an SSL-enabled Oracle WebLogic Server.

See Also: Section 6.5.1.1 for details about enabling inbound SSL on Oracle WebLogic Server.

For a list of all the configuration tools, see Section 6.2.4.

This section contains these topics:

- Oracle Enterprise Manager Fusion Middleware Control
- Oracle WebLogic Server Administration Console
- WLST Command-Line Tool

6.3.1 Oracle Enterprise Manager Fusion Middleware Control

Take these steps:

- Ensure that the SSL port is enabled on the Oracle WebLogic Server instance on which Fusion Middleware Control is deployed, and that the browser (from which you will launch Fusion Middleware Control) trusts the server certificate.
- Now launch Fusion Middleware Control using an SSL-based URL, in the format https://*host:port*.

6.3.2 Oracle WebLogic Server Administration Console

Ensure that the SSL port is enabled on the Oracle WebLogic Server instance. Now launch the administration console by providing the SSL port in the URL. You may get a warning that the certificate is not trusted; accept this certificate and continue.

6.3.3 WLST Command-Line Tool

For details about configuring SSL for WLST, take these steps:

1. Launch the WLST shell.
2. Execute the WLST command:

```
help('connect')
```

Follow the instructions described in the help text to set up the WLST shell in SSL mode.

See Also: Section 6.9 for details about using WLST.

6.4 Configuring SSL for the Web Tier

This section contains these topics:

- Configuring Load Balancers

- Enabling SSL for Oracle Web Cache Endpoints
- Enabling SSL for Oracle HTTP Server Virtual Hosts

Note: ■This discussion applies to the Web Tier in the context of an Oracle WebLogic Server domain. For stand-alone Web Tier installations, see "Configuring Oracle Web Cache for HTTPS Requests" in the *Oracle Fusion Middleware Administrator's Guide for Oracle Web Cache*.

- The order in which these topics appear should not be confused with the sequence in which SSL is enabled (which varies depending on topology). Rather, they are arranged in order starting with the most front-ending component.

6.4.1 Configuring Load Balancers

Use the instructions specific to your load-balancing device to configure load balancers in your Oracle Fusion Middleware environment.

6.4.2 Enabling SSL for Oracle Web Cache Endpoints

This section explains how to enable SSL for Oracle Web Cache listening endpoints using Fusion Middleware Control and WLST.

6.4.2.1 Enable Inbound SSL for Oracle Web Cache Using Fusion Middleware Control

You can SSL-enable inbound traffic to Oracle Web Cache listening endpoints using these steps:

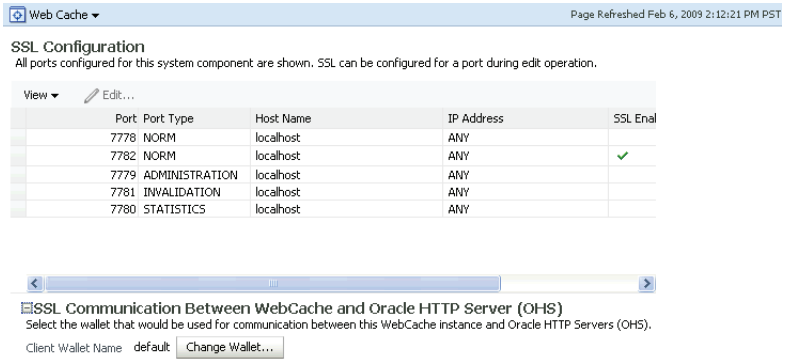
Note: This information applies only to inbound communication; for information about SSL-enabling outbound traffic from Oracle Web Cache to Oracle HTTP Server, see Section 6.4.2.3.

---

1. Select the Oracle Web Cache instance in the navigation pane on the left.
2. Create a wallet, if necessary, by navigating to **Oracle Web Cache**, then **Security**, then **Wallets**.

   For details about wallet creation and maintenance, see Chapter 8.
3. Navigate to **Oracle Web Cache**, then **Security**, then **SSL Configuration**.

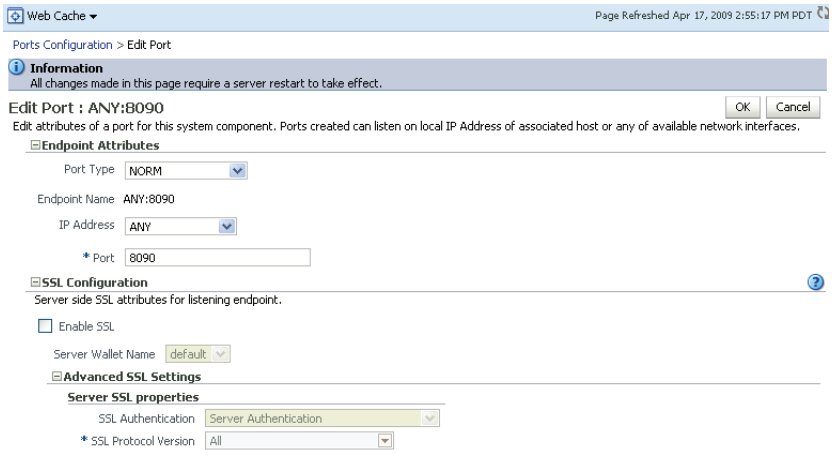
   The SSL Configuration page contains two sets of information:



The top table shows the inbound settings for a list of listening endpoints. A check in the **SSL Enabled** column indicates that the endpoint is configured for SSL.

The bottom portion of the page shows outbound SSL configuration. For more information about outbound SSL, see Section 6.4.2.3.

4. Select an endpoint, and click **Edit**.



The Edit Port page appears. This page contains two sections—a top portion with general details like port and IP address, and a bottom section that configures SSL parameters.

5. To disable SSL, uncheck **Enable SSL**; restart the component instance by navigating to Oracle Web Cache, then **Control**, then **Restart**.

6. To enable SSL for this endpoint, check **Enable SSL**. Next, enter SSL configuration parameters:

   - Select an Oracle wallet from the drop-down list.

     Note: Ensure that the wallet contains the server certificate and its corresponding CA certificate.

   - Select the type of SSL authentication.
   - Select the protocol version (the available options are determined by your choice of authentication).

7. Click **OK**.

8. On Windows platforms only, open Windows Explorer and navigate to your cwallet.sso file. Under properties, security, add `SYSTEM` in "group or user names".
9. Restart the Oracle Web Cache instance by navigating to **Oracle Web Cache**, then **Control**, then **Restart.**

> **See Also:** Section 8.4.1.3, "Sharing Wallets Across Instances"

6.4.2.2 Enable Inbound SSL for Oracle Web Cache Using WLST

You can enable SSL for inbound traffic to Oracle Web Cache using the WLST command-line tool.

SSL-Enable Oracle Web Cache Inbound in server-auth Mode Using WLST

Take these steps:

> **See Also:** See Section 6.9 for details about using WLST commands, including the definition of each command parameter shown in this procedure.

1. Determine the listening endpoints for this Oracle Web Cache instance by running the following command:

```
listListeners('inst1','wc1')
```

   This command will list all the listening endpoints for this instance; select the one that needs to be configured for SSL. For example, select the endpoint named `CACHE.index1.LISTEN.index1`.

> **See Also:** Section 6.9 for details about using WLST.

2. Configure the listening endpoint with SSL properties:

```
configureSSL('inst1',
   'wc1',
   'webcache',
   'CACHE.index1.LISTEN.index1')
```

> **Note:**
>
> - `configureSSL` uses defaults for all SSL attributes; see Table 6–5 for details.
> - You may also specify a properties file as a parameter to `configureSSL`; see Table 6–4 for details.

3. On Windows platforms only, open Windows Explorer and navigate to your cwallet.sso file. Under properties, security, add `SYSTEM` in "group or user names".

6.4.2.3 Enable Outbound SSL for Oracle Web Cache Using Fusion Middleware Control

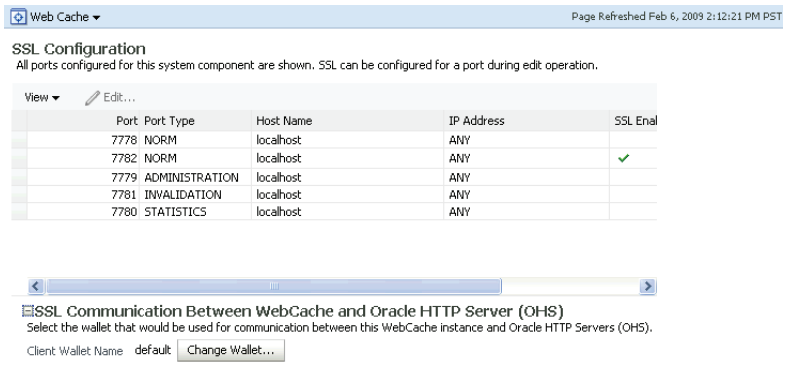
Outbound Oracle Web Cache refers to traffic from Oracle Web Cache to Oracle HTTP Server.

There are two aspects to set up SSL for outbound traffic from Oracle Web Cache: selecting a wallet for outbound SSL and configuring SSL.

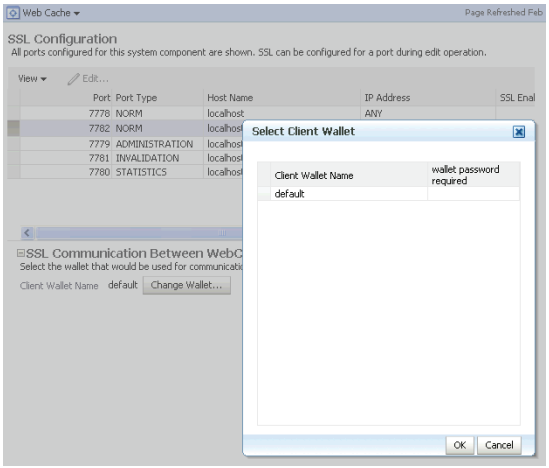
Wallet Selection

Take these steps:

1. Navigate to **Oracle Web Cache**, then **Security**, then **SSL Configuration**.



2. At the bottom of the page, click **Change Wallet** to display the available wallets for this listener.



   Note: The root CA certificate(s) that signed the certificate (for OHS or other component to which Webcache is connecting) must be loaded into this wallet. See Section 8.4.7.5 for details.

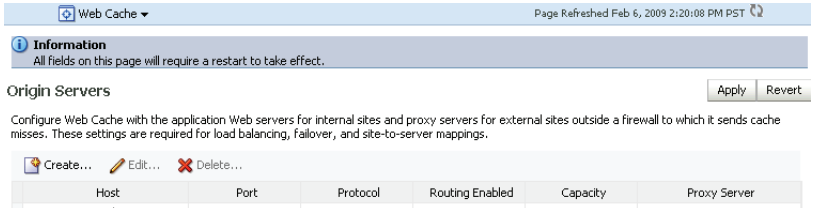
3. Select the desired wallet for outbound SSL and click **OK**.

SSL Configuration

Take these steps:

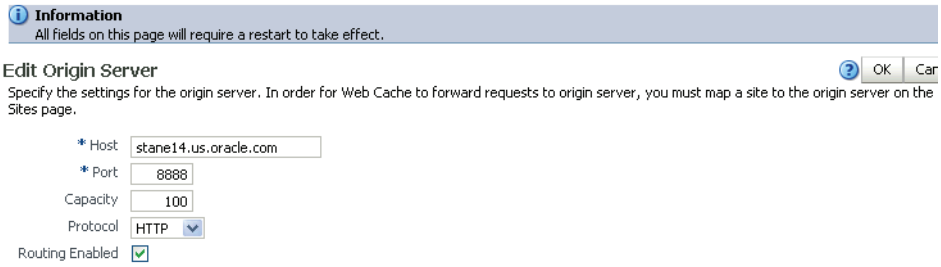
1. Navigate to the Oracle Web Cache instance, then **Administration**, then **Origin Servers**.

   This page displays the Oracle HTTP Servers with which this Oracle Web Cache instance can communicate. For example, if Oracle Web Cache can talk to two different Oracle HTTP Servers you would see two rows in the table.



In this example, the Oracle Web Cache instance is currently configured for non-SSL communication to the origin server over this host and port.

2. To enable SSL for outbound traffic to this origin server, select the row and click **Edit**.
3. The Edit Origin Server page appears:



4. Use the Protocol drop-down box to change the protocol to `https`.
5. Click **OK**.
6. On Windows platforms only, open Windows Explorer and navigate to your cwallet.sso file. Under properties, security, add `SYSTEM` in "group or user names".
7. Restart the Oracle Web Cache instance by navigating to **Oracle Web Cache**, then **Control**, then **Restart.**

   Oracle Web Cache is now configured to communicate to the origin server over SSL.

Note: When editing the origin server settings on this page, ensure that Oracle HTTP Server is listening at this port in SSL mode.

6.4.2.4 Specify the Wallet for Outbound SSL from Oracle Web Cache Using WLST

See Also: See Section 6.9 for details about using WLST commands, including the definition of each command parameter shown in this procedure.

To change the wallet in use for outbound SSL connections from Oracle Web Cache, use a command like the following:

```
configureSSL('inst1',
   'wc1',
   'webcache',
   'CACHE.index0.CLIENTSSL',
   'property-file.prop')
```

where:

- `inst1` is the name of the application server instance
- `wc1` is the name of the Oracle Web Cache instance
- `webcache` is the component type
- `CACHE.index0.CLIENTSSL` is the listener name for client SSL
- `property-file.prop` contains:

  `KeyStore=`*`wallet-path`*

6.4.3 Enabling SSL for Oracle HTTP Server Virtual Hosts

This section shows how to manage SSL configuration for Oracle HTTP Server virtual hosts operating in an Oracle WebLogic Server environment.

> **Note:** For Oracle HTTP Server in standalone mode, see *Oracle Fusion Middleware Administrator's Guide for Oracle HTTP Server*.

For inbound traffic:

- Section 6.4.3.1 (using Fusion Middleware Control)
- Section 6.4.3.2 (using WLST)

For outbound traffic:

- Section 6.4.3.3 (using either Fusion Middleware Control or WLST)

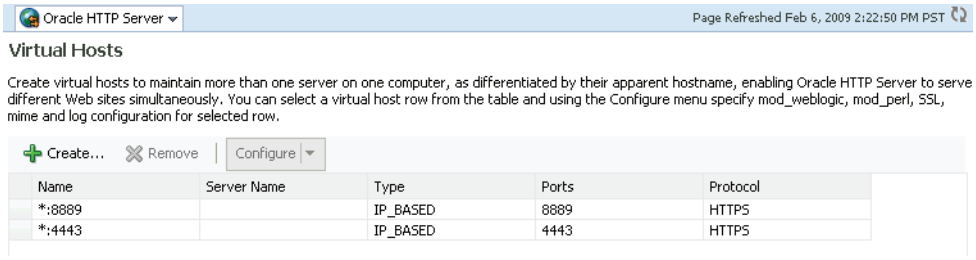
6.4.3.1 Enable SSL for Inbound Requests to Oracle HTTP Server Virtual Hosts Using Fusion Middleware Control

You can SSL-enable inbound traffic to Oracle HTTP Server virtual hosts using these steps:

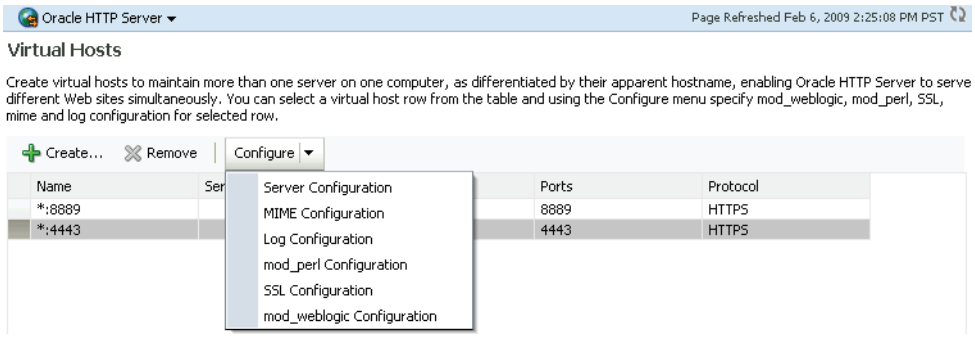
1. Select the Oracle HTTP Server instance in the navigation pane on the left.
2. Create a wallet, if necessary, by navigating to **Oracle HTTP Server**, then **Security**, then **Wallets**.

   For details about wallet creation and maintenance, see Chapter 8.
3. Navigate to **Oracle HTTP Server**, then **Administration**, then **Virtual Hosts**.

   This page shows what hosts are currently configured, and whether they are configured for http or https.



4. Select the virtual host you wish to update, and click **Configure**, then **SSL Configuration**. (Note: If creating a new virtual host, see *Oracle Fusion Middleware Administrator's Guide for Oracle HTTP Server*.)

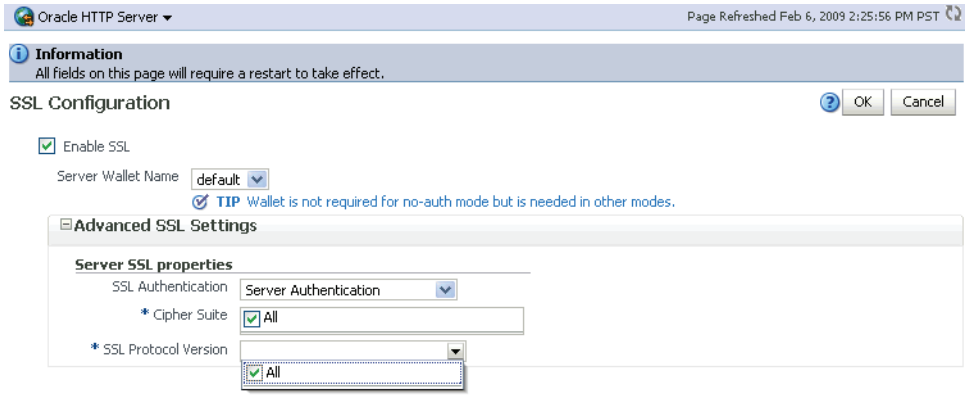


The SSL Configuration page appears.

5. You can convert an https port to http by simply unchecking **Enable SSL**.

   To configure SSL for a virtual host that is currently using http:

   - Check the **Enable SSL** box.
   - Select a wallet from the drop-down list.



   - From the Server SSL properties, select the SSL authentication type, cipher suites to use, and the SSL protocol version.

Note: The default values are appropriate in most situations.

Note: ■This assumes that the certificate is available in Fusion Middleware Control. If it was created through orapki or Oracle Wallet Manager, import it first as explained in Section 8.4.4.9.

- The choice of authentication type determines the available cipher suites, and the selected cipher suites determine the available protocol versions. For more information about ciphers and protocol versions, see Section 6.9.28.

---

6. Click **OK** to apply the changes.
7. On Windows platforms only, open Windows Explorer and navigate to your cwallet.sso file. Under properties, security, add `SYSTEM` in "group or user names".

8. Restart the Oracle HTTP Server instance by navigating to **Oracle HTTP Server**, then **Control**, then **Restart.**
9. Open a browser session and connect to the port number that was SSL-enabled.

6.4.3.2 Enable SSL for Inbound Requests to Oracle HTTP Server Virtual Hosts Using WLST

Take these steps:

1. Determine the virtual hosts for this **Oracle HTTP Server** instance by running the following command:

   ```
   listListeners('inst1','ohs1' )
   ```

   This command lists all the virtual hosts for this instance; select the one that needs to be configured for SSL. For example, you can select vhost1. For details about this command, see *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

2. Configure the virtual host with SSL properties:

   ```
   configureSSL('inst1',
      'ohs1',
      'ohs',
      'vhost1')
   ```

   Note:

   - `configureSSL` uses defaults for all SSL attributes; see Table 6–5 for details.
   - You could also specify a properties file as a parameter to `configureSSL`. See Table 6–4 for details about the parameters. See `configureSSL` in the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference* for examples of how to use a properties file.
   - For details about this command, see Section 6.9.

3. On Windows platforms only, open Windows Explorer and navigate to your cwallet.sso file. Under properties, security, add `SYSTEM` in "group or user names".

6.4.3.3 Enable SSL for Outbound Requests from Oracle HTTP Server

You enable SSL for outbound requests from Oracle HTTP Server by configuring `mod_wl_ohs`.

One-way SSL

The steps are as follows:

1. Generate a custom keystore for Oracle WebLogic Server (see Section 6.5.1) containing a certificate.
2. Import the certificate used by Oracle WebLogic Server from Step 1 into the Oracle HTTP Server wallet as a trusted certificate. You can use any available utility such as WLST or Fusion Middleware Control for this task. (*Note*: The wallet mentioned here is the one that the Oracle HTTP Server listen port uses for SSL. The trusted (root) CA certificate that signed the Oracle WebLogic Server certificate must exist in this wallet. For details on importing trusted certificates see Section 8.3.5.)

3. Edit the Oracle HTTP Server configuration file *INSTANCE_HOME*/config/OHS/ohs1/ssl.conf and add the following line to the SSL configuration under mod_weblogic:

```
WlSSLWallet  "$(ORACLE_INSTANCE}/config/COMPONENT_TYPE/COMPONENT_
NAME/keystores/default"
```

   where default is the name of the Oracle HTTP Server wallet in Step 2.

   Here is an example of how the configuration should look:

```
<IfModule mod_weblogic.c>
WebLogicHost myweblogic.server.com
WebLogicPort 7002
MatchExpression *.jsp
SecureProxy On
WlSSLWallet "$(ORACLE_INSTANCE)/config/OHS/ohs1/keystores/default"
</IfModule>
```

   Save the file and exit.

4. On Windows platforms only, open Windows Explorer and navigate to your cwallet.sso file. Under properties, security, add SYSTEM in "group or user names".
5. Restart Oracle HTTP Server to activate the changes. See *Oracle Fusion Middleware Administrator's Guide for Oracle HTTP Server* for details.
6. Ensure that your Oracle WebLogic Server instance is configured to use the custom keystore generated in Step 1, and that the alias points to the alias value used in generating the certificate. Restart the Oracle WebLogic Server instance.

Two-way SSL

mod_wl_ohs also supports two-way SSL communication. To configure two-way SSL:

1. Perform Steps 1 through 4 of the preceding procedure for one-way SSL.
2. Generate a trust store, trust.jks, for Oracle WebLogic Server.

   The keystore created for one-way SSL (Step 1 of the preceding procedure) could also be used to store trusted certificates, but it is recommended that you create a separate truststore for this procedure.
3. Export the user certificate from the Oracle HTTP Server wallet, and import it into the truststore created in Step 2.

   You can use any available utility such as WLST or Fusion Middleware Control for export, and the keytool utility for import.
4. From the Oracle WebLogic Server Administration Console, select the **Keystores** tab for the server being configured.
5. Set the custom trust store with the trust.jks file location of the trust store that was created in Step 2 (use the full name).
6. Set the keystore type as JKS, and set the passphrase used to create the keystore.
7. Under the **SSL** tab, ensure that Trusted Certificate Authorities is set as **from Custom Trust Keystore**.
8. Ensure that Oracle WebLogic Server is configured for two-way SSL. For details, see "Configuring SSL" in *Oracle Fusion Middleware Securing Oracle WebLogic Server*.

6.5 Configuring SSL for the Middle Tier

Using SSL in the middle tier includes:

- SSL-enabling the application server
- SSL-enabling components and applications running on the application server

This section addresses mid-tier SSL configuration and contains these topics:

- Configuring SSL for Oracle WebLogic Server
- Configuring SSL for Oracle SOA Suite
- Configuring SSL for Oracle WebCenter Portal
- Configuring SSL for Oracle Identity and Access Management
- SSL-Enable Oracle Reports, Forms, Discoverer, and Portal
- Client-Side SSL for Applications

6.5.1 Configuring SSL for Oracle WebLogic Server

This section describes configuration for the application server.

6.5.1.1 Inbound SSL to Oracle WebLogic Server

For information and details about implementing SSL to secure Oracle WebLogic Server, see *Oracle Fusion Middleware Securing Oracle WebLogic Server*.

6.5.1.2 Outbound SSL from Oracle WebLogic Server

This section describes how to SSL-enable outbound connections from Oracle WebLogic Server.

- Outbound SSL from Oracle Platform Security Services to LDAP
- Outbound SSL from Oracle Platform Security Services to Oracle Database
- Outbound SSL from LDAP Authenticator to LDAP
- Outbound SSL to Database

6.5.1.2.1 Outbound SSL from Oracle Platform Security Services to LDAP This section explains how to configure SSL (needs server- and client-side) for policy store and credential store connections to an LDAP directory. Anonymous and one-way SSL is supported.

See *Oracle Fusion Middleware Application Security Guide* for details about the `jps-config.xml` file referenced in this section.

Anonymous SSL (Server-side)

Start the LDAP server in anonymous authentication mode.

For Oracle Internet Directory, see Section 6.6.1.1.

If using another directory, consult your LDAP server documentation for information on this task.

Anonymous SSL (Client-side)

In your `jps-config.xml` file, you must set the protocol to `ldaps` and specify the appropriate port for the property `ldap.url`. This information needs to be updated for policy store, credential store, key store and any other service instances that use `ldap.url`.

One-Way SSL (Server-side)

Prerequisite: LDAP server in SSL Server Authentication Mode.

For details on this procedure, see the *Oracle Fusion Middleware Administrator's Guide for Oracle Internet Directory*.

One-Way SSL (Client-side)

The following must be in place for the client-side configuration:

1. The JVM needs to know where to find the trust store that it uses to trust certificates from LDAP. You do this by setting:

   `-Djavax.net.ssl.trustStore=`*`path_to_jks_file`*

   This property is added either to the JavaSE program, or to the server start-up properties in a JavaEE environment.

2. In your `jps-config.xml` file, you must set the protocol to `ldaps` and specify the appropriate port for the property `ldap.url`. This information needs to be updated for policy store, credential store, key store and any other service instances that use `ldap.url`.

3. Using **keytool**, import the LDAP server's certificate into the trust store specified in step 1.

6.5.1.2.2 Outbound SSL from Oracle Platform Security Services to Oracle Database You can set up a one-way or two-way SSL connection to a database-based OPSS security store.

For details about configuring the database server and clients, see *Oracle Fusion Middleware Application Security Guide*.

6.5.1.2.3 Outbound SSL from LDAP Authenticator to LDAP When you configure an LDAP authenticator in Oracle WebLogic Server, you can specify that connections to the LDAP store should use SSL.

Take these steps to configure the authenticator:

1. Log in to the Oracle WebLogic Server Administration Console.
2. In the left pane, select **Security Realms** and click the name of the realm you are configuring.
3. Select **Providers**, then **Authentication** and click **New**.
4. In the **Name** field, enter a name for the authentication provider.
5. From the **Type** drop-down list, select the type of the Authentication provider and click **OK**.

   For example, if using Oracle Internet Directory, choose `OracleInternetDirectoryAuthenticator`.

6. Select **Providers**, then **Authentication** and click the name of the new authentication provider to complete its configuration.
7. On the Configuration page for the authentication provider, set the desired values on the **Common** and **Provider-Specific** tabs.

   a. Common Tab

      Set the Control Flag to `SUFFICIENT` for all authenticators, including the `DefaultAuthenticator`

   b. Provider-Specific Tab

host: *host-name*

port: *port-number*

principal: cn=orcladmin

credential/confirm: *password*

user base dn: cn=Users,dc=us,dc=oracle,dc=com

group base dn: cn=Groups,dc=us,dc=oracle,dc=com

8. Save your changes and restart the server.

6.5.1.2.4 Outbound SSL to Database Configuring SSL between Oracle WebLogic Server and the database requires two sets of steps:

- Configuring SSL Listener for the Database
- Configuring SSL for the Data Source on Oracle WebLogic Server

Configure an SSL Listener on Oracle Database

To configure the database with an SSL listener, you must specify the server's distinguished name (DN) and TCPS as the protocol in the client network configuration files to enable server DN matching and TCP/IP with SSL connections. Server DN matching prevents the database server from faking its identity to the client during connections by matching the server's global database name against the DN from the server certificate.

You must manually edit the client network configuration files, `tnsnames.ora` and `listener.ora`, to specify the server's DN and the TCP/IP with SSL protocol.

For details, see Section 6.6.3.1.

> **See Also:** Configuring Secure Sockets Layer Authentication in the *Oracle Database Advanced Security Administrator's Guide* at `http://download.oracle.com/docs/cd/E11882_01/network.112/e10746/toc.htm` for more information about configuring SSL for the database listener.

SSL-Enable the Data Source On Oracle WebLogic Server

See Section 6.6.3.2.

6.5.2 Configuring SSL for Oracle SOA Suite

SSL configuration for Oracle SOA Suite varies with the type of connection being secured.

SSL in Oracle WebLogic Server

SSL features in Oracle WebLogic Server include:

- How to set up SSL at the core server.

  For details, see *Oracle Fusion Middleware Securing Oracle WebLogic Server*.

- How to enable SSL for a WebLogic Web service.

  For details, see *Oracle Fusion Middleware Securing WebLogic Web Services for Oracle WebLogic Server*.

SSL for SOA Composites

The following tasks are also needed to secure Oracle SOA Suite applications:

- SSL-protecting SOA composites
- Accessing SSL-protected Web services from within SOA composites

For these and related topics, see the *Oracle Fusion Middleware Administrator's Guide for Oracle SOA Suite and Oracle Business Process Management Suite*.

6.5.3 Configuring SSL for Oracle WebCenter Portal

For information and details about how to implement SSL connections for Oracle WebCenter Portal, see the following topics in the *Oracle Fusion Middleware Administrator's Guide for Oracle WebCenter Portal*:

- Securing the Spaces Connection to Content Server with SSL
- Securing the Browser Connection to Spaces with SSL

6.5.4 Configuring SSL for Oracle Identity and Access Management

You can configure SSL for Oracle Identity and Access Management components residing on the middle tier:

- Configuring SSL for Oracle Directory Integration Platform
- Configuring SSL for Oracle Access Management Identity Federation
- Configuring SSL for Oracle Directory Services Manager

6.5.4.1 Configuring SSL for Oracle Directory Integration Platform

You can configure Oracle Directory Integration Platform to use SSL for communications with connected directories. The *Oracle Fusion Middleware Administrator's Guide for Oracle Directory Integration Platform* provides details about the following SSL tasks for Oracle Directory Integration Platform:

- Configuring Oracle Directory Integration Platform for SSL Mode 2 Server-Only Authentication
- Managing the SSL Certificates of Oracle Internet Directory and Connected Directories
- Bootstrapping in SSL Mode
- Configuring the Third-Party Directory Connector for Synchronization in SSL Mode
- Configuring and Testing Oracle Internet Directory with SSL Server-Side Authentication
- Testing SSL Communication Between Oracle Internet Directory and Microsoft Active Directory

6.5.4.2 Configuring SSL for Oracle Access Management Identity Federation

See "Defining Keystore Settings for Federation in Oracle Access Management Console" in the *Oracle Fusion Middleware Administrator's Guide for Oracle Access Management*.

6.5.4.3 Configuring SSL for Oracle Directory Services Manager

You can configure Oracle Directory Services Manager to use SSL for communications with connected directories. The *Oracle Fusion Middleware Administrator's Guide for Oracle Virtual Directory* provides details about the following SSL tasks for Oracle Directory Services Manager:

- Logging into the Directory Server from Oracle Directory Services Manager Using SSL
- Managing Oracle Directory Services Manager's Key Store
- Storing Oracle Directory Services Manager's Certificate in Oracle Virtual Directory

6.5.5 SSL-Enable Oracle Reports, Forms, Discoverer, and Portal

This section contains these topics:

- SSL for Oracle Reports
- SSL for Oracle Forms
- SSL for Oracle Discoverer
- SSL for Oracle Portal

6.5.5.1 SSL for Oracle Reports

To SSL-enable Oracle Reports, you need to enable SSL on the components front-ending Oracle WebLogic Server.

For example, if you have an Oracle HTTP Server and an Oracle Web Cache front-ending the Oracle WebLogic Server that hosts Oracle Reports, you need to configure the following:

- Inbound SSL for Oracle Web Cache

  See Section 6.4.2.1.

- Inbound SSL for Oracle HTTP Server

  See Section 6.4.3.1.

- Inbound SSL for Oracle WebLogic Server

  See Section 6.5.1.1.

- SSL between Oracle Web Cache and Oracle HTTP Server

  See Section 6.4.2.3.

- SSL between Oracle HTTP Server and Oracle WebLogic Server

  See Section 6.4.3.3.

> **Note:** These steps are necessary only if you wish to set up end-to-end SSL. In most cases, it is sufficient to enable SSL only on the first component getting the request, since the other components are usually within the intranet.
>
> For example, if the request is sent to Oracle Web Cache, you may only need to follow the first step. If the request is sent to Oracle HTTP Server, you may only need to follow the second step. Select the steps as dictated by your topology.

Additionally, Oracle Reports in Fusion Middleware Control accesses the reports servlet for data. If that communication needs to take place over SSL, you must complete the manual procedure described in *Oracle Fusion Middleware Publishing Reports to the Web with Oracle Reports Services*.

6.5.5.2 SSL for Oracle Forms

To SSL-enable Oracle Forms, you need to enable SSL on the components front-ending Oracle WebLogic Server.

For example, if you have an Oracle HTTP Server and an Oracle Web Cache front-ending the Oracle WebLogic Server that hosts Oracle Forms, you need to configure the following:

- Inbound SSL for Oracle Web Cache

  See Section 6.4.2.1.

- Inbound SSL for Oracle HTTP Server

  See Section 6.4.3.1.

- Inbound SSL for Oracle WebLogic Server

  See Section 6.5.1.1.

- SSL between Oracle Web Cache and Oracle HTTP Server

  See Section 6.4.2.3.

- SSL between Oracle HTTP Server and Oracle WebLogic Server

  See Section 6.4.3.3.

> **Note:** These steps are necessary only if you wish to set up end-to-end SSL. In most cases, it is sufficient to enable SSL only on the first component getting the request, since the other components are usually within the intranet.
>
> For example, if the request is sent to Oracle Web Cache, you may only need to follow the first step. If the request is sent to Oracle HTTP Server, you may only need to follow the second step. Select the steps as dictated by your topology.

6.5.5.3 SSL for Oracle Discoverer

Running Oracle Discoverer over `https` requires certain tasks such as enabling SSL for the Oracle HTTP Server virtual host and Oracle Web Cache front-ending the Oracle WebLogic Server that hosts Oracle BI Discoverer, among others.

For details, see Configuring End-to-End Secure Sockets Layer for Discoverer in the *Oracle Fusion Middleware Configuration Guide for Oracle Business Intelligence Discoverer*.

6.5.5.4 SSL for Oracle Portal

Oracle Portal uses a number of different components (such as the Parallel Page Engine, Oracle HTTP Server, and Oracle Web Cache) each of which may act as a client or server in HTTP communication. As a result, each component involving Oracle Portal in the middle tier is individually configured for https.

For details, see the *Oracle Fusion Middleware Administrator's Guide for Oracle Portal*.

6.5.6 Client-Side SSL for Applications

For information on how to write SSL-enabled applications, see "Using SSL Authentication in Java Clients" in *Oracle Fusion Middleware Programming Security for Oracle WebLogic Server*.

For best practices, refer to Section 6.8.2.

6.6 Configuring SSL for the Data Tier

This section contains these topics:

- Enabling SSL on Oracle Internet Directory Listeners
- Enabling SSL on Oracle Virtual Directory Listeners
- Configuring SSL for the Database

6.6.1 Enabling SSL on Oracle Internet Directory Listeners

Out of the box, Oracle Internet Directory nodes are SSL-enabled in no-auth mode.

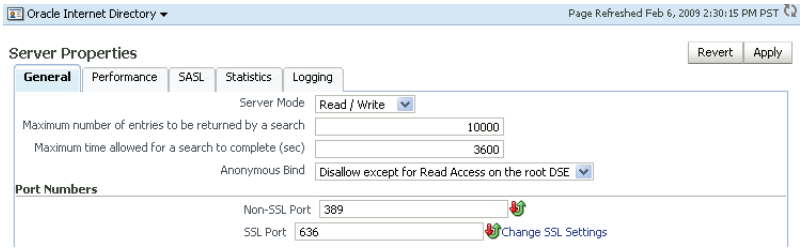
This section explains how to SSL-enable Oracle Internet Directory listeners using Fusion Middleware Control and the WLST command-line tool.

> **See Also:** For details of setting Up Oracle Internet Directory SSL Mutual Authentication Client and Server Authentication), see Note 1311791.1, which is available on My Oracle Support at `https://support.oracle.com/`.

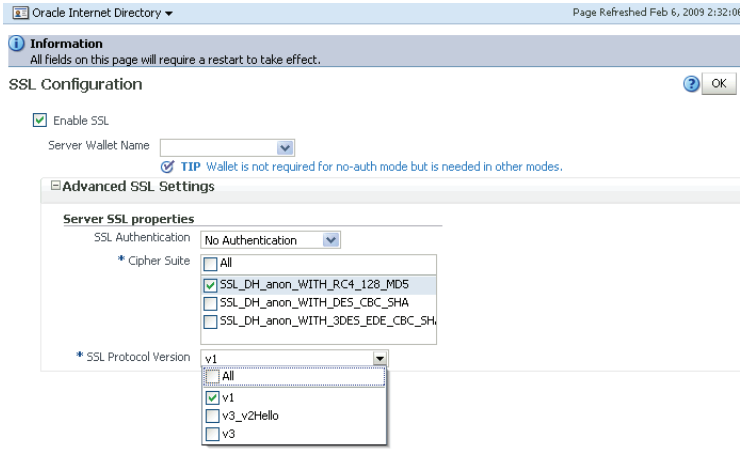
6.6.1.1 Enable Inbound SSL on an Oracle Internet Directory Listener Using Fusion Middleware Control

In this example, the following steps enable SSL in no-auth mode for an instance of Oracle Internet Directory using Fusion Middleware Control:

1. Select the Oracle Internet Directory instance in the navigation pane on the left.
2. Navigate to Oracle Internet Directory, then **Administration**, then **Server Properties**.



3. Click **Change SSL Settings**.
4. On the SSL Settings dialog:



- Select **Enable SSL**.
- Set SSL Authentication to **No Authentication**.
- Set Cipher Suite to **All**.
- Set SSL protocol version to **v3**.
- Click **OK**.

5. Restart the Oracle Internet Directory instance by navigating to **Oracle Internet Directory**, then **Control**, then **Restart.**
6. To verify that the instance is correctly SSL-enabled, execute an `ldapbind` command of the form:

```
ldapbind -D cn=orcladmin
   -U 1
   -h host
   -p SSL_port
```

> **Notes:** `-U 1` represents the no-auth mode.

For Oracle Internet Directory listeners in a stand-alone environment, see *Oracle Fusion Middleware Administrator's Guide for Oracle Internet Directory*.

SSL Enabling in Other Authentication Modes

The steps for SSL-enabling in other authentication modes are the same, except that in the SSL Settings dialog, you would set the appropriate authentication type.

> **Note:** Other authentication types need an Oracle wallet.

6.6.1.2 Enabling Inbound SSL on an Oracle Internet Directory Listener Using WLST

Configure the listener with SSL properties in `no-auth` mode as follows:

> **Note:** The Oracle Internet Directory port name is always `sslport1`.

```
configureSSL('inst1',
```

```
'oid1',
'oid',
'sslport1')
```

Note:

- `configureSSL` can use defaults for all SSL attributes; see Table 6–5 for details.
- We could also specify a properties file as a parameter to `configureSSL`; see Table 6–4 for details.
- See also Section 6.9.

SSL Enabling in Other Authentication Modes

You can do this by running the `configureSSL` command with a properties file as parameter and specifying an appropriate authentication type parameter value. For details, see the *Oracle Fusion Middleware Administrator's Guide for Oracle Internet Directory*.

6.6.1.3 Enabling Outbound SSL from Oracle Internet Directory to Oracle Database

Two sets of procedures are needed to configure SSL connections from Oracle Internet Directory to Oracle Database:

- Configure SSL for the Database
- Configure Outbound Oracle Internet Directory

Configure SSL for the Database

The steps to configure Oracle Database for SSL are described in Section 6.6.3.1.

Configure Outbound Oracle Internet Directory

Take these steps to configure SSL for outbound traffic from Oracle Internet Directory to Oracle Database:

1. Stop the Oracle Internet Directory server instances whose outbound traffic to the database is to be configured with SSL using this `opmnctl` syntax:

   ```
   $ORACLE_INSTANCE/bin/opmnctl stopproc ias-component=componentName
   ```

   For example:

   ```
   $ORACLE_INSTANCE/bin/opmnctl stopproc ias-component=oid1
   ```

2. Configure Security Socket Layer authentication on the database to which the Oracle Internet Directory server instance is connecting.

   For details, see *Oracle Database Advanced Security Administrator's Guide*.

3. Restart the database/listener as required.

4. Start Oracle Internet Directory server instances using this `opmnctl` syntax:

   ```
   $ORACLE_INSTANCE/bin/opmnctl startproc ias-component=componentName
   ```

   For example:

   ```
   $ORACLE_INSTANCE/bin/opmnctl startproc ias-component=oid1
   ```

Note: Only the no-authentication mode is supported.

6.6.2 Enabling SSL on Oracle Virtual Directory Listeners

This section explains how to enable SSL for an instance of Oracle Virtual Directory.

The *Oracle Fusion Middleware Administrator's Guide for Oracle Virtual Directory* provides additional information on these topics:

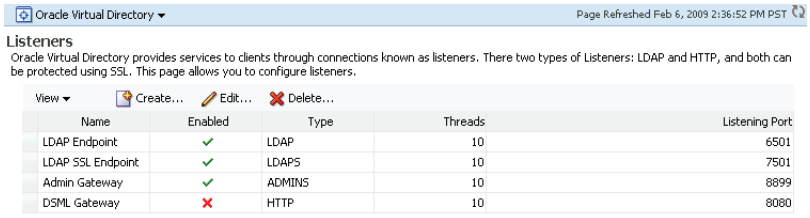
- Configuring SSL for Listeners Using Fusion Middleware Control
- Configuring SSL for Listeners Using WLST
- Configuring a Mutual Authentication SSL Connection Between Oracle Virtual Directory and Oracle Internet Directory

6.6.2.1 Enable SSL for Oracle Virtual Directory Using Fusion Middleware Control

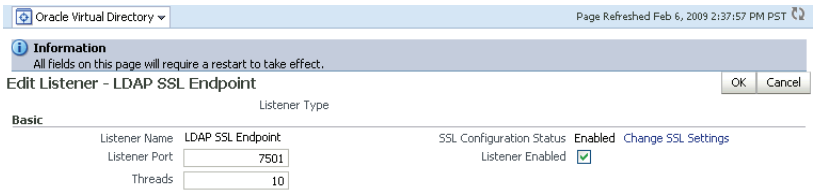
The steps to enable SSL are as follows (the example illustrates the `server-auth` mode):

1. Select the Oracle Virtual Directory instance in the navigation pane on the left.
2. Select a keystore to use for the operation by navigating to **Oracle Virtual Directory**, then **Security**, then **Keystores**

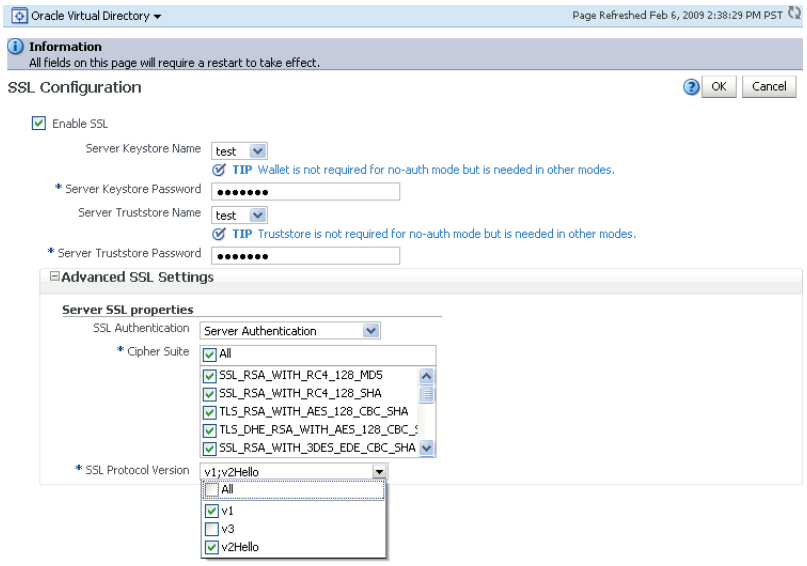
   Choose from the list of keystores that appears. If you need to generate a new keystore, see Section 8.3.3.1 for details.
3. To SSL-enable the listener, navigate to **Oracle Virtual Directory**, then **Administration**, then **Listeners**.
4. Select the LDAP SSL Endpoint listener, and click **Edit**.



   The Edit Listener page appears:



5. Click **Change SSL Settings**.
6. On the SSL Settings dialog:



- Select **Enable SSL**.
- For Server Keystore Name, select the keystore you created in step 2, for example, `OVDtestJks`.
- For Server Keystore Password, type the keystore password you specified in step 2.
- For Server Truststore Name, select the keystore you created in step 2, for example, `OVDtestJks`.
- For Server Truststore Password, type the keystore password you specified in step 2.
- Expand Advanced SSL Settings.
- For SSL authentication, select **Server Authentication**. This is the default setting.
- For Cipher Suite, select the applicable cipher suite, in this example `All`.
- Select a protocol version.
- Click **OK**.

7. Stop and start the Oracle Virtual Directory instance by navigating to **Oracle Virtual Directory**, then **Control**, then **Stop** and **Start.**
8. To verify that the instance is correctly SSL-enabled, execute an `ldapbind` command of the form:

```
ldapbind -D cn=orcladmin
   -U 2
   -h host
   -p SSL_port
-W "file:// DIRECTORY_SSL_WALLET"
```

Note:

- `-U 2` represents the server-auth mode.
- `DIRECTORY_SSL_WALLET` is the path to a wallet file, not including the wallet file name.
- This wallet must exist and must contain the trusted certificate of the CA that issued the server certificate.

SSL Enabling in Other Authentication Modes

The steps for SSL-enabling in other authentication modes are similar, except that in the SSL Settings dialog, you would set the appropriate authentication type.

Note: If configuring SSL for an LDAP listener, SSL communication is verified using `ldapbind`. If it is an http listener, it is verified using a browser.

6.6.2.2 Enabling SSL on an Oracle Virtual Directory Listener Using WLST

Take these steps to configure the listener in server-auth mode:

1. Determine the listeners for this Oracle Virtual Directory instance by running the following command:

   ```
   listListeners('inst1','ovd1' )
   ```

   This command lists all the listeners for instance `inst1` and component name `ovd1`; select the one that needs to be configured for SSL. For this example, select **LDAP SSL Endpoint**.

2. Obtain the name of the SSL MBean for the Oracle Virtual Directory listener:

   ```
   getSSLMBeanName('inst1',
      'ovd1',
      'ovd',
      'LDAP SSL Endpoint')
   ```

   This command will return the SSL MBean name for the specified instance, component name, component type, and listener.

3. Set the passwords for the keystore and truststore in the MBean with the following commands:

   ```
   cd ('SSL_MBean_Name')
   set('KeyStorePassword',java.lang.String('password').toCharArray())
   set('TrustStorePassword',java.lang.String('password').toCharArray())
   ```

4. Configure the listener with SSL properties:

   ```
   configureSSL('inst1',
      'ovd1',
      'ovd',
      'LDAP SSL Endpoint')
   ```

Note: Steps 2 and 3 are required only for server-auth and mutual-auth modes.

Enabling SSL in Other Authentication Modes

You can do this by running the `configureSSL` command with a properties file as parameter and specifying appropriate authentication type parameter value. For details, see "Creating and Managing Oracle Virtual Directory Listeners" in the *Oracle Fusion Middleware Administrator's Guide for Oracle Virtual Directory*.

6.6.3 Configuring SSL for the Database

This section contains these topics:

- SSL-Enable Oracle Database
- SSL-Enable a Data Source

6.6.3.1 SSL-Enable Oracle Database

Take these steps to SSL-enable Oracle database:

1. Create a root CA and a certificate for the DB. Here is an example:

> **Note:** Self-signed certificates are not recommended for production use. For information about obtain production wallets, see Section 8.4.8.3.

```
mkdir root
mkdir server

# Create root wallet, add self-signed certificate and export
orapki wallet create -wallet ./root -pwd password
orapki wallet add -wallet ./root -dn CN=root_test,C=US -keysize 2048 -self_
signed -validity 3650 -pwd password
orapki wallet display -wallet ./root -pwd password
orapki wallet export -wallet ./root -dn CN=root_test,C=US -cert
./root/b64certificate.txt -pwd password

#Create server wallet, add self-signed certificate and export
orapki wallet create -wallet ./server -pwd password
orapki wallet add -wallet ./server -dn CN=server_test,C=US -keysize 2048 -pwd
password
orapki wallet display -wallet ./server -pwd password
orapki wallet export -wallet ./server -dn CN=server_test,C=US -request
./server/creq.txt -pwd password

# Import trusted certificates
orapki cert create -wallet ./root -request ./server/creq.txt -cert
./server/cert.txt -validity 3650 -pwd password
orapki cert display -cert ./server/cert.txt -complete
orapki wallet add -wallet ./server -trusted_cert -cert
./root/b64certificate.txt -pwd password
orapki wallet add -wallet ./server -user_cert -cert ./server/cert.txt -pwd
password
orapki wallet create -wallet ./server -auto_login -pwd password}}
```

2. Update `listener.ora`, `sqlnet.ora`, and `tnsnames.ora` for the database.

   a. This example shows the default `listener.ora`:

```
SID_LIST_LISTENER =
(SID_LIST =(SID_DESC =(SID_NAME = PLSExtProc)(ORACLE_HOME = /path_to_O_
```

```
H)(PROGRAM = extproc)))
LISTENER =(DESCRIPTION_LIST =(DESCRIPTION =
(ADDRESS = (PROTOCOL = IPC)(KEY = EXTPROC1))
(ADDRESS = (PROTOCOL = TCP)(HOST = mynode.mycorp.com)(PORT = 1521))
(ADDRESS = (PROTOCOL = TCPS)(HOST = mynode.mycorp.com)(PORT = 2490))
))

WALLET_LOCATION=(SOURCE=(METHOD=FILE)(METHOD_DATA=(DIRECTORY=/wallet_
location)))

SSL_CLIENT_AUTHENTICATION=FALSE}}
```

And here is an updated `listener.ora` file, illustrating a scenario with no client authentication:

```
SID_LIST_LISTENER =
  (SID_LIST =
    (SID_DESC =
      (GLOBAL_DBNAME = dbname)
      (ORACLE_HOME = /path_to_O_H)
      (SID_NAME = sid)
    )
  )

SSL_CLIENT_AUTHENTICATION = FALSE

WALLET_LOCATION =
  (SOURCE =
    (METHOD = FILE)
    (METHOD_DATA =
      (DIRECTORY = /wallet_path)
    )
  )

LISTENER =
  (DESCRIPTION_LIST =
    (DESCRIPTION =
      (ADDRESS = (PROTOCOL = IPC)(KEY = EXTPROC1521))
    )
    (DESCRIPTION =
      (ADDRESS = (PROTOCOL = TCP)(HOST = mynode.mycorp.com)(PORT = 1521))
    )
    (DESCRIPTION =
      (ADDRESS = (PROTOCOL = TCPS)(HOST = mycorp.com)(PORT = 2490))
    )
  )
```

Note that the SSL port has been added.

b. Likewise, a modified `sqlnet.ora` file may look like this:

```
NAMES.DIRECTORY_PATH= (TNSNAMES, EZCONNECT)
SQLNET.AUTHENTICATION_SERVICES=(BEQ,TCPS,NTS)
WALLET_LOCATION=(SOURCE=(METHOD=FILE)(METHOD_DATA=(DIRECTORY=/directory)))
SSL_CLIENT_AUTHENTICATION=FALSE
```

c. A modified `tnsnames.ora` file may look like this:

```
OID =
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL = TCP)(HOST = mynode.mycorp.com)(PORT = 1521))
    (CONNECT_DATA =
```

```
      (SERVER = DEDICATED)
      (SERVICE_NAME = mynode.mycorp.com)
    )
  )

SSL =
  (DESCRIPTION =
    (ADDRESS_LIST =
      (ADDRESS = (PROTOCOL = TCPS)(HOST = mynode.mycorp.com)(PORT = 2490))
    )
    (CONNECT_DATA =
      (SERVICE_NAME = mynode.mycorp.com)
      or
      (SID = mynode.mycorp.com)
    )
    (SECURITY=(SSL_SERVER_CERT_DN=\"CN=server_test,C=US\"))
  )
```

3. Test the connection to the database using the new connect string. For example:

```
$ tnsping ssl
$ sqlplus username/password@ssl
```

See Also: The chapter "Configuring Secure Sockets Layer Authentication" in the *Oracle Database Advanced Security Administrator's Guide*.

6.6.3.2 SSL-Enable a Data Source

Take these steps to configure your data sources on Oracle WebLogic Server to use SSL.

1. Create a truststore and add the root certificate (which is created when SSL-enabling the database) as a trusted certificate to the truststore.
2. In the Oracle WebLogic Server Administration Console, navigate to the **Connection pool** tab of the data source that you are using.

Note: The data source can be an existing source such as an Oracle WebCenter Portal data source, or a new data source. See Creating a JDBC Data Source in *Oracle Fusion Middleware Configuring and Managing JDBC Data Sources for Oracle WebLogic Server* for details.

The properties you need to specify in the **JDBC Properties** text box depend on the type of authentication you wish to configure.

- If you will require client authentication (two-way authentication):

```
javax.net.ssl.keyStore=..(password of the keystore)
       javax.net.ssl.keyStoreType=JKS
       javax.net.ssl.keyStorePassword=...(password of the keystore)
       javax.net.ssl.trustStore=...(the truststore location on the disk)
       javax.net.ssl.trustStoreType=JKS
       javax.net.ssl.trustStorePassword=...(password of the truststore)
```

- If you will require no client authentication:

```
javax.net.ssl.trustStore=...(the truststore location on the disk)
       javax.net.ssl.trustStoreType=JKS
       javax.net.ssl.trustStorePassword=...(password of the truststore)
```

3. In the URL text box, enter the JDBC connect string. Ensure that the protocol is TCPS and that SSL_SERVER_CERT_DN contains the full DN of the database certificate.

   Use the following syntax if tnsnames.ora uses "SERVICE_NAME":

   ```
   jdbc:oracle:thin:@(DESCRIPTION=(ADDRESS_
   LIST=(ADDRESS=(PROTOCOL=TCPS)(HOST=host-name)(PORT=port-number)))(CONNECT_
   DATA=(SERVICE_NAME=service))(SECURITY=(SSL_SERVER_CERT_DN="CN=server_
   test,C=US")))
   ```

   Use the following syntax if tnsnames.ora uses "SID":

   ```
   jdbc:oracle:thin:@(DESCRIPTION=(ADDRESS_
   LIST=(ADDRESS=(PROTOCOL=TCPS)(HOST=host-name)(PORT=port-number)))(CONNECT_
   DATA=(SID=service))(SECURITY=(SSL_SERVER_CERT_DN="CN=server_test,C=US")))
   ```

4. Test and verify the connection. Your data source is now configured to use SSL.

6.7 Advanced SSL Scenarios

This section explains how to handle additional SSL configuration scenarios beyond the basic topologies described earlier:

- Hardware Security Modules and Accelerators
- CRL Integration with SSL
- Oracle Fusion Middleware FIPS 140-2 Settings

For details and examples of the commands used in this section see Section 6.9.

6.7.1 Hardware Security Modules and Accelerators

A Hardware Security Module (HSM) is a physical plug-in card or an external security device that can be attached to a computer to provide secure storage and use of sensitive content.

> **Note:** This discussion applies only to Oracle HTTP Server, Oracle Web Cache, and Oracle Internet Directory, which are the system components supporting HSM.

Oracle Fusion Middleware supports PKCS#11-compliant HSM devices that provide a secure storage for private keys.

Take these steps to implement SSL for a component using a PKCS#11 wallet:

1. Install the HSM libraries on the machine where the component is running. This is a one-time task and is device-dependent.
2. Next, create a wallet using Oracle Wallet Manager (OWM) or the `orapki` command-line tool. Note the following:
   a. Choose `PKCS11` as the wallet type.
   b. Specify the device-specific PKCS#11 library used to communicate with the device. This library is part of the HSM software.

      On Linux, the library is located at:

      ```
      For LunaSA (Safenet):  /usr/lunasa/lib/libCryptoki2.so
      For nCipher:  /opt/nfast/toolkits/pkcs11/libcknfast.so
      ```

On Windows, the library is located at:

```
For LunaSA (Safenet): C:\Program Files\LunaSA\cryptoki.dll
```

3. Now follow the standard procedure for obtaining third-party certificates, that is, creating a certificate request, getting the request approved by a Certificate Authority (CA), and installing the certificate signed by that CA.

   The wallet you set up is used like any other wallet.

4. Verify the wallet with the `orapki` utility. Use the following command syntax:

   ```
   orapki wallet p11_verify [-wallet [wallet]] [-pwd password]
   ```

   See Also: Appendix H, "Oracle Wallet Manager and orapki" for details about `orapki`

5. Configure SSL on your component listener using the `configureSSL` WLST command, providing a properties file as input. Your properties file should specify the full path of the PKCS#11 wallet directory on the machine where the component is running. (*Note*: Do not save the PKCS#11 wallet in the instance home directory. Only wallets created and managed through Fusion Middleware Control or WLST should reside in the instance home.)

   A sample properties file could look like this:

   ```
   SSLEnabled=true
   AuthenticationType=Server
   PKCS11Wallet=/tmp/lunasa/wallet
   ```

   Note: You must use the WLST command `configureSSL` to configure the PKCS11 wallet. You cannot do this task using Fusion Middleware Control or any other tool.

6.7.2 CRL Integration with SSL

Note:

- This discussion applies only to Oracle HTTP Server and Oracle Web Cache in the context of an Oracle WebLogic Server environment. For SSL configuration in standalone components, see *Oracle Fusion Middleware Administrator's Guide for Oracle HTTP Server* and *Oracle Fusion Middleware Administrator's Guide for Oracle Web Cache*.
- CRL validation is managed through WLST; you cannot perform this task through Fusion Middleware Control.

Components that use SSL can optionally turn on certificate validation using a certificate revocation list (CRL). This allows them to validate the peer certificate in the SSL handshake and ensure that it is not on the list of revoked certificates issued by the Certificate Authority (CA).

This section describes how to configure a component to use CRL-based validation, and how to create and set up CRLs on the file system.

6.7.2.1 Configuring CRL Validation for a Component

Configure SSL on your component listener using the `configureSSL` WLST command, providing a properties file as input.

The properties file must be set up as follows:

1. The CertValidation attribute must be set to `url`.
2. The CertValidationPath attribute must be of the form `file://file_path` or `dir://directory_path`.
   - Use the first format if you are using a single CRL file for certificate validation. This CRL file should contain a concatenation of all CRLs.
   - Use the second format if you are specifying a directory path that contains multiple CRL files in hashed form.

     See Section 6.7.2.2 on how to create CRLs in hashed form.

In this example, the properties file specifies a single CRL file:

```
SSLEnabled=true
AuthenticationType=Server
CertValidation=crl
KeyStore=ohs1
CertValidationPath=file:///tmp/file.crl
```

In this example, the properties file specifies a directory path to multiple CRL files:

```
SSLEnabled=true
AuthenticationType=Server
KeyStore=ohs1
CertValidation=crl
CertValidationPath=dir:///tmp
```

6.7.2.2 Manage CRLs on the File System

Note: LDAP-based CRLs or CRL distribution points are not supported.

You use the `orapki` command-line tool to manage CRLs on the file system. For details on this topic, see Section H.2.5.

CRL Renaming to Hashed Form

If specifying a CRL storage location, the CRL must be renamed. This enables CRLs to be loaded in an efficient manner at runtime. This operation creates a symbolic link to the actual CRL file. On Windows, the CRL is copied to a file with a new name.

To rename a CRL:

```
orapki crl hash
[-crl [url|filename]] [-wallet wallet] [-symlink directory]
[-copy directory] [-summary] [-pwd password]
```

For example:

```
orapki crl hash -crl nzcrl.txt -symlink wltdir -pwd password
```

If the CRL file name is specified at runtime, multiple CRLs can be concatenated in that file. The CRL created in this example is in Base64 format, and you can use a text editor to concatenate the CRLs.

CRL Creation

> **Note:** CRL creation and Certificate Revocation are for test purposes and only used in conjunction with self-signed certificates. For production use, obtain production certificates from well-known CAs and obtain the CRLs from those authorities.

To create a CRL:

```
orapki crl create
[-crl [url|filename]] [-wallet [cawallet]] [-nextupdate [days]] [-pwd password]
```

For example:

```
orapki crl create
-crl nzcrl.crl -wallet rootwlt -nextupdate 3650 -pwd password
```

Certificate Revocation

Revoking a certificate adds the certificate's serial number to the CRL.

To revoke a certificate:

```
orapki crl revoke
[-crl [url|filename]] [-wallet [cawallet]] [-cert   [revokecert]] [-pwd password]
```

For example:

```
orapki crl revoke
-crl nzcrl.txt -wallet rootwlt -cert cert.txt -pwd password
```

6.7.2.3 Test a Component Configured for CRL Validation

To test that a component is correctly configured for CRL validation, take these steps:

1. Set up a wallet with a certificate to be used in your component.
2. Generate a CRL with this certificate in the revoked certificates list. Follow the steps outlined in Section 6.7.2.2.
3. Configure your component to use this CRL. Follow the steps outlined in Section 6.7.2.1.
4. The SSL handshake should fail when this revoked certificate is used.

6.7.3 Oracle Fusion Middleware FIPS 140-2 Settings

This section describes how to configure Oracle Fusion Middleware components to comply with the FIPS 140-2 advanced security standard. Topics include:

- FIPS-Configurable Products
- Setting the SSLFIPS_140 Parameter
- Selecting Cipher Suites
- Other Configuration Parameters

See Also: For more information about this standard, refer to the Cryptographic Modules Validation Program Web site at:

http://csrc.nist.gov/groups/STM/index.html

6.7.3.1 FIPS-Configurable Products

Any product using the Oracle SSL SDK can be configured to run in the FIPS mode. Specifically, you can configure the following Oracle Fusion Middleware components:

- Oracle HTTP Server
- Oracle Web Cache
- Oracle Internet Directory

6.7.3.2 Setting the SSLFIPS_140 Parameter

You can configure these components to run in the `FIPS` mode by setting the `SSLFIPS_140` parameter to `TRUE` in the `fips.ora` file:

```
SSLFIPS_140=TRUE
```

This file does not exist out-of-the-box and has to be created. Locate `fips.ora` either in the `$ORACLE_HOME/ldap/admin` directory, or in the directory pointed to by the `FIPS_HOME` environment variable.

The SSLFIPS_140 parameter is set to `FALSE` by default. You must set it to `TRUE` for `FIPS` mode operation.

6.7.3.3 Selecting Cipher Suites

A cipher suite is a set of authentication, encryption, and data integrity algorithms used for exchanging messages between network nodes. During an SSL handshake, for example, the two nodes negotiate to see which cipher suite they will use when transmitting messages back and forth.

Only the following cipher suites are approved for use in FIPS mode:

```
SSL_RSA_WITH_3DES_EDE_CBC_SHA
SSL_DH_anon_WITH_3DES_EDE_CBC_SHA
TLS_RSA_WITH_AES_128_CBC_SHA
TLS_RSA_WITH_AES_256_CBC_SHA
```

Any other ciphers should not be used while running in FIPS mode.

You can configure one or more of these ciphers using comma-separated values. These should be specified in the SSL properties file for the key 'Ciphers' in the `WLST configureSSL` command, or through Fusion Middleware Control.

See Section 6.9.28 for details about specifying the SSL properties file with the `configureSSL` command.

6.7.3.4 Other Configuration Parameters

The minimum key size for enabling `FIPS` mode is 1024 bits. You need to ensure that the keys used in `FIPS` mode are at least 1024 bits. This is because the certificate in the wallet used by components like Oracle HTTP Server, Oracle Web Cache, and Oracle Internet Directory must have a minimum public key size of 1024 bits.

You can only use wallets created using Oracle tools like `SSLConfig`, Oracle Wallet Manager, or `orapki`. Third-party PKCS#12 wallet files cannot be used in `FIPS` mode.

6.8 Best Practices for SSL

This section outlines some best practices for Oracle Fusion Middleware component administrators and application developers. It contains these topics:

- Best Practices for Administrators
- Best Practices for Application Developers

6.8.1 Best Practices for Administrators

Best practices for system administrators include the following:

- Use self-signed wallets only in test environment. You should obtain a CA signed certificate in the wallet before moving to production environment. For details, see Chapter 8.
- It is recommended that components (at least in the Web tier) use certificates that have the system hostname or virtual host or site name as the DN. This allows browsers to connect in SSL mode without giving unsettling warning messages.
- A minimum key size of 1024 bits is recommended for certificates used for SSL. Higher key size provides more security but at the cost of reduced performance. Pick an appropriate key size value depending on your security and performance requirements.
- Lack of trust is one of the most common reasons for SSL handshake failures. Ensure that the client trusts the server (by importing the server CA certificate into the client keystore) before starting SSL handshake. If client authentication is also required, then the reverse should also be true.

6.8.2 Best Practices for Application Developers

The following practices are recommended:

- Use Java Key Store (JKS) to store certificates for your Java EE applications.
- Externalize SSL configuration parameters like keystore path, truststore path, and authentication type in a configuration file, rather than embedding these values in the application code. This allows you the flexibility to change SSL configuration without having to change the application itself.

6.9 WLST Reference for SSL

Starting with 11*g* Release 2 (11.1.2), WLST commands have been added to manage Oracle wallets and JKS keystores and to configure SSL for Oracle Fusion Middleware components.

Use the commands listed in Table 6–1,Table 6–2, and Table 6–3 for this task.

> **See Also:** Section 8.2 for important instructions on how to launch the WLST shell to run SSL-related commands. Do not launch the WLST interface from any other location.

Note: All WLST commands for SSL configuration must be run in online mode.

You can obtain help for each command by issuing:

```
help('command_name')
```

Certain commands require parameters like instance name, ias-component and process type. You can obtain this information with the command:

```
$ORACLE_INSTANCE/bin/opmnctl status
```

Table 6–1 WLST Commands for SSL Configuration

Use this command...	To...	Use with WLST...
configureSSL	Set the SSL attributes for a component listener.	Online
getSSL	Display the SSL attributes for a component listener.	Online

Table 6–2 WLST Commands for Oracle Wallet Management

Use this command...	To...	Use with WLST...
addCertificateRequest	Generate a certificate signing request in an Oracle wallet.	Online
addSelfSignedCertificate	Add a self-signed certificate to an Oracle wallet.	Online
changeWalletPassword	Change the password to an Oracle wallet.	Online
createWallet	Create an Oracle wallet.	Online
deleteWallet	Delete an Oracle wallet.	Online
exportWallet	Export an Oracle wallet to a file.	Online
exportWalletObject	Export an object (for example, a certificate) from an Oracle wallet to a file.	Online
getWalletObject	Display a certificate or other object present in an Oracle wallet.	Online
importWallet	Import an Oracle wallet from a file.	Online
importWalletObject	Import a certificate or other object from a file to an Oracle wallet.	Online
listWalletObjects	List all objects (such as certificates) present in an Oracle wallet.	Online
listWallets	List all Oracle wallets configured for a component instance.	Online
removeWalletObject	Remove a certificate or other object from a component instance's Oracle wallet.	Online

Table 6–3 WLST Commands for Java Keystore (JKS) Management

Use this command...	To...	Use with WLST...
changeKeyStorePassword	Change the password to a JKS keystore.	Online
createKeyStore	Create a JKS keystore.	Online
deleteKeyStore	Delete a JKS keystore.	Online
exportKeyStore	Export a JKS keystore to a file.	Online
exportKeyStoreObject	Export an object (for example, a certificate) from a JKS keystore to a file.	Online

Table 6–3 (Cont.) WLST Commands for Java Keystore (JKS) Management

Use this command...	To...	Use with WLST...
generateKey	Generate a keypair in a JKS keystore.	Online
getKeyStoreObject	Display a certificate or other object present in a JKS keystore.	Online
importKeyStore	Import a JKS keystore from a file.	Online
importKeyStoreObject	Import a certificate or other object from a file to a JKS keystore.	Online
listKeyStoreObjects	List all objects (for example, certificates) present in a JKS keystore.	Online
listKeyStores	List all JKS keystores configured for a component instance.	Online
removeKeyStoreObject	Remove a certificate or other object from a component instance's JKS keystore.	Online

Note: `WLST` allows you to import certificates only in PEM format.

6.9.1 addCertificateRequest

Online command that generates a certificate signing request in an Oracle wallet.

6.9.1.1 Description

This command generates a certificate signing request in Base64 encoded PKCS#10 format in an Oracle wallet for a component instance (Oracle HTTP Server, Oracle WebCache or Oracle Internet Directory). To get a certificate signed by a certificate authority (CA), send the certificate signing request to your CA.

6.9.1.2 Syntax

```
addCertificateRequest('instName', 'compName', 'compType', 'walletName',
'password', 'DN', 'keySize')
```

Argument	Definition
instName	Specifies the name of the application server instance.
compName	Specifies the name of the component instance.
compType	Specifies the type of component. Valid values are 'ohs', 'oid', and 'webcache'.
walletName	Specifies the name of the wallet file.
password	Specifies the password of the wallet.
DN	Specifies the Distinguished Name of the key pair entry.
keySize	Specifies the key size in bits.

6.9.1.3 Example

The following command generates a certificate signing request with DN `cn=www.acme.com` and key size `1024` in `wallet1`, for Oracle Internet Directory instance `oid1`, in application server instance `inst1`:

```
wls:/mydomain/serverConfig> addCertificateRequest('inst1', 'oid1',
'oid','wallet1', 'password', 'cn=www.acme.com', '1024',)
```

6.9.2 addSelfSignedCertificate

Online command that adds a self-signed certificate.

6.9.2.1 Description

This command creates a key pair and wraps it in a self-signed certificate in an Oracle wallet for the specified component instance (Oracle HTTP Server, Oracle WebCache or Oracle Internet Directory). Only keys based on the RSA algorithm are generated.

6.9.2.2 Syntax

```
addSelfSignedCertificate('instName', 'compName', 'compType', 'walletName',
                         'password', 'DN', 'keySize')
```

Argument	Definition
instName	Specifies the name of the application server instance.
compName	Specifies the name of the component instance.
compType	Specifies the type of component. Valid values are 'ohs', 'oid', and 'webcache'.
walletName	Specifies the name of the wallet file.
password	Specifies the password of the wallet.
DN	Specifies the Distinguished Name of the key pair entry.
keySize	Specifies the key size in bits.

6.9.2.3 Example

The following command adds a self-signed certificate with DN `cn=www.acme.com`, key size `1024` to `wallet1`, for Oracle Internet Directory instance `oid1`, in application server instance inst1:

```
wls:/mydomain/serverConfig> addSelfSignedCertificate('inst1', 'oid1',
'oid','wallet1', 'password', 'cn=www.acme.com', '1024')
```

6.9.3 changeKeyStorePassword

Online command that changes the keystore password.

6.9.3.1 Description

This command changes the password of a Java Keystore (JKS) file for an Oracle Virtual Directory instance.

6.9.3.2 Syntax

```
changeKeyStorePassword('instName', 'compName', 'compType', 'keystoreName',
'currPassword', 'newPassword')
```

Argument	Definition
instName	Specifies the name of the application server instance.
compName	Specifies the name of the component instance.

Argument	Definition
compType	Specifies the type of component. Valid value is 'ovd'.
keystoreName	Specifies the filename of the keystore.
currPassword	Specifies the current keystore password.
newPassword	Specifies the new keystore password.

6.9.3.3 Example

The following command changes the password of file `keys.jks` for Oracle Virtual Directory instance `ovd1` in application server instance `inst1`:

```
wls:/mydomain/serverConfig> changeKeyStorePassword('inst1', 'ovd1',
'ovd','keys.jks', 'currpassword', 'newpassword')
```

6.9.4 changeWalletPassword

Online command that changes the password of an Oracle wallet.

6.9.4.1 Description

This command changes the password of an Oracle wallet for the specified component instance (Oracle HTTP Server, Oracle WebCache or Oracle Internet Directory). This command is only applicable to password-protected wallets.

6.9.4.2 Syntax

```
changeWalletPassword('instName', 'compName', 'compType',
'walletName','currPassword', 'newPassword')
```

Argument	Definition
instName	Specifies the name of the application server instance.
compName	Specifies the name of the component instance.
compType	Specifies the type of component. Valid values are 'oid', 'ohs', and 'webcache'.
walletName	Specifies the filename of the wallet.
currPassword	Specifies the current wallet password.
newPassword	Specifies the new wallet password.

6.9.4.3 Example

The following command changes the password for `wallet1` from `currpassword` to `newpassword` for Oracle HTTP Server instance `ohs1` in application server instance `inst1`:

```
wls:/mydomain/serverConfig> changeWalletPassword('inst1', 'ohs1', 'ohs','wallet1',
'currpassword', 'newpassword')
```

6.9.5 configureSSL

Online command that sets SSL attributes.

6.9.5.1 Description

This command sets the SSL attributes for a component listener. The attributes are specified in a properties file format (name=value). If a properties file is not provided, or it does not contain any SSL attributes, default attribute values are used.

For details about the format of properties files, see Section 6.9.28.

6.9.5.2 Syntax

```
configureSSL('instName', 'compName', 'compType', 'listener', 'filePath')
```

Argument	Definition
instName	Specifies the name of the application server instance.
compName	Specifies the name of the component instance.
compType	Specifies the type of component. Valid values are 'oid', 'ovd', ohs', and 'webcache'.
listener	Specifies the name of the component listener to be configured for SSL.
filePath	Specifies the absolute path of the properties file containing the SSL attributes to set.

6.9.5.3 Examples

The following command configures SSL attributes specified in the properties file `/tmp/ssl.properties` for Oracle Virtual Directory instance `ovd1` in application server instance `inst1`, for listener `listener1`:

```
wls:/mydomain/serverConfig> configureSSL('inst1', 'ovd1', 'ovd',
'listener1','/tmp/ssl.properties')
```

The following command configures SSL attributes without specifying a properties file. Since no file is provided, the default SSL attribute values are used:

```
wls:/mydomain/serverConfig> configureSSL('inst1', 'ovd1', 'ovd', 'listener2')
```

6.9.6 createKeyStore

Online command that creates a JKS keystore.

6.9.6.1 Description

This command creates a Java keystore (JKS) for the specified Oracle Virtual Directory instance. For keystore file location and other information, see Section 8.3.6.1.

6.9.6.2 Syntax

```
createKeyStore('instName', 'compName', 'compType', 'keystoreName', 'password')
```

Argument	Definition
instName	Specifies the name of the application server instance.
compName	Specifies the name of the component instance.
compType	Specifies the type of component. Valid value is 'ovd'.
keystoreName	Specifies the filename of the keystore file to be created.
password	Specifies the keystore password.

6.9.6.3 Example

The following command creates JKS file `keys.jks` with the password *`password`* for Oracle Virtual Directory instance `ovd1` in application server instance `inst1`:

```
wls:/mydomain/serverConfig> createKeyStore('inst1', 'ovd1', 'ovd','keys.jks',
'password')
```

6.9.7 createWallet

Online command that creates an Oracle wallet.

6.9.7.1 Description

This command creates an Oracle wallet for the specified component instance (Oracle HTTP Server, Oracle WebCache or Oracle Internet Directory). Wallets can be of password-protected or auto-login type. For wallet details, see Chapter 8.

6.9.7.2 Syntax

```
createWallet('instName', 'compName', 'compType', 'walletName', 'password')
```

Argument	Definition
instName	Specifies the name of the application server instance.
compName	Specifies the name of the component instance.
compType	Specifies the type of component. Valid values are 'oid', 'ohs', and 'webcache'.
walletName	Specifies the name of the wallet file to be created.
password	Specifies the wallet password.

6.9.7.3 Examples

The following command creates a wallet named `wallet1` with password `password`, for Oracle HTTP Server instance `ohs1` in application server instance `inst1`:

```
wls:/mydomain/serverConfig> createWallet('inst1', 'ohs1', 'ohs','wallet1',
'password')
```

The following command creates an auto-login wallet named `wallet2` for Oracle WebCache instance `wc1`, in application server instance `inst1`:

```
wls:/mydomain/serverConfig> createWallet('inst1', 'wc1', 'webcache','wallet2', '')
```

6.9.8 deleteKeyStore

Online command that deletes a keystore.

6.9.8.1 Description

This command deletes a keystore for a specified Oracle Virtual Directory instance.

6.9.8.2 Syntax

```
deleteKeyStore('instName', 'compName', 'compType', 'keystoreName')
```

Argument	Definition
instName	Specifies the name of the application server instance.

Argument	Definition
compName	Specifies the name of the component instance.
compType	Specifies the type of component. Valid value is 'ovd'.
keystoreName	Specifies the name of the keystore file to delete.

6.9.8.3 Example

The following command deletes JKS file `keys.jks` for Oracle Virtual Directory instance `ovd1` in application server instance `inst1`:

```
wls:/mydomain/serverConfig> deleteKeyStore('inst1', 'ovd1', 'ovd','keys.jks')
```

6.9.9 deleteWallet

Online command that deletes an Oracle wallet.

6.9.9.1 Description

This command deletes an Oracle wallet for the specified component instance (Oracle HTTP Server, Oracle WebCache or Oracle Internet Directory).

6.9.9.2 Syntax

```
deleteWallet('instName', 'compName', 'compType', 'walletName')
```

Argument	Definition
instName	Specifies the name of the application server instance.
compName	Specifies the name of the component instance.
compType	Specifies the type of component. Valid values are 'oid', 'ohs', and 'webcache'.
walletName	Specifies the name of the wallet file to be deleted.

6.9.9.3 Example

The following command deletes a wallet named `wallet1` for Oracle HTTP Server instance `ohs1` in application server instance `inst1`:

```
wls:/mydomain/serverConfig> deleteWallet('inst1', 'ohs1', 'ohs','wallet1')
```

6.9.10 exportKeyStore

Online command that exports the keystore to a file.

6.9.10.1 Description

This command exports a keystore, configured for the specified Oracle Virtual Directory instance, to a file under the given directory. The exported filename is the same as the keystore name.

6.9.10.2 Syntax

```
exportKeyStore('instName', 'compName', 'compType', 'keystoreName',
               'password', 'path')
```

Argument	Definition
instName	Specifies the name of the application server instance.
compName	Specifies the name of the component instance.
compType	Specifies the type of component. Valid value is 'ovd'.
keystoreName	Specifies the name of the keystore file.
password	Specifies the password of the keystore.
path	Specifies the absolute path of the directory under which the keystore is exported.

6.9.10.3 Example

The following command exports the keystore `keys.jks` for Oracle Virtual Directory instance `ovd1` to file `keys.jks` under `/tmp`:

```
wls:/mydomain/serverConfig> exportKeyStore('inst1', 'ovd1', 'ovd', 'keys.jks',
'password', '/tmp')
```

6.9.11 exportKeyStoreObject

Online command that exports an object from a keystore to a file.

6.9.11.1 Description

This command exports a certificate signing request, certificate/certificate chain, or trusted certificate present in a Java keystore (JKS) to a file for the specified Oracle Virtual Directory instance. The certificate signing request is generated before exporting the object. The alias specifies the object to be exported.

6.9.11.2 Syntax

```
exportKeyStoreObject('instName', 'compName', 'compType', 'keystoreName',
'password', 'type', 'path', 'alias')
```

Argument	Definition
instName	Specifies the name of the application server instance.
compName	Specifies the name of the component instance.
compType	Specifies the type of component. Valid value is 'ovd'.
keystoreName	Specifies the name of the keystore file.
password	Specifies the password of the keystore.
type	Specifies the type of the keystore object to be exported. Valid values are 'CertificateRequest', 'Certificate', 'TrustedCertificate' and 'TrustedChain'.
path	Specifies the absolute path of the directory under which the object is exported as a file named base64.txt.
alias	Specifies the alias of the keystore object to be exported.

6.9.11.3 Examples

The following command generates and exports a certificate signing request from the key-pair indicated by alias `mykey` in `keys.jks`, for Oracle Virtual Directory instance

ovd1 in application server instance inst1. The certificate signing request is exported under the directory /tmp:

```
wls:/mydomain/serverConfig> exportKeyStoreObject('inst1', 'ovd1',
'ovd','keys.jks', 'password', 'CertificateRequest', '/tmp','mykey')
```

The following command exports a certificate or certificate chain indicated by alias mykey in keys.jks, for Oracle Virtual Directory instance ovd1, in application server instance inst1. The certificate or certificate chain is exported under the directory /tmp:

```
wls:/mydomain/serverConfig> exportKeyStoreObject('inst1', 'ovd1',
'ovd','keys.jks', 'password', 'Certificate', '/tmp','mykey')
```

The following command exports a trusted certificate indicated by alias mykey in keys.jks, for Oracle Virtual Directory instance ovd1, in application server instance inst1. The trusted certificate is exported under the directory /tmp:

```
wls:/mydomain/serverConfig> exportKeyStoreObject('inst1', 'ovd1',
'ovd','keys.jks', 'password', 'TrustedCertificate', '/tmp','mykey')
```

6.9.12 exportWallet

Online command that exports an Oracle wallet.

6.9.12.1 Description

This command exports an Oracle wallet, configured for a specified component instance (Oracle HTTP Server, Oracle WebCache or Oracle Internet Directory), to files under the given directory. If the exported file is an auto-login only wallet, the file name is cwallet.sso. If it is password-protected wallet, two files are created—ewallet.p12 and cwallet.sso.

6.9.12.2 Syntax

```
exportWallet('instName', 'compName', 'compType', 'walletName','password', 'path')
```

Argument	Definition
instName	Specifies the name of the application server instance.
compName	Specifies the name of the component instance.
compType	Specifies the type of component. Valid values are 'oid', 'ohs', and 'webcache'.
walletName	Specifies the name of the wallet file.
password	Specifies the password of the wallet.
path	Specifies the absolute path of the directory under which the object is exported.

6.9.12.3 Examples

The following command exports auto-login wallet wallet1 for Oracle Internet Directory instance oid1 to file cwallet.sso under /tmp:

```
wls:/mydomain/serverConfig> exportWallet('inst1', 'oid1', 'oid',
'wallet1','','/tmp')
```

The following command exports password-protected wallet `wallet2` for Oracle Internet Directory instance `oid1` to two files, `ewallet.p12` and `cwallet.sso`, under `/tmp`:

```
wls:/mydomain/serverConfig> exportWallet('inst1', 'oid1', 'oid', 'wallet2',
'password', '/tmp')
```

6.9.13 exportWalletObject

Online command that exports a certificate or other wallet object to a file.

6.9.13.1 Description

This command exports a certificate signing request, certificate, certificate chain or trusted certificate present in an Oracle wallet to a file for the specified component instance (Oracle HTTP Server, Oracle WebCache or Oracle Internet Directory). DN indicates the object to be exported.

6.9.13.2 Syntax

```
exportWalletObject('instName', 'compName', 'compType', 'walletName', 'password',
'type', 'path', 'DN')
```

Argument	Definition
instName	Specifies the name of the application server instance.
compName	Specifies the name of the component instance.
compType	Specifies the type of component. Valid values are 'ohs', 'oid', and 'webcache'.
walletName	Specifies the name of the wallet file.
password	Specifies the password of the wallet.
type	Specifies the type of wallet object to be exported. Valid values are 'CertificateRequest', 'Certificate', 'TrustedCertificate' or 'TrustedChain'.
path	Specifies the absolute path of the directory under which the object is exported as a file base64.txt.
DN	Specifies the Distinguished Name of the wallet object being exported.

6.9.13.3 Examples

The following command exports a certificate signing request with DN `cn=www.acme.com` in `wallet1`, for Oracle Internet Directory instance `oid1`, in application server instance `inst1`. The certificate signing request is exported under the directory `/tmp`:

```
wls:/mydomain/serverConfig> exportWalletObject('inst1', 'oid1', 'oid','wallet1',
'password', 'CertificateRequest', '/tmp','cn=www.acme.com')
```

The following command exports a certificate with DN `cn=www.acme.com` in `wallet1`, for Oracle Internet Directory instance `oid1`, in application server instance `inst1`. The certificate or certificate chain is exported under the directory `/tmp`:

```
wls:/mydomain/serverConfig> exportWalletObject('inst1', 'oid1', 'oid','wallet1',
'password', 'Certificate', '/tmp','cn=www.acme.com')
```

The following command exports a trusted certificate with DN `cn=www.acme.com` in `wallet1`, for Oracle Internet Directory instance `oid1`, in application server instance `inst1`. The trusted certificate is exported under the directory `/tmp`:

```
wls:/mydomain/serverConfig> exportWalletObject('inst1', 'oid1', 'oid','wallet1',
'password', 'TrustedCertificate', '/tmp','cn=www.acme.com')
```

The following command exports a certificate chain with DN `cn=www.acme.com` in `wallet1`, for Oracle Internet Directory instance `oid1`, in application server instance `inst1`. The certificate or certificate chain is exported under the directory `/tmp`:

```
wls:/mydomain/serverConfig> exportWalletObject('inst1', 'oid1', 'oid','wallet1',
'password', 'TrustedChain', '/tmp','cn=www.acme.com')
```

6.9.14 generateKey

Online command that generates a key pair in a Java keystore.

6.9.14.1 Description

This command generates a key pair in a Java keystore (JKS) for Oracle Virtual Directory. It also wraps the key pair in a self-signed certificate. Only keys based on the RSA algorithm are generated.

6.9.14.2 Syntax

```
generateKey('instName', 'compName', 'compType', 'keystoreName', 'password', 'DN',
'keySize', 'alias', 'algorithm')
```

Argument	Definition
instName	Specifies the name of the application server instance.
compName	Specifies the name of the component instance.
compType	Specifies the type of component. Valid value is 'ovd'.
keystoreName	Specifies the name of the keystore.
password	Specifies the password of the keystore.
DN	Specifies the Distinguished Name of the key pair entry.
keySize	Specifies the key size in bits.
alias	Specifies the alias of the key pair entry in the keystore.
algorithm	Specifies the key algorithm. Valid value is 'RSA'.

6.9.14.3 Examples

The following command generates a key pair with DN `cn=www.acme.com`, key size `1024`, algorithm `RSA` and alias `mykey` in `keys.jks`, for Oracle Virtual Directory instance `ovd1` in application server instance `inst1`:

```
wls:/mydomain/serverConfig> generateKey('inst1', 'ovd1', 'ovd','keys.jks',
'password', 'cn=www.acme.com', '1024', 'mykey', 'RSA')
```

The following command is the same as above, except it does not explicitly specify the key algorithm:

```
wls:/mydomain/serverConfig> generateKey('inst1', 'ovd1', 'ovd','keys.jks',
'password', 'cn=www.acme.com', '1024', 'mykey')
```

6.9.15 getKeyStoreObject

Online command that shows details about a keystore object.

6.9.15.1 Description

This command displays a specific certificate or trusted certificate present in a Java keystore (JKS) for Oracle Virtual Directory. The keystore object is indicated by its index number, as given by the `listKeyStoreObjects` command. It shows the certificate details including DN, key size, algorithm, and other information.

6.9.15.2 Syntax

```
getKeyStoreObject('instName', 'compName', 'compType', 'keystoreName', 'password',
'type', 'index')
```

Argument	Definition
instName	Specifies the name of the application server instance.
compName	Specifies the name of the component instance.
compType	Specifies the type of component. Valid value is 'ovd'.
keystoreName	Specifies the name of the keystore file.
password	Specifies the password of the keystore.
type	Specifies the type of the keystore object to be listed. Valid values are 'Certificate' and 'TrustedCertificate'.
index	Specifies the index number of the keystore object as returned by the `listKeyStoreObjects` command.

6.9.15.3 Examples

The following command shows a trusted certificate with index `1` present in `keys.jks`, for Oracle Virtual Directory instance `ovd1`, in application server instance `inst1`:

```
wls:/mydomain/serverConfig> getKeyStoreObject('inst1', 'ovd1', 'ovd','keys.jks',
'password', 'TrustedCertificate', '1')
```

The following command shows a certificate with index `1` present in `keys.jks`, for Oracle Virtual Directory instance `ovd1`, in application server instance `inst1`:

```
wls:/mydomain/serverConfig> getKeyStoreObject('inst1', 'ovd1', 'ovd','keys.jks',
'password', 'Certificate', '1')
```

6.9.16 getSSL

Online command that lists the configured SSL attributes.

6.9.16.1 Description

This command lists the configured SSL attributes for the specified component listener. For Oracle Internet Directory, the listener name is always `sslport1`.

6.9.16.2 Syntax

```
getSSL('instName', 'compName', 'compType', 'listener')
```

Argument	Definition
instName	Specifies the name of the application server instance.
compName	Specifies the name of the component instance.

Argument	Definition
compType	Specifies the type of component. Valid values are 'ovd', 'oid', 'ohs', and 'webcache'.
listener	Specifies the name of the component listener.

6.9.16.3 Example

The following command shows the SSL attributes configured for Oracle Internet Directory instance `oid1`, in application server instance `inst1`, for listener `sslport1`:

```
wls:/mydomain/serverConfig> getSSL('inst1', 'oid1', 'oid', 'sslport1')
```

6.9.17 getWalletObject

Online command that displays information about a certificate or other object in an Oracle wallet.

6.9.17.1 Description

This command displays a specific certificate signing request, certificate or trusted certificate present in an Oracle wallet for the specified component instance (Oracle HTTP Server, Oracle WebCache or Oracle Internet Directory). The wallet object is indicated by its index number, as given by the `listWalletObjects` command. For certificates or trusted certificates, it shows the certificate details including DN, key size, algorithm and other data. For certificate signing requests, it shows the subject DN, key size and algorithm.

6.9.17.2 Syntax

```
getWalletObject('instName', 'compName', 'compType', 'walletName', 'password',
'type', 'index')
```

Argument	Definition
instName	Specifies the name of the application server instance.
compName	Specifies the name of the component instance.
compType	Specifies the type of component. Valid values are 'ohs', 'oid', and 'webcache'.
walletName	Specifies the name of the wallet file.
password	Specifies the password of the wallet.
type	Specifies the type of wallet object to be exported. Valid values are 'CertificateRequest', 'Certificate', and 'TrustedCertificate'.
index	Specifies the index number of the wallet object as returned by the `listWalletObjects` command.

6.9.17.3 Examples

The following command shows certificate signing request details for the object with index `0` present in `wallet1`, for Oracle Internet Directory instance `oid1`, in application server instance `inst1`:

```
wls:/mydomain/serverConfig> getKeyStoreObject('inst1', 'oid1',
'oid','wallet1','password', 'CertificateRequest', '0')
```

The following command shows certificate details for the object with index `0` present in `wallet1`, for Oracle Internet Directory instance `oid1`, in application server instance `inst1`:

```
wls:/mydomain/serverConfig> getKeyStoreObject('inst1', 'oid1',
'oid','wallet1','password', 'Certificate', '0')
```

The following command shows trusted certificate details for the object with index `0`, present in `wallet1`, for Oracle Internet Directory instance `oid1`, in application serverinstance `inst1`:

```
wls:/mydomain/serverConfig> getKeyStoreObject('inst1', 'oid1',
'oid','wallet1','password', 'TrustedCertificate', '0')
```

6.9.18 importKeyStore

Online command that imports a keystore from a file.

6.9.18.1 Description

This command imports a Java keystore (JKS) from a file to the specified Oracle Virtual Directory instance for manageability. The component instance name must be unique.

6.9.18.2 Syntax

```
importKeyStore('instName', 'compName', 'compType', 'keystoreName',
'password', 'filePath')
```

Argument	Definition
instName	Specifies the name of the application server instance.
compName	Specifies the name of the component instance.
compType	Specifies the type of component. Valid value is 'ovd'.
keystoreName	Specifies the name of the keystore being imported. This name must be unique for this component instance.
password	Specifies the password of the keystore.
filePath	Specifies the absolute path of the keystore file to be imported.

6.9.18.3 Example

The following command imports the keystore `/tmp/keys.jks` as `file.jks` into Oracle Virtual Directory instance `ovd1`. Subsequently, the keystore is managed through the name `file.jks`:

```
wls:/mydomain/serverConfig> importKeyStore('inst1', 'ovd1', 'ovd', 'file.jks',
'password', '/tmp/keys.jks')
```

6.9.19 importKeyStoreObject

Online command that imports an object from a file to a keystore.

6.9.19.1 Description

This command imports a certificate, certificate chain, or trusted certificate into a Java keystore (JKS) for Oracle Virtual Directory, assigning it the specified alias which must be unique in the keystore. If a certificate or certificate chain is being imported, the alias must match that of the corresponding key-pair.

6.9.19.2 Syntax

```
importKeyStoreObject('instName', 'compName', 'compType', 'keystoreName',
'password', 'type', 'filePath', 'alias')
```

Argument	Definition
instName	Specifies the name of the application server instance.
compName	Specifies the name of the component instance.
compType	Specifies the type of component. Valid value is 'ovd'.
keystoreName	Specifies the name of the keystore.
password	Specifies the password of the keystore.
type	Specifies the type of the keystore object to be imported. Valid values are 'Certificate' and 'TrustedCertificate'.
filePath	Specifies the absolute path of the file containing the keystore object.
alias	Specifies the alias to assign to the keystore object to be imported.

6.9.19.3 Examples

The following command imports a certificate or certificate chain from file `cert.txt` into `keys.jks`, using alias `mykey` for Oracle Virtual Directory instance `ovd1`, in application server instance `inst1`. The file `keys.jks` must already have an alias `mykey` for a key-pair whose public key matches that in the certificate being imported:

```
wls:/mydomain/serverConfig> > importKeyStoreObject('inst1', 'ovd1',
'ovd','keys.jks', 'password', 'Certificate','/tmp/cert.txt', 'mykey')
```

The following command imports a trusted certificate from file `trust.txt` into `keys.jks` using alias `mykey1`, for Oracle Virtual Directory instance `ovd1` in application server instance `inst1`:

```
wls:/mydomain/serverConfig> importKeyStoreObject('inst1', 'ovd1',
'ovd','keys.jks', 'password', 'TrustedCertificate','/tmp/trust.txt', 'mykey1')
```

6.9.20 importWallet

Online command that imports an Oracle wallet from a file.

6.9.20.1 Description

This command imports an Oracle wallet from a file to the specified component instance (Oracle HTTP Server, Oracle WebCache, or Oracle Internet Directory) for manageability. If the wallet being imported is an auto-login wallet, the file path must point to `cwallet.sso`; if the wallet is password-protected, it must point to `ewallet.p12`. The wallet name must be unique for the component instance.

6.9.20.2 Syntax

```
importWallet('instName', 'compName', 'compType', 'walletName', 'password',
'filePath')
```

Argument	Definition
instName	Specifies the name of the application server instance.
compName	Specifies the name of the component instance.

Argument	Definition
compType	Specifies the type of component. Valid values are 'ohs', 'oid', and 'webcache'.
walletName	Specifies the name of the wallet being imported. The name must be unique for the component instance.
password	Specifies the password of the wallet.
filePath	Specifies the absolute path of the wallet file being imported.

6.9.20.3 Examples

The following command imports the auto-login wallet file `/tmp/cwallet.sso` as `wallet1` into Oracle Internet Directory instance `oid1`. Subsequently, the wallet is managed with the name `wallet1`. No password is passed since it is an auto-login wallet:

```
wls:/mydomain/serverConfig> importWallet('inst1', 'oid1', 'oid', 'wallet1', '',
'/tmp/cwallet.sso')
```

The following command imports password-protected wallet `/tmp/ewallet.p12` as `wallet2` into Oracle Internet Directory instance `oid1`. Subsequently, the wallet is managed with the name `wallet2`. The wallet password is passed as a parameter:

```
wls:/mydomain/serverConfig> importWallet('inst1', 'oid1', 'oid', 'wallet2',
'password', '/tmp/ewallet.p12')
```

6.9.21 importWalletObject

Online command that imports a certificate or other object into an Oracle wallet.

6.9.21.1 Description

This command imports a certificate, trusted certificate or certificate chain into an Oracle wallet for the specified component instance (Oracle HTTP Server, Oracle WebCache component or Oracle Internet Directory). When importing a certificate, use the same wallet file from which the certificate signing request was generated.

6.9.21.2 Syntax

```
importWalletObject('instName', 'compName', 'compType', 'walletName', 'password',
'type', 'filePath')
```

Argument	Definition
instName	Specifies the name of the application server instance.
compName	Specifies the name of the component instance.
compType	Specifies the type of component. Valid values are 'ohs', 'oid', and 'webcache'.
walletName	Specifies the name of the wallet file.
password	Specifies the password of the wallet.
type	Specifies the type of wallet object to be imported. Valid values are 'Certificate', 'TrustedCertificate' and 'TrustedChain'.
filePath	Specifies the absolute path of the file containing the wallet object.

6.9.21.3 Examples

The following command imports a certificate chain in PKCS#7 format from file `chain.txt` into `wallet1`, for Oracle Internet Directory instance `oid1`, in application server instance `inst1`:

```
wls:/mydomain/serverConfig> importWalletObject('inst1', 'oid1', 'oid','wallet1',
'password', 'TrustedChain','/tmp/chain.txt')
```

The following command imports a certificate from file `cert.txt` into `wallet1`, for Oracle Internet Directory instance `oid1`, in application server instance `inst1`:

```
wls:/mydomain/serverConfig> > importWalletObject('inst1', 'oid1', 'oid','wallet1',
'password', 'Certificate','/tmp/cert.txt')
```

The following command imports a trusted certificate from file `trust.txt` into `wallet1`, for Oracle Internet Directory instance `oid1`, in application server instance `inst1`:

```
wls:/mydomain/serverConfig> importWalletObject('inst1', 'oid1', 'oid','wallet1',
'password', 'TrustedCertificate','/tmp/trust.txt')
```

6.9.22 listKeyStoreObjects

Online command that lists the contents of a keystore.

6.9.22.1 Description

This command lists all the certificates or trusted certificates present in a Java keystore (JKS) for Oracle Virtual Directory.

6.9.22.2 Syntax

```
listKeyStoreObjects('instName', 'compName', 'compType', 'keystoreName',
'password', 'type')
```

Argument	Definition
instName	Specifies the name of the application server instance.
compName	Specifies the name of the component instance.
compType	Specifies the type of component. Valid value is 'ovd'.
keystoreName	Specifies the name of the keystore file.
password	Specifies the password of the keystore.
type	Specifies the type of keystore object to be listed. Valid values are 'Certificate' and 'TrustedCertificate'.

6.9.22.3 Examples

The following command lists all trusted certificates present in `keys.jks`, for Oracle Virtual Directory instance `ovd1`, in application server instance `inst1`:

```
wls:/mydomain/serverConfig> listKeyStoreObjects('inst1', 'ovd1', 'ovd','keys.jks',
'password', 'TrustedCertificate')
```

The following command lists all certificates present in `keys.jks`, for Oracle Virtual Directory instance `ovd1`, in application server instance `inst1`:

```
wls:/mydomain/serverConfig> listKeyStoreObjects('inst1', 'ovd1', 'ovd','keys.jks',
'password', 'Certificate')
```

6.9.23 listKeyStores

Online command that lists all the keystores for a component.

6.9.23.1 Description

This command lists all the Java keystores (JKS) configured for the specified Oracle Virtual Directory instance.

6.9.23.2 Syntax

```
listKeyStores('instName', 'compName', 'compType')
```

Argument	Definition
instName	Specifies the name of the application server instance.
compName	Specifies the name of the component instance
compType	Specifies the type of component. Valid value is 'ovd'.

6.9.23.3 Example

The following command lists all keystores for Oracle Virtual Directory instance `ovd1` in application server instance `inst1`:

```
wls:/mydomain/serverConfig> listKeyStores('inst1', 'ovd1', 'ovd')
```

6.9.24 listWalletObjects

Online command that lists all objects in an Oracle wallet.

6.9.24.1 Description

This command lists all certificate signing requests, certificates, or trusted certificates present in an Oracle wallet for the specified component instance (Oracle HTTP Server, Oracle WebCache or Oracle Internet Directory).

6.9.24.2 Syntax

```
listWalletObjects('instName', 'compName', 'compType', 'walletName', password',
'type')
```

Argument	Definition
instName	Specifies the name of the application server instance.
compName	Specifies the name of the component instance.
compType	Specifies the type of component. Valid values are 'ohs','oid', and 'webcache'.
walletName	Specifies the name of the wallet file.
password	Specifies the password of the wallet.
type	Specifies the type of wallet object to be listed. Valid values are 'CertificateRequest', 'Certificate', and 'TrustedCertificate'.

6.9.24.3 Examples

The following command lists all certificate signing requests in `wallet1`, for Oracle Internet Directory instance `oid1`, in application server instance `inst1`:

```
wls:/mydomain/serverConfig> > listWalletObjects('inst1', 'oid1',
'oid','wallet1','password', 'CertificateRequest')
```

The following command lists all certificates in `wallet1`, for Oracle Internet Directory instance `oid1`, in application server instance `inst1`:

```
wls:/mydomain/serverConfig> listWalletObjects('inst1', 'oid1',
'oid','wallet1','password', 'Certificate')
```

The following command lists all trusted certificates in `wallet1`, for Oracle Internet Directory instance `oid1`, in application server instance `inst1`:

```
wls:/mydomain/serverConfig> listWalletObjects('inst1', 'oid1',
'oid','wallet1','password', 'TrustedCertificate')
```

6.9.25 listWallets

Online command that lists all wallets configured for a component instance.

6.9.25.1 Description

This command displays all the wallets configured for the specified component instance (Oracle HTTP Server, Oracle WebCache or Oracle Internet Directory), and identifies the auto-login wallets.

6.9.25.2 Syntax

```
listWallets('instName', 'compName', 'compType')
```

Argument	Definition
instName	Specifies the name of the application server instance.
compName	Specifies the name of the component instance
compType	Specifies the type of component. Valid values are 'ohs','oid', and 'webcache'.

6.9.25.3 Example

The following command lists all wallets for Oracle Internet Directory instance `oid1` in application server instance `inst1`:

```
wls:/mydomain/serverConfig> > listWallets('inst1', 'oid1', 'oid')
```

6.9.26 removeKeyStoreObject

Online command that removes an object from a keystore.

6.9.26.1 Description

This command removes a certificate request, certificate, trusted certificate, or all trusted certificates from a Java keystore (JKS) for Oracle Virtual Directory. Use an alias to remove a specific object; no alias is needed if all trusted certificates are being removed.

6.9.26.2 Syntax

```
removeKeyStoreObject('instName', 'compName', 'compType', 'keystoreName',
'password', 'type', 'alias')
```

Argument	Definition
instName	Specifies the name of the application server instance.
compName	Specifies the name of the component instance.
compType	Specifies the type of component. Valid value is 'ovd'.
keystoreName	Specifies the name of the keystore file.
password	Specifies the password of the keystore.
type	Specifies the type of the keystore object to be removed. Valid values are 'Certificate', 'TrustedCertificate' or 'TrustedAll'.
alias	Specifies the alias of the keystore object to be removed.

6.9.26.3 Examples

The following command removes a certificate or certificate chain denoted by alias `mykey` in `keys.jks`, for Oracle Virtual Directory instance `ovd1`, in application server instance `inst1`:

```
wls:/mydomain/serverConfig> removeKeyStoreObject('inst1', 'ovd1',
'ovd','keys.jks', 'password', 'Certificate','mykey')
```

The following command removes a trusted certificate denoted by alias `mykey` in `keys.jks`, for Oracle Virtual Directory instance `ovd1`, in application server instance `inst1`:

```
wls:/mydomain/serverConfig> removeKeyStoreObject('inst1', 'ovd1',
'ovd','keys.jks', 'password', 'TrustedCertificate','mykey')
```

The following command removes all trusted certificates in `keys.jks`, for Oracle Virtual Directory instance `ovd1`, in application server instance `inst1`. Since no alias is required, the value `None` is passed for that parameter:

```
wls:/mydomain/serverConfig> removeKeyStoreObject('inst1', 'ovd1',
'ovd','keys.jks', 'password', 'TrustedAll',None)
```

6.9.27 removeWalletObject

Online command that removes a certificate or other object from an Oracle wallet.

6.9.27.1 Description

This command removes a certificate signing request, certificate, trusted certificate or all trusted certificates from an Oracle wallet for the specified component instance (Oracle HTTP Server, Oracle WebCache or Oracle Internet Directory). DN is used to indicate the object to be removed.

6.9.27.2 Syntax

```
removeWalletObject('instName', 'compName', 'compType', 'walletName', 'password',
'type', 'DN')
```

Argument	Definition
instName	Specifies the name of the application server instance.
compName	Specifies the name of the component instance.
compType	Specifies the type of component. Valid values are 'ohs','oid', and 'webcache'.

Argument	Definition
walletName	Specifies the name of the wallet file.
password	Specifies the password of the wallet.
type	Specifies the type of the keystore object to be removed. Valid values are 'CertificateRequest', 'Certificate', 'TrustedCertificate' or 'TrustedAll'.
DN	Specifies the Distinguished Name of the wallet object to be removed.

6.9.27.3 Examples

The following command removes all trusted certificates from `wallet1`, for Oracle Internet Directory instance `oid1`, in application server instance `inst1`. It is not necessary to provide a DN, so you pass null (denoted by `None`) for the DN parameter:

```
wls:/mydomain/serverConfig> removeWalletObject('inst1', 'oid1', 'oid','wallet1',
'password', 'TrustedAll',None)
```

The following command removes a certificate signing request indicated by DN `cn=www.acme.com` from `wallet1`, for Oracle Internet Directory instance `oid1`, in application server instance `inst1`:

```
wls:/mydomain/serverConfig> removeWalletObject('inst1', 'oid1', 'oid','wallet1',
'password', 'CertificateRequest','cn=www.acme.com')
```

The following command removes a certificate indicated by DN `cn=www.acme.com` from `wallet1`, for Oracle Internet Directory instance `oid1`, in application server instance `inst1`:

```
wls:/mydomain/serverConfig> removeWalletObject('inst1', 'oid1', 'oid','wallet1',
'password', 'Certificate','cn=www.acme.com')
```

The following command removes a trusted certificate indicated by DN `cn=www.acme.com` from `wallet1`, for Oracle Internet Directory instance `oid1`, in application server instance `inst1`:

```
wls:/mydomain/serverConfig> removeWalletObject('inst1', 'oid1', 'oid','wallet1',
'password', 'TrustedCertificate','cn=www.acme.com')
```

6.9.28 Properties Files for SSL

SSL configuration employs certain properties files for use with the WLST `configureSSL` command. The files contain parameters to specify the desired SSL configuration, such as authentication type, cipher values, and SSL version.

You can use descriptive names if you need to manage multiple properties files for different components. For example, you could have properties files named `ohs-ssl-properties.prop` or `ovd-ssl-properties.prop`.

6.9.28.1 Structure of Properties Files

All the SSL properties files have a consistent structure.

Table 6–4 provides details about the key-value structure and usage of these files.

Table 6–4 Parameters in Properties File

Key	Mandatory?	Allowed Values for Oracle HTTP Server, Oracle Internet Directory, and Oracle Web Cache	Allowed Values for Oracle Virtual Directory	Usage
SSLEnabled	No	true false	true false	Either value
Ciphers	No	SSL_RSA_WITH_RC4_128_MD5 SSL_RSA_WITH_RC4_128_SHA SSL_RSA_WITH_3DES_EDE_CBC_SHA SSL_RSA_WITH_DES_CBC_SHA SSL_DH_anon_WITH_RC4_128_MD5 SSL_DH_anon_WITH_DES_CBC_SHA SSL_DH_anon_WITH_3DES_EDE_CBC_SHA TLS_RSA_WITH_AES_128_CBC_SHA TLS_RSA_WITH_AES_256_CBC_SHA	One of more of the ciphers allowed by the JSSE provider. For the complete list of ciphers allowed by JDK 1.5, see Appendix A of the following guide: http://java.sun.com/j2se/1.5.0/docs/guide/security/jsse/JSSERefGuide.html	One or more comma separated values
SSLVersions	No	nzos_Version_3_0 nzos_Version_3_0_With_2_0_Hello nzos_Version_1_0	TLSv1 SSLv2Hello (cannot be specified alone, must specify at least one other version) SSLv3	One or more comma separated values
CertValidation	No	none crl	N/A	Either value
CertValidation Path	No	file://crl_file_path dir://crl_dir_path	N/A	Path of the CRL file, or directory containing CRL files
KeyStore	No	Valid wallet name	Valid keystore name	
TrustStore	No	N/A	Valid truststore name	
AuthenticationType	No	None Server Optional Mutual	None Server Optional Mutual	Any one value

Table 6–5 shows the default values:

Table 6–5 Default Values of Parameters

Key	Default Value for Oracle HTTP Server	Default Value for Oracle Web Cache	Default Value for Oracle Internet Directory	Default Value for Oracle Virtual Directory
SSLEnabled	true	true	true	true
Ciphers	null	null	null	null
SSLVersions	null	null	null	null
CertValidation	none	none	-	-
CertValidation Path	null	null	-	-

Table 6–5 (Cont.) Default Values of Parameters

Key	Default Value for Oracle HTTP Server	Default Value for Oracle Web Cache	Default Value for Oracle Internet Directory	Default Value for Oracle Virtual Directory
KeyStore	default	default	null	keys.jks
TrustStore	-	-	-	keys.jks
Authentication Type	Server	Server	none	Server

Note:

- At least one `DH_anon` cipher must be used in SSL no-auth mode. For all other modes, at least one `RSA` cipher must be used.
- The value of the `KeyStore` parameter must be specified when configuring SSL in server-auth, mutual-auth, or optional client auth.
- If only `AES` ciphers have been specified, the `SSLVersions` parameter must contain `TLSv1` or `nzos_Version_1_0`.
- If you are doing CRL-based validation, the value of the `CertValidation` parameter should be `crl` and the value of the `CertValidationPath` parameter should point to the CRL file/directory.

6.9.28.2 Examples of Properties Files

Some examples demonstrating the use of the properties files follow.

Example 1: Basic Properties File

```
SSLEnabled=true
AuthenticationType=None
CertValidation=none
```

This properties file specifies no authentication mode, and default values will be used during SSL configuration for ciphers and SSL version. Keystore and truststore properties are not specified since the authentication type is `None`. For other authentication types, keystore must be specified.

Example 2: Basic Properties File

```
SSLEnabled=
AuthenticationType=None
CertValidation=none
```

This properties file is exactly the same as above, except that `SSLEnabled` is explicitly specified without any value. This is the same as not specifying the key at all. In both cases, the default value will be used.

Therefore, all the following three settings have the same meaning:

- The setting:

  ```
  SSLEnabled=true
  ```

  Here the value `true` is explicitly specified.

- The setting:

  ```
  SSLEnabled=
  ```

  Since no value is mentioned here, the default value of `SSLEnabled` (`true`) is used.

- The key `SSLEnabled` is not present in the properties file.

  Since the key is not present, its default value (`true`) is used.

Example 3: Properties File with Version for OHS

```
SSLEnabled=true
AuthenticationType=Mutual
SSLVersion=nzos_Version_3_0
CertValidation=crl
CertValidationPath=file:///tmp/file.crl
KeyStore=ohs1
```

This properties file has:

- Default values for ciphers
- Keystore
- SSL version v3
- CRL validation turned on
- Mutual Authentication mode

Example 4: Properties File with Ciphers for Oracle Virtual Directory

```
AuthenticationType=Server
Ciphers=SSL_RSA_WITH_RC4_128_MD5
SSLVersion=SSLv3,SSLv2Hello
KeyStore=ovdidentity.jks
TrustStore=ovdtrust.jks
SSLEnabled=true
```

This properties file contains:

- Specific cipher value
- SSL Version
- Server authentication mode

7

Using the SSL Automation Tool

This chapter contains the following sections:

7.1 Introduction to the SSL Automation Tool

The Oracle SSL Automation Tool enables you to configure multiple components in a domain using a domain-specific CA certificate.

The task of enabling SSL in a deployment can be intimidating and cumbersome for administrators. Manual configuration of SSL generally requires an administrator to have some expertise in several areas, such as:

- SSL as a technology
- Low-level tools available to perform SSL configuration and administration
- Best security practices

The Oracle SSL Automation Tool replaces manual procedures and simplifies SSL configuration. It enables you to generate a central, self-signed CA certificate, configure component servers with that certificate, and provide the CA certificate as a trusted certificate to multiple clients. It ensures that a network of trust is established in a consistent manner on all clients and servers, and can be used for both outward facing connections and for connections within the DMZ.

The SSL Automation Tool is based on a trust model, which introduces the concept of SSL Domains. An SSL domain is the security environment in which all the SSL components are deployed with the same CA signed certificates. Each SSL Domain has associated with it a self-signed Domain CA. All components within this SSL Domain implicitly trust the Domain CA. Additionally, this Domain CA can generate SSL Server Certificates for the server components deployed within that SSL Domain. If the server components in one SSL Domain (A) need to be trusted by a client component in another SSL Domain (B), then only the Domain CA certificate from (A) need be imported and trusted by the client component in SSL Domain (B).

The tool consists of a series of shell scripts: three main SSL scripts and several component-specific scripts.

Table 7–1 lists the main scripts.

Table 7–1 Main Scripts

Script	Function
`SSLGenCA.sh`	Generates the CA certificate and stores it in an LDAP directory
`SSLServerConfig.sh`	Configures the servers
`SSLClientConfig.sh`	Configures the clients

The server and client configuration scripts invoke component-specific scripts, depending on the value of an option that you specify on the command line when you invoke the main script.

The scripts use the LDAP Policy Store present in a deployment to centrally store the SSL Domain CA wallets. These SSL Domain CA wallets are protected by LDAP access controls, with access granted only to members of the SSL Administrators group. You must be a member of the group to run the scripts.

The SSL Automation Tool provides the following benefits:

- It provides a consistent set of interfaces for consumption by administrators.
- It removes the propagation of self-signed certificates and reduces the number of relevant trust points, which are now limited to SSL Domain CAs.
- It ensures that only properly authorized SSL Administrators are allowed to perform SSL related administrative tasks.
- It allows support for additional components to be added incrementally without the need for fundamental change.

7.2 Prerequisites

Before you attempt to use this tool, ensure that you have performed the tasks described in this section.

7.2.1 Setting up Oracle Fusion Middleware Environment

All the components of your Oracle Fusion Middleware environment must be up and running before you invoke the scripts to configure SSL on those components.

If your components are running on Windows platforms, you must obtain and install Cygwin from `http://www.cygwin.com` before you can use the scripts. Set the `ORACLE_HOME` environment variable in the Cygwin shell. For example:

```
export ORACLE_HOME='C:/rc8/fmwhome/Oracle_Home/'
```

7.2.2 Assembling Required Information

Make sure you have the values of the following variables listed in Table 7–2 and Table 7–3 available before you invoke the SSL scripts.

Table 7–2 Domain-Level Information Variables for SSL Automation Tool

Variable
`HOSTNAME`
`ORACLE_HOME` (Fusion Middleware)
`ORACLE_COMMON`

Table 7–2 (Cont.) Domain-Level Information Variables for SSL Automation Tool

Variable
MIDDLEWARE_HOME
DOMAIN_NAME
DOMAIN_HOME
DOMAIN_ADMINISTRATOR_USERNAME
DOMAIN_ADMINISTRATION_PASSWORD
DOMAIN_HOST_NAME
ADMINSERVER_PORT
DOMAIN_ADMINISTRATOR_USERNAME
DOMAIN_ADMINISTRATION_PASSWORD
INSTANCE_HOME
INSTANCE_NAME

Table 7–3 Component-Specific Information Variables for SSL Automation Tool

Variable
OVD_NAME
OVD_PORT
OID_NAME
OID_PORT
OID_SSL_PORT
OID_ADMIN
OID_ADMIN_PASSWORD
DB_HOST
DB_PORT
DB_SERVICE_NAME
DB_SID

7.3 Generating the CA Certificate

You invoke the CA certificate generating script `SSLGenCA.sh` to initialize and create an SSL Domain and generate the SSL Domain CA. Run the script only once for the whole SSL domain. If you run it again, you must configure all the servers and clients with the newly-generated CA wallet. An SSL domain is the security environment in which all the SSL components will be deployed with the same CA signed certificates.

Enter a shell that is set up with the default environment for an Oracle Fusion Middleware installation.

To run this script, you need the following information:

- Connection information (host and port) for the LDAP directory used by the deployment
- Administrator credentials that enable you to access that LDAP directory
- The name of the SSL Domain

Execute this command:

```
$ORACLE_COMMON_HOME/oracle_common/bin/SSLGenCA.sh
```

Provide information when prompted.

This script performs the following tasks:

- Creates a Demo Signing CA wallet for use in the domain.
- Extracts the public Demo CA Certificate from the CA wallet.
- Uploads the wallet and the certificate to LDAP and stores them in the entry: `cn=demoCA,`*`Deployment_SSL_Domain`*.
- Creates an access group in LDAP: `cn=sslAdmins, cn=demoCA,`*`Deployment_SSL_Domain`* and grants that group administrative privileges to the parent container. All other entities are denied access. (Add users to the group to give access.)

  The Demo CA Certificate is now available for download by an anonymous or authenticated user.
- The Demo CA Wallet password is stored locally in an obfuscated wallet for future use. Its path is: `$ORACLE_HOME/credCA/castore`.

  As administrator, you must secure this wallet so that only SSL administrators can read it.

7.3.1 Example: Generating a Certificate

This example shows a run of SSLGenCA.sh to generate a new CA wallet and store it in the Policy Store (LDAP server).

```
$ SSLGenCA.sh

SSL Certificate Authority Generation Script: Release 11.1.1.4.0 - Production
Copyright (c) 2010 Oracle.  All rights reserved.

*********************************************************************
********** This tool will generate a self-signed CA wallet ***********
********** and store it in a central LDAP directory        ***********
********** for IDM and FA SSL setup and provisioning       ***********
*********************************************************************
>>>Enter the LDAP hostname [host1.example.com]:
>>>Enter the LDAP port [3060]: 20040
>>>Enter the admin user [cn=orcladmin]
>>>Enter password for cn=orcladmin:
>>>Enter the LDAP sslDomain where your CA will be stored [idm]:
>>>Enter a password to protect your CA wallet:
>>>Enter confirmed password for your CA wallet:

Generate a new CA Wallet...
Create SSL Domains Container for cn=idm,cn=sslDomains...
Storing the newly generated CA to the LDAP...
Setup ACL to protect the CA wallet...
The newly generated CA is stored in LDAP entry cn=demoCA,cn=idm,cn=sslDomains
successfully
```

7.4 Configuring a Component Server

You configure a server by invoking the `SSLServerConfig.sh` script. This script uses the SSL Domain CA to generate a Server Certificate. Then the script passes control to a component specific configuration script, which picks up the generated Server Certificate and configures the component to accept SSL connections.

To run this script, you need the following information:

- Connection information (host and port) for the LDAP directory used by the deployment.
- Administrator credentials that enable you to access that LDAP directory.
- Server name. This can be either the WebLogic Administration Server or a Managed Server.

Before invoking the script, enter a shell that is set up with the default environment for an Oracle Fusion Middleware installation. The location of the script is: `$ORACLE_COMMON_HOME/oracle_common/bin/SSLServerConfig.sh` The syntax for the script is:

```
SSLServerConfig.sh -component [oid|ovd|oam|wls] [-v]
```

Specify one and only one component. Depending on the component you specify, `SSLServerConfig.sh` invokes a component-specific script. Component-specific server scripts have names of the form *`COMPONENT_NAME`*`_SSL_Server_Config.sh`.

If you specify the component option `wls`, the script configures all Java EE components on the named server. Java EE components include Oracle Identity Navigator, Oracle Access Manager 10g, Oracle Identity Manager, and Oracle Access Management Identity Federation.

To configure Oracle Internet Directory, Oracle Virtual Directory, or Oracle Access Manager 10g, use the appropriate component option, as shown in Table 7–4.

Table 7–4 Component Options to SSLServerConfig.sh

Component Option	Script Invoked	Component Configured
`wls`	`WLS_SSL_Server_Config.sh`	Oracle WebLogic Server and Java EE components
`oid`	`OID_SSL_Server_Config.sh`	Oracle Internet Directory server
`ovd`	`OVD_SSL_Server_Config.sh`	Oracle Virtual Directory server
`oam`	`OAM_SSL_Server_Config.sh`	Oracle Access Manager 10*g* Access Server

Provide information when prompted.

If you are using the `oid` or `ovd` option, and your Oracle Internet Directory or Oracle Virtual Directory host is not the same as your WebLogic Server host (in a high availability environment, for example), you must run the server script on the Oracle Internet Directory or Oracle Virtual Directory host.

This script performs the following tasks:

- Downloads the Demo Signing CA generated in Section 7.3 and stores it in `$ORACLE_HOME/rootCA`.
- Executes the component-specific script *`COMPONENT_NAME`*`_SSL_Server_Config.sh`, if appropriate.

The component-specific script performs the following tasks:

- Generates a server certificate based on the Demo Signing CA Wallet.
- Imports the certificate into the component-specific wallet/keystore.
- Configures the component instance for SSL Server-Auth, based on the new server certificate in the component specific wallet/keystore.

7.4.1 Example: Configuring a WebLogic Server and Java EE Components

```
$ ./SSLServerConfig.sh -component wls

Server SSL Automation Script: Release 11.1.1.4.0 - Production
Copyright (c) 2010 Oracle.  All rights reserved.

Downloading the CA wallet from the central LDAP location...
>>>Enter the LDAP Hostname [host1.example.com]:
>>>Enter the LDAP port [3060]: 16468
>>>Enter an admin user DN [cn=orcladmin]
>>>Enter password for cn=orcladmin:
>>>Enter the sslDomain for the CA [idm]:
>>>Enter a password to protect your SSL wallet/keystore:
>>>Enter confirmed password for your SSL wallet/keystore:
>>>Enter password for the CA wallet:
>>>Searching the LDAP for the CA usercertificate ...
Importing the CA certifcate into trust stores...
>>>Searching the LDAP for the CA userpkcs12 ...

Invoking Weblogic SSL Server Configuration Script...
Enter attribute values for your certificate DN
>>>Country Name 2 letter code [US]:
>>>State or Province Name [California]:
>>>Locality Name(eg, city) []:Belmont
>>>Organization Name (eg, company) [mycompany]:Oracle
>>>Organizational Unit Name (eg, section) [wls-20101123115644]:wls-admin
>>>Common Name (eg, hostName.domainName.com) [host1.example.com]:
The subject DN is
 cn=host1.example.com,ou=wls-admin,O=Oracle,l=Belmont,st=California,c=US
>>>Import the existing CA at /mw784/im7335/rootCA/cacert.der into keystore...
>>>Import the server certificate at /mw784/im7335/rootCA/keystores/wls/cert.txt
into kstore...
Configuring SSL for your WLS server instance...
>>>Enter your WLS domain home directory: /mw784/user_projects/domains/imdomain8017
>>>Enter your WLS server instance name [AdminServer]
Enter SSL Listen Port: [7002] 7778
>>>Enter weblogic admin port: [7001] 19249
>>>Enter weblogic admin user: [weblogic]
>>>Enter password for weblogic:
>>>Enter your keystore name [identity.jks]: id.jks
/mw784/im7335/rootCA/keystores/wls
/mw784/user_projects/domains/imdomain8017/keystores/id.jks
Configuring WLS AdminServer ...
Running /mw784/im7335/common/bin/wlst.sh
/mw784/im7335/rootCA/keystores/wls/wlssvr.py...
Your WLS server has been set up successfully
```

7.4.2 Example: Configuring an Oracle Internet Directory Server Component

```
$ ./SSLServerConfig.sh -component oid
Server SSL Automation Script: Release 11.1.1.4.0 - Production
```

```
Copyright (c) 2010 Oracle.  All rights reserved.

Downloading the CA wallet from the central LDAP location...
>>> Enter the LDAP Hostname [host1.example.com]:
>>> Enter the LDAP port [3060]: 16468
>>> Enter an admin user DN [cn=orcladmin]
>>> Enter password for cn=orcladmin:
>>> Enter the sslDomain for the CA [idm]:
>>> Enter a password to protect your SSL wallet/keystore:
>>> Enter confirmed password for your SSL wallet/keystore:
>>> Enter password for the CA wallet:
>>> Searching the LDAP for the CA usercertificate ...
Importing the CA certifcate into trust stores...
>>> Searching the LDAP for the CA userpkcs12 ...

Invoking OID SSL Server Configuration Script...
Enter attribute values for your certificate DN
>>> Country Name 2 letter code [US]:
>>> State or Province Name [California]:
>>> Locality Name(eg, city) []:Belmont
>> Organization Name (eg, company) [mycompany]:Example
>>> Organizational Unit Name (eg, section) [oid-20101118211946]:
>>> Common Name (eg, hostName.domainName.com) [host1.example.com]:
The subject DN is
cn=host1.example.com,ou=oid-20101118211946,O=Example,l=Belmont,st=California,c=US

Creating an Oracle SSL Wallet for oid instance...
/mw784/im7335/../oracle_common/bin
>>> Enter your OID component name: [oid1] Enter the weblogic admin port: [7001]
19249
>>> Enter the weblogic admin server host [host1.example.com]  host1.example.com
>>> Enter the weblogic admin user: [weblogic]
>>> Enter weblogic password:
>>> Enter your AS instance name:[asinst_1] iminst8017
>>> Enter an SSL wallet name for OID component [oid_wallet1]
Checking the existence of oid_wallet1 in the OID server...
Configuring the newly generated Oracle Wallet with your OID component...
Do you want to restart your OID component?[y/n]y

Do you want to test your SSL set up?[y/n]y
>>> Please enter your OID ssl port:[3131] 16180
Please enter the oid hostname:[host1] host1.example.com
>>> Invoking /mw784/im7335/bin/ldapbind -h host1.example.com -p 16180 -U 2 -D
cn=orcladmin ...
Bind successful

Your oid1 SSL server has been set up successfully
```

7.4.3 Example: Configuring an Oracle Virtual Directory Server Component

```
$ ./SSLServerConfig.sh -component ovd
Server SSL Automation Script: Release 11.1.1.4.0 - Production
Copyright (c) 2010 Oracle.  All rights reserved.

Downloading the CA wallet from the central LDAP location...
>>> Enter the LDAP Hostname [host1.example.com]:
>>> Enter the LDAP port [3060]: 16468
>>> Enter an admin user DN [cn=orcladmin]
>>> Enter password for cn=orcladmin:
>>> Enter the sslDomain for the CA [idm]:
```

```
>>> Enter a password to protect your SSL wallet/keystore:
>>> Enter confirmed password for your SSL wallet/keystore:
>>> Enter password for the CA wallet:
Searching the LDAP for the CA usercertificate ...
Importing the CA certifcate into trust stores...
>>> Searching the LDAP for the CA userpkcs12 ...

Invoking OVD SSL Server Configuration Script...
Enter attribute values for your certificate DN
>>> Country Name 2 letter code [US]:
>>> State or Province Name [California]:
>>> Locality Name(eg, city) []:redwood
>>> Organization Name (eg, company) [mycompany]:
>>> Organizational Unit Name (eg, section) [ovd-20101118212540]:
>>> Common Name (eg, hostName.domainName.com) [host1.example.com]:
The subject DN is
cn=host1.example.com,ou=ovd-20101118212540,l=redwood,st=California,c=US
>>> Import the existing CA at /mw784/im7335/rootCA/cacert.der into keystore...
>>> Import the server certificate at /mw784/im7335/rootCA/keystores/ovd/cert.txt
into kstore...
>>> Enter your OVD instance name [ovd1]
>>> Enter your Oracle instance [asinst_1]: iminst8017
>>> Enter the weblogic admin server host [host1.example.com]  host1.example.com
>>> Enter weblogic admin port: [7001] 19249
>>> Enter weblogic admin user: [weblogic]
>>> Enter password for weblogic:
>>> Enter your keystore name [ovdks1.jks]:

Checking the existence of ovdks1.jks in the OVD...
Configuring ovdks1.jks for ovd1 listener...
Do you want to restart your OVD instance?[y/n]y

Do you want to test your OVD SSL set up?[y/n]y

Please enter your OVD ssl port:[3131] 24888
Please enter the OVD hostname:[host1] host1.example.com

/mw784/im7335/bin/ldapbind -h host1.example.com -p 24888 -U 2 -D =orcladmin ...
Bind successfully to OVD SSL port 24888
Your  SSL server has been set up successfully
```

7.4.4 Example: Configuring an Oracle Access Manager 10g Access Server Component

```
$ SSLServerConfig.sh -component oam

Server SSL Automation Script: Release 11.1.1.4.0 - Production
Copyright (c) 2010 Oracle.  All rights reserved.


Downloading the CA wallet from the central LDAP location...
>>>Enter the LDAP Hostname [host1.example.com]:
>>>Enter the LDAP port [3060]: 16625
>>>Enter an admin user DN [cn=orcladmin]
>>>Enter password for cn=orcladmin:
>>>Enter the ssl domain name [idm]:
>>>Searching the LDAP for the CA usercertificate ...
>>>Searching the LDAP for the CA userpkcs12 ...


Invoking OAM SSL Server Configuration Script...
```

```
>>>Enter your OAM10 Access Server install location: [e.g.
/scratch/aime/OAM10/access] /scratch/install/OAM10/access
****************************************************************
*** CA root cert has been converted from DER to PEM format.  ***
****************************************************************
****************************************************************
*** This script will first invoke configureAAAServer tool to ***
*** reconfig AAA server in cert mode, and then generate a    ***
*** certificate request. Please select 3(Cert), 1(request a  ***
*** certificate), and enter pass phrase for the first 3      ***
*** prompts. Otherwise, this script is not guaranteed to     ***
*** work properly.                                           ***
****************************************************************

 Please enter the Mode in which you want the Access Server to run : 1(Open)
2(Simple) 3(Cert) : 3

 Do you want to request a certificate (1) or install a certificate (2) ? : 1

 Please enter the Pass phrase for this Access Server :

 Do you want to store the password in the file ? : 1(Y) 2(N) : 1

 Preparing to generate certificate.  This may take up to 60 seconds.  Please wait.
Generating a 1024 bit RSA private key
.++++++
.....................++++++
writing new private key to '/scratch/install/OAM10/access/oblix/config/aaa_
key.pem'
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [US]:US
State or Province Name (full name) [Some-State]:California
Locality Name (eg, city) []:Redwood Shores
Organization Name (eg, company) [Some-Organization Pty Ltd]:Example
Organizational Unit Name (eg, section) []:OAM
Common Name (eg, hostName.domainName.com) []:host1.example.com
Email Address []:
writing RSA key

 Your certificate request is in file :
/scratch/install/OAM10/access/oblix/config/aaa_req.pem
 Please get your certificate request signed by the Certificate Authority.
 On obtaining your certificate, please place your certificate in
'/scratch/install/OAM10/access/oblix/config/aaa_cert.pem' file and the certificate
authority's certificate for the corresponding component (for example: WebGate,
AXML Server) in '/scratch/install/OAM10/access/oblix/config/aaa_chain.pem' file.
 Once you have your certificate placed at the above mentioned location, please
follow the instructions on how to start the Access Server.

 More Information on setting up Access Server in Certificate mode can be obtained
from the Setup Installation Guide.
 Access Server mode has been re-configured successfully.
```

```
 Please note that new security mode will take effect only after the security mode
for this Access Server is changed to 'cert' from the Access Manager System
Console.
 Do you want to specify or update the failover information ? : 1(Y) 2(N) :

 Please restart your Access Server by executing the
'/scratch/install/OAM10/access/oblix/apps/common/bin/restart_access_server'
program from command line once you have placed your certificates at the above
mentioned location.

 Press enter key to continue ...


****************************************************************
*** Now we will sign the certificate request using CA cert.  ***
****************************************************************

>>>Enter the CA wallet password:

Certificate request (aaa_req.pem) has been converted to orapki acceptable format
in /scratch/install/WT/Oracle_WT1/rootCA/OAM

The certificate has been signed by the root CA

****************************************************************
*** OAM server certificate have been installed into Access   ***
*** Server config directory.                                  ***
****************************************************************

****************************************************************
*** Restarting AAA Server ...                                 ***
****************************************************************

Do you want to restart your Access Server? [y/n] y

Access Server has been started/restarted

****************************************************************
*** Your OAM10 Access Server has been setup successfully in  ***
*** cert mode.                                                ***
****************************************************************
```

7.5 Configuring a Client

You configure a client by invoking the script SSLClientConfig.sh. The script retrieves the SSL Domain CA then passes control to a component-specific script to import it and perform any additional configuration steps required.

To run this script, you need the following information:

- Connection information (host and port) for the LDAP directory used by the deployment
- Administrator credentials that enable you to access that LDAP directory
- The name of the SSL deployment, for example: `idm`, `fmw`

Before invoking the script, enter a shell that is set up with the default environment for an Oracle Fusion Middleware installation. The location of the script is: `$ORACLE_COMMON_HOME/oracle_common/bin/SSLClientConfig.sh` The syntax for the script is:

```
SSLClientConfig.sh -component [cacert|wls|webgate] [-v]
```

Depending on the `-component` option specified, `SSLClientConfig.sh` may invoke a component script listed in Table 7–5. The component-specific client scripts have names of the form *COMPONENT_NAME*`_SSL_Client_Config.sh`.

Table 7–5 Component Options to SSLClientConfig.sh

Component Option	Script Invoked	Component Configured
`cacert`	None	Other SSL Clients
`wls`	`WLS_SSL_Client_Config.sh`	Oracle WebLogic clients and Java EE components.
`webgate`	`OAMWG_SSL_Client_Config.sh`	Oracle Access Manager 10*g* WebGate

Provide information when prompted.

The client script performs the following tasks:

- Downloads the CA certificate or wallet from the LDAP server in the SSL Domain.
- Creates the related Java Trust Store, Oracle Wallet, or Java Keystore for the Oracle Identity Manager or Oracle Access Manager client.
- Imports the Signing CA certificate as a trusted certificate into the relevant trust stores, wallet, or keystore.

  For WebGate clients, it creates a full Java KeyStore with a private certificate, a client certificate, and the CA signing certificate.

  For other client components, which only need a common trust store or wallet, the script imports the CA certificate into the newly generated trust store.

7.5.1 Example: Downloading the CA Certificate for SSL Clients

```
$ ./SSLClientConfig.sh -component cacert

SSL Automation Script: Release 11.1.1.4.0 - Production
Copyright (c) 2010 Oracle.  All rights reserved.

Downloading the CA certificate from a central LDAP location
Creating a common trust store in JKS and Oracle Wallet formats ...
Configuring SSL clients with the common trust store...
Make sure that your LDAP server is currently up and running.

Downloading the CA certificate from the LDAP server...
>>> Enter the LDAP hostname [host1.example.com]: Enter the LDAP port: [3060]?
16468
>>> Enter your LDAP user [cn=orcladmin]:
>>> Enter password for cn=orcladmin:
>>> Enter the sslDomain for the CA [idm]:
Searching the LDAP for the CA usercertificate ...
Importing the CA certifcate into trust stores...
>>> The common trust store in JKS format is located at
/mw784/im7335/rootCA/keystores/tmp/trust.jks
>>> The common trust store in Oracle wallet format is located at
/mw784/im7335/rootCA/keystores/tmp/ewallet.p12
Generate trust store for the CA cert at cn=idm,cn=sslDomains
>>> Enter a password to protect your truststore:
```

```
>>> Enter confirmed password for your truststore:

Updating the existing /mw784/im7335/rootCA/keystores/common/trust.jks...
Importing the CA certifcate into trust stores...
>>> The common trust store in JKS format is located at
/mw784/im7335/rootCA/keystores/common/trust.jks
>>> The common trust store in Oracle wallet format is located at
/mw784/im7335/rootCA/keystores/common/ewallet.p12
```

7.5.2 Example: Downloading the Certificate and Configuring a WebLogic Client

```
$ ./SSLClientConfig.sh -component wls

SSL Automation Script: Release 11.1.1.4.0 - Production
Copyright (c) 2010 Oracle.  All rights reserved.

Downloading the CA certificate from a central LDAP location
Creating a common trust store in JKS and Oracle Wallet formats ...
Configuring SSL clients with the common trust store...
Make sure that your LDAP server is currently up and running.

Downloading the CA certificate from the LDAP server...
>>> Enter the LDAP hostname [host1.example.com]:
>>> Enter the LDAP port: [3060]? 16468
>>> Enter your LDAP user [cn=orcladmin]:
>>> Enter password for cn=orcladmin:
>>> Enter the sslDomain for the CA [idm]:
>>> Searching the LDAP for the CA usercertificate ...
Importing the CA certifcate into trust stores...
>>> The common trust store in JKS format is located at
/mw784/im7335/rootCA/keystores/tmp/trust.jks
>>> The common trust store in Oracle wallet format is located at
/mw784/im7335/rootCA/keystores/tmp/ewallet.p12
Invoking Weblogic SSL Client Configuration Script...
>>> Enter a password to protect your truststore:
>>> Enter confirmed password for your truststore:

Updating the existing /mw784/im7335/rootCA/keystores/wls/trust.jks...
Importing the CA certifcate into trust stores...
>>> The common trust store in JKS format is located at
/mw784/im7335/rootCA/keystores/wls/trust.jks
>>> The common trust store in Oracle wallet format is located at
/mw784/im7335/rootCA/keystores/wls/ewallet.p12
cat: /mw784/im7335/rootCA/cacert_tmp.txt: No such file or directory
Configuring SSL Trust for your WLS server instance...
>>> Enter your trust store name: [trust.jks]mytrust.jkds
>>> Enter your WLS domain home directory: /mw784/user_
projects/domains/imdomain8017
>>> Enter your WLS server instance name [AdminServer]
>>> Enter weblogic admin port: [7001] 19249
>>> Enter weblogic admin user: [weblogic]
>>> Enter password for weblogic:
>>> Copy /mw784/im7335/rootCA/keystores/wls/trust.jks to /mw784/user_
projects/domains/imdomain8017/servers/AdminServer/keystores/mytrust.jkds...
Configuring WLS AdminServer ...
Running /mw784/im7335/common/bin/wlst.sh
/mw784/im7335/rootCA/keystores/wls/wlscln.py...
Your WLS server has been set up successfully
```

7.5.3 Example: Downloading the Certificate and Configuring a WebGate Client

```
$ SSLClientConfig.sh -component webgate
Script started on Thu 28 Oct 2010 10:23:38 AM PDT

SSL Automation Script: Release 11.1.1.4.0 - Production
Copyright (c) 2010 Oracle.  All rights reserved.

Downloading the CA certificate from a central LDAP location
Creating a common trus store in JKS and Oracle Wallet formats ...
Configuring SSL clients with the common trust store...
Make sure that your LDAP server is currently up and running.

Downloading the CA certificate from the LDAP server...
>>>Enter the LDAP hostname [host1.example.com]:
>>>Enter the LDAP port: [3060]? 16625
>>>Enter your LDAP user [cn=orcladmin]:
>>>Enter password for cn=orcladmin:
>>>Enter the sslDomain for the CA [idm]:
>>>Searching the LDAP for the CA usercertificate ...
Invoking Webgate SSL Client Configuration Script...
>>>Searching the LDAP for the CA userpkcs12 ...
>>>Enter your 10g WebGate install location: [e.g. /scratch/aime/wg10/access]
/scratch/install/OAM10/cwg/access

****************************************************************
*** CA root cert has been converted from DER to PEM format.  ***
****************************************************************

>>>Enter WebGate ID: wg7777

>>>Enter WebGate Password:

>>>Enter the Access Server Host Name [host1.example.com]:

>>>Enter the Access Server Port [6021]:

>>>Enter Access Server ID: aaa1

>>>Enter WebGate Pass Phrase:

****************************************************************
*** This script will first invoke configureWebGate tool to   ***
*** reconfig webgate in cert mode, and then generate a       ***
*** certificate request.                                     ***
****************************************************************


 Preparing to generate certificate.  This may take up to 60 seconds.  Please wait.
Generating a 1024 bit RSA private key
..............++++++
.....................++++++
writing new private key to '/scratch/install/OAM10/cwg/access/oblix/config/aaa_
key.pem'
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
```

```
-----
Country Name (2 letter code) [US]:US
State or Province Name (full name) [Some-State]:California
Locality Name (eg, city) []:Redwood Shores
Organization Name (eg, company) [Some-Organization Pty Ltd]:Example
Organizational Unit Name (eg, section) []:OAM
Common Name (eg, hostName.domainName.com) []:host1.example.com
Email Address []:
writing RSA key

 Your certificate request is in file :
/scratch/install/OAM10/cwg/access/oblix/config/aaa_req.pem

 Please get your certificate request signed by the Certificate Authority

 On obtaining your certificate, please place your certificate in
'/scratch/install/OAM10/cwg/access/oblix/config/aaa_cert.pem' file and Access
Server's CA certificate in '/scratch/install/OAM10/cwg/access/oblix/config/aaa_
chain.pem' file

 Once you have your certificate placed at the above mentioned location, please run
'/scratch/install/OAM10/cwg/access/oblix/tools/configureWebGate/configureWebGate'
program

 More Information on setting up Web Gate in Certificate mode can be obtained from
the Setup Installation Guide

 Press enter key to continue ...

****************************************************************
*** Now we will sign the certificate request using CA cert.  ***
****************************************************************

>>>Enter the CA wallet password:

Certificate request (aaa_req.pem) has been converted to orapki acceptable format
in /scratch/install/WT/Oracle_WT1/rootCA/WEBGATE

The certificate has been signed by the root CA

****************************************************************
*** WebGate certificate have been installed into WebGate     ***
*** config directory.                                        ***
****************************************************************

****************************************************************
*** Testing connection to AAA Server ...                     ***
*** (Make sure AAA Server is up and running.)                ***
****************************************************************

 Preparing to connect to Access Server.  Please wait.

 Web Gate installed Successfully.

****************************************************************
*** Restarting OHS ...                                       ***
****************************************************************

Do you want to restart your OHS webserver? [y/n] y
```

```
>>>Enter ORACLE_HOME for your OHS webtier install [e.g. /scratch/aime/WT/Oracle_
WT1]: /scratch/install/WT/Oracle_WT1

>>>Enter ORACLE_INSTANCE for your OHS webtier instance [e.g.
/scratch/aime/WT/Oracle_WT1/instances/instance1]: /scratch/install/WT/Oracle_
WT1/instances/instance1

>>>Enter OHS component id [ohs1]:

OHS instance has been started/restarted

*****************************************************************
*** Your 10g WebGate has been setup successfully in cert     ***
*** mode.                                                     ***
*****************************************************************
```

8

Managing Keystores, Wallets, and Certificates

This chapter explains how to use Oracle Fusion Middleware security features to administer keystores, wallets, and certificates. It contains these sections:

- Key and Certificate Storage in Oracle Fusion Middleware
- Command-Line Interface for Keystores and Wallets
- JKS Keystore Management
- Wallet Management

8.1 Key and Certificate Storage in Oracle Fusion Middleware

Private keys, digital certificates, and trusted CA certificates are stored in keystores. This section describes the keystores available in Oracle Fusion Middleware and contains these topics:

- Types of Keystores
- Keystore Management Tools

8.1.1 Types of Keystores

Oracle Fusion Middleware provides two types of keystores for keys and certificates:

- JKS Keystore and Truststore
- Oracle Wallet

8.1.1.1 JKS Keystore and Truststore

A JKS keystore is the default JDK implementation of Java keystores provided by Sun Microsystems. In 11*g* Release 2 (11.1.2), all Java components and Java EE applications use the JKS-based keystore and truststore.

You use a JKS-based keystore for the following:

- Oracle WebLogic Server
- Oracle Virtual Directory
- Applications deployed on Oracle WebLogic Server, including:
  - Oracle SOA Suite
  - Oracle WebCenter Portal

In Oracle Fusion Middleware, you can use graphical user interface or command-line tools to create, import, export, and delete a Java keystore and the certificates contained in the keystore. See Section 8.1.2, "Keystore Management Tools" for details.

While creating a keystore, you can pre-populate it with a keypair wrapped in a self-signed certificate; such a keystore is typically used in development and testing phases.

The other choice is to generate a certificate signing request for a keypair, so that you can request a signed certificate back from a Certificate Authority (CA). Once the CA sends the certificate back, it is imported into the keystore; the keystore now contains a trusted certificate, since it comes from a trusted third-party. Such a keystore is typically used in production environments.

Keystores are always password-protected.

8.1.1.2 Oracle Wallet

An Oracle wallet is a container that stores your credentials, such as certificates, trusted certificates, certificate requests, and private keys. You can store Oracle wallets on the file system or in LDAP directories such as Oracle Internet Directory. Oracle wallets can be auto-login or password-protected wallets.

You use an Oracle Wallet for the following components:

- Oracle HTTP Server
- Oracle Web Cache
- Oracle Internet Directory

> **Note:** Wallets configured for Oracle Internet Directory must have auto-login enabled.

In Oracle Fusion Middleware, you can use graphical user interface or command-line tools to create, import, export and delete a wallet and the certificates contained in the wallet. See Section 8.1.2, "Keystore Management Tools" for details.

When creating a wallet, you can pre-populate it with a self-signed certificate; such a wallet is called a test wallet and is typically used in development and testing phases.

The other choice is to create a certificate request, so that you can request a signed certificate back from a Certificate Authority (CA). Once the CA sends the certificate back, it is imported into the wallet; such a wallet is called a third-party wallet.

Either the test wallet or the third-party wallet may be password-protected, or may be configured to not require a password, in which case it is called an auto-login wallet.

8.1.2 Keystore Management Tools

Oracle Fusion Middleware provides these options for keystore operations:

- WLST, a command-line interface for JKS keystores and wallets
- orapki, a command-line tool for wallets
- Fusion Middleware Control, a graphical user interface
- Oracle Wallet Manager, a stand-alone graphical user interface for wallets, recommended for managing PKCS#11 wallets. Also see the discussion titled *Using Oracle Wallet Manager in a Stand-alone Environment* at the end of this section.

This table shows the type of keystore used by each component, and the tool(s) available to manage the keystore:

Component/Application	Type of Keystore	Tasks	Tool
Oracle HTTP Server Oracle Web Cache Oracle Internet Directory	Oracle Wallet	Create Wallet, Create Certificate Request, Delete Wallet, Import Certificate, Export Certificate, Enable SSL	Fusion Middleware Control, WLST Oracle Wallet Manager and orapki for PKCS#11, PKCS#12, and Hardware Security Modules (HSM)-based wallets. Also for environments where Fusion Middleware Control and WLST are not available (such as a stand-alone upgrade of these components without a domain).
Oracle Virtual Directory	JKS-based Keystore	Create KeyStore, Create Certificate Request, Delete KeyStore, Import Certificate, Export Certificate, Enable SSL	Fusion Middleware Control, WLST
Oracle SOA Suite	JKS-based Keystore	All Keystore operations	JDK Keytool
Oracle WebCenter Portal	JKS-based Keystore	All Keystore operations	JDK Keytool
Oracle WebLogic Server	JKS-based Keystore	All Keystore operations	JDK Keytool
Oracle WebLogic Server	JKS-based Keystore	Enable SSL	Oracle WebLogic Server Administration Console
All Java EE applications (for example Oracle Directory Integration Platform, Oracle Directory Services Manager)	JKS-based Keystore	All Keystore operations	JDK Keytool

See Also: For details about using keytool, see *Oracle Fusion Middleware Securing Oracle WebLogic Server*.

Note: Pre-11*g* wallets (corresponding to 10*g* Release 10.1.2 and 10.1.3 formats) are supported in 11*g* Release 2 (11.1.2).

About Importing DER-encoded Certificates

You cannot use Fusion Middleware Control or the `WLST` command-line tool to import DER-encoded certificates or trusted certificates into an Oracle wallet or a JKS keystore. Use these tools instead:

- To import DER-encoded certificates or trusted certificates into an Oracle wallet, use:
  - Oracle Wallet Manager or
  - `orapki` command-line tool
- To import DER-encoded certificates or trusted certificates into a JKS keystore, use the `keytool` utility.

Using a Keystore Not Created with WLST or Fusion Middleware Control

If an Oracle wallet or JKS keystore was created with tools such as `orapki` or `keytool`, it must be imported prior to use. Specifically:

- For Oracle HTTP Server, Oracle Web Cache, and Oracle Internet Directory, if a wallet was created using `orapki` or Oracle Wallet Manager, in order to view or manage it in Fusion Middleware Control you must first import it with either Fusion Middleware Control or the WLST `importWallet` command. For details, see Section 8.4.4.9 and Section 8.4.4.10.
- For Oracle Virtual Directory, if a keystore was created using `keytool`, in order to view or manage it in Fusion Middleware Control you must first import it with either Fusion Middleware Control or the WLST `importKeyStore` command.

Copying Keystores to File System Not Supported

Creating, renaming, or copying keystores directly to any directory on the file system is not supported. Any existing pre-11*g* keystore or wallet that you wish to use must be imported using either Fusion Middleware Control or the WLST utility.

Using Oracle Wallet Manager in a Stand-alone Environment

In a stand-alone environment, such as a stand-alone Web Tier installation, you can use Oracle Wallet Manager to create and manage wallets.

For details about Oracle Wallet Manager, including its use for PKCS#12 wallets, and wallet and certificate lifecycle, see the chapter "Using Oracle Wallet Manager", in the *Oracle Advanced Security Administrator's Guide*:

`http://download.oracle.com/docs/cd/E11882_01/network.112/e10746/asowalet.htm`

Additional Information

Details about the tools are provided in these sections:

- Command-Line Interface for Keystores and Wallets
- JKS Keystore Management
- Wallet Management
- Appendix H, "Oracle Wallet Manager and orapki"

8.2 Command-Line Interface for Keystores and Wallets

Oracle Fusion Middleware provides a set of `wlst` scripts to create and manage JKS keystores and Oracle wallets, and to manipulate their stored objects.

How to Launch the WLST Command-Line Interface

When running SSL `WLST` commands, you must invoke the `WLST` script from the Oracle Common home. See Section 3.5.1.1 for more information.

> **Note:** All SSL-related WLST commands require you to launch the script from the above-mentioned location only.

This brings up the WLST shell. Connect to a running Oracle WebLogic Server instance by specifying the user name, password, and connect URL. After connecting, you are

now ready to run SSL-related WLST commands as explained in the subsequent sections.

8.3 JKS Keystore Management

This section describes the typical life cycle of keystores and certificates, and how to use Oracle Fusion Middleware tools to create and maintain keystores and certificates. It includes these topics:

- About Keystores and Certificates
- Managing the Keystore Life Cycle
- Common Keystore Operations
- Managing the Certificate Life Cycle
- Common Certificate Operations
- Keystore and Certificate Maintenance

8.3.1 About Keystores and Certificates

Keys and certificates are used to digitally sign and verify data and achieve authentication, integrity, and privacy in network communications.

A Java keystore (JKS) is a protected database that holds keys and certificates for the organization. Oracle Fusion Middleware utilizes JKS keystores for Oracle Virtual Directory, for applications deployed in Oracle WebLogic Server, and for Oracle WebLogic Server itself.

Access to a keystore requires a password which is defined at the time the keystore is created, by the person who creates the keystore, and which can only be changed by providing the current password.

In addition, each private key in a keystore can be secured by its own password.

This section contains these topics:

- Sharing Keystores Across Instances
- Keystore Naming Conventions

8.3.1.1 Sharing Keystores Across Instances

Oracle recommends that you do not share keystores between component instances or Oracle instances, since each keystore represents a unique identity.

The exception to this is an environment with a cluster of component instances, in which case keystore sharing would be an acceptable practice.

Note that no management tools or interfaces are available to facilitate keystore sharing. However, you can export a keystore from one instance and import it into another instance.

8.3.1.2 Keystore Naming Conventions

Follow these naming conventions for your JKS keystores:

- Do not use a name longer than 256 characters.
- Do not use any of the following characters in a keystore name:

```
| ; , ! @ # $ ( ) < > / \ " ' ` ~ { } [ ] = + & ^ space tab
```

Note: Observe this rule even if your operating system supports the character.

- Do not use non-ascii characters in a keystore name.
- Additionally, follow the operating system-specific rules for directory and file names.

8.3.2 Managing the Keystore Life Cycle

Typical life cycle events for a JKS keystore are as follows:

- The keystore is created. Keystores can be created directly, or by importing a keystore file from the file system.
- The list of available keystores are viewed and specific keystores selected for update.
- Keystores are updated or deleted. Update operations require that the keystore password be entered.
- The keystore password can be changed.
- The keystore can be deleted.
- Keystores can be exported and imported.

8.3.3 Common Keystore Operations

This section explains the following keystore operations:

- Creating a Keystore Using Fusion Middleware Control
- Creating a Keystore Using WLST
- Exporting a Keystore Using Fusion Middleware Control
- Exporting a Keystore Using WLST
- Deleting a Keystore Using Fusion Middleware Control
- Deleting a Keystore Using WLST
- Importing a Keystore Using Fusion Middleware Control
- Importing a Keystore Using WLST
- Changing the Keystore Password Using Fusion Middleware Control
- Changing the Keystore Password Using WLST

8.3.3.1 Creating a Keystore Using Fusion Middleware Control

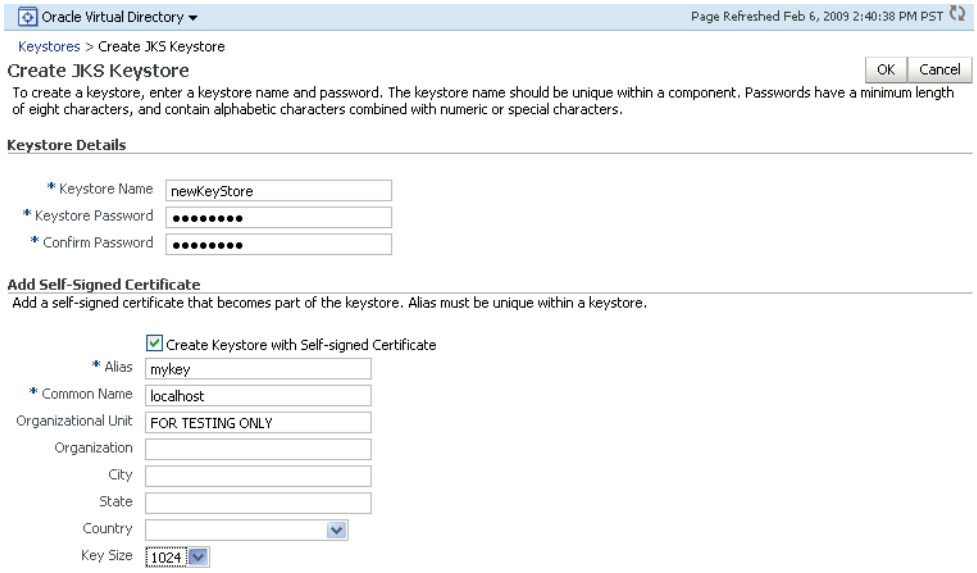
Take these steps to create a keystore:

1. Log in to the domain of interest using Fusion Middleware Control.
2. From the navigation pane, locate your component instance.
3. Navigate to *component_name*, then **Security**, then **Keystores**. For example, navigate to Oracle Virtual Directory, then **Security**, then **Keystores**.

Note: The component type is displayed at the top of the page, adjacent to the Topology icon.

4. The Java Keystore page appears. On this page you can create, update, and delete keystores, and perform other keystore management tasks.
5. Click **Create**. The Create Keystore dialog appears.
6. Provide keystore details such as name and password.

   You can also request a self-signed certificate in this dialog, and fill in the alias name and DN information.



Note: If you want to use this keystore only to store trusted certificates, you can uncheck the Create Self-Signed Certificate checkbox. This will create a keystore with no keypair.

7. Click **Submit**. The new keystore appears in the list of Java keystores.

8.3.3.2 Creating a Keystore Using WLST

Assuming the instance name is `inst1`, use this command to create a keystore:

```
createKeyStore('inst1', 'ovd5', 'ovd', 'newKeyStore', 'password')
```

where `password` is the password for this keystore.

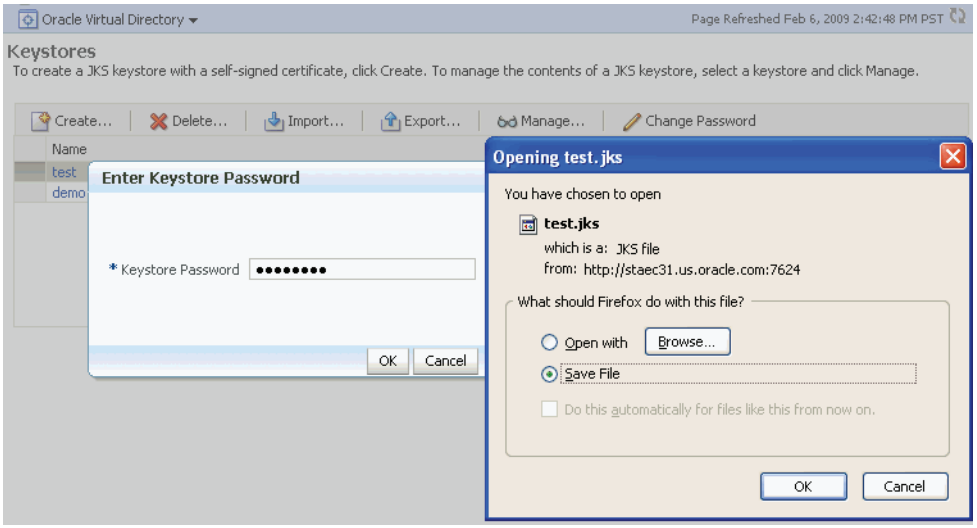
See Also:
- Section 6.9.6, "createKeyStore".
- The discussion at the start of Section 6.9, which explains how you can obtain the parameter values needed to execute the commands.

8.3.3.3 Exporting a Keystore Using Fusion Middleware Control

If multiple Oracle Virtual Directory instances want to share the same keystore file, this can be achieved by exporting the keystore from one instance and importing it into the other instances.

Take these steps to export a keystore:

1. Navigate to the Java Keystores page for the component instance, as explained in Section 8.3.3.1, "Creating a Keystore Using Fusion Middleware Control."
2. Select the desired keystore from the list of stores.
3. Click **Export**.
4. A dialog box appears in which you must enter the keystore password to continue.
5. Specify a file system location, and click **OK**.



See Also: Section 8.3.3.7, "Importing a Keystore Using Fusion Middleware Control"

8.3.3.4 Exporting a Keystore Using WLST

Assuming the instance name is `inst1`, use this command to export a keystore:

```
exportKeyStore('inst1', 'ovd5', 'ovd', 'test', 'password', '/tmp')
```

where `password` is the password for this keystore.

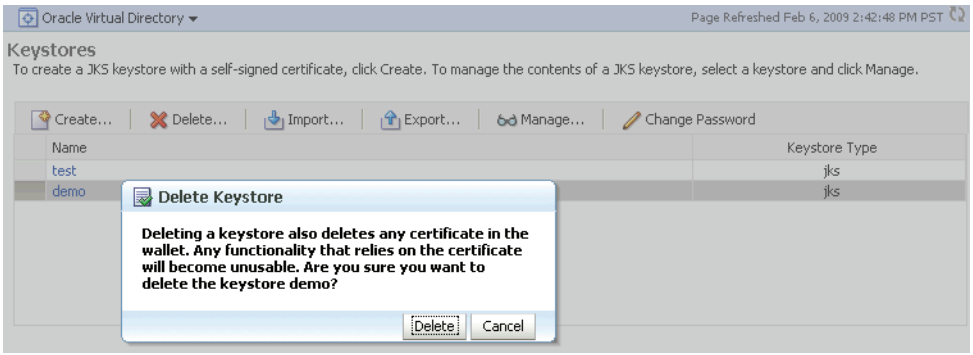
This command exports the keystore into a file named `test` under the directory `/tmp`.

See Also: Section 6.9.10, "exportKeyStore".

8.3.3.5 Deleting a Keystore Using Fusion Middleware Control

Take these steps to delete a keystore:

1. Navigate to the Java Keystores page for the component instance, as explained in Section 8.3.3.1, "Creating a Keystore Using Fusion Middleware Control."
2. Select the desired keystore from the list of stores.
3. Click **Delete**.
4. A dialog box appears to request confirmation of the delete request.



5. Click **Delete**.

8.3.3.6 Deleting a Keystore Using WLST

Assuming the application server instance name is `inst1`, use this command to delete a keystore:

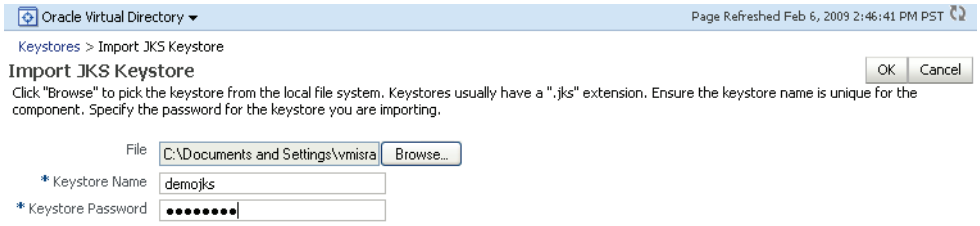
```
deleteKeyStore('inst1', 'ovd5', 'ovd', 'demo')
```

where the component type is `ovd`, the component instance is `ovd5`, and the keystore is named `demo`.

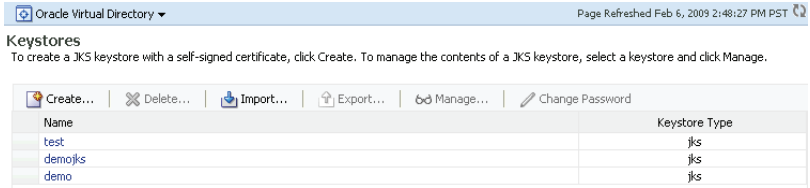
See Also: Section 6.9.8, "deleteKeyStore".

8.3.3.7 Importing a Keystore Using Fusion Middleware Control

1. Navigate to the Java Keystores page for the component instance, as explained in Section 8.3.3.1, "Creating a Keystore Using Fusion Middleware Control."
2. Click **Import**.
3. The Import Keystore dialog box appears.
4. Browse the file system to locate the keystore file.
5. Provide a name for the keystore. Enter the keystore password.



6. Click **OK**.
7. The imported keystore appears in the list of Java keystores.



8.3.3.8 Importing a Keystore Using WLST

Assuming the instance name is inst1, use this command to import a keystore:

```
importKeyStore('inst1', 'ovd5', 'ovd', 'demojks', 'password', '/tmp/demojks.jks')
```

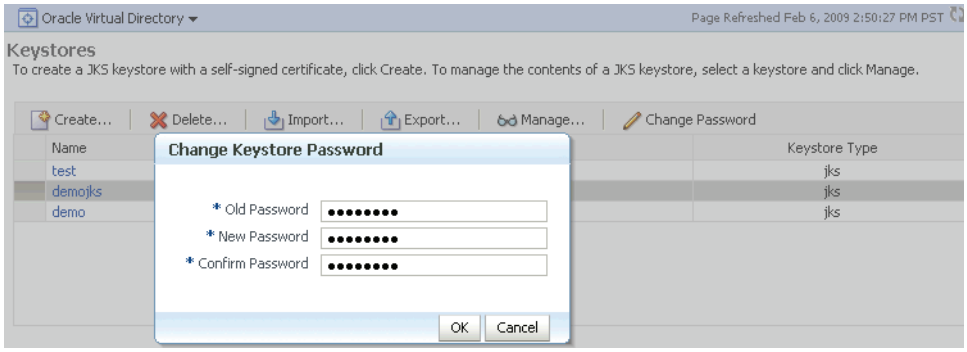
where `password` is the password for this keystore.

See Also: Section 6.9.18, "importKeyStore".

8.3.3.9 Changing the Keystore Password Using Fusion Middleware Control

Take these steps to change a keystore password:

1. Navigate to the Java Keystores page for the component instance, as explained in Section 8.3.3.1, "Creating a Keystore Using Fusion Middleware Control."
2. Select a keystore and click **Change Password**.
3. A dialog box appears on which you must enter the current password and enter a new password. The new password must be entered a second time to confirm.



4. Click **OK** to change the password. In future, any operations performed on this keystore or its certificates will require the use of the new password.

8.3.3.10 Changing the Keystore Password Using WLST

Assuming the instance name is `inst1`, use this command to change the keystore password:

```
changeKeyStorePassword('inst1', 'ovd5', 'ovd', 'demojks', 'current_password',
'new_password')
```

where `current_password` is the current password for this keystore, and `new_password` is the new password.

See Also: Section 6.9.3, "changeKeyStorePassword".

8.3.4 Managing the Certificate Life Cycle

Typical life cycle events for a certificate residing in a keystore are as follows:

- A self-signed certificate is automatically created for the keypair.
- A certificate signing request (CSR) is generated, and can then be exported to a file.
- Certificates are imported into the keystore. A certificate can either be pasted into a text box or imported from the file system. You can import both user certificates and trusted certificates (also known as CA certificates) in this way.
- Certificates or trusted certificates are exported from the keystore out to a file.

- Certificates or trusted certificates are deleted from the keystore.

8.3.5 Common Certificate Operations

This section describes the following common certificate operations:

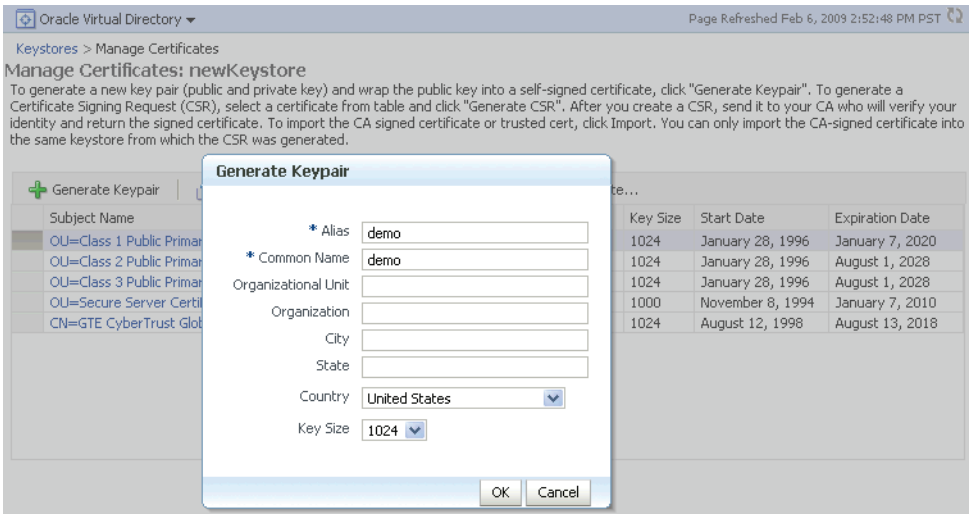
- Generating a New Key for the Keystore Using Fusion Middleware Control
- Generating a New Key for the Keystore Using WLST
- Generating a Certificate Signing Request Using Fusion Middleware Control
- Generating a Certificate Signing Request Using WLST
- Importing a Certificate or Trusted Certificate into a Keystore Using Fusion Middleware Control
- Importing a Certificate or Trusted Certificate into a Keystore Using WLST
- Exporting a Certificate or Trusted Certificate from the Keystore Using Fusion Middleware Control
- Exporting a Certificate or Trusted Certificate from the Keystore Using WLST
- Deleting a Certificate or Trusted Certificate from the Keystore Using Fusion Middleware Control
- Deleting a Certificate or Trusted Certificate from the Keystore Using WLST
- Converting a Self-Signed Certificate to a Third-Party Certificate Using Fusion Middleware Control
- Converting a Self-Signed Certificate to a Third-Party Certificate Using WLST

8.3.5.1 Generating a New Key for the Keystore Using Fusion Middleware Control

To generate a new key (that is, a new self-signed certificate) for a keystore:

1. Navigate to the Java Keystores page for the component instance, as explained in Section 8.3.3.1, "Creating a Keystore Using Fusion Middleware Control."
2. Select the keystore from the list of stores.
3. A dialog box appears in which you must enter the keystore password to continue.
4. The Manage Certificates page appears. Here, you can manage both types of keystore entries, that is, certificates and trusted certificates.
5. Click the **Generate Keypair** button.
6. In the Generate Keypair dialog, enter the details for the new key and click **OK**.

Example: Generating a Key Pair



When you complete these steps, a new public-private key pair is generated for the keystore, and the public key is wrapped in a self-signed certificate.

While these steps generate a new keypair for an existing keystore, you can also generate a new keypair when creating the keystore itself. For details, see Section 8.3.3.1, "Creating a Keystore Using Fusion Middleware Control."

8.3.5.2 Generating a New Key for the Keystore Using WLST

Assuming the instance name is `inst1`, use this command to generate a new key for a keystore:

```
generateKey('inst1', 'ovd5', 'ovd', 'newKeystore', 'password', 'subject_dn', 'key_size', 'alias')
```

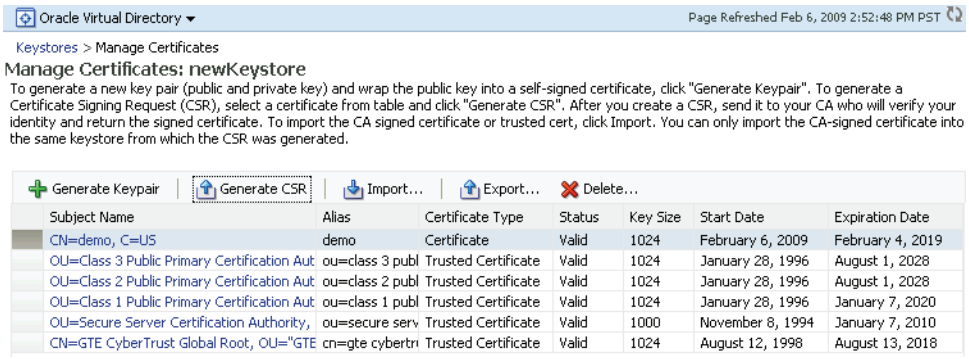
where `password` is the password for this keystore, `subject_dn` is the distinguished name by which the key pair is generated, `key_size` is the key size in bits, and `alias` is the key alias.

> **See Also:** See Section 6.9 for details about using WLST commands.

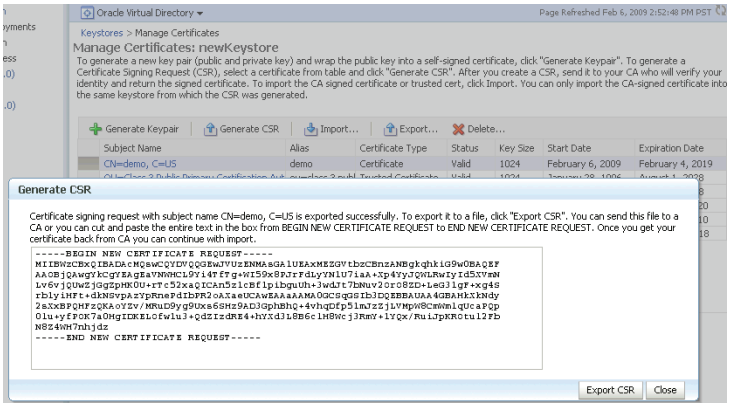
8.3.5.3 Generating a Certificate Signing Request Using Fusion Middleware Control

Take these steps to create a Certificate Signing Request (CSR):

1. From the navigation pane, locate your component instance.
2. Navigate to *component_name*, then **Security**, then **Keystores**.
3. Select the desired keystore from the list of stores.
4. A dialog box appears in which you must enter the keystore password to continue.
5. The Manage Certificates page appears. Select the self-signed certificate for which you want to generate the CSR and click **Generate CSR**.



6. A dialog box appears, showing the generated signing request. You can either:
   - Copy the CSR from the dialog box and past it to a file.
   - Click the **Export CSR** button to directly save it to a file.



After the CSR is exported, you can send it to a Certificate Authority (CA) to generate a certificate.

8.3.5.4 Generating a Certificate Signing Request Using WLST

Assuming the instance name is `inst1`, use this command to generate and export a CSR:

```
exportKeyStoreObject('inst1', 'ovd5', 'ovd', 'newKeystore', 'password',
'CertificateRequest', '/tmp', 'alias')
```

where `password` is the password for this keystore, `/tmp` is the path under which the certificate request is generated in BASE64 format in the file `base64.txt`, and `alias` is the alias of the key pair that is used to generate the certificate request.

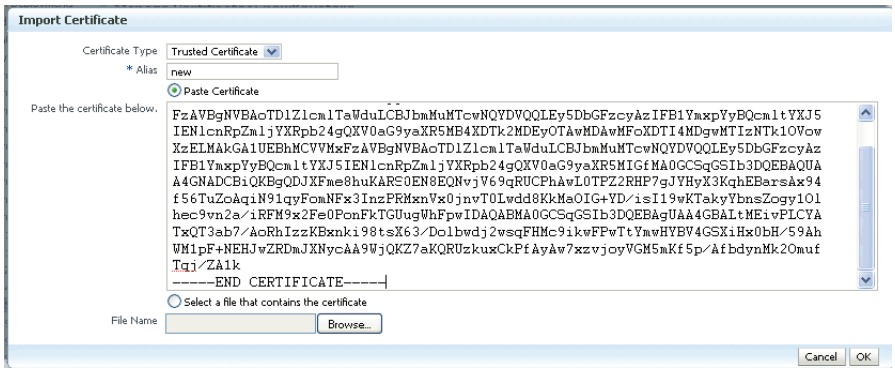
See Also: See Section 6.9 for details about using WLST commands.

8.3.5.5 Importing a Certificate or Trusted Certificate into a Keystore Using Fusion Middleware Control

Note: You cannot use Fusion Middleware Control to import DER-encoded certificates or trusted certificates into a JKS keystore; use the `keytool` utility for this task.

Take these steps to import a certificate, or a trusted certificate, into a keystore:

1. From the navigation pane, locate your component instance.
2. Navigate to *component_name*, then **Security**, then **Keystores**.
3. Select the desired keystore from the list of stores.
4. A dialog box appears in which you must enter the keystore password to continue.
5. The Manage Certificates page appears. Click the **Import** button.
6. A dialog box appears with which you can either:
   - Paste the Base-64 encoded contents of a certificate or trusted certificate into the keystore directly.
   - Select a certificate or trusted certificate file from the file system.

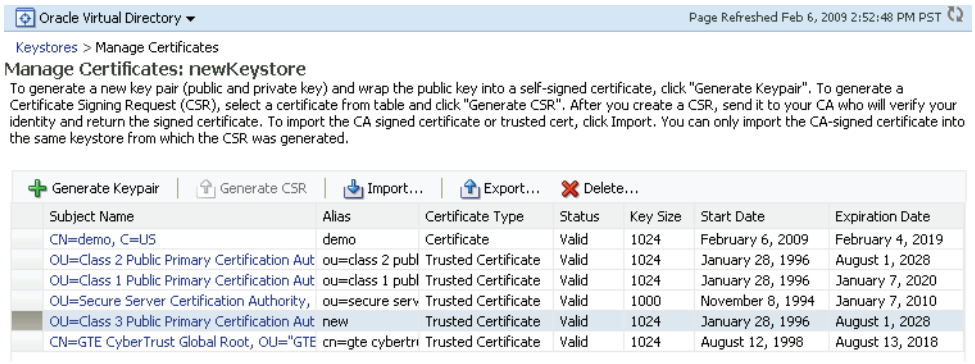


   You need to specify an alias while importing a certificate.

   When importing a certificate, the alias should match the alias of the corresponding keypair.

   When importing a trusted certificate, the alias should be unique in the keystore.

7. Click **OK**. The Manage Certificates page appears, showing the newly imported certificate or trusted certificate.



8.3.5.6 Importing a Certificate or Trusted Certificate into a Keystore Using WLST

Note: You cannot use the `WLST` command-line tool to import DER-encoded certificates or trusted certificates into a JKS keystore; use the `keytool` utility for this purpose.

Assuming the instance name is `inst1`, use this command to import a certificate into a keystore:

```
importKeyStoreObject('inst1', 'ovd5', 'ovd', 'newKeystore', 'password',
'Certificate', '/tmp/cert.txt', 'alias')
```

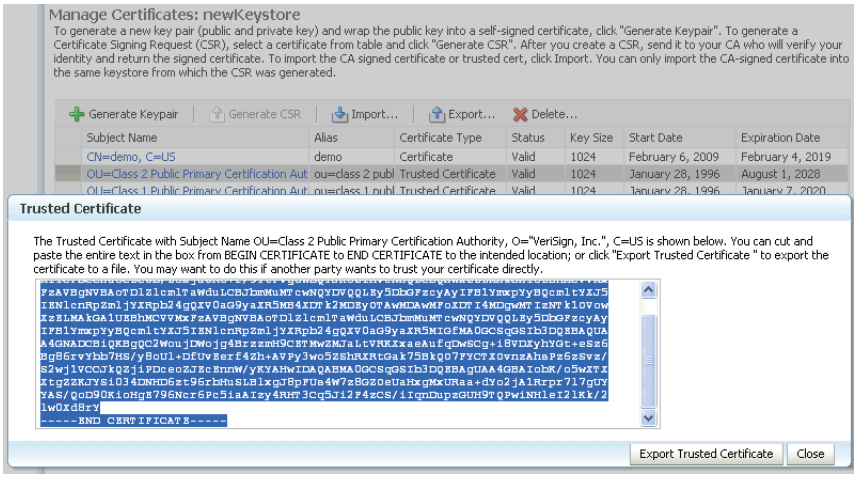
where `password` is the password for this keystore, `/tmp/cert.txt` is the file that contains BASE64 encoded certificate, and `alias` is the alias by which this certificate is imported. Note that this alias must be same as that of the key pair that was used to generate this certificate request.

See Also: See Section 6.9 for details about using WLST commands.

8.3.5.7 Exporting a Certificate or Trusted Certificate from the Keystore Using Fusion Middleware Control

Take these steps to export a certificate or trusted certificate from the keystore:

1. From the navigation pane, locate your component instance.
2. Navigate to *component_name*, then **Security**, then **Keystores**.
3. Select the desired keystore from the list of stores.
4. A dialog box appears in which you must enter the keystore password to continue.
5. The Manage Certificates page appears. Click **Export**.
6. A dialog box appears which shows the Base64 encoded certificate or trusted certificate. You can either copy the contents from the text box and paste it to a file, or select the **Export** button to save it directly to a file.



8.3.5.8 Exporting a Certificate or Trusted Certificate from the Keystore Using WLST

Assuming the instance name is `inst1`, use this command to export a certificate:

```
exportKeyStoreObject('inst1', 'ovd5', 'ovd', 'newKeystore', 'password',
'Certificate', '/tmp', 'alias')
```

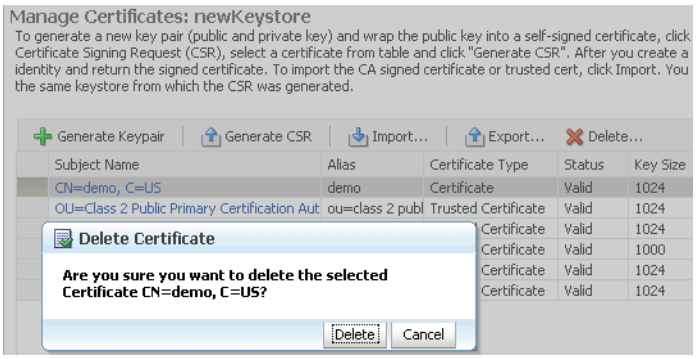
where `password` is the password for this keystore, `/tmp` is the path under which the certificate is generated in BASE64 format in the file `base64.txt`, and `alias` is the alias of the certificate being exported.

See Also: See Section 6.9 for details about using WLST commands.

8.3.5.9 Deleting a Certificate or Trusted Certificate from the Keystore Using Fusion Middleware Control

Take these steps to delete a certificate, or a trusted certificate, from a keystore:

1. From the navigation pane, locate your component instance.
2. Navigate to *component_name*, then **Security**, then **Keystores**.
3. Select the desired keystore from the list of stores.
4. A dialog box appears in which you must enter the keystore password to continue.
5. The Manage Certificates page appears. Select the certificate or trusted certificate to be deleted, and Click **Delete**.
6. A dialog box appears asking you to confirm the choice. Select **OK** to confirm.



8.3.5.10 Deleting a Certificate or Trusted Certificate from the Keystore Using WLST

Assuming the application server instance name is `inst1`, use this command to delete a certificate:

```
removeKeyStoreObject('inst1', 'ovd5', 'ovd', 'newKeystore', 'password',
'Certificate', 'alias')
```

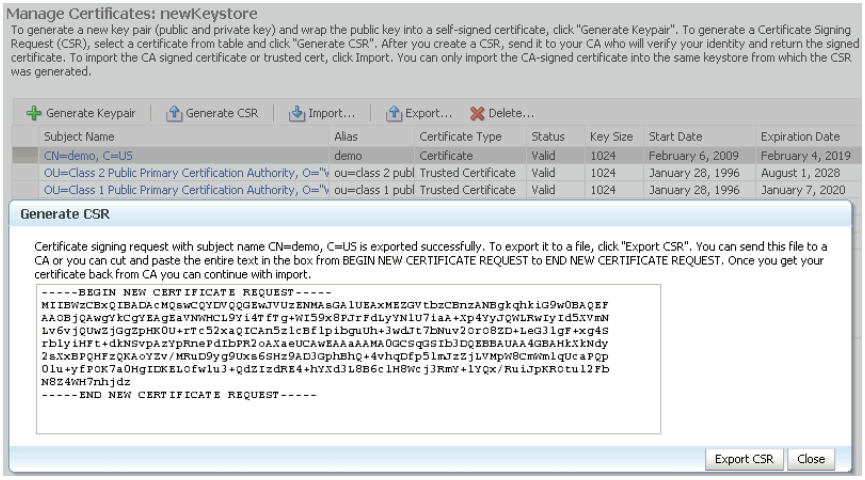
where `password` is the password for this keystore and `alias` is the alias of the certificate being deleted.

See Also: See Section 6.9 for details about using WLST commands.

8.3.5.11 Converting a Self-Signed Certificate to a Third-Party Certificate Using Fusion Middleware Control

Take these steps to convert a self-signed certificate, residing in a keystore, into a third-party certificate:

1. From the navigation pane, locate your component instance.
2. Navigate to *component_name*, then **Security**, then **Keystores**.
3. Select the keystore that contains the self-signed certificate from the list of stores.
4. A dialog box appears in which you must enter the keystore password; click **OK** to continue.
5. The Manage Certificates page appears.
6. A new certificate request must be generated for the self-signed certificate that is to be converted. Select the certificate and click **Generate CSR**. In this example, the request is made for the self-signed certificate with alias `demo`.



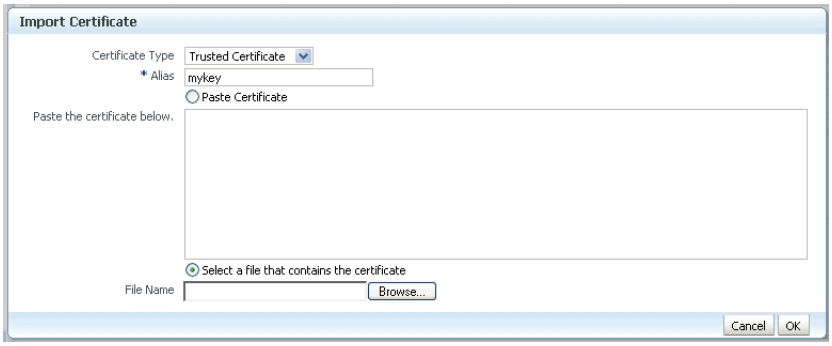
The certificate request is displayed.

7. You can either:
   - Copy and paste the Base64-encoded certificate request to a file.
   - Export it directly to a file with the **Export CSR** button.
8. Submit the certificate request file to a certificate authority (CA).
9. The CA signs the certificate request and generates a certificate. The CA will return you one of the following:
   - A single file containing both the newly generated certificate and its own CA certificate in `pkcs7` format
   - Two files, one containing the newly generated certificate and a second containing its own CA certificate
10. Use **Import** to import these files into your keystore:
    - If you received a single file from the CA, import it as a certificate, using an alias that matches the alias of the self-signed certificate you are replacing (from Step 6)
    - If you received two files:
      – Import the file containing the CA certificate as a trusted certificate (use an alias that is unique in the keystore)
      – Import the certificate file as a certificate (using an alias that matches the alias of the self-signed certificate you are replacing

---

Note: The order is important: you must import the trusted certificate first, followed by the certificate.

The CA returned a single file, which is imported as a certificate:



11. After import, the certificate issued by the CA replaces the self-signed certificate.

8.3.5.12 Converting a Self-Signed Certificate to a Third-Party Certificate Using WLST

Use these steps to convert a self-signed certificate to a third-party certificate (that is, one signed by a certificate authority):

1. Generate and export a CSR.

```
exportKeyStoreObject('inst1', 'ovd5', 'ovd', 'jks1', '<password>',
'CertificateRequest', '/tmp', 'mykey')
```

2. Submit the CSR `/tmp/base64.txt` to a certificate authority. The CA will return a newly generated certificate and its own certificate, either as one file in PKCS#7 format or as two separate files.

3. If you receive a single file from the CA, run the command:

```
importKeyStoreObject('inst1', 'ovd5', 'ovd', 'jks1', 'password', 'Certificate',
'/tmp/cert.txt', 'alias')
```

   where `password` is the password for this keystore, `/tmp/cert.txt` is the file that the CA returned and contains the BASE64 encoded PKCS#7, and `alias` is the alias by which this certificate is imported. Note that this alias must match that of the key pair that was used to generate the certificate request.

   If you receive two files from the CA, import the CA certificate first as a trusted certificate, followed by the newly generated certificate:

```
importKeyStoreObject('inst1', 'ovd5', 'ovd', 'jks1', 'password',
'TrustedCertificate', '/tmp/cacert.txt', 'unique_alias')
```

   where `unique_alias` is a unique alias by which the trusted certificate is imported.

```
importKeyStoreObject('inst1', 'ovd5', 'ovd', 'jks1', 'password', 'Certificate',
'/tmp/cert.txt', 'alias')
```

   where `password` is the password for this keystore, `/tmp/cert.txt` is the file that the CA returned and contains BASE64 encoded certificate, `/tmp/cacert.txt` is the file containing the BASE64 encoded CA certificate, and `alias` is the alias by which this certificate is imported. Note that this alias must match that of the key pair that was used to generate the certificate request.

See Also: See Section 6.9 for details about using WLST commands.

8.3.6 Keystore and Certificate Maintenance

This section contains the following administration topics:

- Location of Keystores
- Replacing Expiring Certificates
- Effect of Host Name Change on Keystores

8.3.6.1 Location of Keystores

The root directory for Oracle Virtual Directory keystores is located in `$ORACLE_INSTANCE/config/OVD/`*`ovd_instance_name`*`/keystores`.

This root directory will contain all the JKS files.

A sample structure, assuming there are two keystores named `keys.jks` and `trust.jks` respectively, would look like this:

`ORACLE_INSTANCE``/config/OVD/`*`ovd_instance_name`*`/keystores/keys.jks`
`ORACLE_INSTANCE``/config/OVD/`*`ovd_instance_name`*`/keystores/trust.jks`

8.3.6.2 Replacing Expiring Certificates

An expiring certificate should be replaced before it actually expires to avoid or reduce application downtime.

The steps for replacing an expiring certificate are as follows:

1. Generate a certificate request from the keystore (use the same key-pair for which the current expiring certificate was issued).
2. Provide this certificate request to the third-party Certificate Authority (CA) for certificate issuance. The validity date of the new certificate should be earlier than the expiration date of the current certificate. This overlap is recommended to reduce downtime.

> **Note:** Steps 1 and 2 are not required when the third-party CA already maintains the certificate request in a repository. In that case, simply ask the CA to issue a new certificate for that certificate request.

3. Import the newly issued certificate into the keystore using the same alias as that of the key-pair.
4. If the new certificate was issued by a CA other than the one that issued the original certificate, you may also need to import the new CA's trusted certificate before importing the newly issued certificate.

8.3.6.3 Effect of Host Name Change on Keystores

Typically, the certificate DN is based on the host name of the server where the keystore is used.

For example, if a keystore is being created for the Oracle Virtual Directory server on host my.example.com, then the DN of the certificate in this Oracle Virtual Directory keystore will be something like:

`"CN=my.example.com,O=`*`organization name`*`"`

This synchronization is required because most clients do host name verification during the SSL handshake.

Clients that perform host name verification include Web browsers and Oracle HTTP Client, among others. If the host name of the server does not match that of the certificate DN:

- A clear warning is displayed (in the case of browser clients).
- There may be SSL handshake failure (in the case of other clients).

Thus, whenever you have a keystore on a server that is accepting requests from clients, you must ensure that whenever the host name of this server changes, you also update the certificate in the keystore.

This can be done by requesting a new certificate with a new DN (based on the new host name).

For a Production Keystore

The steps are:

1. Generate a new request with the new DN (based on a new host name). See Section 8.3.5.3 for details.
2. Send this request to a certificate authority (CA).
3. Get back a new certificate from the CA.
4. Import the new certificate with the same alias as the key-pair for which certificate request was generated. See Section 8.3.5.5 for details.

For a Self-signed Keystore

The steps are:

1. Delete the existing keystore. See Section 8.3.3.5 for details.
2. Create a new keystore with a key-pair using the new DN (based on the new host name). See Section 8.3.3.1 for details.

For Both Keystore Types

For both production and self-signed keystores, once the new certificate is available in the keystore, ensure that it is imported into all the component keystores where it needs to be trusted. For example, if the HTTP listener on Oracle Virtual Directory was SSL-enabled and its certificate changed due to a host name change, then you need to import its new certificate into the client keystore or browser repository so that it can trust its new peer.

8.4 Wallet Management

This section contains the following topics:

- About Wallets and Certificates
- Accessing the Wallet Management Page in Fusion Middleware Control
- Managing the Wallet Life Cycle
- Common Wallet Operations
- Managing the Certificate Life Cycle
- Accessing the Certificate Management Page for Wallets in Fusion Middleware Control
- Common Certificate Operations

- Wallet and Certificate Maintenance

8.4.1 About Wallets and Certificates

This section contains the following topics:

- Password-Protected and Autologin Wallets
- Self-Signed and Third-Party Wallets
- Sharing Wallets Across Instances
- Wallet Naming Conventions

8.4.1.1 Password-Protected and Autologin Wallets

You can create two types of wallets:

- Auto-login wallet

  This is an obfuscated form of a PKCS#12 wallet that provides PKI-based access to services and applications without requiring a password at runtime. You can also add to, modify, or delete the wallet without needing a password. File system permissions provide the necessary security for auto-login wallets.

  Note: In previous releases, you could create a wallet with a password and then enable auto-login to create an obfuscated wallet. With 11*g* Release 2 (11.1.2), auto-login wallets are created without a password. When using such a wallet, you do not need to specify a password.

  If using an auto-login wallet without a password, specify a null password ("") in the `ldapbind` command.

  Older type of wallets (such as Release 10*g* wallets) will continue to work as they did earlier.

- Password-protected wallet

  As the name suggests, this type of wallet is protected by a password. Any addition, modification, or deletion to the wallet content requires a password.

  Every time a password-protected wallet is created, an auto-login wallet is automatically generated. However, this auto-login wallet is different from the user-created auto-login wallet described in the previous bullet. While the user-created wallet can even be updated at configuration time without a password, an automatically generated auto-login wallet is a read-only wallet that does not allow direct updates. Modifications to the wallet must occur through the password protected file (by providing a password), at which time the auto-login wallet is regenerated.

  The purpose of this system-generated auto-login wallet is to provide PKI-based access to services and applications without requiring a password at runtime, while still requiring a password at configuration time.

  Note: Wallets configured for Oracle Internet Directory must have auto-login enabled.

8.4.1.2 Self-Signed and Third-Party Wallets

Self-signed wallets contain certificates for which the issuer is the same as the subject. These wallets are typically created for use within an intranet environment where trust is not a high priority. Each self-signed wallet has its own unique issuer; hence, in an environment with multiple components and wallets, the trust management tasks increase n-fold.

When created through Fusion Middleware Control, a self-signed wallet is valid for five years.

Third-party wallets contain certificates that are issued by well known CA's. The functionality and security remain the same as for self-signed wallets, but the use of third-party certificates provides added trust because the issuers are well known, so they are already trusted by most clients.

Difference Between Self-Signed and Third-Party Wallets

From a functional and security perspective, a self-signed certificate is comparable to one issued by a third party. The only difference is that a self-signed certificate is not trusted.

8.4.1.3 Sharing Wallets Across Instances

Oracle recommends that you do not share wallets between component instances or Oracle instances, since each wallet represents a unique identity.

The exception to this is an environment with a cluster of component instances, in which case wallet sharing would be an acceptable practice.

Note that no management tools or interfaces are available to facilitate wallet sharing. However, you can export a wallet from one instance and import it into another instance. See Section 8.4.4 for details of wallet export and import.

8.4.1.4 Wallet Naming Conventions

Follow these naming conventions for your Oracle wallets:

- Do not use a name longer than 256 characters.
- Do not use any of the following characters in a wallet name:

  ```
  | ; , ! @ # $ ( ) < > / \ " ' ` ~ { } [ ] = + & ^ space tab
  ```

  Note: Observe this rule even your operating system supports the character.

- Do not use non-ascii characters in a wallet name.
- Additionally, follow the operating system-specific rules for directory and file names

Due to the way data is handled in an LDAP directory such as Oracle Internet Directory, wallet names are not case-sensitive.

Thus, it is recommended that you use case-insensitive wallet names (preferably, using all lower case letters). For example, if you have created a wallet named `UPPER`, do not create another wallet named `upper`; doing so could cause confusion during wallet management operations.

8.4.2 Accessing the Wallet Management Page in Fusion Middleware Control

An Oracle wallet is associated with the component where it is utilized. To locate a component instance:

1. Log into Fusion Middleware Control using administrator credentials.
2. Select the domain of interest.

   Note: You can use Setup to discover a specific Oracle WebLogic Server domain to work with.

3. From the navigation pane, locate the instance (for example, an OHS instance) that will use the wallet. Click on the instance.

   The component type now appears on the upper left of the page adjacent to the Farm drop-down.

4. Select the component type drop-down (for example, Oracle HTTP Server).

   If the component is not started, start it by right-clicking to open the component menu, press **Control**, then **Start Up**.

5. Navigate to **Security**, then **Wallets**.
6. The Wallets page appears.

On the Wallets page, you can:

- Create a wallet.
- Delete a wallet.
- Import a wallet.
- Export a wallet.

8.4.3 Managing the Wallet Life Cycle

Typical life cycle events for an Oracle wallet are as follows:

- The wallet is created. Wallets can be created directly, or by importing a wallet file from the file system.
- The list of available wallets are viewed and specific wallets selected for update.
- Wallets are updated or deleted. Update operations for password-protected wallets require that the wallet password be entered.
- The wallet password can be changed for password-protected wallets.
- The wallet can be deleted.
- Wallets can be exported and imported.

8.4.4 Common Wallet Operations

This section describes the steps required to perform a range of wallet management functions, including:

- Creating a Wallet Using Fusion Middleware Control
- Creating a Wallet Using WLST
- Creating a Self-Signed Wallet Using Fusion Middleware Control

- Creating a Self-Signed Wallet Using WLST
- Changing a Self-Signed Wallet to a Third-Party Wallet Using Fusion Middleware Control
- Changing a Self-Signed Wallet to a Third-Party Wallet Using WLST
- Exporting a Wallet Using Fusion Middleware Control
- Exporting a Wallet Using WLST
- Importing a Wallet Using Fusion Middleware Control
- Importing a Wallet Using WLST
- Deleting a Wallet Using Fusion Middleware Control
- Deleting a Wallet Using WLST

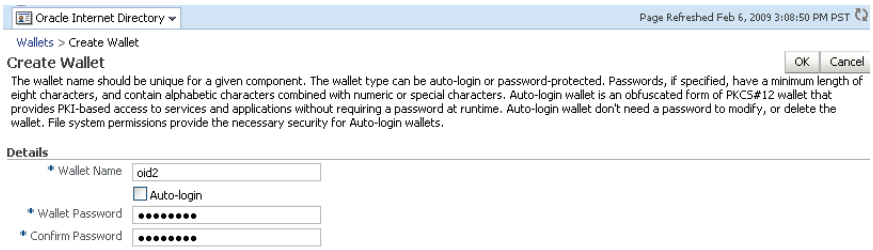
8.4.4.1 Creating a Wallet Using Fusion Middleware Control

Take these steps to a wallet:

1. Navigate to the Wallets page for your component instance. See Section 8.4.2, "Accessing the Wallet Management Page in Fusion Middleware Control."
2. Click **Create**.
3. The Create Wallet page appears.
4. Enter a wallet name.
5. Check or uncheck the **Autologin** box, depending on whether your wallet will be an auto-login wallet. The default is an auto-login wallet.

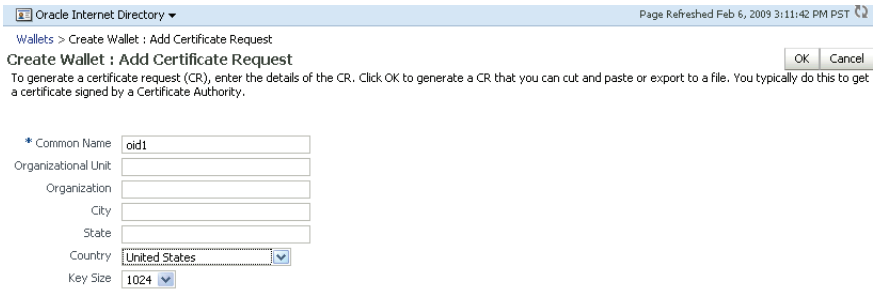
   See Section 8.4.1.1, "Password-Protected and Autologin Wallets" for details.

   Note: Wallets configured for Oracle Internet Directory must have auto-login enabled.



6. Click **Submit**.
7. At this point, you must choose whether to add a certificate request (CR) at this time. If you choose not to do so, you can always add the CR later; see Section 8.4.7.1, "Adding a Certificate Request Using Fusion Middleware Control."

   In this example, we choose to add a CR:



> **Note:** The common name entered here should match the host name of the Oracle HTTP Server to which clients will connect; this helps to prevent problems of the type mentioned in Section 8.4.8.2.

8. Click **Finish**.
9. There are two options for the CR:
   - Copy and paste the Base64-encoded certificate request from the text box to a file
   - Export it directly to a file with the **Export Certificate Request** button.
10. A message appears confirming the wallet creation.

8.4.4.2 Creating a Wallet Using WLST

> **Note:** The WLST commands described in this chapter use Oracle Internet Directory as the example component. The same commands can be executed for Oracle HTTP Server or Oracle Web Cache by changing the third parameter from `oid` to `ohs` or `webcache` respectively.

Assuming the instance name is `inst1`, use this command to create a wallet:

```
createWallet('inst1', 'oid1', 'oid', 'oid2', 'password')
```

where `oid2` is the wallet name and password is the password for this wallet. If an auto-login wallet needs to be created, the password should be specified as '' (that is, no text between the quotes).

> **See Also:** Section 6.9.7, "createWallet".

> **Note:** Wallets configured for Oracle Internet Directory must have auto-login enabled.

8.4.4.3 Creating a Self-Signed Wallet Using Fusion Middleware Control

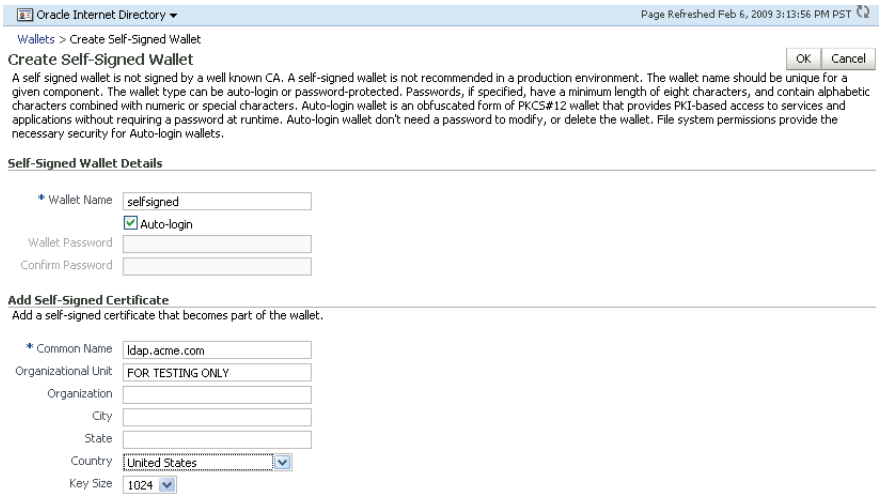
Take these steps to create a self-signed wallet:

> **See Also:** Section 8.4.1.2, "Self-Signed and Third-Party Wallets"

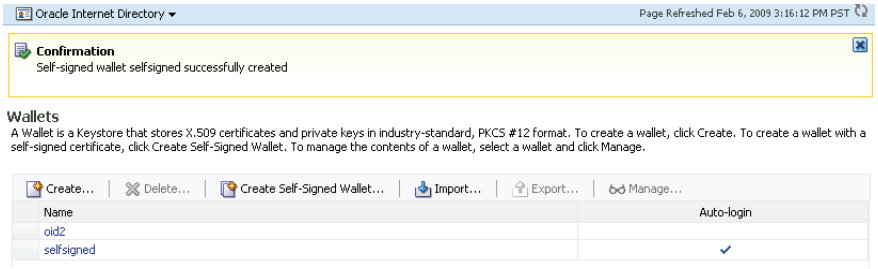
1. Navigate to the Wallets page for your component instance. See Section 8.4.2, "Accessing the Wallet Management Page in Fusion Middleware Control."
2. Click **Create Self-Signed Wallet**.
3. On the Self-Signed Wallet page, enter data to create the wallet. This includes:
   - The wallet name
   - Whether this is an auto-login wallet

     See Also: Section 8.4.1.1, "Password-Protected and Autologin Wallets"

   - The DN information
   - The key size



4. Click **Submit**.
5. A confirmation message is displayed and the new wallet appears in the list of wallets.



Note: Wallets configured for Oracle Internet Directory must have auto-login enabled.

8.4.4.4 Creating a Self-Signed Wallet Using WLST

Assuming the instance name is `inst1`, use these commands to create a self-signed wallet:

```
createWallet('inst1', 'oid1', 'oid', 'oid2', 'password')
addSelfSignedCertificate('inst1', 'oid1', 'oid', 'oid2', 'password', 'subject_dn',
```

```
'key_size')
```

where `oid2` is the wallet name, `subject_dn` is the distinguished name of the self-signed certificate, `key_size` is the key size in bits and `password` is the password for this wallet. If an auto-login wallet needs to be created, the password should be specified as '' (that is, with no text between the quotes).

See Also:

- Section 6.9.7, "createWallet"
- Section 6.9.2, "addSelfSignedCertificate"

Note: Wallets configured for Oracle Internet Directory must have auto-login enabled.

8.4.4.5 Changing a Self-Signed Wallet to a Third-Party Wallet Using Fusion Middleware Control

For steps to convert a self-signed wallet into a third-party wallet, see Section 8.4.8.3, "Changing a Self-Signed Wallet to a Third-Party Wallet."

8.4.4.6 Changing a Self-Signed Wallet to a Third-Party Wallet Using WLST

For steps to convert a self-signed wallet into a third-party wallet, see Section 8.4.8.3, "Changing a Self-Signed Wallet to a Third-Party Wallet."

8.4.4.7 Exporting a Wallet Using Fusion Middleware Control

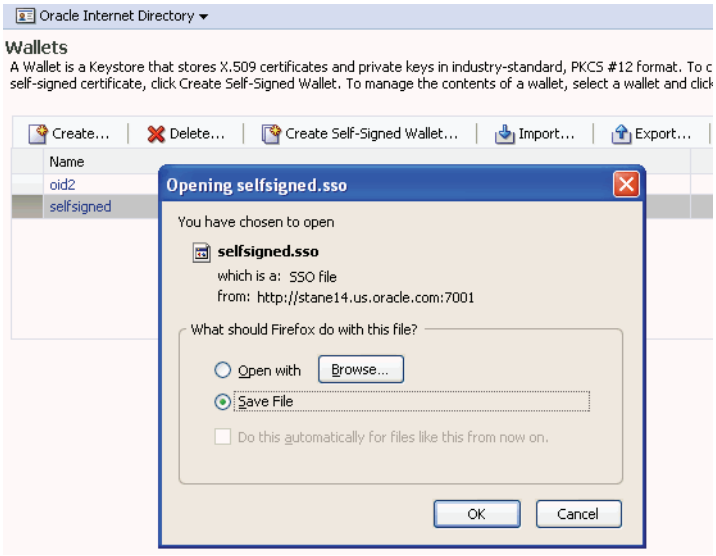
Take these steps to export a wallet:

1. Navigate to the Wallets page for your component instance. See Section 8.4.2, "Accessing the Wallet Management Page in Fusion Middleware Control."
2. Select the row corresponding to the wallet of interest.

Note: Do not click on the wallet name itself; this opens the wallet for certificate management operations.

---

3. Click **Export**.
4. The Export Wallet page appears.
5. Enter the filename and the location where the wallet is to be exported.
6. Click **OK**.



8.4.4.8 Exporting a Wallet Using WLST

Assuming the instance name is `inst1`, use this command to export a wallet:

```
exportWallet('inst1', 'oid1', 'oid', 'selfsigned', 'password', '/tmp')
```

where `password` is the password for this wallet (specify '' as password for auto-login wallet).

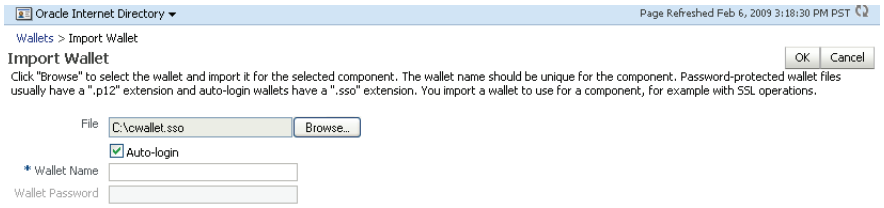
If it is an auto-login wallet, this command will export the wallet into a file named cwallet.sso under the directory `/tmp`. If it is a password-protected wallet, there will be two files created under `/tmp`, namely `ewallet.p12` and `cwallet.sso`.

See Also: Section 6.9.12, "exportWallet".

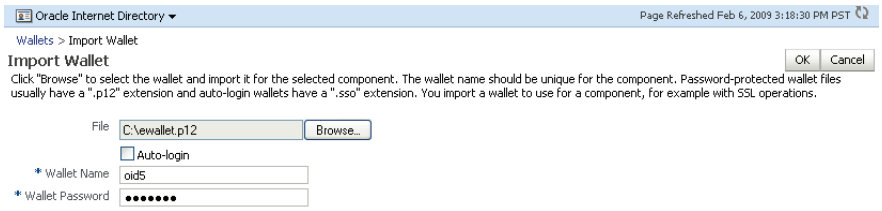
8.4.4.9 Importing a Wallet Using Fusion Middleware Control

Take these steps to import a wallet:

1. Navigate to the Wallets page for your component instance. See Section 8.4.2, "Accessing the Wallet Management Page in Fusion Middleware Control".
2. Click **Import**.
3. The Import Wallet page appears.
4. If this is an auto-login wallet, check the box and enter the wallet name. No password is required.



5. If this is not an auto-login wallet, uncheck the auto-login box. Specify both the wallet name and password.



6. Click **OK**. The wallet is imported into the repository.

8.4.4.10 Importing a Wallet Using WLST

Assuming the instance name is `inst1`, use this command to import a wallet:

```
importWallet('inst1', 'oid1', 'oid', 'oid5', 'password', '/tmp/ewallet.p12')
```

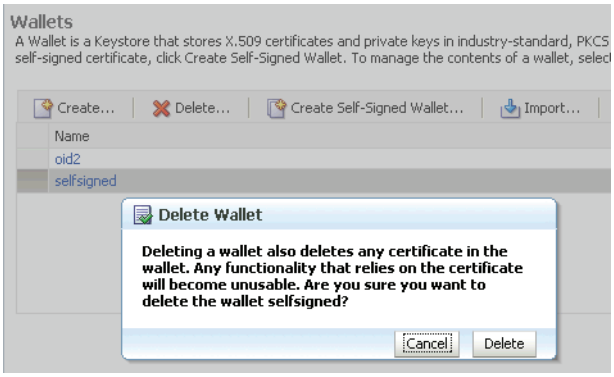
where `password` is the password of the wallet being imported and `/tmp/ewallet.p12` contains the wallet file (if there are two files `ewallet.p12` and `cwallet.sso`, point to `ewallet.p12`). Point to `cwallet.sso` only if it is an auto-login wallet - in this case, the password should be specified as ".

See Also: Section 6.9.20, "importWallet".

8.4.4.11 Deleting a Wallet Using Fusion Middleware Control

Take these steps to delete a wallet:

1. Navigate to the Wallets page for your component instance. See Section 8.4.2, "Accessing the Wallet Management Page in Fusion Middleware Control."
2. Select the row corresponding to the wallet of interest.
3. Click **Delete**.



4. The wallet is deleted and no longer appears on the list of wallets.

8.4.4.12 Deleting a Wallet Using WLST

Assuming the instance name is `inst1`, use this command to delete a wallet:

```
deleteWallet('inst1', 'oid1', 'oid', 'selfsigned')
```

See Also: Section 6.9.9, "deleteWallet".

8.4.5 Managing the Certificate Life Cycle

The complete certificate life cycle, starting from wallet creation, includes these actions:

1. Create an empty wallet (that is, a wallet that does not contain a certificate request).
2. Add a certificate request to the wallet.
3. Export the certificate request.
4. Use the certificate request to obtain the corresponding certificate.
5. Import trusted certificates.
6. Import the certificate.

These steps are needed to generate a wallet with a third-party trusted certificate. For details about this task, see Section 8.4.7.9, "Converting a Self-Signed Certificate into a Third-Party Certificate Using Fusion Middleware Control."

> **See Also:** Section 8.4.6, "Accessing the Certificate Management Page for Wallets in Fusion Middleware Control"

8.4.6 Accessing the Certificate Management Page for Wallets in Fusion Middleware Control

An Oracle wallet is associated with the component where it is utilized.

To locate a component instance:

1. Log into Fusion Middleware Control using administrator credentials.
2. Select the domain of interest.

> **Note:** You can use Setup to discover a specific Oracle WebLogic Server domain to work with.

3. Use the navigation pane to locate the instance (for example, an Oracle HTTP Server instance) that will use the wallet. *Note*: Oracle HTTP Server must be running so that subsequent steps will work.

- Use the navigation pane to locate the instance (for example, an Oracle HTTP Server instance) that will use the wallet. *Note*: Oracle HTTP Server must be running so that subsequent steps will work.

After locating your component instance, there are two ways you can access a wallet's certificate management page in Fusion Middleware Control:

- Go to the `Wallets` page, select the line for the wallet of interest and click **Manage**.
- Go to the `Wallets` page, locate the wallet of interest, and click on the wallet name.

On the Certificate Management page, you can:

- Add a certificate request.
- Export a certificate request, a certificate, or a trusted certificate.
- Import a certificate or a trusted certificate.
- Delete a certificate request, a certificate, or a trusted certificate.

8.4.7 Common Certificate Operations

This section describes the following common certificate operations:

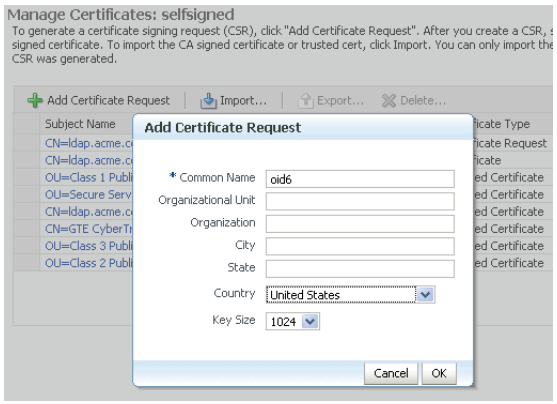
- Adding a Certificate Request Using Fusion Middleware Control

- Adding a Certificate Request Using WLST
- Exporting a Certificate, Certificate Request, or a Trusted Certificate Using Fusion Middleware Control
- Exporting a Certificate, Certificate Request, or a Trusted Certificate Using WLST
- Importing a Certificate or a Trusted Certificate Using Fusion Middleware Control
- Importing a Certificate or a Trusted Certificate Using WLST
- Deleting a Certificate Request, a Certificate, or a Trusted Certificate Using Fusion Middleware Control
- Deleting a Certificate Request, a Certificate, or a Trusted Certificate Using WLST
- Converting a Self-Signed Certificate into a Third-Party Certificate Using Fusion Middleware Control
- Converting a Self-Signed Certificate into a Third-Party Certificate Using WLST

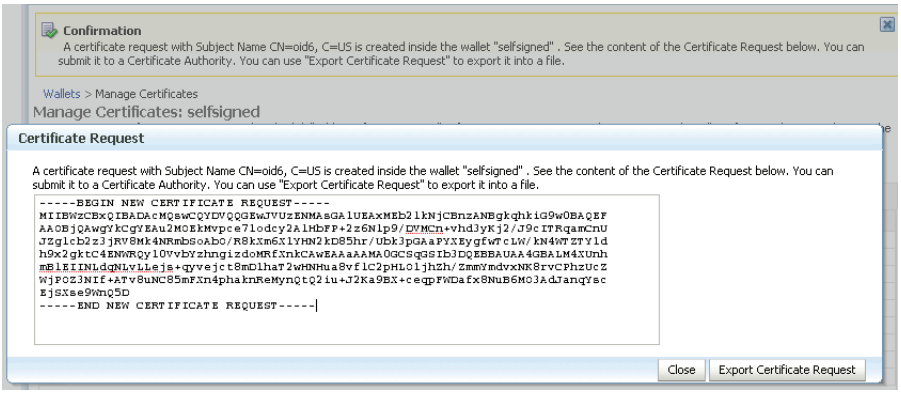
8.4.7.1 Adding a Certificate Request Using Fusion Middleware Control

It is possible to add a certificate request at the time you create the wallet, but if it was not done at that time, you can do so using the following steps:

1. Navigate to the Certificate Management page. See Section 8.4.6, "Accessing the Certificate Management Page for Wallets in Fusion Middleware Control."
2. Click **Add Certificate Request**.
3. A dialog box appears where you enter the CRs DN values:



   Fields marked with an asterisk (*) are required. *Note*: The common name must be the host name that clients will use to access the component.
4. Click **OK**.
5. The new CR is generated and a dialog box appears with the CR in the text box. You can either:
   - Copy and paste the Base64-encoded certificate request to a file.
   - Export it directly to a file with the **Export Certificate Request** button.



8.4.7.2 Adding a Certificate Request Using WLST

Assuming the instance name is `inst1`, use this command to add a certificate request for a wallet:

```
addCertificateRequest('inst1', 'oid1', 'oid', 'selfsigned', 'password', 'subject_dn', 'key_size')
```

where `password` is the password for this wallet, `subject_dn` is the distinguished name by which the certificate request is generated and `key_size` is the key size in bits.

See Also: Section 6.9.1

8.4.7.3 Exporting a Certificate, Certificate Request, or a Trusted Certificate Using Fusion Middleware Control

Take these steps to export a certificate, a certificate request (CR), or a trusted certificate:

1. Navigate to the Certificate Management page. See Section 8.4.6, "Accessing the Certificate Management Page for Wallets in Fusion Middleware Control."
2. Select the certificate, CR, or trusted certificate and click **Export**.
3. A dialog box appears with the certificate, CR, or trusted certificate in the text box. You can either:
   - Copy and paste the Base64-encoded certificate to a file.
   - Export it directly to a file with the **Export Certificate** or **Export Trusted Certificate** button.

8.4.7.4 Exporting a Certificate, Certificate Request, or a Trusted Certificate Using WLST

Assuming the instance name is `inst1`, use this command to export a certificate request:

```
exportWalletObject('inst1', 'oid1', 'oid', 'selfsigned', 'password',
'CertificateRequest', '/tmp', 'subject_dn')
```

where `password` is the password for this wallet, `/tmp` is the path under which the certificate request is exported in BASE64 format in the file `base64.txt`, and `subject_dn` is the distinguished name of the certificate request that is exported.

To export a certificate or trusted certificate, replace `CertificateRequest` in the above command with `Certificate` or `TrustedCertificate`.

See Also: Section 6.9.13

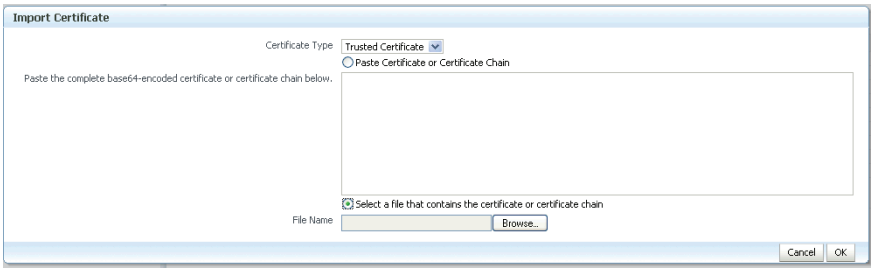
8.4.7.5 Importing a Certificate or a Trusted Certificate Using Fusion Middleware Control

Note: You cannot use Fusion Middleware Control to import DER-encoded certificates or trusted certificates into an Oracle wallet. Use one of these tools instead:

- Oracle Wallet Manager or
- `orapki` command-line tool

Take these steps to import a certificate or a trusted certificate:

1. Navigate to the Certificate Management page. See Section 8.4.6, "Accessing the Certificate Management Page for Wallets in Fusion Middleware Control."
2. Click **Import**.
3. In the Import Certificate dialog, you can select either a certificate or a trusted certificate.
4. There are two ways to do the import:
   - Paste the Base64-encoded certificate or trusted certificate in the text box.
   - Use the file selector to browse your file system to locate a file containing the Base64-encoded certificate or trusted certificate.



5. Click **OK**.

8.4.7.6 Importing a Certificate or a Trusted Certificate Using WLST

Note: You cannot use the `WLST` command-line tool to import DER-encoded certificates or trusted certificates into an Oracle wallet. Use one of these tools instead:

- Oracle Wallet Manager or
- `orapki` command-line tool

Assuming the instance name is `inst1`, use this command to import a certificate into a wallet:

```
importWalletObject('inst1', 'oid1', 'oid', 'selfsigned', 'password',
'Certificate', '/tmp/cert.txt')
```

where `password` is the password for this wallet and `/tmp/cert.txt` is the file that contains BASE64 encoded certificate.

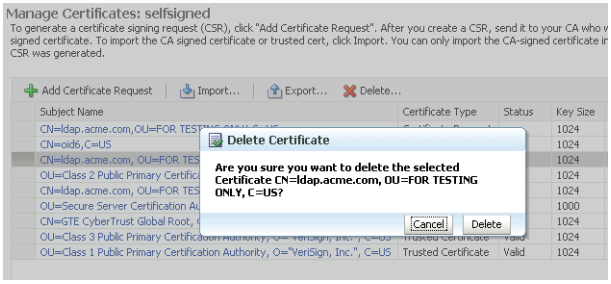
To import a trusted certificate, replace `Certificate` in the above command with `TrustedCertificate`.

See Also: Section 6.9.21

8.4.7.7 Deleting a Certificate Request, a Certificate, or a Trusted Certificate Using Fusion Middleware Control

Take these steps to delete a CR, a certificate, or a trusted certificate:

1. Navigate to the Certificate Management page. See Section 8.4.6, "Accessing the Certificate Management Page for Wallets in Fusion Middleware Control."
2. Select the row containing the certificate request, certificate or trusted certificate.
3. Click **Delete**.
4. A dialog box appears, requesting confirmation.



5. Click **Yes**.
6. The object no longer appears in the Manage Certificates list.

8.4.7.8 Deleting a Certificate Request, a Certificate, or a Trusted Certificate Using WLST

Assuming the instance name is inst1, use this command to delete a certificate:

```
removeWalletObject('inst1', 'oid1', 'oid', 'selfsigned', 'password',
'Certificate', 'subject_dn')
```

where `password` is the password for this wallet and `subject_dn` is the distinguished name of the certificate being deleted.

To delete a certificate request or trusted certificate, replace `Certificate` in the above command with `CertificateRequest` or `TrustedCertificate`.

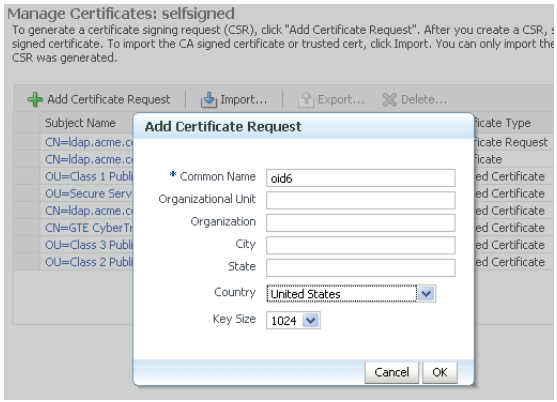
See Also: Section 6.9

8.4.7.9 Converting a Self-Signed Certificate into a Third-Party Certificate Using Fusion Middleware Control

A self-signed certificate residing in a wallet can be converted into a third-party certificate signed by a certificate authority (CA). Take these steps to perform the task:

Note: The steps are illustrated for use with Oracle Internet Directory, and similar steps are applicable for generating wallets to use with Oracle HTTP Server and Oracle Web Cache.

1. From the navigation pane, locate your component instance.
2. Navigate to *component_name*, the **Security**, then **Wallets**.
3. From the list of wallets, select the wallet that contains the self-signed certificate.
4. The Manage Certificates page appears. It contains the list of certificates in the wallet.
5. A new certificate request must be generated for the self-signed certificate that is to be converted. Select the self-signed certificate and click **Add Certificate Request**. A dialog box appears:



Note: The common name entered here should match the host name of the server to which clients will connect; this helps to prevent problems of the type mentioned in Section 8.4.8.2.

6. Enter the certificate request (CR) details and click **OK**.

   The CR is generated. You can either:

   - Copy and paste the Base64-encoded certificate request to a file.
   - Export it directly to a file with the **Export Certificate Request** button.

7. Submit the certificate request file to a certificate authority to generate a certificate. This is an offline procedure that you can execute in accordance with your local policy for obtaining certificates.
8. The CA signs the certificate request and generates a certificate. The CA will return you one of the following:

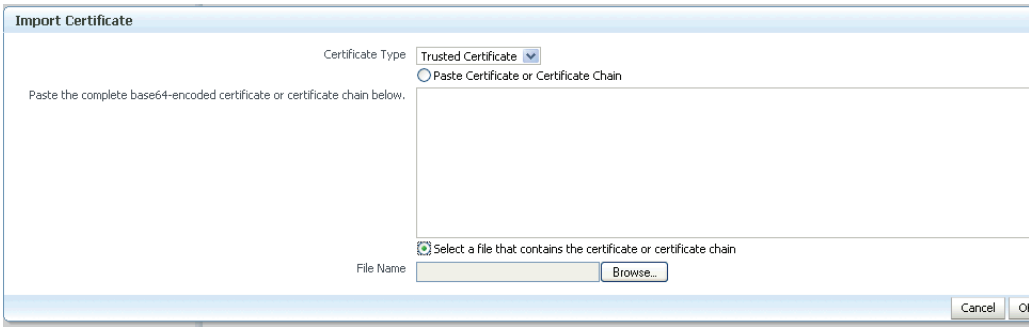
   - A single file containing both the newly generated certificate and its own CA certificate in `pkcs7` format
   - Two files, one containing the newly generated certificate and a second containing its own CA certificate (or certificates, if there is a chain)

9. Use **Import** to import these files into your wallet:

- If you received a single file from the CA, import it as a trusted certificate, using an alias that matches the alias of the self-signed certificate you are replacing (from Step 3).
- If you received two files:
  - Import the file containing the CA certificate as a trusted certificate (use an alias that is unique in the wallet).
  - Import the certificate file as a certificate (using an alias that matches the alias of the self-signed certificate you are replacing).

Note: The order is important: you must import the trusted certificate first, followed by the certificate.

The CA returned a single file, which is imported as a trusted certificate:



10. After import, the certificate issued by the CA replaces the self-signed certificate.

8.4.7.10 Converting a Self-Signed Certificate into a Third-Party Certificate Using WLST

See Also: Section 6.9

Follow these steps to convert a self signed certificate to a third-party certificate using WLST:

1. Add a certificate request, for example:

```
addCertificateRequest('inst1', 'oid1', 'oid', 'selfsigned', 'password',
'subject_dn', 'key_size')
```

2. Export the certificate request:

```
exportWalletObject('inst1', 'oid1', 'oid', 'selfsigned', 'password',
'CertificateRequest', '/tmp', 'subject_dn')
```

3. Submit the certificate request `/tmp/base64.txt` to a certificate authority. The CA will return a newly generated certificate and its own certificate, either as one file in PKCS#7 format or as two separate files.
4. If you receive a single file from the CA, run the following command

```
importWalletObject('inst1', 'oid1', 'oid', 'selfsigned', 'password',
'TrustedChain', '/tmp/cert.txt')
```

where `password` is the password for this wallet and `/tmp/cert.txt` is the file that the CA returned and contains BASE64 encoded PKCS#7.

If you receive two files from the CA, import the CA certificate first as a trusted certificate, followed by the newly generated certificate.

```
importWalletObject('inst1', 'oid1', 'oid', 'selfsigned', 'password',
'TrustedCertificate', '/tmp/cacert.txt')

importWalletObject('inst1', 'oid1', 'oid', 'selfsigned', 'password',
'Certificate', '/tmp/cert.txt')
```

where `password` is the password for this wallet, `/tmp/cert.txt` is the file that the CA returned and contains BASE64 encoded certificate and `/tmp/cacert.txt` is the file containing the BASE64 encoded CA certificate.

8.4.8 Wallet and Certificate Maintenance

This section contains the following administration topics:

- Location of Wallets
- Effect of Host Name Change on a Wallet
- Changing a Self-Signed Wallet to a Third-Party Wallet
- Replacing an Expiring Certificate in a Wallet

8.4.8.1 Location of Wallets

This section describes the location of wallets for different components.

Root Directory for an Oracle Internet Directory Wallet

The root directory for wallets is `$ORACLE_INSTANCE/OID/admin`.

This root directory will contain subdirectories with wallet names; these subdirectories will contain the actual wallet files.

For example, assuming there are two wallets named `oid1` and `oid2`, respectively; `oid1` is a password-protected wallet, and `oid2` is an auto-login-only wallet. A sample structure could look like:

```
$ORACLE_INSTANCE/OID/admin/oid1/cwallet.sso
$ORACLE_INSTANCE/OID/admin/oid1/ewallet.p12
$ORACLE_INSTANCE/OID/admin/oid2/cwallet.sso
```

Root Directory for an Oracle HTTP Server Wallet

The root directory for wallets is `$ORACLE_INSTANCE/config/OHS/`*`ohs_instance_name`*`/keystores`.

This root directory contains subdirectories with wallet names; these subdirectories contain the actual wallet files.

For example, assuming there are two wallets named ohs1 and ohs2, respectively; ohs1 is a password-protected wallet, and ohs2 is an auto-login-only wallet. A sample structure could look like:

```
$ORACLE_INSTANCE/config/OHS/ohs_instance1/keystores/ohs1/cwallet.sso
$ORACLE_INSTANCE/config/OHS/ohs_instance1/keystores/ohs1/ewallet.p12
$ORACLE_INSTANCE/config/OHS/ohs_instance1/keystores/ohs2/cwallet.sso
```

Root Directory for an Oracle Web Cache Wallet

The root directory for wallets is `$ORACLE_INSTANCE/config/WebCache/`*`webcache_instance_name`*`/keystores`.

This root directory will contain subdirectories with wallet names; these subdirectories will contain the actual wallet files.

For example, assuming there are two wallets named `wc1` and `wc2`, respectively; `wc1` is a password-protected wallet, and `wc2` is an auto-login-only wallet. A sample structure could look like this:

```
$ORACLE_INSTANCE/config/WebCache/webcache_instance1/keystores/wc1/cwallet.sso
$ORACLE_INSTANCE/config/WebCache/webcache_instance1/keystores/wc1/ewallet.p12
$ORACLE_INSTANCE/config/WebCache/webcache_instance1/keystores/wc2/cwallet.sso
```

8.4.8.2 Effect of Host Name Change on a Wallet

Typically, the wallet DN is based on the host name of the server where the wallet is used.

For example, if a wallet is being created for the Oracle HTTP Server my.example.com, then the DN of the certificate in this Oracle HTTP Server wallet will be something like "CN=my.example.com,O=`organization name`".

This synchronization is required because most clients do host name verification during the SSL handshake.

Clients that perform host name verification include Web browsers and Oracle HTTPClient, among others. If the host name of the server does not match that of the certificate DN, then:

- A clear warning will be displayed (in the case of browser clients).
- There may be SSL handshake failure (in the case of other clients).

Thus, when you have a wallet on a server that is accepting requests from clients, you must ensure that whenever the host name of this server changes, you also update the certificate in the wallet.

You can do this by requesting a new certificate with a new DN (based on the new host name).

For a Production Wallet

The steps are:

- Generate a new request with the new DN (based on new host name).
- Send this request to a certificate authority (CA).
- Get back a new certificate from the CA.
- Delete the older certificate and certificate request from the wallet.
- Import the new certificate.

See Section 8.4.4 for details about these operations.

For a Self-signed Wallet

The steps are:

- Delete the existing wallet.
- Create a new wallet with a self-signed certificate using the new DN (based on the new host name).

See Section 8.4.4 for details about these operations.

For both production and self-signed wallets, once the new certificate is available in the wallet, you need to ensure that it is imported into all the component wallets where it needs to be trusted. For example, if Oracle WebLogic Server is SSL-enabled and the certificate for Oracle WebLogic Server changed due to a host name change, then you need to import its new certificate into the Oracle HTTP Server wallet so that it can trust its new peer.

8.4.8.3 Changing a Self-Signed Wallet to a Third-Party Wallet

You can convert a self-signed wallet into a third-party wallet, one that contains certificates signed by a trusted Certificate Authority (CA).

Assuming a self-signed wallet named `MYWallet`, containing a certificate with DN as "`CN=my.example.com,O=example`", take these steps to convert it into a third-party wallet:

1. Remove the user certificate "`CN=my.example.com,O=example`" from the wallet.
2. Remove the trusted certificate "`CN=my.example.com,O=example`" from the wallet (this has the same DN as the user certificate, but is a separate entity nonetheless).
3. Export the certificate request "`CN=my.example.com,O=example`" from the wallet and save it to a file.
4. Give this certificate request file to a third-party certificate authority (CA) such as Verisign.
5. The CA will return one of the following:
   - A user certificate file and its own certificate file
   - A single file with a certificate chain consisting of a user certificate and its own certificate
6. Import the above file(s) into the wallet.

See Section 8.4.4 for details about these operations.

8.4.8.4 Replacing an Expiring Certificate in a Wallet

An expiring certificate should be replaced before it actually expires to avoid or reduce application downtime.

The steps for replacing an expiring certificate are as follows:

1. Export the certificate request from the wallet (this is the same request for which the current expiring certificate was issued).
2. Provide this certificate request to the third-party Certificate Authority (CA) for certificate issuance. The validity date of the new certificate should be earlier than the expiration date of the current certificate. This overlap is recommended to reduce downtime.

> **Note:** Steps 1 and 2 are not required when the third-party CA already maintains the certificate request in a repository. In that case, simply request the CA to issue a new certificate for that certificate request.

3. Remove the existing certificate (the one that is about to expire) from the wallet.

4. Import the newly issued certificate into the wallet.

   To reduce downtime, remove the previous certificate and import the new certificate in the overlap period when the new certificate has become valid and the older one has not yet expired.

5. If the new certificate was issued by a CA other than the one that issued the original certificate, you may also need to import the new CA's trusted certificate before importing the newly issued certificate.

See Section 8.4.4 for details about these operations.

Part IV

Deploying Applications

This part describes the deployment process and how to deploy applications to Oracle Fusion Middleware.

Part IV contains the following chapters:

9 Understanding the Deployment Process

Before you deploy Oracle Fusion Middleware applications, such as Java EE applications or SOA Composite applications, you should understand the deployment process, such as designing and developing applications and deploying those applications to Managed Servers.

This chapter describes the following topics:

- What Is a Deployer?
- General Procedures for Moving from Application Design to Production Deployment
- Diagnosing Typical Problems

9.1 What Is a Deployer?

A **deployer** is responsible for deploying applications, such as Java EE applications, ADF applications, SOA Composite applications, or WebCenter Portal applications, to WebLogic Server instances or clusters.

A user who is functioning as a deployer should be granted the Oracle WebLogic Server deployer security role. The deployer security role allows deployment operations, as well as viewing the server configuration and changing startup and shutdown classes. To grant this role to a user, use the Oracle WebLogic Server Administration Console. See "Managing Security Roles" in the Oracle WebLogic Server Administration Console Help for more information.

9.2 General Procedures for Moving from Application Design to Production Deployment

This section describes the general procedures involved in moving from application design and development to deployment in a production environment. It contains the following topics:

- Designing and Developing an Application
- Deploying an Application to Managed Servers
- Automating the Migration of an Application to Other Environments

9.2.1 Designing and Developing an Application

In many cases, developers use Oracle JDeveloper to create their applications. Oracle JDeveloper is an integrated development environment (IDE) for building service-oriented applications using the latest industry standards for Java, XML, Web

services, portlets, and SQL. JDeveloper supports the complete software development life cycle, with integrated features for modeling, coding, debugging, testing, profiling, tuning, and deploying applications.

In this environment, you use the integrated Oracle WebLogic Server, which is packaged with Oracle JDeveloper for testing your applications.

For information about developing your applications, see:

- *Oracle Fusion Middleware Developing Applications for Oracle WebLogic Server*
- *Oracle Fusion Middleware Fusion Developer's Guide for Oracle Application Development Framework*
- *Oracle Fusion Middleware Developer's Guide for Oracle WebCenter Portal*
- *Oracle Fusion Middleware Developer's Guide for Oracle SOA Suite*

9.2.2 Deploying an Application to Managed Servers

After you have designed and tested your application with the integrated Oracle WebLogic Server, you can deploy the application to a Managed Server instance. For example, you may have installed Oracle WebLogic Server and configured a domain, including a Managed Server, in your production environment and you want to deploy the application to that Managed Server.

The following books provide specific information about deploying the different types of applications:

- For Java EE applications, see *Oracle Fusion Middleware Deploying Applications to Oracle WebLogic Server*
- For Oracle ADF, see the *Oracle Fusion Middleware Administrator's Guide for Oracle Application Development Framework*
- For Oracle WebCenter Portal, see the *Oracle Fusion Middleware Administrator's Guide for Oracle WebCenter Portal*
- For Oracle SOA Suite, see the *Oracle Fusion Middleware Developer's Guide for Oracle SOA Suite*

This section provides an outline of the major steps involved when you migrate your application from the integrated Oracle WebLogic Server to an environment separate from the development environment. Those general steps are:

1. Package the application. For Java EE, ADF, and WebCenter Portal applications, you package the application in an EAR file. For Oracle SOA Suite, you package the application into a JAR or ZIP file.

   For information about packaging the application, see:

   - For Java EE applications: "Preparing Applications and Modules for Deployment" in the *Oracle Fusion Middleware Deploying Applications to Oracle WebLogic Server*
   - For Oracle ADF: "What You May Need to Know About EAR Files and Packaging" in the *Oracle Fusion Middleware Fusion Developer's Guide for Oracle Application Development Framework*
   - For Oracle WebCenter Portal: "Packaging a WebCenter Portal Application" in the *Oracle Fusion Middleware Developer's Guide for Oracle WebCenter Portal*
   - For Oracle SOA Suite: "Understanding the Packaging Impact" in the *Oracle Fusion Middleware Developer's Guide for Oracle SOA Suite*

2. Set up your environment. This includes:

   - Installing and configuring a domain and a Managed Server that is configured with the correct domain template. For example, if you are deploying an Oracle SOA Suite application, the Managed Server must use the Oracle SOA Suite domain template. The appropriate domain template is applied when you create the domain using the Configuration Wizard. Alternatively, you can extend a domain to use another domain template, as described in Section 19.2.

     For more information about installing and configuring for specific components, see:

     – For Oracle ADF: "How to Install the ADF Runtime to the WebLogic Installation" in the *Oracle Fusion Middleware Administrator's Guide for Oracle Application Development Framework*

     – For Oracle WebCenter Portal: "Installing Oracle WebCenter Portal" and "Configuring Oracle WebCenter Portal" in the *Oracle Fusion Middleware Installation Guide for Oracle WebCenter Portal*

     – For Oracle SOA Suite: "Installing Oracle SOA Suite and Oracle Business Process Management Suite" and "Configuring Oracle SOA Suite and Oracle Business Process Management Suite" in the *Oracle Fusion Middleware Installation Guide for Oracle SOA Suite and Oracle Business Process Management Suite*

   - Creating any necessary schemas in an existing database. See the *Oracle Fusion Middleware Repository Creation Utility User's Guide*.

   - Registering the MDS Repository with the Oracle WebLogic Server domain, if your application uses the MDS Repository. For example, Oracle SOA Suite and Oracle WebCenter Portal applications require MDS. Some ADF applications involve customizations using MDS. See Section 14.3.2.1.1 for information about registering the MDS Repository.

3. If your application uses a database, set up the JDBC data sources.

   For more information about setting up the JDBC data sources, see:

   - For pure Java EE applications: *Oracle Fusion Middleware Configuring and Managing JDBC Data Sources for Oracle WebLogic Server*

   - For Oracle ADF: "How to Create a JDBC Data Source for Oracle WebLogic Server" in the *Oracle Fusion Middleware Administrator's Guide for Oracle Application Development Framework*

   - For Oracle WebCenter Portal: "Choosing the Data Source" in the *Oracle Fusion Middleware Administrator's Guide for Oracle WebCenter Portal*

   - For Oracle SOA Suite: "Creating Data Sources and Queues" in the *Oracle Fusion Middleware Developer's Guide for Oracle SOA Suite*

4. For Oracle SOA Suite, create connection factories and connection pooling. For more information, see "Creating Connection Factories and Connection Pooling" in the *Oracle Fusion Middleware Developer's Guide for Oracle SOA Suite*.

5. Create a connection to the target Managed Server.

   From Oracle JDeveloper, you can deploy your applications to Managed Server instances that reside outside JDeveloper. To do this, you must first create a connection to the server instance to which you want to deploy your application.

   For more information, see:

- For Oracle ADF: "How to Create a Connection to the Target Application Server" in the *Oracle Fusion Middleware Fusion Developer's Guide for Oracle Application Development Framework*
- For Oracle WebCenter Portal: "Creating a WebLogic Managed Server Connection" in the *Oracle Fusion Middleware Developer's Guide for Oracle WebCenter Portal*
- For Oracle SOA Suite: "Creating an Application Server Connection" in the *Oracle Fusion Middleware Developer's Guide for Oracle SOA Suite*

6. For Oracle SOA Suite, create a SOA-MDS connection, if the SOA composite application shares metadata with other composites. See "Creating a SOA-MDS Connection" in the *Oracle Fusion Middleware Developer's Guide for Oracle SOA Suite*.
7. Create a configuration plan or deployment plan, which contains information about environment-specific values, such as JDBC connection strings or host names of various servers. For more information, see:
   - For pure Java EE applications: "Creating a New Deployment Plan to Configure an Application" in the *Oracle Fusion Middleware Deploying Applications to Oracle WebLogic Server*
   - For Oracle SOA Suite: "Introduction to Configuration Plans" in the *Oracle Fusion Middleware Developer's Guide for Oracle SOA Suite*
8. Migrate application security, such as credentials, identities, and policies. For more information, see:
   - For pure Java EE applications: "Migrating Security Data" in the *Oracle Fusion Middleware Securing Oracle WebLogic Server*
   - For Oracle ADF: "Preparing the Secure Application for Deployment" in the *Oracle Fusion Middleware Fusion Developer's Guide for Oracle Application Development Framework*
   - For Oracle WebCenter Portal: "Managing WebCenter Portal Application Security" in the *Oracle Fusion Middleware Administrator's Guide for Oracle WebCenter Portal*
   - For Oracle SOA Suite: "Enabling Security" in the *Oracle Fusion Middleware Developer's Guide for Oracle SOA Suite*
9. Create a deployment profile. A **deployment profile** packages or archives a custom ADF, WebCenter Portal, or SOA application and associated files so that the application can be deployed to an Oracle WebLogic Server Managed Server instance. Deployment profiles are created at the project and application level.

   For more information, see:
   - For Oracle ADF: "How to Create Deployment Profiles" in the *Oracle Fusion Middleware Fusion Developer's Guide for Oracle Application Development Framework*
   - For Oracle WebCenter Portal: "Creating Deployment Profiles" in the *Oracle Fusion Middleware Developer's Guide for Oracle WebCenter Portal*
   - For Oracle SOA Suite: "Optionally Creating a Project Deployment Profile" in the *Oracle Fusion Middleware Developer's Guide for Oracle SOA Suite*
10. Migrate Oracle JDeveloper extensions for Oracle SOA Suite and Oracle WebCenter Portal. Table 9–1 shows the extensions and where they are documented:

Table 9–1 Oracle JDeveloper Extensions

Component	Extension	See
Oracle WebCenter Portal	WebCenter Portal extensions	"Creating and Provisioning a WebLogic Managed Server Instance" in the *Oracle Fusion Middleware Developer's Guide for Oracle WebCenter Portal*
Oracle SOA Suite	SOA extensions	"Installing Additional Oracle Fusion Middleware Design Time Components" in the *Oracle Fusion Middleware Installation Guide for Oracle JDeveloper*

11. Deploy the application to a Managed Server.

    For more information, see:

    - For pure Java EE applications: "Exporting an Application for Deployment to New Environments" in the *Oracle Fusion Middleware Deploying Applications to Oracle WebLogic Server*
    - For Oracle ADF: "Deploying the Application" in the *Oracle Fusion Middleware Fusion Developer's Guide for Oracle Application Development Framework*
    - For Oracle WebCenter Portal: "Deploying the Application to a WebLogic Managed Server" in the *Oracle Fusion Middleware Administrator's Guide for Oracle WebCenter Portal*
    - For Oracle SOA Suite: "Deploying SOA Composite Applications" in the *Oracle Fusion Middleware Developer's Guide for Oracle SOA Suite*

9.2.3 Automating the Migration of an Application to Other Environments

You can automate the migration of an application by using WLST or ant scripts. This makes it easier to deploy your application to multiple environments or Managed Servers and to deploy updated versions of the application.

For more information about using scripts to migrate an application to other environments, see:

- For pure Java EE applications: "Using the WebLogic Scripting Tool" in the *Oracle Fusion Middleware Oracle WebLogic Scripting Tool*
- For Oracle ADF: "Deploying Using Scripts and Ant" in the *Oracle Fusion Middleware Administrator's Guide for Oracle Application Development Framework*
- For Oracle WebCenter Portal: "Creating and Provisioning a WebLogic Managed Server Instance" in the *Oracle Fusion Middleware Developer's Guide for Oracle WebCenter Portal*
- For Oracle SOA Suite: "Managing SOA Composite Applications with Scripts" in the *Oracle Fusion Middleware Developer's Guide for Oracle SOA Suite*

9.3 Diagnosing Typical Problems

The following describes some of the typical problems that you may encounter when you deploy an application to a Managed Server:

- Connection information. Ensure that you have correctly configured the connection to the target Managed Server. See Steps 4, 5, and 6 in Section 9.2.2.

- Oracle JDeveloper extensions. Ensure that you have migrated any Oracle JDeveloper extensions. See Table 9–1.
- Data sources. Ensure that you have correctly configured JDBC data sources. See Step 3 in Section 9.2.2.
- Security configuration. Ensure that you have migrated application security, such as credentials, identities, and policies. See Step 8 in Section 9.2.2.

In addition, see the "Troubleshooting Common Deployment Errors" in the *Oracle Fusion Middleware Developer's Guide for Oracle SOA Suite* for information about troubleshooting SOA applications.

10 Deploying Applications

Deployment is the process of packaging application files as an archive file and transferring them to a target application server. This chapter describes how to deploy applications, such as Java EE or SOA applications, to Oracle Fusion Middleware.

It contains the following topics:

10.1 Overview of Deploying Applications

Oracle WebLogic Server provides a Java EE-compliant infrastructure for deploying, undeploying, and redeploying Java EE-compliant applications and modules.

The following topics describe:

10.1.1 What Types of Applications Can You Deploy?

You can deploy the following into Oracle WebLogic Server:

- A complete Java EE application packaged as an Enterprise Archive (EAR) file.
- Standalone modules packaged as Java Archive files (JARs) containing Web services, Enterprise JavaBeans (EJBs), application clients (CARs), or resource adapters (RARs).
- An ADF application. Oracle Application Development Framework (Oracle ADF) is an end-to-end application framework that builds on Java Platform, Enterprise Edition (Java EE) standards, and open-source technologies to simplify and accelerate implementing service-oriented applications.

- An Oracle SOA Suite composite application. A SOA composite application is a single unit of deployment that greatly simplifies the management and lifecycle of SOA applications.
- An Oracle WebCenter Portal application. WebCenter Portal applications differ from traditional Java EE applications in that they support run-time customization, including the application's pages, the portlets contained within these pages, and document libraries.

A Metadata Archive (MAR) is a compressed archive of selected metadata, such as the application-level deployment profile, for an application. A MAR is used to deploy metadata content to the metadata service (MDS) repository. The following application types use a MAR as a container for content that is deployed to the MDS Repository: ADF applications, SOA composite applications, and Oracle WebCenter Portal applications.

Note: If your application uses password indirection in the application-level data source, you cannot use Fusion Middleware Control to deploy the application. The section "Deploying an Application to an EAR File to run on Oracle WebLogic Server" in the Oracle JDeveloper Help describes how to change the settings of the application to be able to deploy the application using Fusion Middleware Control.

You can use Fusion Middleware Control, Oracle WebLogic Server Administration Console, Oracle JDeveloper, or the command line to deploy, undeploy, or redeploy an application. Which method you use depends on the type of application, as described in Table 10–1.

Table 10–1 Tools to Deploy Applications

Type of Application	Tools to Use
Pure Java EE application	Oracle WebLogic Server Administration Console
	Fusion Middleware Control: Deployment Wizard
	Oracle JDeveloper
	WLST command line
ADF application	Fusion Middleware Control: Deployment Wizard
	Oracle JDeveloper
	WLST command line
SOA Composite application	Fusion Middleware Control: SOA Composite Deployment Wizard
	Oracle JDeveloper
	WLST command line
WebCenter Portal application	Fusion Middleware Control: Deployment Wizard
	Oracle JDeveloper
	WLST command line

If your application uses an MDS Repository, you must register the repository with the Oracle WebLogic Server domain before you deploy your application. Applications such as custom Java EE applications developed by your organization and some Oracle Fusion Middleware component applications, such as Oracle B2B and Oracle Web

Services Manager, use an MDS Repository. For information about the MDS Repository and registering the repository, see Section 14.3.

> **Note:** If your application contains an application-level credential store, and you are moving the application from a test to a production environment, you must reassociate the credential store, as described in "Reassociating the Domain Policy Store" in the *Oracle Fusion Middleware Application Security Guide*.

10.1.2 Understanding Deployment, Redeployment, and Undeployment

When you deploy an application, you deploy it to the application server for the first time.

When you redeploy an application, you can:

- Redeploy a new version of the application; the previous version is still available, but the state is set to "Retired."

  This is known as the production redeployment strategy. Oracle WebLogic Server automatically manages client connections so that only new client requests are directed to the new version. Clients already connected to the application during the redeployment continue to use the older version of the application until they complete their work, at which point Oracle WebLogic Server automatically retires the older application.

- Redeploy the same version of the application or redeploy an application that is not assigned a version; the application version you select is replaced with the new deployment.

- Redeploy a previous version of the application; the earlier, retired version is set to "Active" and the later version is set to "Retired."

When you undeploy an application, Oracle WebLogic Server stops the application and removes staged files from target servers. It does not remove the original source files used for deployment.

10.2 Understanding and Managing Data Sources

The following topics describe data sources and how to manage them:

- Understanding Data Sources
- Creating and Managing JDBC Data Sources

10.2.1 Understanding Data Sources

A **data source** is a Java object that application components use to obtain connections to a relational database. Specific connection information, such as the URL or user name and password, are set on a data source object as properties and do not need to be explicitly defined in an application's code. This abstraction allows applications to be built in a portable manner, because the application is not tied to a specific back-end database. The database can change without affecting the application code.

Applications use the Java Naming and Directory Interface (JNDI) API to access a data source object. The application uses a JNDI name that is bound to the data source object. The JNDI name is logical and can be mapped to any data source object. Like data source properties, using JNDI provides a level of abstraction, since the underlying

data source object can change without any changes required in the application code. The end result is the details of accessing a database are transparent to the application.

> **See Also:** *Oracle Fusion Middleware Configuring and Managing JDBC Data Sources for Oracle WebLogic Server* for more information about data sources

When you configure certain Oracle Fusion Middleware components, such as Oracle SOA Suite or Oracle WebCenter Portal, using the Oracle WebLogic Server Configuration Wizard, you specify the data source connection information. If the components use the MDS Repository, the Configuration Wizard prepends "mds-" to the data source name to indicate that the data source is a system data source used by MDS Repository.

> **See Also:** *Oracle Fusion Middleware Creating Domains Using the Configuration Wizard* for information about specifying data sources with the Configuration Wizard

If you are using Oracle Real Application Clusters (Oracle RAC) or Oracle Fusion Middleware Cold Failover Cluster, you must configure one of the following types of data sources:

- Multi data sources

  To use multi data sources, you must use the Oracle WebLogic Server Administration Console. Note that if you create a multi data source and you add an existing MDS data source to it, the data source you added is no longer considered a valid MDS Repository. The repository is not displayed in Fusion Middleware Control or Oracle WebLogic Server Administration Console. For example, the MDS Repository is not listed in the Fusion Middleware Control navigation pane and is not displayed as a choice for a target metadata repository when you deploy an application.

- GridLink data sources

  To use GridLink data sources, you can use the Oracle WebLogic Server Administration Console or Fusion Middleware Control, as described in Section 10.2.2.5.

  > **See Also:** *Oracle Fusion Middleware Configuring and Managing JDBC Data Sources for Oracle WebLogic Server* for more information about configuring multi data sources and GridLink data sources

10.2.2 Creating and Managing JDBC Data Sources

You can create and manage JDBC data sources using the following management tools:

- The Oracle WebLogic Server Administration Console
- The WebLogic Scripting Tool (WLST)
- Fusion Middleware Control

To create an MDS data source manually, you should use Fusion Middleware Control or WLST to set the correct attributes for the data source. The MDS data source is displayed in the navigation pane in Fusion Middleware Control and in the domain structure in the Administration Console. If your application uses an MDS Repository, you must register the repository with the Oracle WebLogic Server domain before you deploy your application. For information about the MDS Repository and registering the repository, see Section 14.3.

Note: When you create the data source, you must use the MDS schema created by the Repository Creation Utility (RCU), not other schemas.

Although it is not recommended, you can also use the Oracle WebLogic Server Administration Console to create a MDS data source. If you do, note the following:

- You must prefix the data source name with "mds-" if you intend it to be used with MDS Repository.
- You must target the data source to the Administration Server and to all Managed Servers to which you are deploying applications that need the data source.
- You must turn off global transactions.

See Also:

- *Oracle Fusion Middleware Configuring and Managing JDBC Data Sources for Oracle WebLogic Server* for information about creating and managing a data source using the Oracle WebLogic Server Administration Console or WLST
- *Oracle Fusion Middleware Configuring and Managing JDBC Data Sources for Oracle WebLogic Server* for more information about configuring multiple data sources

For information creating and managing JDBC data sources with Fusion Middleware Control, see the following topics:

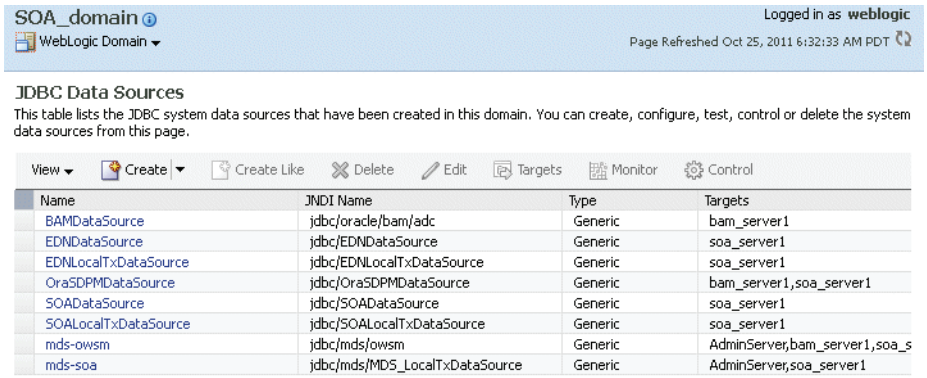
- Creating a JDBC Data Source Using Fusion Middleware Control
- Editing a JDBC Data Source Using Fusion Middleware Control
- Monitoring a JDBC Data Source Using Fusion Middleware Control
- Controlling a JDBC Data Source Using Fusion Middleware Control
- Creating a GridLink Data Source Using Fusion Middleware Control

10.2.2.1 Creating a JDBC Data Source Using Fusion Middleware Control

To create a JDBC data source using Fusion Middleware Control:

1. From the navigation pane, expand the farm, then **WebLogic Domain.**
2. Select the domain to display the Domain home page.
3. From the **WebLogic Domain** menu, choose **JDBC Data Sources.**

   The JDBC Data Sources page is displayed, as shown in the following figure:



4. From **Create,** select **Generic Data Source.**
5. Follow the instructions in the wizard to set the properties of the data source and to target the data source for one or more of the Managed Servers in the domain.

   For help on individual fields and properties, use your mouse to give focus to a field. Fusion Middleware Control displays a popup definition of the field.

   Note that the data source properties you define in Fusion Middleware Control are similar to those you define when creating data sources in the Oracle WebLogic Server Administration Console. As a result, you can also refer to "Creating a JDBC Data Source" in *Oracle Fusion Middleware Configuring and Managing JDBC Data Sources for Oracle WebLogic Server* for more information about the data source properties.

10.2.2.2 Editing a JDBC Data Source Using Fusion Middleware Control

To edit an existing JDBC data source using Fusion Middleware Control:

1. From the navigation pane, expand the farm, then **WebLogic Domain**.
2. Select the domain to display the Domain home page.
3. From the **WebLogic Domain** menu, choose **JDBC Data Sources**.

   The JDBC Data Sources page is displayed.
4. Select the data source that you want to edit.
5. Click **Edit** to display the Edit JDBC Data Source page.
6. Use the tabs on this page to modify the properties of the selected data source.

   For help on individual fields and properties, use your mouse to give focus to a field. Fusion Middleware Control displays a popup definition of the field.

   Note that the data source properties you edit in Fusion Middleware Control are similar to those you edit when editing data sources in the Oracle WebLogic Server Administration Console. As a result, you can also refer to "Creating a JDBC Data Source" in *Oracle Fusion Middleware Configuring and Managing JDBC Data Sources for Oracle WebLogic Server* for more information about the data source properties.

10.2.2.3 Monitoring a JDBC Data Source Using Fusion Middleware Control

To monitor a JDBC data source using Fusion Middleware Control:

1. From the navigation pane, expand the farm, then **WebLogic Domain**.
2. Select the domain to display the Domain home page.
3. From the **WebLogic Domain** menu, choose **JDBC Data Sources**.

The JDBC Data Sources page is displayed.

4. Select the data source that you want to monitor.
5. Click **Monitoring** to display the Monitor JDBC Data Source page.

   This page shows the current instances of the selected data source.

   Note that only data sources that are targeted to a running Managed Server are shown on this page. If a specific data source is not listed on the monitoring page, then edit the data source to be sure it is targeted to a running Managed Server.
6. For each data source instance, review the performance metrics.

   For information on how to get help on individual performance metrics, see "Viewing Performance Metrics Using Fusion Middleware Control" in the *Oracle Fusion Middleware Performance and Tuning Guide*.

10.2.2.4 Controlling a JDBC Data Source Using Fusion Middleware Control

To start, stop, suspend, resume, or clear the statement cache for a JDBC data source using Fusion Middleware Control:

1. From the navigation pane, expand the farm, then **WebLogic Domain**.
2. Select the domain to display the Domain home page.
3. From the **WebLogic Domain** menu, choose **JDBC Data Sources**.

   The JDBC Data Sources page is displayed.
4. Select the data source that you want to edit.
5. Click **Control** to display the Control JDBC Data Source page.

   Note that only data sources that are targeted to a running Managed Server are shown on this page. If a specific data source is not listed on the control page, edit the data source to be sure that it is targeted to a running Managed Server.
6. Click Start, Stop, Force Stop, Resume, Suspend, Force Suspend, Shrink, Reset, or Clear Statement Cache to control or change the state of the selected JDBC data source.

   Note that the commands you select on this page are similar to those available when you are managing data sources in the Oracle WebLogic Server Administration Console. Refer to "Managing WebLogic JDBC Resources" in *Oracle Fusion Middleware Configuring and Managing JDBC Data Sources for Oracle WebLogic Server* for more information about the JDBC data source control options.

10.2.2.5 Creating a GridLink Data Source Using Fusion Middleware Control

A single GridLink data source provides connectivity between WebLogic Server and an Oracle Database service targeted to an Oracle RAC cluster. For detailed information about GridLink data sources, see "Creating a GridLink Data Source" in *Oracle Fusion Middleware Configuring and Managing JDBC Data Sources for Oracle WebLogic Server*.

To create a Grid Link data source using Fusion Middleware Control:

1. From the navigation pane, expand the farm, then **WebLogic Domain**.
2. Select the domain to display the Domain home page.
3. From the **WebLogic Domain** menu, choose **JDBC Data Sources**.

   The JDBC Data Sources page is displayed.
4. From **Create,** select **GridLink Data Source.**

5. Follow the instructions in the wizard to set the properties of the data source and to target the data source for one or more of the Managed Servers in the domain.

   For help on individual fields and properties, use your mouse to give focus to a field. Fusion Middleware Control displays a popup definition of the field.

   Note that the data source properties you define in Fusion Middleware Control are similar to those you define when creating data sources in the Oracle WebLogic Server Administration Console. As a result, you can also refer to "Creating a GridLink Data Source" in *Oracle Fusion Middleware Configuring and Managing JDBC Data Sources for Oracle WebLogic Server* for more information about the data source properties.

10.3 Deploying, Undeploying, and Redeploying Java EE Applications

You can use Fusion Middleware Control, Oracle WebLogic Server Administration Console, Oracle JDeveloper, or the command line to deploy, undeploy, or redeploy a Java EE application. The following topics describe using Fusion Middleware Control and the command line to accomplish these tasks:

- Deploying Java EE Applications
- Undeploying Java EE Applications
- Redeploying Java EE Applications

> **See Also:** *Oracle Fusion Middleware Deploying Applications to Oracle WebLogic Server* for information about deploying using Oracle WebLogic Server Administration Console and for more information about using the WLST command line

10.3.1 Deploying Java EE Applications

You can deploy an application to a WebLogic Server Managed Server instance or a cluster. This section describes how to deploy an application to a Managed Server.

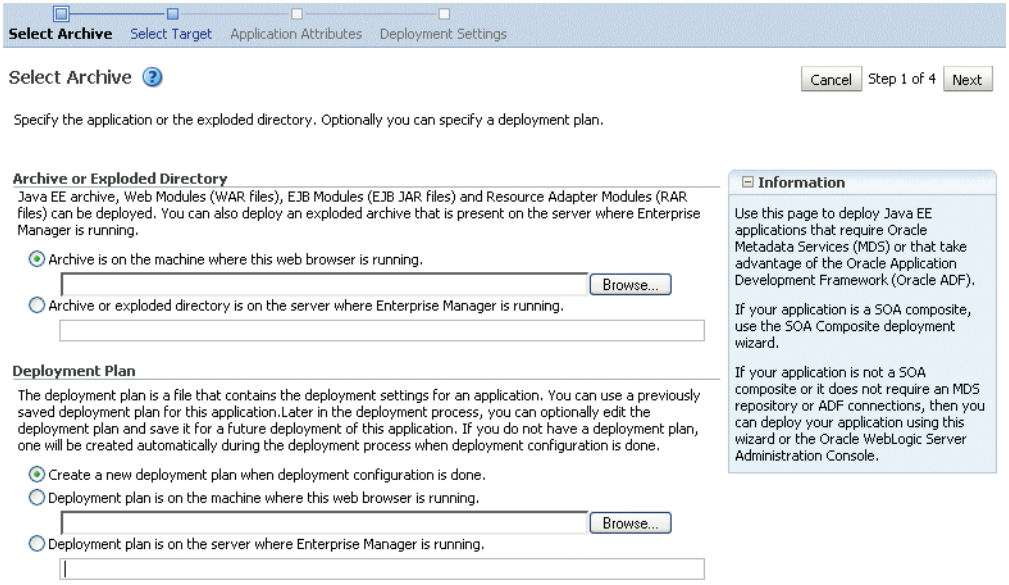
10.3.1.1 Deploying Java EE Applications Using Fusion Middleware Control

To deploy a Java EE application to a Managed Server using Fusion Middleware Control:

1. From the navigation pane, expand the farm, then **WebLogic Domain,** and then the domain.
2. Select the server in which you want to deploy the application.

   The server home page is displayed.
3. From the WebLogic Server menu, choose **Application Deployment,** then **Deploy.**

   The Deployment Wizard, Select Archive page is displayed, as shown in the following figure:



4. In the Archive or Exploded Directory section, you can select one of the following:
   - **Archive is on the machine where this browser is running.** Enter the location of the archive or click **Browse** to find the archive file.
   - **Archive or exploded directory is on the server where Enterprise Manager is running.** Enter the location of the archive or click **Browse** to find the archive file.
5. In the Deployment Plan section, you can select one of the following:
   - **Create a new deployment plan when deployment configuration is done.**
   - **Deployment plan is on the machine where this web browser is running.** If you select this option, enter the path to the plan.
   - **Deployment plan is on the server where Enterprise Manager is running.** If you select this option, enter the path to the plan.
6. Click **Next.**

   The Select Target page is displayed.
7. Select the target to which you want to deploy the application. The Administration Server, Managed Servers, and clusters are listed. You can select a cluster, one or more Managed Servers in the cluster, or a Managed Server that is not in a cluster. Although the Administration Server is shown in the list of targets, you should not deploy an application to it. The Administration Server is intended only for administrative applications such as the Oracle WebLogic Server Administration Console.
8. Click **Next.**

   The Application Attributes page is displayed.
9. In the Application Attributes section, for **Application Name,** enter the application name.
10. In the Context Root of Web Modules section, if the Web module does not have the context root configured in the application.xml file, you can specify the context root for your application. The **context root** is the URI for the Web module. Each Web module or EJB module that contains Web services may have a context root.

11. In the Distribution section, you can select one of the following:
    - **Distribute and start application (servicing all requests)**
    - **Distribute and start application in admin mode (servicing only admin requests)**
    - **Distribute only**
12. You can expand Other Options, which provides the following options:
    - Use the defaults defined by the deployment's targets. Recommended selection.
    - Copy this application onto every target. During deployment, the files are copied automatically to the Managed Servers to which the application is targeted.
13. Click **Next.**

    The Deployment Wizard, Deployment Settings page is displayed.
14. On this page, you can perform common tasks before deploying your application or you can edit the deployment plan or save it to a disk.

    See Section 10.8 for more detailed information about these tasks.

    Depending on the type of application, in the Deployment Tasks section, you can:
    - Configure Web modules: Click **Go to Task** in the Configure Web Modules row. The Configure Web Modules page is displayed. Click **Configure General Properties** to view and edit the general configuration for the Web Module or **Map Resource References** to map the resource references.

      For example, you can change the session invalidation interval or the maximum age of session cookies.
    - Configure EJB modules: Click **Go to Task** in the Configure EJB modules row to set standard EJB deployment descriptor properties. The Configure EJB Modules page is displayed. Click **Configure EJB Properties** to view and edit the general configuration for the EJBs or **Map Resource References** to map the resource preferences.

      For example, you can configure the maximum number of beans in the free pool or the network access point.
    - Configure application security: Click **Go to Task** in the Configure Application Security row. Depending on what type of security is used, different pages are displayed, as described in Section 10.8.
    - Configure persistence: Click **Go to Task** in the Configure Persistence row to configure Java Persistent API (JPA) persistence units.
15. Expand **Deployment Plan.**

    You can edit and save the deployment plan, if you choose. If you edit the deployment plan and change descriptor values, those changes are saved to the deployment plan. In addition, the following configurations are saved to the deployment plan:
    - Application attributes
    - Web module configuration
    - EJB configuration

    Application attributes related to MDS are stored in the file adf-config.xml. Application security attributes are stored in weblogic-application.xml.

Fusion Middleware Control updates the relevant files and repackages the .ear file.

16. Click **Deploy.**

    Fusion Middleware Control displays processing messages.

17. When the deployment is completed, click **Close.**

To deploy an application to multiple servers at the same time, navigate to the domain. Then, from the WebLogic Domain menu, select **Application Deployment,** then **Deploy.** The deployment wizard displays a page where you can select the servers.

To deploy an application to a cluster, select the cluster. Then, from the Cluster menu, select **Application Deployment,** then **Deploy.**

10.3.1.2 Deploying Java EE Applications Using WLST

You can deploy an application using the WLST command line. To deploy a Java EE application when WLST is connected to the Administration Server, you use the WLST command `deploy`, using the following format:

```
deploy(app_name, path [,targets] [,stageMode] [,planPath] [,options])
```

You must invoke the deploy command on the computer that hosts the Administration Server.

For example, to deploy the application mainWebApp:

```
deploy("myApp","/scratch/applications/wlserver_
10.3/samples/server/examples/build/mainWebApp")
```

You can also deploy the application using the weblogic.deployer, as shown in the following example:

```
java weblogic.Deployer -adminurl http://localhost:7001
   -user username -password password -deploy
   -name myApp c:\localfiles\mainWebApp
   -plan c:\localfiles\productionEnvPlan.xml
```

See Also:

- "Deployment Tools" in *Oracle Fusion Middleware Deploying Applications to Oracle WebLogic Server* for more information about using WLST to deploy applications
- *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*

10.3.2 Undeploying Java EE Applications

You can undeploy an application or a specific version of an application from a WebLogic Server Managed Server instance or a cluster. This section describes how to undeploy an application from a Managed Server. If an application has been deployed to multiple servers, when you undeploy it using Fusion Middleware Control, the application is undeployed from all the servers.

10.3.2.1 Undeploying Java EE Applications Using Fusion Middleware Control

To undeploy a Java EE application from a Managed Server using Fusion Middleware Control:

1. From the navigation pane, expand **Application Deployments.**
2. Select the application to undeploy.

The application home page is displayed.

3. From the Application Deployment menu, choose **Application Deployment,** then **Undeploy.**

   The confirmation page is displayed.

4. Click **Undeploy.**

   Processing messages are displayed.

5. When the operation completes, click **Close.**

Alternatively, you can navigate to the domain, Managed Server, or cluster. Then, from the target's menu, choose **Application Deployment,** then **Undeploy.** In the Select Application page, select the application you want to undeploy.

10.3.2.2 Undeploying Java EE Applications Using WLST

You can undeploy an application using the WLST command line. To undeploy a Java EE application when WLST is connected to the Administration Server, you use the WLST command `undeploy`, using the following format:

```
undeploy(app_name, path [,targets] [,options])
```

You must invoke the undeploy command on the computer that hosts the Administration Server.

For example, to undeploy the application businessApp from all target servers and specify that WLST wait 60,000 ms for the process to complete:

```
wls:/mydomain/serverConfig> undeploy('businessApp', timeout=60000)
```

10.3.3 Redeploying Java EE Applications

You can redeploy a new version of an updated application, redeploy the same version, or redeploy a non-versioned application. You can redeploy an application to a cluster or a Managed Server. This section describes how to redeploy an application to a Managed Server.

10.3.3.1 Redeploying Java EE Applications Using Fusion Middleware Control

To redeploy a Java EE application to a Managed Server using Fusion Middleware Control:

1. From the navigation pane, expand the farm, then **Application Deployments.**
2. Select the application.

   The application home page is displayed.

3. From the Application Deployment menu, choose **Application Deployment,** and then **Redeploy.**

   The Select Application page is displayed.

4. Click **Next.**
5. In the Archive or Exploded Directory section, you can select one of the following:
   - **Archive is on the machine where this browser is running.** Enter the location of the archive or click **Browse** to find the archive file.

- **Archive or exploded directory is on the server where Enterprise Manager is running.** Enter the location of the archive or click **Browse** to find the archive file.

6. In the Deployment Plan section, you can select one of the following:
   - **Create a new deployment plan when deployment configuration is done.**
   - **Deployment plan is on the machine where this web browser is running.** Enter the path to the plan or click **Browse** to find the plan file.
   - **Deployment plan is on the server where Enterprise Manager is running.** Enter the path to the plan or click **Browse** to find the plan file.
7. Click **Next.**

   The Application Attributes page is displayed.
8. Click **Next.**

   The Deployment Wizard, Deployment Settings page is displayed.
9. On this page, you can perform common tasks before deploying your application or you can edit the deployment plan or save it to a disk. Depending on the type of application, in the Deployment Tasks section, you can:
   - Configure Web modules
   - Configure application security
   - Configure EJB modules
   - Configure persistence

   See Section 10.8 for detailed information about these tasks.
10. Expand **Deployment Plan.**

    You can edit and save the deployment plan, if you choose. If you edit the deployment plan and change descriptor values, those changes are saved to the deployment plan. In addition, the following configurations are saved to the deployment plan:
    - Application attributes
    - Web module configuration
    - EJB configuration

    Application attributes related to MDS are stored in the file adf-config.xml. Application security attributes are stored in weblogic-application.xml.

    Fusion Middleware Control updates the relevant files and repackages the .ear file.
11. Click **Redeploy.**

    Processing messages are displayed.
12. When the operation completes, click **Close.**

To redeploy an application to a cluster, select the cluster. Then, from the target's menu, select **Application Deployment,** then **Redeploy.**

10.3.3.2 Redeploying Java EE Applications Using WLST

You can redeploy an application using the WLST command line. To redeploy a Java EE application when WLST is connected to the Administration Server, you use the WLST command `redeploy`, using the following format:

```
redeploy(app_name [,planpath] [,options])
```

You must invoke the redeploy command on the computer that hosts the Administration Server.

For example, to redeploy the application businessApp from all target servers:

```
redeploy('businessApp')
```

10.4 Deploying, Undeploying, and Redeploying Oracle ADF Applications

Oracle ADF is an end-to-end application framework that builds on Java Platform, Enterprise Edition (Java EE) standards and open-source technologies to simplify and accelerate implementing service-oriented applications.

You can use Fusion Middleware Control, Oracle WebLogic Server Administration Console, Oracle JDeveloper, or the command line to deploy, undeploy, or redeploy an Oracle ADF application. The following topics describe using Fusion Middleware Control, the Administration Console, and the command line to accomplish these tasks:

- Deploying Oracle ADF Applications
- Undeploying Oracle ADF Applications
- Redeploying Oracle ADF Applications

> **See Also:** *Oracle Fusion Middleware Fusion Developer's Guide for Oracle Application Development Framework* for information on developing ADF applications and for deploying them using Oracle JDeveloper

10.4.1 Deploying Oracle ADF Applications

You can deploy an application to a WebLogic Server Managed Server instance or a cluster. This section describes how to deploy an application to a Managed Server. This example assumes that you have created an .ear file containing the ADF application.

10.4.1.1 Deploying ADF Applications Using Fusion Middleware Control

To deploy an Oracle ADF application using Fusion Middleware Control:

1. From the navigation pane, expand the farm, then **WebLogic Domain,** and then the domain.
2. Select the server in which you want to deploy the application.

   The server home page is displayed.
3. From the WebLogic Server menu, choose **Application Deployment,** then **Deploy.**

   The Deployment Wizard, Select Archive page is displayed.
4. In the Archive or Exploded Directory section, you can select one of the following:
   - **Archive is on the machine where this browser is running.** Enter the location of the archive or click **Browse** to find the archive file.
   - **Archive or exploded directory is on the server where Enterprise Manager is running.** Enter the location of the archive or click **Browse** to find the archive file.
5. In the Deployment Plan section, you can select one of the following:
   - **Create a new deployment plan when deployment configuration is done.**

- **Deployment plan is on the machine where this web browser is running.** Enter the path to the plan.
- **Deployment plan is on the server where Enterprise Manager is running.** Enter the path to the plan.

6. Click **Next.**

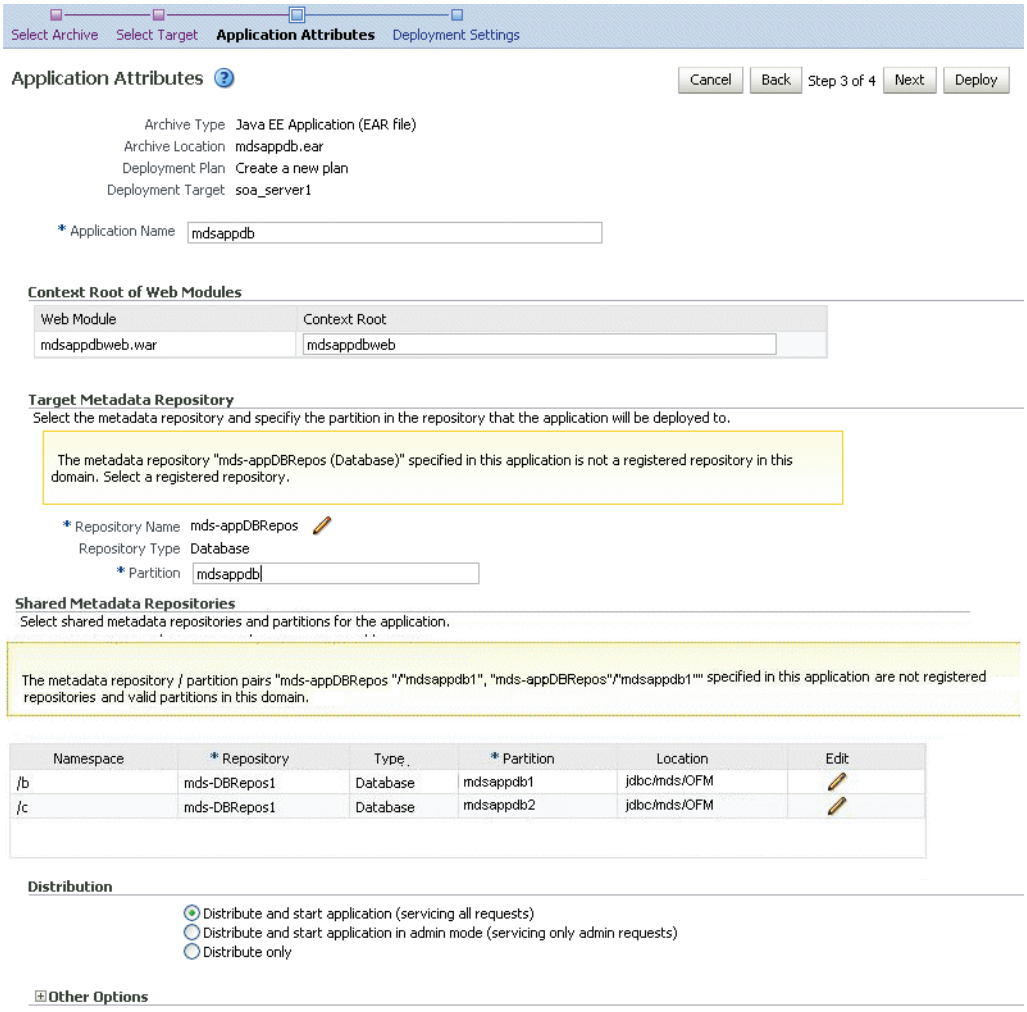
   The Select Target page is displayed.

7. Select the target to which you want to deploy the application.

   You can select a cluster, one or more Managed Servers in the cluster, or a Managed Server that is not in a cluster.

8. Click **Next.**

   The Application Attributes page is displayed, as shown in the following figure:



9. In the Application Attributes section, for **Application Name,** enter the application name.

10. In the Context Root of Web Modules section, if the Web module does not have the context root configured in the application.xml file, you can specify the context root for your application. The context root is the URI for the Web module. Each Web module or EJB module that contains Web services may have a context root.

11. In the Target Metadata Repository section, you can choose the repository and partition for this application. If the partition name is not specified in the adf-config.xml file, the application name plus the version is used as the default partition name. This ensures that the partition used is unique in the domain so that the metadata for different applications are not accidentally imported into the same repository partition and overwrite each other. Typically, each application's metadata is deployed to its own partition.

    - To change the repository, click the icon next to the **Repository Name.** In the Metadata Repositories dialog box, select the repository and click **OK.**
    - To change the partition, enter the partition name in **Partition Name.** Oracle recommends that you create a new partition for each application. If you enter a name of a partition that does not exist, the partition is created.

    The adf-config.xml file in the .ear file is updated with the new information.

    If the partition or repository specified in the application is not valid in the domain, Fusion Middleware Control displays a message.

12. If the application's adf-config.xml file archive contains MDS configuration for an MDS shared repository, the Shared Metadata Repository section is displayed. In this section, you can choose the repository and partition for this application. If the partition or repository specified in the application is not valid in the domain, Fusion Middleware Control displays a message.

    If you change the repository or partition, the adf-config.xml file in the .ear file is updated with the new information.

13. In the Distribution section, you can select one of the following:

    - **Distribute and start application (servicing all requests)**
    - **Distribute and start application in admin mode (servicing only admin requests)**
    - **Distribute only**

14. Click **Next.**

    The Deployment Wizard, Deployment Settings page is displayed.

15. On this page, you can perform common tasks before deploying your application or you can edit the deployment plan or save it to a disk. Depending on the type of application, in the Deployment Tasks section, you can:

    - Configure Web modules: Click **Go to Task** in the Configure Web Modules row. The Configure Web Modules page is displayed. Click **Configure General Properties** to view and edit the general configuration for the Web Module or **Map Resource References** to map the resource references.

      For example, you can change the session invalidation interval or the maximum age of session cookies.

    - Configure EJB modules: Click **Go to Task** in the Configure EJB modules row to set standard EJB deployment descriptor properties. The Configure EJB Modules page is displayed. Click **Configure EJB Properties** to view and edit the general configuration for the EJBs or **Map Resource References** to map the resource preferences.

      For example, you can configure the maximum number of beans in the free pool or the network access point.

- Configure application security: Click **Go to Task** in the Configure Application Security row. Depending on what type of security is used, different pages are displayed, as described in Section 10.8.
- Configure persistence: Click **Go to Task** in the Configure Persistence row to configure Java Persistent API (JPA) persistence units.
- Configure ADF Connections: To modify the ADF connections, click **Go to Task** in the Configure ADF Connections row. The Configure ADF Connections page is displayed, showing the current connection information. To modify a connection type, click the **Edit** icon for a particular row. For example, you can modify the connection information for an external application. For more information about ADF connections, see *Oracle Fusion Middleware Fusion Developer's Guide for Oracle Application Development Framework*.

For more information about these options, see Section 10.8.

16. Expand **Deployment Plan.**

    You can edit and save the deployment plan, if you choose.

17. Click **Deploy.**

    Fusion Middleware Control displays processing messages.

18. When the deployment is completed, click **Close.**

10.4.1.2 Deploying ADF Applications Using WLST or the Administration Console

You can deploy an ADF application using the WLST command line or the Oracle WebLogic Server Administration Console.

Take the following steps:

1. If your application uses an MDS Repository, you must configure the application archive (.ear) file before you deploy your application. You must provide the repository information for the deploy target repository and any shared metadata repositories using the WLST getMDSArchiveConfig command. The repository specified must already be registered with the domain before deploying the application. The following example show how to use this command to get the MDSArchiveConfig and call the setAppMetadataRepository method to set the deploy target repository. Otherwise, your application will fail to start.

   ```
   wls:/offline> archive = getMDSArchiveConfig(fromLocation='/tmp/App1.ear')
   wls:/offline> archive.setAppMetadataRepository(repository='AppRepos1',
                 partition='partition1', type='DB', jndi='mds-jndi1')
   ```

   The operation places the changes in the MDS configuration portion of the adf-config.xml file in the archive file.

2. Save the changes to the original .ear file, using the following command:

   ```
   wls:/offline> archive.save()
   ```

3. Deploy the application.

   To deploy an application when WLST is connected to the Administration Server, you use the WLST command `deploy`, using the following format:

   ```
   deploy(app_name, path [,targets] [,stageMode] [,planPath] [,options])
   ```

   You must invoke the deploy command on the computer that hosts the Administration Server.

For example, to deploy the application myApp:

```
deploy("myApp","/scratch/applications/myApp", targets='myserver',
timeout=120000))
```

See Also:

- "Deployment Tools" in *Oracle Fusion Middleware Deploying Applications to Oracle WebLogic Server* for more information about using WLST to deploy applications
- *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*

To deploy the application using the Oracle WebLogic Server Administration Console:

1. If you have not already done so, in the Change Center of the Administration Console, click **Lock & Edit**.
2. In the left pane of the Administration Console, select **Deployments.**
3. In the right pane, click **Install.**

See Also: The Help in the Oracle WebLogic Server Administration Console

10.4.2 Undeploying Oracle ADF Applications

To undeploy an Oracle ADF application using Fusion Middleware Control:

1. From the navigation pane, expand **Application Deployments,** then the application to undeploy.

   The application home page is displayed.
2. From the Application Deployment menu, choose **Application Deployment,** then **Undeploy.**

   The confirmation page is displayed.
3. Click **Undeploy.**

   Processing messages are displayed.
4. When the operation completes, click **Close.**

Alternatively, you can navigate to the domain, Managed Server, or cluster. Then, from the target's menu, choose **Application Deployment,** then **Undeploy.** In the Select Application page, select the application you want to undeploy.

Note that when you undeploy an application, documents stored in the MDS partition are not deleted.

10.4.3 Redeploying Oracle ADF Applications

When you redeploy an application, if the application contains a Metadata Archive (MAR), the contents of the MAR is imported to the application's metadata repository only if the MAR is changed. If the MAR is unchanged from previous deployment of the application, it is ignored.

To redeploy an Oracle ADF application using Fusion Middleware Control:

1. From the navigation pane, expand the farm, then **Application Deployments.**
2. Select the application.

The application home page is displayed.

3. From the Application Deployment menu, choose **Application Deployment,** and then **Redeploy.**

   The Select Application page is displayed.

4. Click **Next.**

   The Select Archive page is displayed.

5. In the Archive or Exploded Directory section, you can select one of the following:

   - **Archive is on the machine where this browser is running.** Enter the location of the archive or click **Browse** to find the archive file.
   - **Archive or exploded directory is on the server where Enterprise Manager is running.** Enter the location of the archive or click **Browse** to find the archive file.

6. In the Deployment Plan section, you can select one of the following:

   - **Create a new deployment plan when deployment configuration is done.**
   - **Deployment plan is on the machine where this web browser is running.** Enter the path to the plan.
   - **Deployment plan is on the server where Enterprise Manager is running.** Enter the path to the plan.

7. Click **Next.**

   The Application Attributes page is displayed.

8. In the Application Attributes section, for **Application Name,** enter the application name.

9. In the Context Root of Web Modules section, if the Web module does not have the context root configured in the application.xml file, you can specify the context root for your application. The context root is the URI for the Web module. Each Web module or EJB module that contains Web services may have a context root.

10. The Target Metadata Repository section is displayed. In this section, you can choose the repository and partition for this application:

    - To change the repository, click the icon next to the **Repository Name.** In the Metadata Repositories dialog box, select the repository and click **OK.**
    - To change the partition, enter the partition name in **Partition Name.** Oracle recommends that you create a new partition for each application. If you enter a name of a partition that does not exist, the partition is created.

11. If the application's adf-config.xml file archive contains MDS configuration for an MDS shared repository, the Shared Metadata Repository section is displayed. In this section, you can choose the repository and partition for this application.

12. Click **Next.**

    The Deployment Settings page is displayed.

13. On this page, you can perform common tasks before deploying your application or you can edit the deployment plan or save it to a disk. In the Deployment Tasks section, you can:

    - Configure Web modules
    - Configure application security

- Configure persistence

See Section 10.8 for detailed information about these options.

14. Expand **Deployment Plan.**

    You can edit and save the deployment plan, if you choose.

15. Click **Deploy.**

    Fusion Middleware Control displays processing messages.

16. When the deployment is completed, click **Close.**

17. In the Confirmation page, click **Redeploy.**

10.5 Deploying, Undeploying, and Redeploying SOA Composite Applications

SOA composite applications consist of the following:

- Service components such as Oracle Mediator for routing, BPEL processes for orchestration, human tasks for workflow approvals, business rules for designing business decisions, and complex event processing for queries of event streams
- Binding components (services and references) for connecting SOA composite applications to external services, applications, and technologies

These components are assembled together into a SOA composite application. This application is a single unit of deployment that greatly simplifies the management and lifecycle of SOA applications.

You can use Fusion Middleware Control, Oracle JDeveloper, or the command line to deploy, undeploy, or redeploy a SOA application. The following topics describe using Fusion Middleware Control to accomplish these tasks:

- Deploying SOA Composite Applications
- Undeploying SOA Composite Applications
- Redeploying SOA Composite Applications

> **See Also:** *Oracle Fusion Middleware Administrator's Guide for Oracle SOA Suite and Oracle Business Process Management Suite*

10.5.1 Deploying SOA Composite Applications

When you deploy a SOA composite application, the deployment extracts and activates the composite application in the SOA Infrastructure.

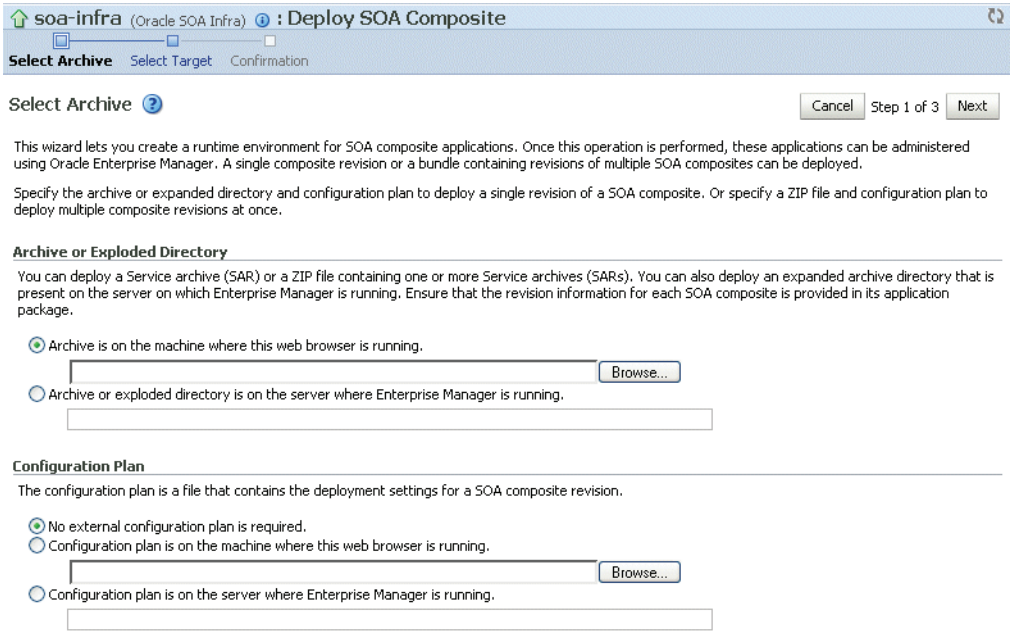
You can deploy SOA composite applications from Fusion Middleware Control with the Deploy SOA Composite wizard:

1. From the navigation pane, expand the farm, then **SOA,** and then select **soa-infra.**

   The SOA Infrastructure home page is displayed.

2. From the SOA Infrastructure menu, choose **SOA Deployment,** then **Deploy.**

   The Deployment Wizard, Select Archive page is displayed, as shown in the following figure:



3. In the Archive or Exploded Directory section, specify the archive of the SOA composite application to deploy. The archive contains the project files of the application to be deployed (for example, **HelloWorld_rev1.0.jar** for a single archive or **OrderBooking_rev1.0.zip** for multiple archives).
4. In the Configuration Plan section, optionally specify the configuration plan to include with the archive. The configuration plan enables you to define the URL and property values to use in different environments. During process deployment, the configuration plan is used to search the SOA project for values that must be replaced to adapt the project to the next target environment.
5. Click **Next.**

   The Select Target page appears.
6. In the SOA Partition section, select the partition into which to deploy this SOA composite application. Partitions enable you to logically group SOA composite applications into separate sections. Note that even if there is only one partition available, you must explicitly select it. Once deployed, a composite cannot be transferred to a different partition.
7. Click **Next.**

   The Confirmation page appears.
8. Review your selections.
9. Select whether or not to deploy the SOA composite application as the default revision. The default revision is instantiated when a new request comes in.
10. Click **Deploy.**

    Processing messages are displayed.
11. When deployment has completed, close the confirmation box.

> **See Also:** "Deploying Applications" in the *Oracle Fusion Middleware Administrator's Guide for Oracle SOA Suite and Oracle Business Process Management Suite* for complete information about deploying SOA Composite applications

10.5.2 Undeploying SOA Composite Applications

You can undeploy SOA composite applications from Fusion Middleware Control with the Undeploy SOA Composite wizard:

1. From the navigation pane, expand the farm, then **SOA,** and then select **soa-infra.**

   The SOA Infrastructure home page is displayed.

2. From the SOA Infrastructure menu, choose **SOA Deployment,** then **Undeploy.**
3. Select the composite to undeploy and click **Next.**
4. Review your selections. If you are satisfied, click **Undeploy.**

   Processing messages are displayed.

5. When undeployment has completed, close the confirmation window.

> **See Also:** "Undeploying Applications" in the *Oracle Fusion Middleware Administrator's Guide for Oracle SOA Suite and Oracle Business Process Management Suite* for complete information about undeploying SOA Composite applications

10.5.3 Redeploying SOA Composite Applications

You can redeploy SOA composite applications from Fusion Middleware Control with the Redeploy SOA Composite wizard:

1. From the navigation pane, expand the farm, then **SOA,** and then select **soa-infra.**

   The SOA Infrastructure home page is displayed.

2. From the SOA Infrastructure menu, choose **SOA Deployment,** then **Redeploy.**

   The Select Composite page is displayed.

3. Select the composite that you want to redeploy.
4. Click **Next.**

   The Select Archive page appears.

5. In the Archive or Exploded Directory section, select the location of the SOA composite application revision you want to redeploy.
6. In the Configuration Plan section, optionally specify the configuration plan to include with the archive.
7. Click **Next.**

   The Confirmation page appears.

8. Select whether or not to redeploy the SOA composite application as the default revision.
9. Click **Redeploy.**

   Processing messages are displayed.

10. When redeployment has completed, click **Close**.

> **See Also:** "Redeploying Applications" in the *Oracle Fusion Middleware Administrator's Guide for Oracle SOA Suite and Oracle Business Process Management Suite* for complete information about redeploying SOA Composite applications

10.6 Deploying, Undeploying, and Redeploying WebCenter Portal Applications

Oracle WebCenter Portal applications differ from traditional Java EE applications in that they support run-time customization, such as the application's pages, the portlets contained within these pages, and the document libraries. Customizations are stored as follows:

- WebCenter Portal application customizations are stored in Oracle Metadata Services (MDS), which is installed in a database.
- Portlet Producer customizations (or preferences) are usually stored in a database preference store.

You can use Fusion Middleware Control, Oracle JDeveloper, or the command line to deploy, undeploy, or redeploy a WebCenter Portal application. The following topics describe using Fusion Middleware Control to accomplish these tasks:

- Deploying WebCenter Portal Applications
- Undeploying WebCenter Portal Applications
- Redeploying WebCenter Portal Applications

> **See Also:** *Oracle Fusion Middleware Administrator's Guide for Oracle WebCenter Portal*

10.6.1 Deploying WebCenter Portal Applications

To deploy your application to a Managed Server that resides outside JDeveloper, you must first create an application deployment plan. In Oracle JDeveloper, first create a project-level deployment profile and then an application-level deployment profile. The project-level deployment profile is packaged as a Web Application Archive (WAR) file. The application-level deployment profile is packaged as a Metadata Archive (MAR). A single MAR can contain metadata content of multiple projects. MAR files are used to deploy metadata content to the MDS Repository. For information about creating deployment plans with Oracle JDeveloper, see the *Oracle Fusion Middleware Developer's Guide for Oracle WebCenter Portal*.

For complete information about deploying Oracle WebCenter Portal applications, see "Deploying WebCenter Portal Framework Applications" in the *Oracle Fusion Middleware Administrator's Guide for Oracle WebCenter Portal.*

To deploy an Oracle WebCenter Portal application to a Managed Server using Fusion Middleware Control:

1. From the navigation pane, expand the farm, then **WebLogic Domain.**
2. Select the domain in which you want to deploy the application.

   The server home page is displayed.
3. From the WebLogic Domain menu, select **Application Deployment,** then **Deploy.**

   The Deployment Wizard, Select Archive page is displayed.
4. In the Archive or Exploded Directory section, you can select one of the following:
   - **Archive is on the machine where this browser is running.** Enter the location of the archive or click **Browse** to find the archive file.

- **Archive or exploded directory is on the server where Enterprise Manager is running.** Enter the location of the archive or click **Browse** to find the archive file.

5. In the Deployment Plan section, you can select one of the following:
   - **Create a new deployment plan when deployment configuration is done.**
   - **Deployment plan is on the machine where this web browser is running.** Enter the path to the plan.
   - **Deployment plan is on the server where Enterprise Manager is running.** Enter the path to the plan.
6. Click **Next.**

   The Select Target page is displayed.
7. Select the target to which you want to deploy the application.

   You can select a cluster, one or more Managed Servers in the cluster, or a Managed Server that is not in a cluster.
8. Click **Next.**

   The Application Attributes page is displayed.
9. In the Application Attributes section, for **Application Name,** enter the application name.
10. In the Context Root of Web Modules section, specify the context root for your application if you have not specified it in application.xml. The context root is the URI for the Web module. Each Web module or EJB module that contains Web services may have a context root.
11. In the Target Metadata Repository section, you can choose the repository and partition for this application. If the partition or repository specified in the application is not valid in the domain, Fusion Middleware Control displays a message.
    - To change the repository, click the icon next to the **Repository Name.** In the Metadata Repositories dialog box, select the repository and click **OK.**
    - To change the partition, enter the partition name in **Partition Name.** Oracle recommends that you create a new partition for each application. If you enter a name of a partition that does not exist, the partition is created.

      Each application must have a unique partition in the repository.
12. Click **Next.**

    The Deployment Wizard, Deployment Settings page is displayed.
13. On this page, you can perform common tasks before deploying your application or you can edit the deployment plan or save it to a disk. Depending on the type of application, in the Deployment Tasks section, you can:
    - Configure Web modules: Click **Go to Task** in the Configure Web Modules row. The Configure Web Modules page is displayed. Click **Configure General Properties** to view and edit the general configuration for the Web Module or **Map Resource References** to map the resource references.

      For example, you can change the session invalidation interval or the maximum age of session cookies.

- Configure EJB modules: Click **Go to Task** in the Configure EJB modules row to set standard EJB deployment descriptor properties. The Configure EJB Modules page is displayed. Click **Configure EJB Properties** to view and edit the general configuration for the EJBs or **Map Resource References** to map the resource preferences.

  For example, you can configure the maximum number of beans in the free pool or the network access point.
- Configure application security: Click **Go to Task** in the Configure Application Security row. Depending on what type of security is used, different pages are displayed, as described in Section 10.8.
- Configure persistence: Click **Go to Task** in the Configure Persistence row to configure Java Persistent API (JPA) persistence units.
- Configure ADF Connections: To modify the ADF connections, click **Go to Task** in the Configure ADF Connections row. The Configure ADF Connections page is displayed, showing the current connection information. To modify a connection type, click the **Edit** icon for a particular row.

See Section 10.8 for more detailed information about these options.

14. Expand **Deployment Plan.**

    You can edit and save the deployment plan, if you choose.
15. Click **Deploy.**

    Fusion Middleware Control displays processing messages.
16. When the deployment is completed, click **Close.**

10.6.2 Undeploying WebCenter Portal Applications

To undeploy a WebCenter Portal application:

1. From the navigation pane, expand **Application Deployments,** then the application to undeploy.

   The application home page is displayed.
2. From the Application Deployment menu, select **Application Deployment,** then **Undeploy.**

   The confirmation page is displayed.
3. Click **Undeploy.**

   Processing messages are displayed.
4. When the operation completes, click **Close.**

Alternatively, you can navigate to the domain, Managed Server, or cluster. Then, from the target's menu, choose **Application Deployment,** then **Undeploy.** In the Select Application page, select the application you want to undeploy.

10.6.3 Redeploying WebCenter Portal Applications

To redeploy a WebCenter Portal application:

1. From the navigation pane, expand the farm, then **WebLogic Domain.** and then the domain.
2. Select the server in which you want to redeploy the application.

The server home page is displayed.

3. From the WebLogic Server menu, select **Application Deployment,** then **Redeploy.**

   The Select Application page is displayed. You can only redeploy applications that are versioned. If the application is not versioned, you must undeploy, then redeploy.

4. Select the application to redeploy.

5. Click **Next.**

   The Select Archive page is displayed.

6. In the Archive or Exploded Directory section, you can select one of the following:

   - **Archive is on the machine where this browser is running.** Enter the location of the archive or click **Browse** to find the archive file.
   - **Archive or exploded directory is on the server where Enterprise Manager is running.** Enter the location of the archive or click **Browse** to find the archive file.

7. In the Deployment Plan section, you can select one of the following:

   - **Create a new deployment plan when deployment configuration is done**
   - **Deployment plan is on the machine where this web browser is running.** Enter the path to the plan.
   - **Deployment plan is on the server where Enterprise Manager is running.** Enter the path to the plan.

8. Click **Next.**

   The Application Attributes page is displayed.

9. In the Application Attributes section, for **Application Name,** enter the application name.

10. In the Context Root of Web Modules section, specify the context root for your application if you have not specified it in application.xml. The context root is the URI for the Web module. Each Web module or EJB module that contains Web services may have a context root.

11. In the Target Metadata Repository section, select the MDS Repository and for **Partition Name,** enter a partition name. Be careful to use the same repository connection and partition name that you used when you originally deployed the application. If you do not, all customizations are lost.

12. Click **Next.**

    The Deployment Settings page is displayed.

13. On this page, you can perform common tasks before deploying your application or you can edit the deployment plan or save it to a disk. In the Deployment Tasks section, you can:

    - Configure Web modules
    - Configure application security
    - Configure persistence

    See Section 10.8 for detailed information about these options.

14. Expand **Deployment Plan.**

You can edit and save the deployment plan, if you choose.

15. Click **Redeploy.**

    Fusion Middleware Control displays processing messages.

16. When the deployment is completed, click **Close.**

10.7 Managing Deployment Plans

A **deployment plan** is a client-side aggregation of all the configuration data needed to deploy an archive into Oracle WebLogic Server. A deployment plan allows you to easily deploy or redeploy an application using a saved set of configuration settings.

A new deployment plan is created by default if you do not apply an existing deployment plan to an application at the time of deployment, as described in Section 10.3.1. Once created, you can save a deployment plan as a file and reuse it for redeploying the application or for deploying other applications.

However, if you change the configuration of an application after it is deployed (for example, if you modify the MDS configuration of an application), then any existing deployment plans you saved no longer represent the configuration settings of the deployed application.

In such a situation, you can fetch a new deployment plan that more closely represents the configuration of the deployed application.

To fetch the deployment plan of an application that is currently deployed:

1. From the navigation pane, expand the farm, then **WebLogic Domain**.
2. Select the domain.

   The WebLogic Domain page is displayed.

3. From the **WebLogic Domain** menu, choose **Application Deployment,** then **Fetch Deployment Plan**.

   The Fetch Deployment Plan page is displayed.

4. Select an application from the list of currently deployed applications.
5. Select a location where you want to save the deployment plan, and click **Fetch**.

   You can save the plan to the computer where the Web browser is running or to the computer where Fusion Middleware Control is running.

6. In the resulting dialog box, specify a directory location for the saved deployment plan.

   You can now use this deployment plan to later deploy or redeploy the application using the configuration currently in use by the application.

Alternatively, you can edit a deployment plan on the Deployment Settings page of the Application Deployment wizard.

10.8 About the Common Deployment Tasks in Fusion Middleware Control

When you deploy an application using Fusion Middleware Control, you can use the Deployment Settings page of the Deployment wizard to perform specific deployment configuration tasks before the application is deployed.

The following describes the deployment tasks that can appear on the Deployment Settings page, depending on the type of application you are deploying.

Configure Web modules

This deployment task is available when you are deploying any application that includes a Web module. In most cases, this means the application contains a Web application deployment descriptor (web.xml or weblogic.xml); however, a Web module can also be identified by annotations in the Java code of the application.

You can use this deployment task to set standard Web application deployment descriptor properties, such as:

- Session validation interval
- Maximum age of session cookies

Configure EJBs

This deployment task is available for any application that includes an EJB module. In most cases, this means the application contains an EJB deployment descriptor (ejb-jar.xml or weblogic-ejb-jar.xml); however, an EJB module can also be identified by annotations in the Java code of the application.

You can use this deployment task to set standard EJB deployment descriptor properties, such as:

- The maximum number of beans in the free pool
- The EJB network access point

Configure Application Security

This deployment task is available for all application types. However, the options available when you select this task vary depending on the existence of the following files in the application:

- jazn-data.xml

  If the jazn-data.xml file exists in the application, then you can:

  - Append, overwrite, or ignore policy migration.
    - If you are deploying the application for the first time, then select **Append**.
    - If the application was previously deployed and the application authorization policy exists, then select **Append**, or select **Ignore** to keep the application authorization policy.
    - To overwrite the previous policy, then select **Overwrite**.
  - Specify the Application stripe ID, if the stripe ID is inconsistent with the one defined in the migration options.
  - Specify that policies are removed when the application is undeployed.

- cwallet.sso

  If an cwallet.sso file exists in the application, then you can set additional application credential migration options.

If the application contains both files, the page displays both sections.

For more information about the settings available when you select the Configure Application Security deployment task, see "Deploying Java EE and Oracle ADF Applications with Fusion Middleware Control" in the *Oracle Fusion Middleware Application Security Guide*.

If neither of these files exists in the application, then you can use this task to determine how user roles and policies will be defined when the application is deployed. For

example, you can choose to use only the roles and policies defined in the deployment descriptors, or you can choose to use only the roles and policies defined on the server. The Configure Application Security page displays the following options:

- **Deployment Descriptors Only:** Use only roles and policies that are defined in the deployment descriptors.
- **Custom Roles:** Use roles that are defined in the Administration Console; use policies that are defined in the deployment descriptor.
- **Custom Roles and Policies:** Use only roles and policies that are defined in the Administration Console.
- **Advanced:** Use a custom model that you have configured on the realm's configuration page.

Configure persistence

This deployment task is available for applications that contain one or more persistence.xml files. Using this task, you can configure the Java Persistent API (JPA) persistence units for the application.

You can view details about each persistence unit and define a Java Transaction API (JTA) data source or non-JTA data source for each persistence unit.

Configuring the data sources for persistence units can be useful for applications that take advantage of Oracle TopLink. For more information, refer to the *Oracle Fusion Middleware Developer's Guide for Oracle TopLink*.

For more information about how persistence units and the persistence.xml file can be used in Java EE applications, refer to the definition of Persistence Units in the Java EE 5 Tutorial at the following Web site:

`http://download.oracle.com/javaee/5/tutorial/doc/bnbqw.html#bnbrj`

Configure ADF connections

This deployment task is available for applications that use ADF connections. You can modify the connection information for an external application. For more information about ADF connections, see the *Oracle Fusion Middleware Fusion Developer's Guide for Oracle Application Development Framework*.

10.9 Changing MDS Configuration Attributes for Deployed Applications

If your application uses an MDS Repository, you can modify configuration attributes after the application is deployed. To view or modify the attributes, you can use the System MBean Browser or WLST.

Note: Changes to the configuration persist in MDS as customizations. Because these persist as customizations:

- Any changes made to the configuration are retained across application deployments. For example, assume that an application has an ExternalChangeDetectionInterval configuration attribute value set to 40 seconds through Oracle JDeveloper. If you change the ExternalChangeDetectionInterval configuration attribute to 50 seconds, and you redeploy the application, the value of the attribute remains at 50 seconds.
- In a cluster, because all instances of the deployed application point to the same MDS Repository partition, all instances of the application use the same value. If a configuration attribute has been changed for one application instance, all instances of that application in a cluster use the changed value.

The following topics describe how you can change the MDS configuration attributes:

- Changing the MDS Configuration Attributes Using Fusion Middleware Control
- Changing the MDS Configuration Using WLST
- Restoring the Original MDS Configuration for an Application

10.9.1 Changing the MDS Configuration Attributes Using Fusion Middleware Control

To change the MDS configuration attributes of an application, take the following steps:

1. Navigate to the application's home page by expanding the farm, then **Application Deployments.** Then, select an application.

   The application's home page is displayed.

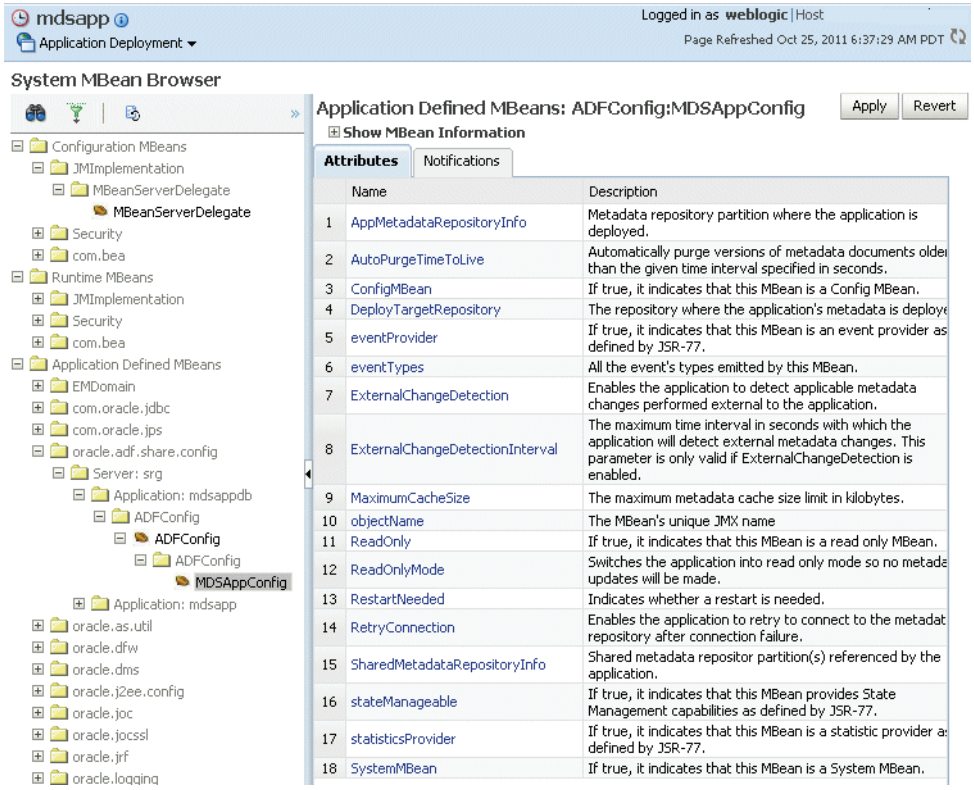
2. From the Application Deployment menu, choose **System MBean Browser.**

   The System MBean Browser page is displayed.

3. Expand **Application Defined MBeans,** then **oracle.adf.share.config,** then **Server:** ***name***, then **Application:** ***name***, then **ADFConfig,** then **ADFConfig,** and **ADFConfig.**

4. Select **MDSAppConfig.**

   The Application Defined MBeans page is displayed, as shown in the following figure:



5. You can view the description and values for the attributes.

   Table 10–2 describes the configuration attributes that are specific to MDS. Note that other attributes, such as ConfigMBean appear in the browser, but these are generic attributes for all MBeans.

Table 10–2 MDS Configuration Attributes for Deployed Applications

Attribute	Description
AppMetadataRepositoryInfo	Read only. Describes the metadata repository partition where the application is deployed.
AutoPurgeTimeToLive	Automatically purge versions of metadata documents older than the given time interval, specified in seconds. Any unlabeled versions older than this time interval are automatically purged on any subsequent update from this application. If the value is not set, versions are not automatically purged.
DeployTargetRepository	The name of the target repository configured for the application.

Table 10–2 (Cont.) MDS Configuration Attributes for Deployed Applications

Attribute	Description
ExternalChangeDetection	Specifies that the MDS Repository is polled to determine if any metadata changes have been performed on other cluster nodes or by other applications. If changes are detected, notifications are sent to applications that share the repository. Multiple applications can share metadata that is deployed to a shared repository. Changes performed by one application to this shared metadata can be detected by the other application. To do this, both the applications should configure the shared repository as part of their application configuration. If the MDS Repository is being used by more than one application in the same JVM, then MDS polls for changes if any of those applications have ExternalChangeDetection set to true. This attribute should only be set to false if the application metadata is never updated or if it is used only by this application and on a single server node. This attribute is applicable only to database-based repositories. The default is true.
ExternalChangeDetectionInterval	The maximum time interval, in seconds, to poll the MDS Repository to determine if there are external metadata changes. This attribute is only valid if ExternalChangeDetection is enabled. If the MDS Repository is shared and being used by more than one application in the same JVM, MDS uses the lowest of the values specified in the different applications for this attribute. As a result, changing the value of this parameter in one application only has an effect if the new value is lower than any values specified in the other applications. The default is 30 seconds.
MaximumCacheSize	The maximum metadata cache size limit, in kilobytes. If the value is 0, caching is disabled. If no value is specified, there is no cache limit. In this case, cached data is stored indefinitely.
ReadOnlyMode	Changes the application to read-only mode, so that no updates can be made to the application's repository partition, including configuration and application metadata.
RetryConnection	Enables the application to retry the connection to the metadata repository after connection failure.
SharedMetadataRepositoryInfo	Read only. Specifies the MDS Repository partition used by the application. Note that an application can use more than one shared metadata repository.

6. To view or modify an attribute, select the attribute.

   The attribute page is displayed.

7. If the attribute is not read-only, you can change the values. For example, for AutoPurgeTimeToLive, you can change the interval, by entering a new value in **Value.**

8. Click **Apply.**

9. Navigate up to ADFConfig (the parent of MDSAppConfig) and select it.

10. In the Operations tab, click **Save.**
11. Click **Invoke.**

10.9.2 Changing the MDS Configuration Using WLST

You can change the MDS configuration of an application using WLST. The following example shows a WLST script that reads and then sets the ReadOnlyMode attribute:

```
"""
Getting ReadOnlyMode Attribute from MBean
"""
connect('username','password','hostname:port')
application = 'application_name'
attribute = 'ReadOnlyMode'
beanName = 'oracle.adf.share.config:ApplicationName='+ application
+',name=MDSAppConfig,type=ADFConfig,Application='+ application
+',ADFConfig=ADFConfig,*'

beanObjectName = ObjectName(beanName)
beans = mbs.queryMBeans(beanObjectName, None)
bean = beans.iterator().next().getObjectName()
custom()
value = mbs.getAttribute(bean, attribute)
print value

"""
Setting ReadOnlyMode Attribute from MBean
"""
attr = Attribute(attribute, Boolean(0))
mbs.setAttribute(bean,attr)
value = mbs.getAttribute(bean, attribute)
print value

"""
Saving the Changes. This is required to persist the changes.
"""

adfConfigName = 'oracle.adf.share.config:ApplicationName='+ application +
',name=ADFConfig,type=ADFConfig,Application='+ application + ',*'
adfConfigObjectName = ObjectName(adfConfigName)
adfConfigMBeans = mbs.queryMBeans(adfConfigObjectName, None)
adfConfigMBean = adfConfigMBeans.iterator().next().getObjectName()
mbs.invoke(adfConfigMBean, 'save', None, None)
```

10.9.3 Restoring the Original MDS Configuration for an Application

To restore the original MDS configuration for an application:

1. Navigate to the application's home page by expanding the farm, then **Application Deployments.** Then, select an application.

   The application's home page is displayed.

2. From the Application Deployment menu, choose **System MBean Browser.**

   The System MBean Browser page is displayed.

3. Expand **Application Defined MBeans,** then **oracle.adf.share.config,** then **Server:** ***name,*** then **Application:** ***name,*** then **ADFConfig,** and then **ADFConfig.**
4. Select the Operations tab.

5. Select **RestoreToOriginalConfiguration.**

   The Operation: restoreToOriginalConfiguration page is displayed.

6. Click **Invoke.**

   Use this operation with caution. It causes all changes made to the original adf-config.xml file to be discarded. The adf-config.xml is restored to the base document.

Part V

Monitoring Oracle Fusion Middleware

This part provides information about how to find information about the cause of an error and its corrective action, to view and manage log files to assist in monitoring system activity and to diagnose problems and how to monitor Oracle Fusion Middleware.

Part V contains the following chapters:

11

Monitoring Oracle Fusion Middleware

This chapter describes how to monitor Oracle Fusion Middleware using Fusion Middleware Control, Oracle WebLogic Server Administration Console, and the command line. It describes the following topics:

- Monitoring the Status of Oracle Fusion Middleware
- Viewing the Performance of Oracle Fusion Middleware
- Viewing the Routing Topology

> **Note:** For information about monitoring servers for IBM WebSphere, see "Managing Oracle Fusion Middleware on IBM WebSphere" in the *Oracle Fusion Middleware Third-Party Application Server Guide*.

11.1 Monitoring the Status of Oracle Fusion Middleware

Monitoring the health of your Oracle Fusion Middleware environment and ensuring that it performs optimally is an important task for the administrator.

Oracle Fusion Middleware provides the following methods for monitoring the status of your environment:

- Oracle WebLogic Server Administration Console: You can monitor the status of Oracle WebLogic Server domains, clusters, servers, Java components, and applications. From the Administration Console, navigate to the entity's page. See "Overview of the Administration Console" in the *Oracle Fusion Middleware Introduction to Oracle WebLogic Server* for information on monitoring using the console.
- Fusion Middleware Control: You can monitor the status of Oracle WebLogic Server domains, clusters, servers, Java components, system components, and applications. Navigate to the entity's home page, for example, to the home page for an Oracle HTTP Server instance.
- The command line: You can monitor the status of your environment using the WLST or opmnctl command lines.

  To monitor the status of Java components, use the WLST `state` command, with the following format:

  ```
  state(name, type)
  ```

  For example, to get the status of the Managed Server server1, use the following command:

  ```
  wls:/mydomain/serverConfig> state('server1','Server')
  ```

```
Current state of "server1": SUSPENDED
```

To monitor the status of system components, use the `opmnctl status` command, with the following format:

```
opmnctl status [scope] [options]
```

For example, to view the status of all processes monitored by OPMN, use the following command:

```
opmnctl status
```

Most of the monitoring tasks in this chapter describe how to monitor using Fusion Middleware Control or the command line.

The following topics provide more detail:

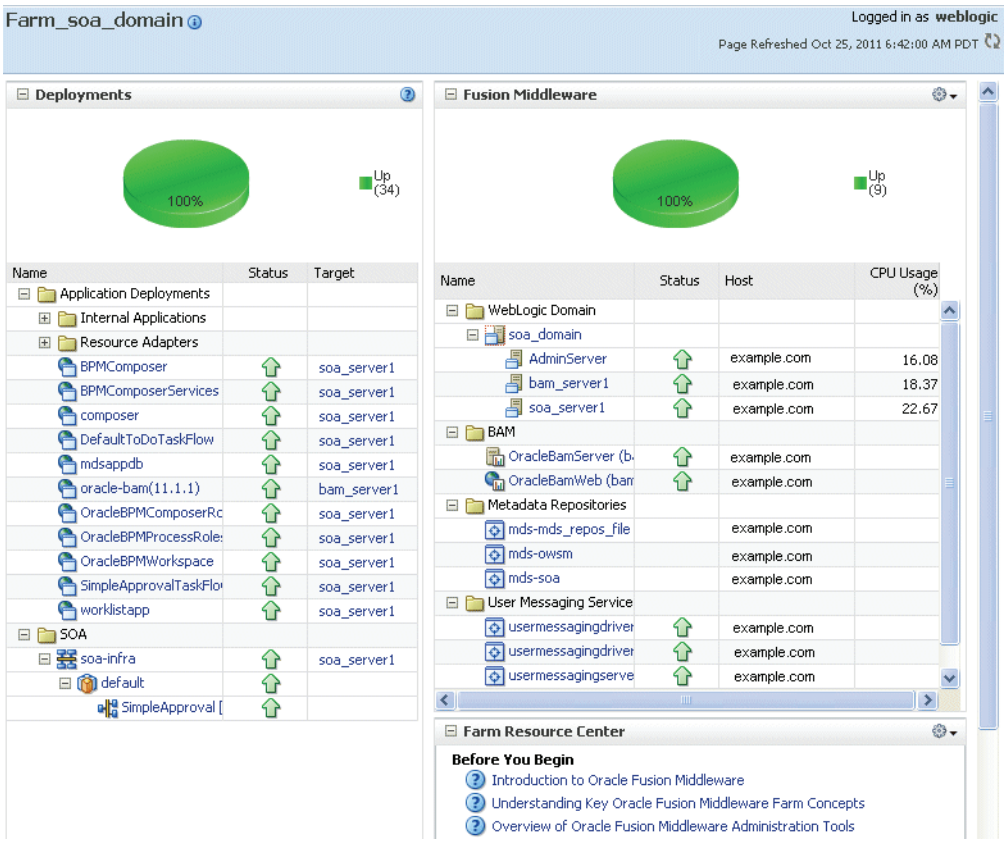
- Viewing General Information
- Monitoring an Oracle WebLogic Server Domain
- Monitoring an Oracle WebLogic Server Administration or Managed Server
- Monitoring a Cluster
- Monitoring a Java Component
- Monitoring a System Component
- Monitoring Java EE Applications
- Monitoring ADF Applications
- Monitoring SOA Composite Applications
- Monitoring Oracle WebCenter Portal Applications
- Monitoring Applications Deployed to a Cluster

11.1.1 Viewing General Information

You can view the overall status of the Oracle Fusion Middleware environment from the home page of the farm using Fusion Middleware Control. This page lists the availability of all components, an application deployment summary, including SOA composites, if any SOA composite applications are deployed.

To view the overall status, from the navigation pane, select the farm.

The farm home page is displayed, as shown in the following figure:



This page shows the following:

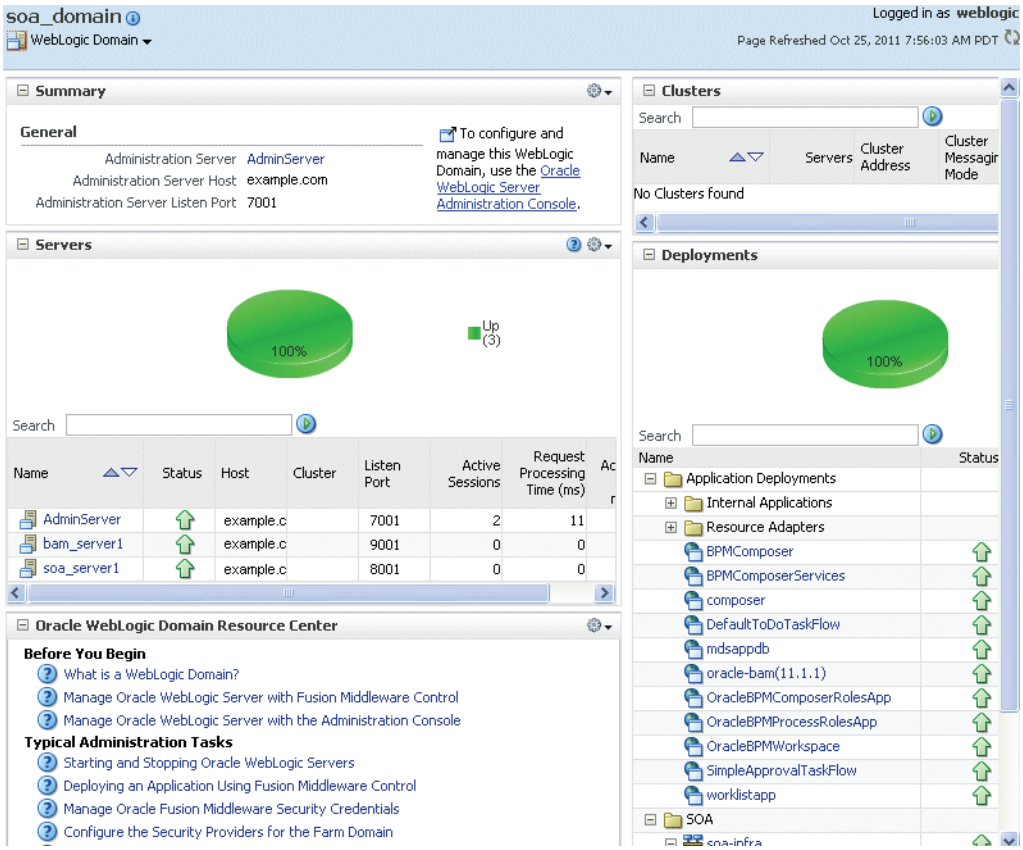
- A list of deployed applications and the status of each
- Lists of domains, the servers within the domains, metadata repositories, and other Oracle Fusion Middleware entities, with the status of each
- A Resource Center with links to relevant documentation

11.1.2 Monitoring an Oracle WebLogic Server Domain

You can view the status of a domain, including the servers, clusters, and deployments in the domain from the domain home page of Fusion Middleware Control:

1. From the navigation pane, expand the farm, then **WebLogic Domain.**
2. Select the domain.

   The domain home page is displayed, as shown in the following figure:



This page shows the following:

- A general summary of the domain, along with a link to the Oracle WebLogic Server Administration Console
- Information about the servers, both the Administration Server and the Managed Servers, in the domain
- Information about the clusters in the domain
- Information about the deployments in the domain
- A Resource Center, which provides links to more information

> **See Also:** "Overview of the Administration Console" in the *Oracle Fusion Middleware Introduction to Oracle WebLogic Server* for information about monitoring an Oracle WebLogic Server domain using the Oracle WebLogic Server Administration Console. The Administration Console provides details about the health and performance of the domain.

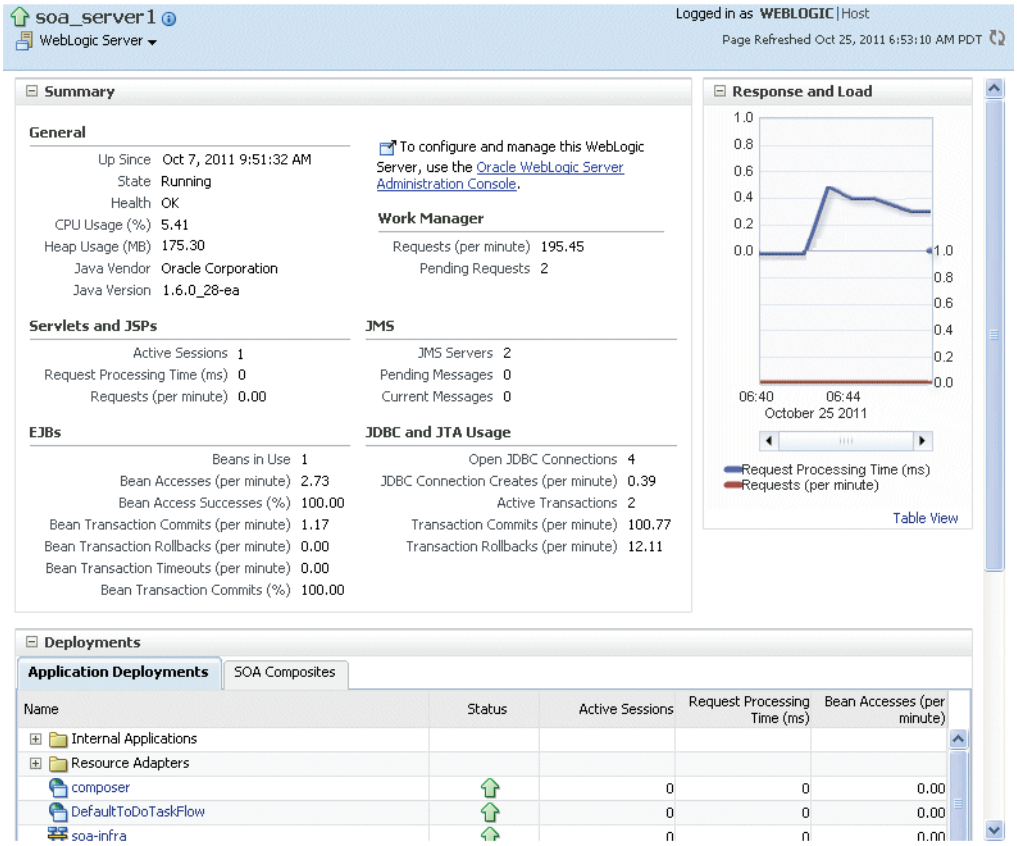
11.1.3 Monitoring an Oracle WebLogic Server Administration or Managed Server

You can view the status of a WebLogic Server Administration Server or Managed Server in Fusion Middleware Control:

1. From the navigation pane, expand the farm, then **WebLogic Domain**, and then the domain.
2. Select the server.

   The server home page is displayed.

The following figure shows the home page for a Managed Server:



This page shows the following:

- A general summary of the server, including its state, and information about the servlets, JSPs, and EJBs running in the server
- Response and load
- Information about the applications deployed to the server

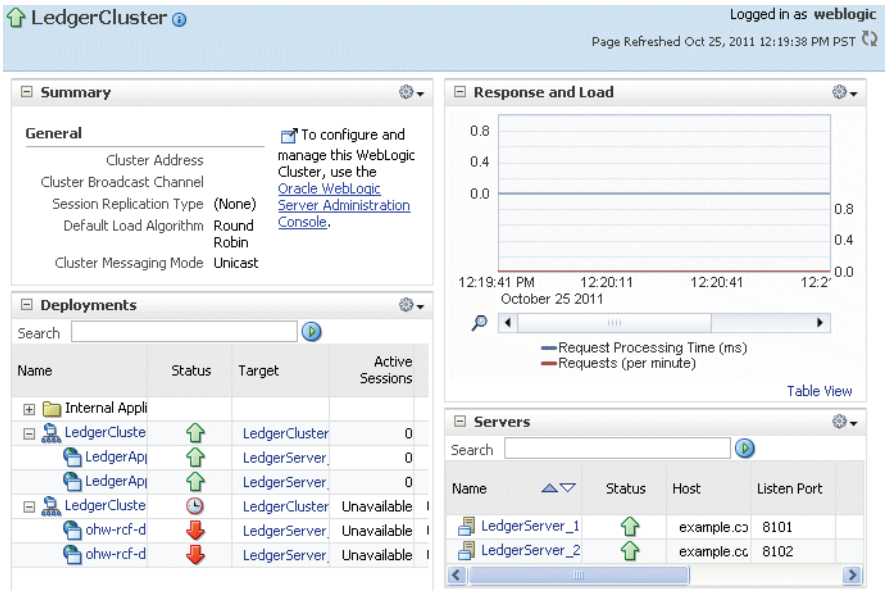
> **See Also:** "Overview of the Administration Console" in the *Oracle Fusion Middleware Introduction to Oracle WebLogic Server* for information about monitoring servers using the Oracle WebLogic Server Administration Console. The Administration Console provides details about the health and performance of the server.

11.1.4 Monitoring a Cluster

You can view the status of a cluster, including the servers and deployments in the cluster using Fusion Middleware Control:

1. From the navigation pane, expand the farm, then **WebLogic Domain**, and then the domain.
2. Select the cluster.

   The cluster page is displayed, as shown in the following figure:



This page shows the following:

- A general summary of the cluster, including the broadcast channel, if appropriate, the load algorithm, and the messaging mode
- A response and load section, which shows the requests per minute and the request processing time
- A deployments section with information about the applications deployed to the cluster
- A server section, with a table listing the servers that are part of the cluster

See Also: "Overview of the Administration Console" in the *Oracle Fusion Middleware Introduction to Oracle WebLogic Server* for information about monitoring a cluster using Oracle WebLogic Server Administration Console. The Administration Console provides details about the health and performance of the cluster.

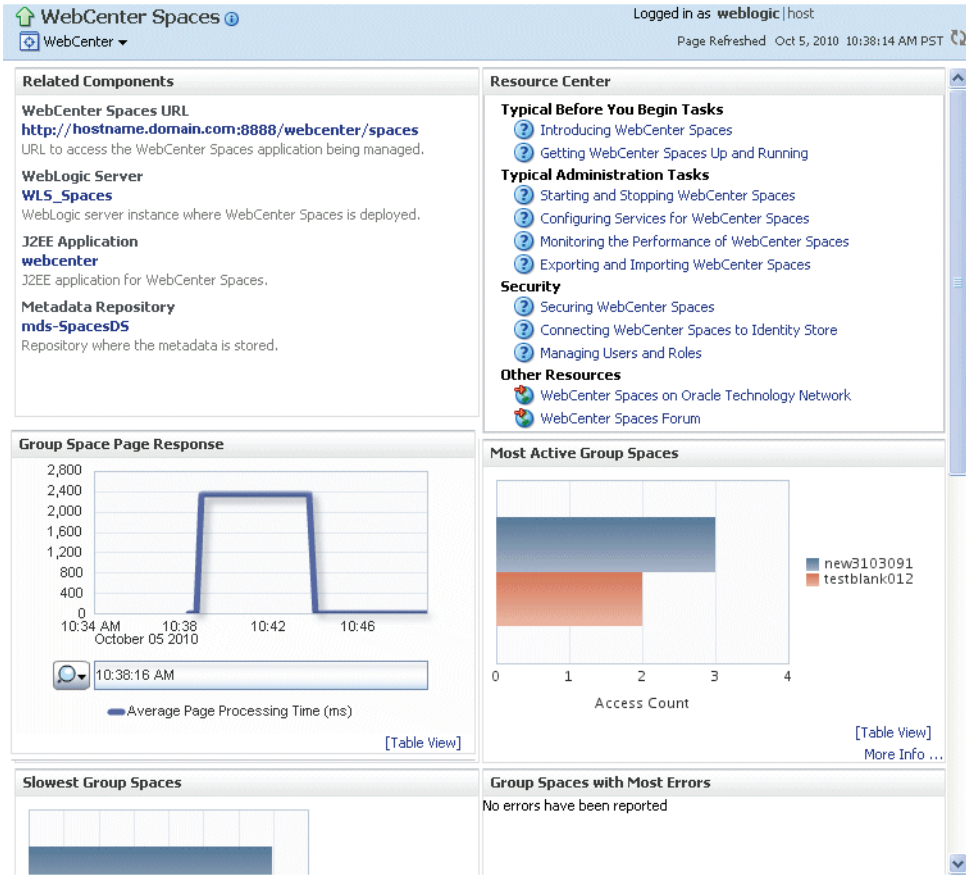
11.1.5 Monitoring a Java Component

You can view the status of a Java component, including whether the component is started, in the component home page in Fusion Middleware Control.

To monitor a Java component, such as WebCenter Portal: Spaces:

1. From the navigation pane, expand the farm, then the type of component, such as WebCenter, then the component, such as Portal, then Spaces.
2. Select the component. For example, select WebCenter Spaces instance.

   The component home page is displayed, as shown in the following figure:



This page shows the following:

- A list of related components
- A Resource Center with links to relevant documentation
- A chart showing the group space page response
- A chart showing the most active group spaces
- A chart showing the slowest group spaces
- A chart showing the group spaces with the most errors

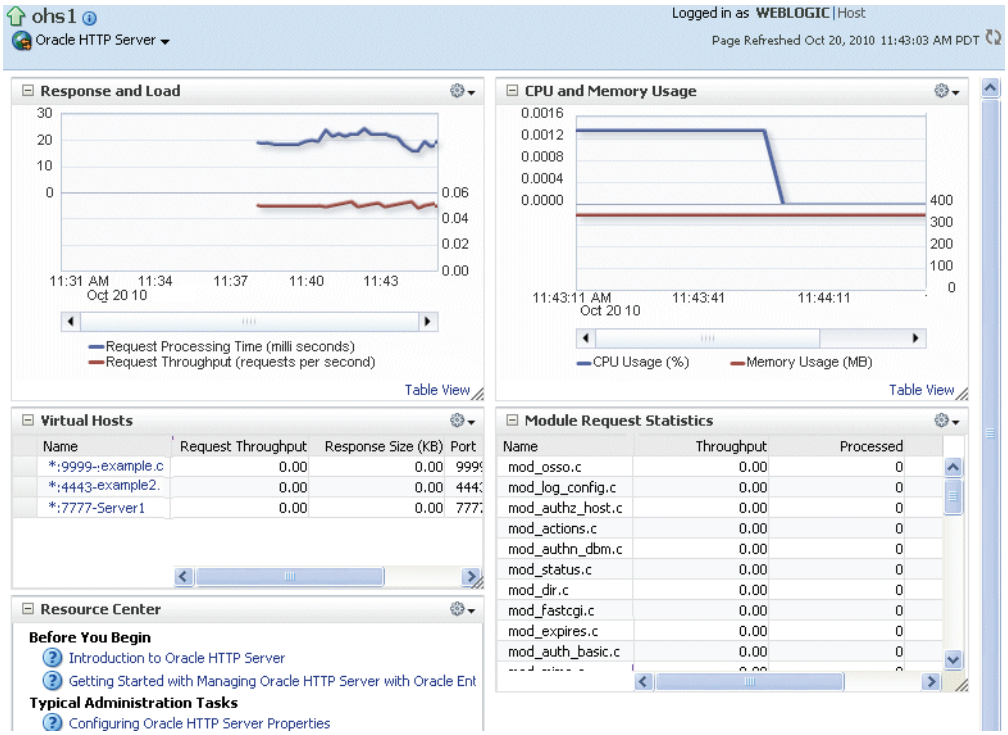
> **See Also:** "Overview of the Administration Console" in the *Oracle Fusion Middleware Introduction to Oracle WebLogic Server* for information about using the Oracle WebLogic Server Administration Console to monitor Java components

11.1.6 Monitoring a System Component

To monitor a system component, such as Oracle HTTP Server:

1. From the navigation pane, expand the farm, then **Web Tier.**
2. Select the component, such as ohs1.

   The component home page is displayed, as shown in the following figure:



This page shows the following:

- A response and load section, which shows the requests per second and the request processing time
- CPU and memory usage
- The virtual hosts, with their names, request throughput and response size.
- Module request statistics, with a list of modules and the throughput for each.: A table listing the processing time for each module
- A Resource Center with links to relevant documentation

11.1.7 Monitoring Java EE Applications

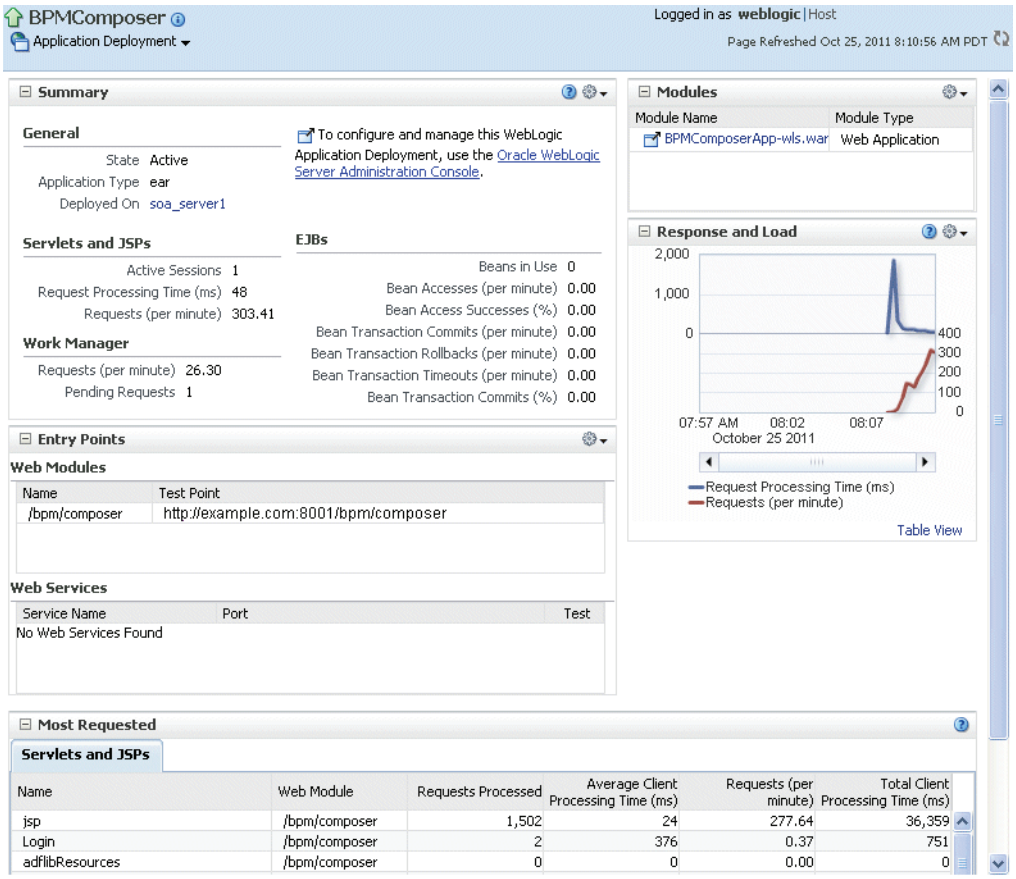
To monitor a Java EE application using Fusion Middleware Control:

1. From the navigation pane, expand the farm, expand **Application Deployments,** then select the application to monitor.

   The application's home page is displayed.

2. In this page, you can view a summary of the application's status, entry points to the application, Web services and modules associated with the application, and the response and load.

   The following figure shows a portion of the application's home page:



This page shows the following:

- A summary of the application, including its state, the Managed Server on which it is deployed, and information about active sessions, active requests, and request processing time
- Entry points, including any Web modules and Web services
- A list of modules with the type of module for each
- Response and load, which shows the requests per minute and the request processing time
- A list of most requested servlets, JSPs, and Web services

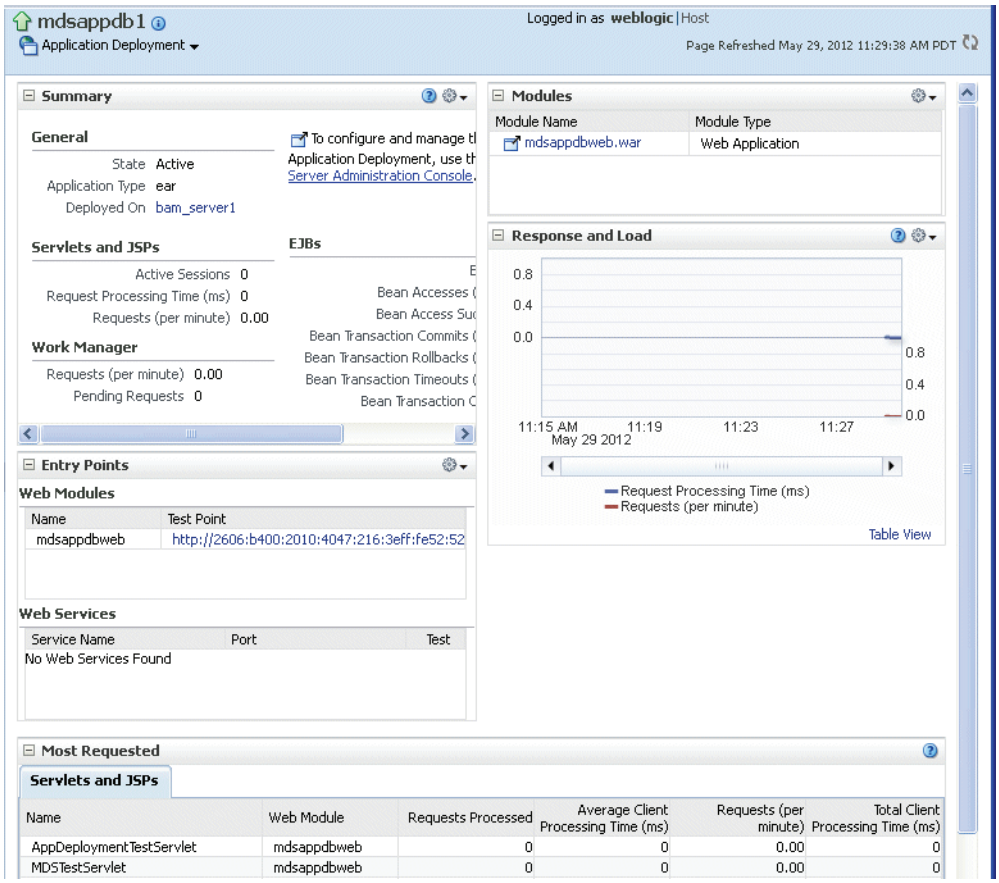
11.1.8 Monitoring ADF Applications

To monitor an ADF application:

1. From the navigation pane, expand the farm, expand **Application Deployments,** then select the application to monitor.

   The application's home page is displayed.

   The following figure shows a portion of the application's home page:



- A summary of the application, including its state, the Managed Server on which it is deployed, and information about active sessions, active requests, and request processing time
- Entry points, including any Web modules and Web services
- A list of modules with the type of module for each
- Response and load, which shows the requests per minute and the request processing time
- A list of most requested servlets and JSPs

2. In this page, you can view a summary of the application's status, entry points to the application, Web services and modules associated with the application, and the response and load.
3. To view performance information about application modules, application module pools, and task flows, from **Application Deployments,** choose **ADF,** then **ADF Performance.** The ADF Performance page is displayed.

   It contains a tab for Application Module Pools, which shows the requests, average creation time, and free instances. It also contains a tab for Task Flows, which shows the request processing time and active taskflows.

11.1.9 Monitoring SOA Composite Applications

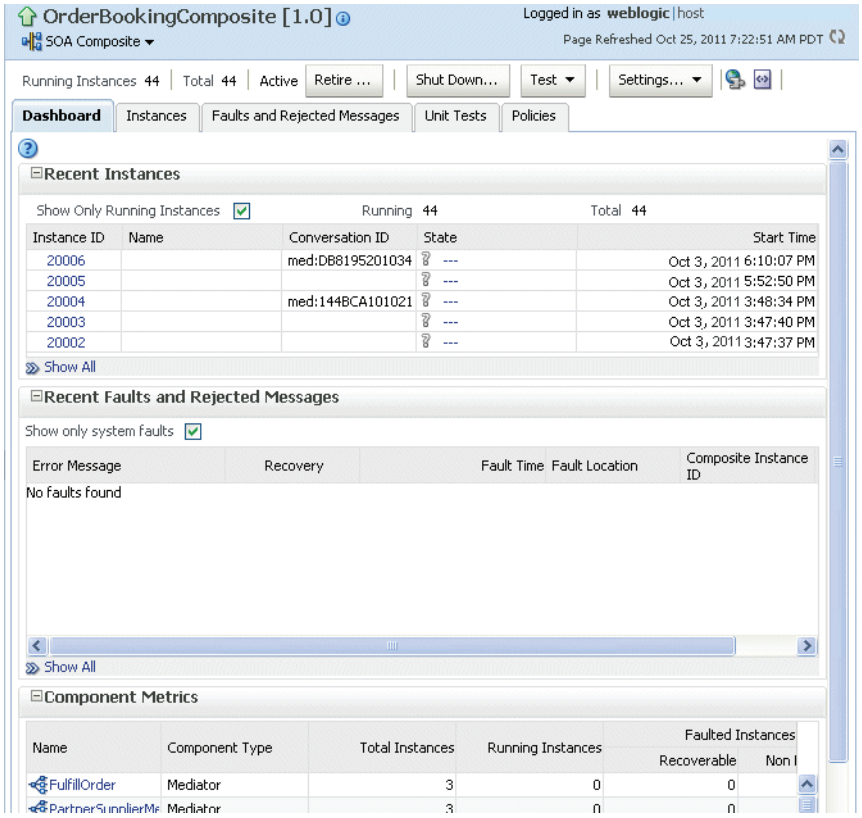
To monitor a SOA composite application:

1. From the navigation pane, expand the farm, expand **SOA,** then **soa-infra.** Select the application to monitor.

The application's home page is displayed.

2. From this page, you can monitor the running instances, faults and rejected messages, and component metrics.

   The following figure shows part of a SOA composite home page:



This page, with the Dashboard tab selected, shows the following:

- The recent instances
- Recent faults and rejected messages
- Component metrics

11.1.10 Monitoring Oracle WebCenter Portal Applications

To monitor an Oracle WebCenter Portal application:

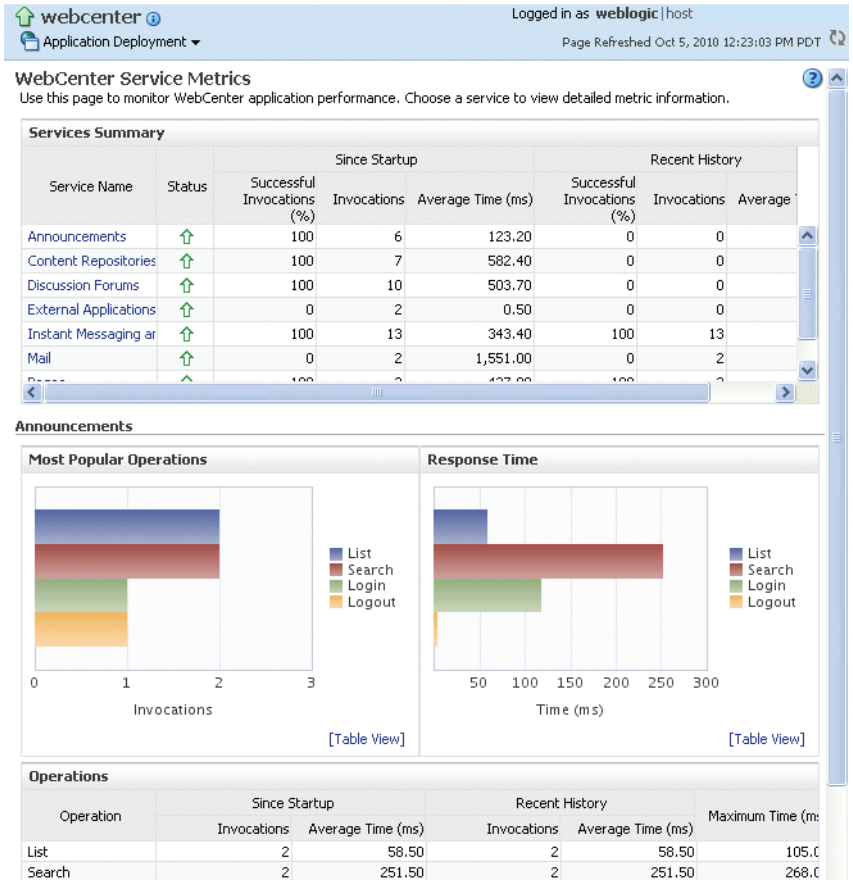
1. From the navigation pane, expand the farm, expand **Application Deployments,** then select the application to monitor.

   The application's home page is displayed.

2. In this page, you can view a summary of the application's status, entry points to the application, Web services and modules associated with the application, and the response and load.

3. For some applications, you can view service metrics. From the Application Deployment menu, choose **Web Center,** then **Service Metrics.**

   The following figure shows the Service Metrics page:



This page shows the following:

- A summary of WebCenter Service metrics, with the status and invocations for each service.
- A chart showing the most popular operations,
- A chart showing response time for the different types of operations
- An Operations section that shows the operations since startup and recent operations

> **See Also:** "Understanding WebCenter Portal Performance Metrics" in *Oracle Fusion Middleware Administrator's Guide for Oracle WebCenter Portal* for more information about service metrics

11.1.11 Monitoring Applications Deployed to a Cluster

If you deploy an application to a cluster, Oracle Fusion Middleware automatically deploys the application to each Managed Server in the cluster. As a result, there is an instance of the application on each server.

There are times when you want to monitor the performance of the application on an individual server, and times when you want to monitor the overall performance of the application across all the servers in the cluster.

For example, normally, you would manage the overall performance of the application to determine if there are any performance issues affecting all users of the application, regardless of which instance users access. If you notice a performance problem, you

can then drill down to a specific instance of the application to determine if the problem is affecting one or all of the application instances in the cluster.

Fusion Middleware Control provides monitoring pages for both of these scenarios:

1. From the navigation pane, expand the farm, expand **Application Deployments**.

   Fusion Middleware lists the applications deployed in the current domain.

   If an application has been deployed to a cluster, Fusion Middleware Control shows a plus sign (+) next to the application to indicate that it represents more than one instance of the application on the cluster.

2. Expand the cluster application to show each instance of the application, as shown in the following figure:

3. Monitor the overall performance of the application on the cluster by clicking the cluster application, or monitor the performance of the application on a single server by clicking one of the application deployment instances.

11.2 Viewing the Performance of Oracle Fusion Middleware

If you encounter a problem, such as an application that is running slowly or is hanging, you can view more detailed performance information, including performance metrics for a particular target, to find out more information about the problem.

Oracle Fusion Middleware automatically and continuously measures run-time performance. The performance metrics are automatically enabled; you do not need to set options or perform any extra configuration to collect them.

Note that Fusion Middleware Control provides real-time data. If you are interested in viewing historical data, consider using Oracle Enterprise Manager Grid Control.

For example, to view the performance of an Oracle WebLogic Server Managed Server:

1. From the navigation pane, expand the farm, then **WebLogic Domain**, and then the domain.
2. Select the server to monitor.

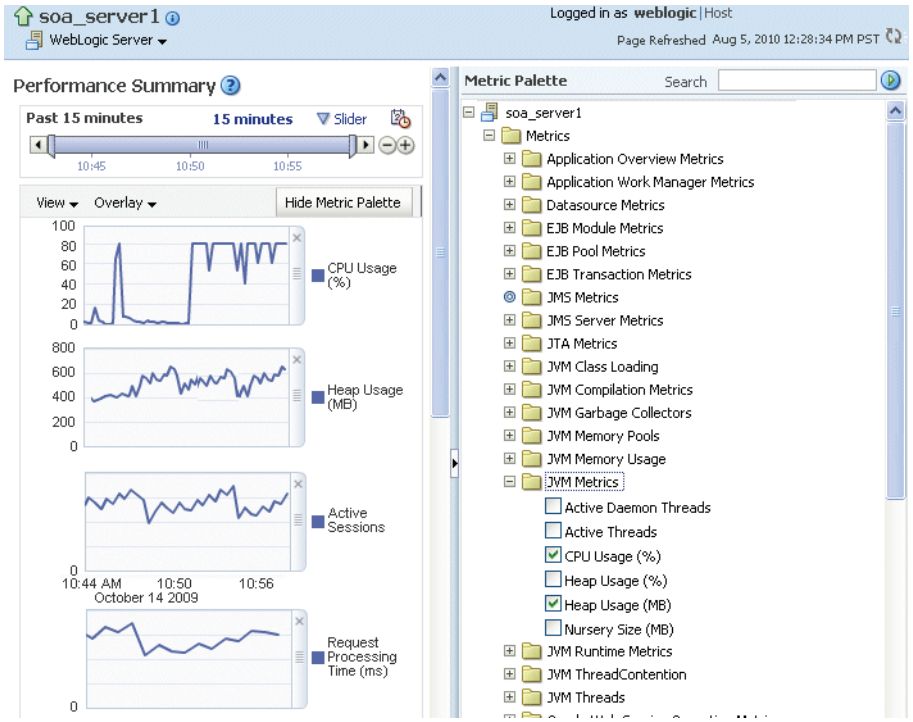
   The Managed Server home page is displayed.

3. From the WebLogic Server menu, choose **Performance Summary.**

   The Performance Summary page is displayed. It shows performance metrics, as well as information about response time and request processing time for applications deployed to the Oracle WebLogic Server.

4. To see additional metrics, click **Show Metric Palette** and expand the metric categories.

   The following figure shows the Performance Summary page with the Metric Palette displayed:



5. Select a metric to add it to the Performance Summary.
6. To overlay another target, click **Overlay**, and select the target. The target is added to the charts, so that you can view the performance of more than one target at a time, comparing their performance.
7. To customize the time frame shown by the charts, you can:
   - Click **Slider** to display a slider tool that lets you specify that more or less time is shown in the charts. For example, to show the past 10 minutes, instead of the past 15 minutes, slide the left slider control to the right until it displays the last 10 minutes.
   - Select the calendar and clock icon. Then, enter the **Start Time** and **End Time.** If there is no data available for those times, a confirmation message displays, explaining the timeline will be automatically adjusted to the time period for which the data is available.

You can also view the performance of a components, such as Oracle HTTP Server or Oracle SOA Suite. Navigate to the component and select **Monitoring,** then **Performance Summary** from the dynamic target menu.

11.3 Viewing the Routing Topology

Fusion Middleware Control provides a Topology Viewer for the farm. The Topology Viewer is a graphical representation of routing relationships across components and elements of the farm. You can easily determine how requests are routed across components. For example, you can see how requests are routed from Oracle Web Cache, to Oracle HTTP Server, to a Managed Server, to a data source.

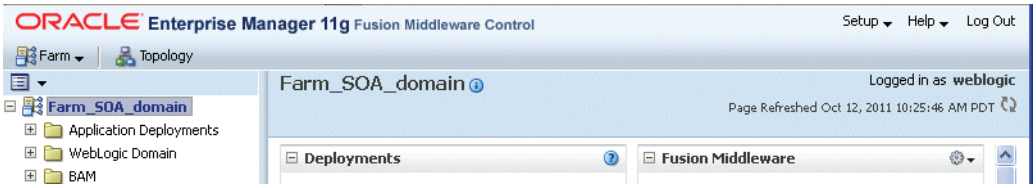
Note: To view relationships between Oracle WebLogic Server, Oracle Web Cache, and Oracle HTTP Server, each target must be running and show its status as Up.

The Topology Viewer enables you to easily monitor your Oracle Fusion Middleware environment. You can see which entities are up and which are down.

You can also print the topology.

To view the topology:

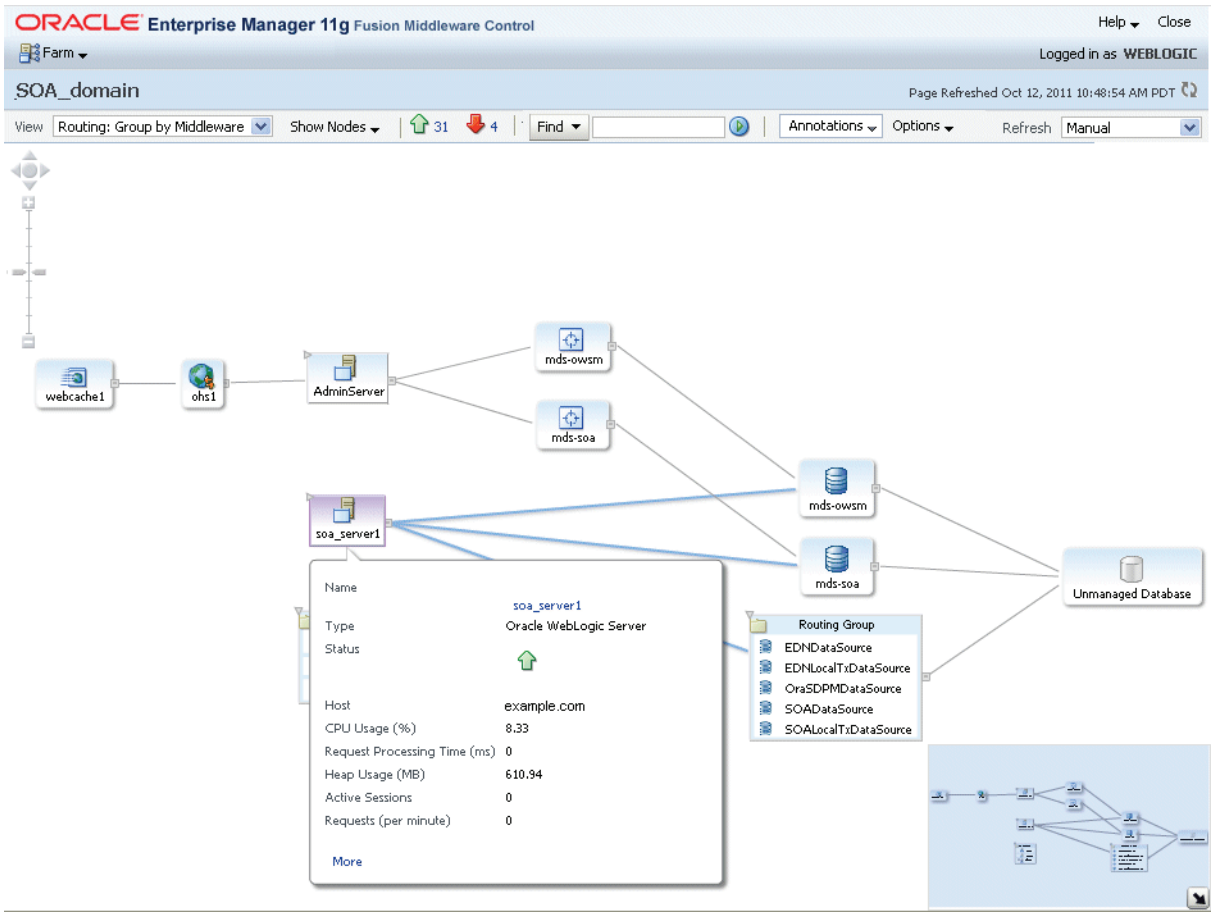
1. Click **Topology,** as shown in the following figure:



   The Topology Viewer is displayed in a separate window.

2. To see information about a particular target, place your mouse over the target. To view additional information, click **More.**

   The following shows the Topology Viewer window, with information about the Managed Server soa_server1:



With Topology Viewer, you can also:

- Choose how to group the routing. From the View menu, you can choose to group by Middleware or by application.

- Choose the types of nodes to show. From the Show Nodes menu, select the types of nodes, such as data sources.
- View the targets by status. Click the green up arrow or the red down arrow at the top of the page. A list of the targets with the specified status is shown.
- Search for a target within the topology. This makes it easier to find a target if you have many targets. Enter the name in the **Find** box, and click **Find.**

  The Find results box is displayed. Click the target name to highlight the target. The topology is repositioned so you can see the target if it was not previously visible in the viewing area.

  You can also specify criteria for the search. From **Find**, choose the one or more types of **Status** or one or more of **Target Type**, or both.
- Hide or show the status or metrics. From **Annotations**, click **Status** or **Metrics**.

  If you select Metrics, one key performance metric for the component is displayed. (You cannot change the metric that is displayed.)
- Reposition the topology and change its orientation:
  - To change the orientation, from the Options menu, choose **Layout,** then **Left to Right** or **Top Down.**
  - To reposition the topology, click in the topology, but not on a target or route. Drag the topology to position it.
  - To change what is visible in the topology view, from the Options menu, choose **Show/Hide Navigator.** Then, drag the shaded section in the navigator window, which is located in the bottom right.
- Navigate to the home page of a target. Right-click the target, and select **Home.**
- Perform operations directly on the target by right-clicking. The right-click target menu is displayed. For example, from this menu, you can start or stop an Oracle WebLogic Server or view additional performance metrics.
- View the routing relationships between components. For example, you can view the routing from Oracle Web Cache to Oracle HTTP Server to Oracle WebLogic Server. Clicking on the line between the two targets displays the URLs used.
- From the **Refresh** dropdown, you can refresh manually, or you can enable automatically refreshing the status and metrics, every minute, every five minutes, or every thirty minutes. By default, the Topology Viewer refreshes the metrics every 5 minutes.

Notes:

- If you use Mozilla Firefox, when you click a link or menu item in the Topology Viewer to navigate back to the main Fusion Middleware Control window, the main window does not always get focus. For example, if you right-click a target node and select View Log Messages from the target menu, the focus remains on the Topology Viewer window. (If you go back to the main window, the Log Messages page is correctly displayed.)

  To workaround this problem, make the following change in Firefox:

  From the Tools menu, select **Options,** and then **Content.** Click **Advanced.** In the Advanced JavaScript Settings dialog box, select **Raise and lower windows.**

- If you use Internet Explorer, turn off the **Always Open Popups in New Tab** option.

12

Managing Log Files and Diagnostic Data

Oracle Fusion Middleware components generate log files containing messages that record all types of events, including startup and shutdown information, errors, warning messages, and access information on HTTP requests. This chapter describes how to find information about the cause of an error and its corrective action and to view and manage log files to assist in monitoring system activity and in diagnosing problems.

It contains the following topics:

- Overview of Oracle Fusion Middleware Logging
- Understanding ODL Messages and ODL Log Files
- Viewing and Searching Log Files
- Configuring Settings for Log Files
- Correlating Messages Across Log Files and Components
- Configuring Tracing

> **Note:** For information about logging for IBM WebSphere, see "Configuring Oracle Fusion Middleware Logging on IBM WebSphere" in the *Oracle Fusion Middleware Third-Party Application Server Guide*.

12.1 Overview of Oracle Fusion Middleware Logging

Most Oracle Fusion Middleware components write diagnostic log files in the Oracle Diagnostic Logging (ODL) format. Log file naming and the format of the contents of log files conforms to an Oracle standard. By default, the diagnostic messages are written in text format.

ODL provides the following benefits:

- The capability to limit the total amount of diagnostic information saved. You can set the level of information saved and you can specify the maximum size of the log file and the log file directory.
- When you reach the specified size, older segment files are removed and newer segment files are saved in chronological fashion.
- Components can remain active, and do not need to be shutdown, when older diagnostic logging files are deleted.

You can view log files using Fusion Middleware Control or the WLST `displayLogs` command, or you can download log files to your local client and view them using another tool (for example, a text editor or another file viewing utility).

Note: Oracle WebLogic Server does not use the ODL format. For information about the Oracle WebLogic Server log format, see *Oracle Fusion Middleware Configuring Log Files and Filtering Log Messages for Oracle WebLogic Server*.

12.2 Understanding ODL Messages and ODL Log Files

Using ODL, diagnostic messages are written to log files and each message includes information, such as the time, component ID, and user.

The following example shows an ODL format error messages from Oracle SOA Suite:

```
[2010-09-23T10:54:00.206-07:00] [soa_server1] [NOTIFICATION] [] [oracle.mds]
 [tid: [STANDBY].ExecuteThread: '1' for queue: 'weblogic.kernel.Default
 (self-tuning)'] [userId: <anonymous>] [ecid:
0000I3K7DCnAhKB5JZ4Eyf19wAgN000001,0]
 [APP: wsm-pm] "Metadata Services: Metadata archive (MAR) not found."
```

In the message, the fields map to the following attributes, which are described in Table 12–1:

- Timestamp, originating: `2010-09-23T10:54:00.206-07:00`
- Organization ID: `soa_server1`
- Message Type: `NOTIFICATION`
- Component ID: `oracle.mds`
- Thread ID: `tid: [STANDBY].ExecuteThread: '1' for queue: 'weblogic.kernel.Default (self-tuning)'`
- User ID: `userId: <anonymous>`
- Execution Context ID: `ecid: 0000I3K7DCnAhKB5JZ4Eyf19wAgN000001,0`
- Supplemental Attribute: `APP: wsm-pm`
- Message Text: `"Metadata Services: Metadata archive (MAR) not found."`

By default, the information is written to the log files in ODL text format. You can change the format to ODL XML format, as described in Section 12.4.4.

Table 12–1 describes the contents of an ODL message. For any given component, the optional attributes may not be present in the generated diagnostic messages.

Table 12–1 ODL Format Message Fields

Attribute Name	Description	Required
Timestamp, Originating (TIME)	The date and time when the message was generated. This reflects the local time zone.	Yes
Timestamp, normalized (time_norm)	The timestamp normalized for clock drift across hosts. This field is used when the diagnostic message is copied to a repository on a different host.	No
Organization ID (org_id)	The organization ID for the originating component.	No
INSTANCE_ID (INST_ID)	The name of the Oracle instance to which the component that originated the message belongs.	No
COMPONENT ID (COMP_ID)	The ID of the component that originated the message.	Yes

Table 12–1 (Cont.) ODL Format Message Fields

Attribute Name	Description	Required
MESSAGE_ID (MSG_ID)	The ID that uniquely identifies the message within the component. The ID consists of a prefix that represents the component, followed by a dash, then a 5-digit number. For example: `OHS-51009`	Yes
MESSAGE_TYPE (MSG_TYPE)	The type of message. Possible values are: INCIDENT_ERROR, ERROR, WARNING, NOTIFICATION, TRACE, and UNKNOWN. See Table 12–3 for information about the message types.	Yes
MESSAGE_LEVEL (MSG_LEVEL)	The message level, represented by an integer value that qualifies the message type. Possible values are from 1 (highest severity) through 32 (lowest severity). See Table 12–3 for information about the message levels.	Yes
HOST_ID (HOST_ID)	The name of the host where the message originated.	No
HOST_NW_ADDR (HOST_ADDR)	The network address of the host where the message originated.	No
MODULE_ID (MODULE)	The ID of the module that originated the message. If the component is a single module, the component ID is listed for this attribute.	Yes
PROCESS_ID (PID)	The process ID for the process or execution unit associated with the message.	No
THREAD_ID (TID)	The ID of the thread that generated the message.	No
USER_ID (USER)	The name of the user whose execution context generated the message.	No
ECID	The Execution Context ID (ECID), which is a global unique identifier of the execution of a particular request in which the originating component participates. You can use the ECID to correlate error messages from different components. See Section 12.5 for information about ECIDs.	Yes
RID	The relationship ID (RID), which distinguishes the work done in one thread on one process, from work done by any other threads on this and other processes, on behalf of the same request. See Section 12.5 for information about RIDs.	No
SUPPL_ATTRS	An additional list of name/value pairs which contain component-specific attributes about the event.	No
MESSAGE TEXT (TEXT)	The text of the message.	Yes
Message Arguments (arg)	A list of arguments bound with the message text.	No
Supplemental Detail	Supplemental information about the event, including more detailed information than the message text.	No

The log file location depends on the type of component:

- For most Java components, the log file location is:

  (UNIX) *MW_HOME*/user_projects/domains/*domain_name*/servers/*server_name*/logs
  (Windows) *MW_HOME*\user_projects\domains*domain_name*\servers*server_name*\logs

  The default name of a log file is *server-name*-diagnostic.log.

- For system components, the default log file location is:

  (UNIX) *ORACLE_INSTANCE*/diagnostics/logs
  (Windows) *ORACLE_INSTANCE*\diagnostics\logs

Table 12–2 shows the log file location for components of Oracle Fusion Middleware.

In the table, *DOMAIN_HOME* refers to the following directory, which is the WebLogic Server domain home:

MW_HOME/user_projects/domains/*domain_name*

In the table, *ORACLE_INSTANCE* refers to the following directory, which is the Oracle instance home:

MW_HOME/*instance_name*

Table 12–2 Log File Location

Component	Log File Location
Fusion Middleware Control	*DOMAIN_HOME*/sysman/log/emoms.log *DOMAIN_HOME*/sysman/log/emoms.trc
Oracle Access Management Identity Federation	*DOMAIN_HOME*/servers/*server_name*/logs/*server-name*-diagnostic.log
Oracle Application Development Framework	*DOMAIN_HOME*/servers/*server_name*/logs/*server-name*-diagnostic.log
Oracle Business Activity Monitoring	*DOMAIN_HOME*/servers/*server_name*/logs/bam-diagnostic.log
Oracle Business Intelligence Discoverer	*DOMAIN_HOME*/servers/*server_name*/logs/discoverer/server/diagnostic.log *DOMAIN_HOME*/servers/*server_name*/logs/discoverer/*server_name*-diagnostic.log *DOMAIN_HOME*/servers/*server_name*/logs/discoverer/diagnostic.log
Oracle Business Process Management	*DOMAIN_HOME*/servers/*server_name*/logs/*server-name*-diagnostic.log
Oracle Directory Integration Platform	*DOMAIN_HOME*/servers/*server_name*/logs/*server-name*-diagnostic.log
Oracle Forms Services	*DOMAIN_HOME*/servers/*server_name*/logs/*server-name*-diagnostic.log *ORACLE_HOME*/j2ee/DevSuite/application-deployments/forms/application.log *ORACLE_INSTANCE*/diagnostics/logs/FormsComponent/forms/* *ORACLE_INSTANCE*/diagnostics/logs/FRComponent/dejvm/*
Oracle Fusion Middleware Audit Framework	*DOMAIN_HOME*/servers/*server_name*/logs/*server-name*-diagnostic.log
Oracle HTTP Server	*ORACLE_INSTANCE*/diagnostics/logs/OHS/*component_name*/*.log
Oracle Information Rights Management	*DOMAIN_HOME*/servers/*server_name*/logs/*server-name*-diagnostic.log
Oracle Internet Directory	*ORACLE_INSTANCE*/diagnostics/logs/OID/*component_name*/oid*.log *ORACLE_INSTANCE*/diagnostics/logs/OID/tools/*.log
Oracle Platform Security Services	*DOMAIN_HOME*/servers/*server_name*/logs/*server-name*-diagnostic.log
Oracle Portal	*DOMAIN_HOME*/servers/*server_name*/logs/*server-name*-diagnostic.log *ORACLE_INSTANCE*/diagnostics/logs/portal
Oracle Reports	*ORACLE_INSTANCE*/diagnostics/logs/ReportsServerComponent/*component_name*/* *ORACLE_INSTANCE*/diagnostics/logs/ReportsBridgeComponent/*component_name*/* *ORACLE_INSTANCE*/diagnostics/logs/ReportsToolsComponent/*component_name*/*
Oracle SOA Suite	*DOMAIN_HOME*/servers/*server_name*/logs/*server-name*-diagnostic.log
Oracle TopLink	*DOMAIN_HOME*/servers/*server_name*/logs/*server-name*-diagnostic.log
Oracle Virtual Directory	*ORACLE_INSTANCE*/diagnostics/logs/OVD/*component_name*/diagnostic.log

Table 12–2 (Cont.) Log File Location

Component	Log File Location
Oracle Web Cache	*ORACLE_INSTANCE*/diagnostics/logs/WebCache/*component_name**.log
Oracle Web Services Manager	*DOMAIN_HOME*/servers/*server_name*/logs/owsm/msglogging *DOMAIN_HOME*/servers/*server_name*/logs/owsm-diagnostic.log
Oracle WebCenter Content: Imaging	*DOMAIN_HOME*/servers/*server_name*/logs/*server-name*-diagnostic.log
Oracle WebCenter Portal	*DOMAIN_HOME*/servers/*server_name*/logs/*server-name*-diagnostic.log
Oracle WebLogic Server	*DOMAIN_HOME*/servers/*server_name*/logs/*server-name*-diagnostic.log
Repository Creation Utility	By default, writes to file specified in RCU_LOG_LOCATION. If not specified, attempts to write to the following locations: 1. *ORACLE_HOME*/rcu/log/*timestamp* 2. /tmp/logdir.*timestamp*

12.3 Viewing and Searching Log Files

You can view, list, and search log files across Oracle Fusion Middleware components. You can view and search log files using Fusion Middleware Control or you can download a log file to your local client and view the log files using another tool. You can also list, view, and search log files using the WLST command-line tool.

This section covers the following topics:

- Viewing Log Files and Their Messages
- Searching Log Files
- Downloading Log Files

Note the following about using the WLST commands to view the log files:

- To use the custom WLST logging commands, you must invoke the WLST script from the Oracle Common home. See Section 3.5.1.1 for more information.
- The log viewing commands work whether you are connected or not connected to a WebLogic server. If you are not connected, you must specify the path in the oracleInstance parameter. You specify either the WebLogic domain home, or the Oracle instance.
- Most of the WLST logging commands require that you are running in the domainRuntime tree. For example, to connect and to run in the domainRuntime tree, use the following WLST commands:

```
connect('username', 'password', 'localhost:port_number')
domainRuntime()
```

> **See Also:** "Logging Custom WLST Commands" in the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*

12.3.1 Viewing Log Files and Their Messages

You can view the log files using Fusion Middleware Control or WLST commands, as described in the following topics:

- Viewing Log Files and Their Messages Using Fusion Middleware Control
- Viewing Log Files and Their Messages Using WLST

12.3.1.1 Viewing Log Files and Their Messages Using Fusion Middleware Control

You can view the messages for all of the entities in a domain, an Oracle WebLogic Server, a component, or an application.

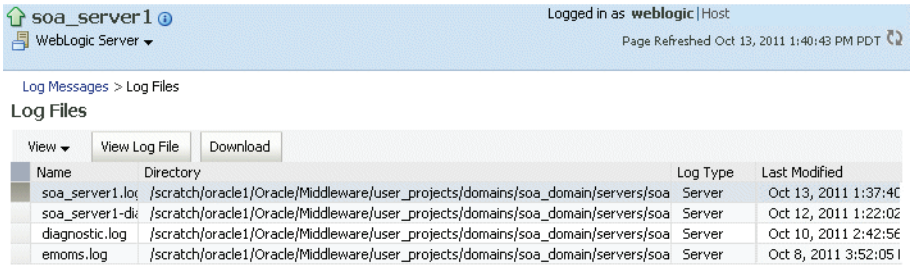
For example, to view the log files and their messages for a Managed Server:

1. From the navigation pane, expand the farm, then **WebLogic Domain**, and then the domain. Right-click the Managed Server name and choose **Logs,** then **View Log Messages.**

   The Log Messages page is displayed.

2. Expand **Selected Targets** and in the row for a particular component or application, click **Target Log Files.**

   The Log Files page is displayed. On this page, you can see a list of log files related to the Managed Server, as shown in the following figure:

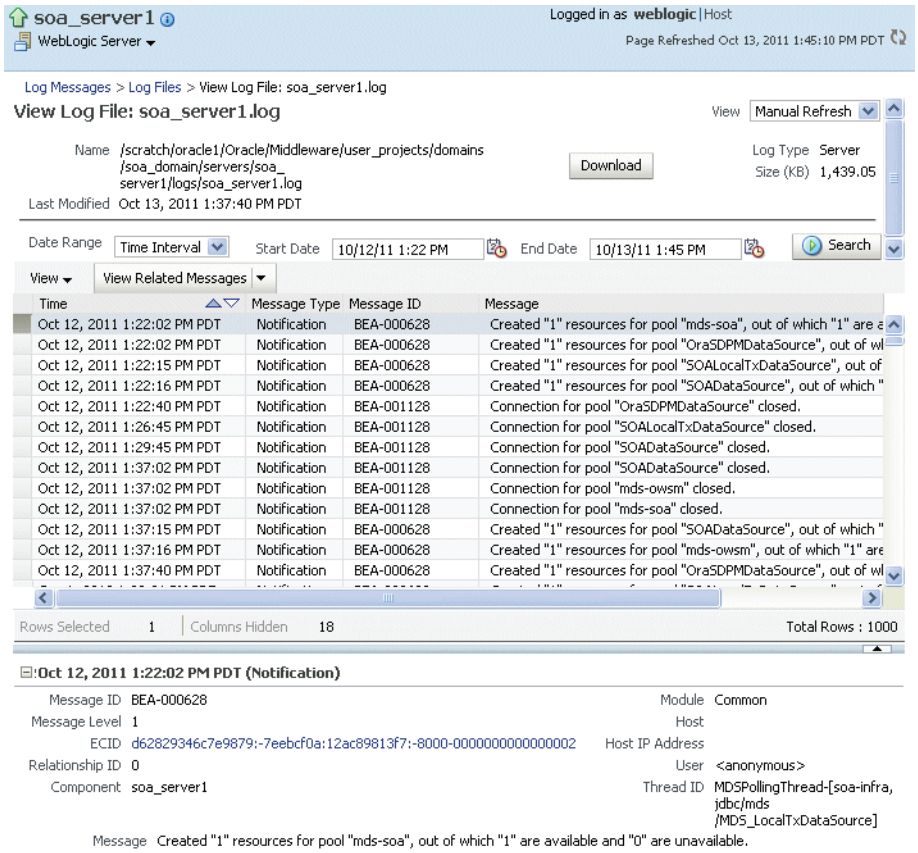


3. Select a file and click **View Log File.**

   The View Log Files page is displayed. On this page, you can view the list of messages.

4. To view the details of a message, select the message.

   The details are displayed in the pane below the listing, as shown in the following figure:



By default, the messages are sorted by time, in ascending order. You can sort the messages by the any of the columns, such as Message Type, by clicking the column name.

5. To view messages that are related by time or ECID, click **View Related Messages** and select **by Time** or **by ECID (Execution Context ID).**

   The Related Messages page is displayed.

12.3.1.2 Viewing Log Files and Their Messages Using WLST

You can list the log files for an Oracle WebLogic Server domain, a server, an Oracle instance, or component using the WLST `listLogs` command.

You can use this command while connected or disconnected. While connected, the default target is the Oracle WebLogic Server domain.

To list the log files, first use the `domainRuntime` command as described in Section 12.3. The following describes how to list and view log files:

- To list all of the log files for the Oracle WebLogic Server soa_server1, use the following command:

```
listLogs(target='server_soa')

file://host/scratch/Oracle/Middleware/user_projects/domains/SOA_
domain/servers/soa_server1/logs/soa_server1.log
2010-09-17 16:40:45                    4.9M soa_server1.log00001
2010-09-17 18:35:35                    4.9M soa_server1.log00002
2010-09-17 20:30:25                    4.9M soa_server1.log00003
...
file://host/scratch/Oracle/Middleware/user_projects/domains/SOA_
```

```
domain/servers/soa_server1/logs/soa_server1-diagnostic.log
2010-09-22 13:53:32                    10M soa_server1-diagnostic-22.log
2010-09-22 19:18:32                    10M soa_server1-diagnostic-23.log
2010-09-23 00:42:32                    10M soa_server1-diagnostic-24.log
2010-09-23 06:07:32                    10M soa_server1-diagnostic-25.log
2010-09-23 11:31:32                    10M soa_server1-diagnostic-26.log
2010-09-23 16:56:32                    10M soa_server1-diagnostic-27.log
2010-09-23 22:20:32                    10M soa_server1-diagnostic-28.log
2010-09-24 03:45:32                    10M soa_server1-diagnostic-29.log
2010-09-24 09:11:32                    10M soa_server1-diagnostic-30.log
2010-09-24 14:08:32                   9.2M soa_server1-diagnostic.log
...
```

- To list the logs for the Oracle HTTP Server ohs1 in the Oracle instance asinst_1, use the following command:

  ```
  listLogs(target='opmn:asinst_1/ohs1')
  ```

- To list the logs while disconnected, you must specify the `oracleInstance` parameter, passing it the path of either the Oracle WebLogic Server domain or the Oracle instance home for the system component. For example, to list the log files for the Managed Server soa_server1:

  ```
  listLogs(oracleInstance='/scratch/Oracle/Middleware/user_projects/domains/SOA_
  domain',
            target='soa_server1')
  ```

- To view the diagnostic messages in log files, use the WLST `displayLogs` command. This command works when you are either connected or disconnected.

  For example, to view the messages generated in the last 10 minutes in the log files for the Oracle WebLogic Server domain, use the following command:

  ```
  displayLogs(last=10)

  [2010-09-05T08:05:29.652-07:00] [soa_server1] [NOTIFICATION] [BEA-000628]
   [Common] [host: hostname] [nwaddr: 10.229.149.27] [tid:
   [ACTIVE].ExecuteThread: '10' for queue: 'weblogic.kernel.Default
   (self-tuning)'] [userId: <WLS Kernel>] [TARGET: /SOA_domain/soa_server1]
   [LOG_FILE: /scratch//Oracle/Middleware/user_projects/domains/SOA_
  domain/servers/soa_server1/logs/soa_server1.log] Created "1" resources for
   pool "SOADataSource", out of which "1" are available and "0" are unavailable.
  [2010-09-05T08:05:29.673-07:00] [soa_server1] [NOTIFICATION] [BEA-000628]
   [Common] [host: hostname] [nwaddr: 10.229.149.27] [tid:
  oracle.integration.platform.blocks.executor.WorkManagerExecutor$1@17f5105]
   [userId: <anonymous>] [TARGET: /SOA_domain/soa_server1] [LOG_FILE:
   /scratch/Oracle/Middleware/user_projects/domains/SOA
  _domain/servers/soa_server1/logs/soa_server1.log] Created "1" resources for
   pool "SOADataSource", out of which "1" are available and "0" are unavailable.
  [2010-09-05T08:05:30.448-07:00] [soa_server1] [NOTIFICATION] [BEA-001128]
   [JDBC] [host: hostname] [nwaddr: 10.229.149.27] [tid:
   oracle.integration.platform.blocks.executor.WorkManagerExecutor$1@17f5105]
   [userId: <anonymous>] [TARGET: /SOA_domain/soa_server1] [LOG_FILE:
   /scratch/Oracle/Middleware/user_projects/domains/SOA
  _domain/servers/soa_server1/logs/soa_server1.log] Connection for pool
   "SOADataSource" closed.
  ```

  The previous command returns the messages sorted by time, in ascending order.

- To display the log files for the Oracle HTTP Server ohs1 in the Oracle instance asinst_1, use the following command:

```
displayLogs(target='opmn:asinst_1/ohs1')
```

You can search the messages by specifying particular criteria and sort the output, as described in Section 12.3.2.

> **See Also:** "Logging Custom WLST Commands" in the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference* for more information about the `listLogs` and `displayLogs` commands

12.3.2 Searching Log Files

You can search for diagnostic messages by time, type of message, and certain log file attributes by using Fusion Middleware Control or WLST commands, as described in the following topics:

- Searching Log Files Using Fusion Middleware Control
- Searching Log Files Using WLST

12.3.2.1 Searching Log Files Using Fusion Middleware Control

You can search for diagnostic messages using standard and supplemental ODL attributes using the Log Messages page of Fusion Middleware Control. By default, this page shows a summary of the logged issues for the last hour.

You can modify the search criteria to identify messages of relevance. You can view the search results in different modes, allowing ease of navigation through large amounts of data.

The following sections describe how to search log files:

- Searching Log Files: Basic Searches
- Searching Log Files: Advanced Searches

12.3.2.1.1 Searching Log Files: Basic Searches This section describes how to perform basic searches for log messages.

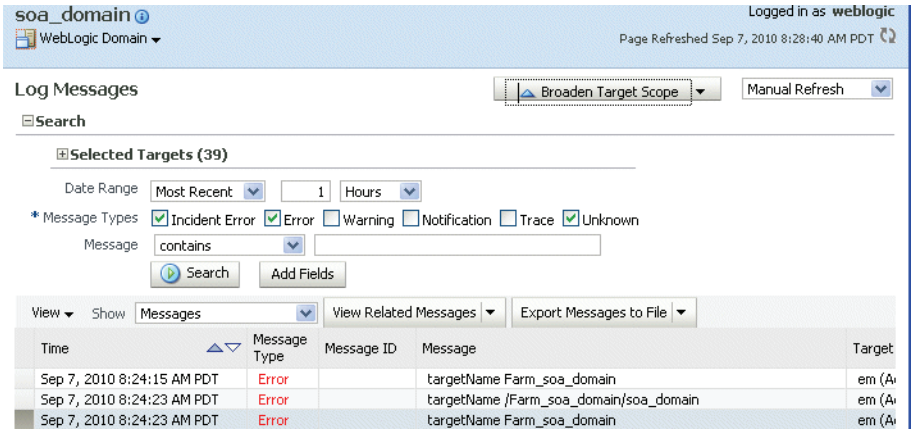
You can search for all of the messages for all of the entities in a domain, an Oracle WebLogic Server, a component, or an application.

For example, to search for messages for a domain:

1. From the WebLogic Domain menu, choose **Logs,** then **View Log Messages.**

   To search for messages for a component or application, select the component or application. Then choose **Logs,** then **View Log Messages** from that target's menu.

   The Log Messages page displays a Search section and a table that shows a summary of the messages for the last hour, as shown in the following figure:



2. In the Date Range section, you can select either:
   - **Most Recent:** If you select this option, select a time, such as 3 hours. The default is 1 hour.
   - **Time Interval:** If you select this option, select the calendar icon for **Start Date.** Select a date and time. Then, select the calendar icon for **End Date.** Select a date and time.
3. In the Message Types section, select one or more of the message types. The types are described in Table 12–3.
4. You can specify more search criteria, as described in Section 12.3.2.1.2.
5. Click **Search.**
6. To help identify messages of relevance, in the table, for **Show,** select one of the following modes:
   - **Messages:** Shows the matching messages.

     To see the details of a particular message, click the message. The details are displayed below the table of messages.

     To view related messages, select a message, then click **View Related Messages** and select **by Time** or **by ECID (Execution Context ID).**
   - **Group by Message Type:** Summarizes the matching messages by grouping them based on message type at the target level. This is the default mode.

     To see the messages, click the count in one of the message type columns. The Messages by Message Type page is displayed. To see the details of a particular message, click the message. The details are displayed below the table of messages.
   - **Group by Message ID:** Summarizes the matching messages by grouping them based on message ID, message type, and module IDs at the target level.

     To see the associated messages, click the count in the **Occurrences** column. The Messages by Message ID page is displayed. To see the details of a particular message, click the message. The details are displayed below the table of messages.

12.3.2.1.2 Searching Log Files: Advanced Searches

This section describes some of the advanced search mechanisms you can use.

You can refine your search criteria using the following controls in the Log Messages page:

- **Message:** You can select an operator, such as **contains** and then enter a value to be matched.
- **Add Fields:** Click this to specify additional criteria, such as Host, which lets you narrow the search to particular hosts. Then click **Add.**

  For each field you add, select an operator, such as **contains** and then enter a value to be matched.
- **Broaden Target Scope:** Click this to expand the search to logs associated with all members of the parent of the target. For example, if you are searching an application's logs, you can expand the search to contain the Managed Server to which the application is deployed.
- **Selected Targets:** Expand this to see the targets that are participating in the search. To add targets, click **Add** and provide information in the dialog box. To remove targets, select the target and click **Remove.**

12.3.2.2 Searching Log Files Using WLST

You can search the log files using the WLST `displayLogs` command. You can narrow your search by specifying criteria, such as time, component ID, message type, or ECID. For example:

- To search for error messages generated in the last 5 minutes, for the Oracle HTTP Server ohs1, use the following command:

```
displayLogs(target='opmn:asinst_1/ohs1', last=5)
```

- To search for error messages generated in the last 10 minutes for the Managed Server soa_server1, use the following command:

```
displayLogs(oracleInstance='/scratch/Oracle/Middleware/user_
projects/domains/soa_domain', target='soa_server1', last=10)
```

You can narrow your search by using the `query` parameter and specifying criteria, such as component ID, message type, or ECID. In the `query` clause, you can specify a query expression with any of the attributes listed in Table 12–1. Some of the criteria you can use are:

- Types of messages. For example, to search for ERROR and INCIDENT_ERROR messages for the Managed Server soa_server1, use the following command:

```
displayLogs(oracleInstance='/scratch/Oracle/Middleware/user_
projects/domains/soa_domain',
          target='soa_server1',
          query='MSG_TYPE eq ERROR or MSG_TYPE eq INCIDENT_ERROR')
```

- A particular ECID. For example, to search for error messages with a particular ECID (0000I3K7DCnAhKB5JZ4Eyf19wAgN000001,0') for the Managed Server soa_server1, use the following command:

```
displayLogs(oracleInstance='/scratch/Oracle/Middleware/user_
projects/domains/soa_domain',
         target='soa_server1',
         query='ecid eq 0000I3K7DCnAhKB5JZ4Eyf19wAgN000001,0')
```

- Component type. For example, to search for messages from Oracle HTTP Server instances, use the following query:

```
displayLogs(query='COMPONENT_ID eq ohs')
```

- Range of time. To search for error messages that occurred within a specified range of time, you specify the attribute TSTZ_ORIGINATING with both `from` and `to` operators, using the following format:

  ```
  displayLogs(query='TSTZ_ORIGINATING from start_time and
                  TSTZ_ORIGINATING to end_time')
  ```

  You specify the date using the following ISO 8601 time format:

  ```
  YYYY-MM-DDThh:mm:ss-hh:mm_offset_from_UTC
  ```

  For example:

  ```
  2010-09-30T12:00:00:0000-08:00
  ```

  For example, to display the error message from between 8:00 a.m. and 11 a.m. on April 17, 2010, use the following command:

  ```
  displayLogs(query='TSTZ_ORIGINATING from 2010-04-17T08:00:00-07:00
             and TSTZ_ORIGINATING to 2010-04-17T11:00:00-07:00')
  ```

- Group messages. To display a count of messages, grouped by specific attributes, use the `groupBy` parameter to the WLST command `displayLogs`. For example, to display the count of WARNING messages by component, use the following command:

  ```
  displayLogs(groupBy=['COMPONENT_ID'], query='MSG_TYPE eq WARNING')
  ```

12.3.3 Downloading Log Files

You can download messages using Fusion Middleware Control or WLST commands, as described in the following topics:

- Downloading Log Files Using Fusion Middleware Control
- Downloading Log Files Using WLST

12.3.3.1 Downloading Log Files Using Fusion Middleware Control

You can download the log messages to a file. You can download either the matching messages from a search or the messages in a particular log file.

To download the matching messages from a search to a file using Fusion Middleware Control:

1. From the navigation pane, expand the farm and select the target, for example by clicking on the domain.
2. From the dynamic target menu, such as the WebLogic Domain menu, choose **Logs,** then **View Log Messages.**

   The Log Messages page is displayed.
3. Search for particular types of messages as described in Section 12.3.2.1.
4. Select a file type by clicking **Export Messages to File** and select one of the following:
   - **As Oracle Diagnostic Log Text (.txt)**
   - **As Oracle Diagnostic Log Text (.xml)**
   - **As Comma-Separated List (.csv)**

   An Opening dialog box is displayed.

5. Select either **Open With** or **Save to Disk.** Click **OK.**

To export specific types of messages or messages with a particular Message ID to a file:

1. From the navigation pane, expand the farm, then **WebLogic Domain,** and then the domain. Select a Managed Server.
2. From the dynamic target menu, choose **Logs,** then **View Log Messages.**

   The Log Messages page is displayed.
3. Search for particular types of messages as described in Section 12.3.2.1.
4. For **Show,** select **Group by Message Type** or **Group by Message ID.**
5. To download the messages into a file, if you selected Group by Message Type, select the link in one of the columns that lists the number of messages, such as the Errors column. If you selected Group by Message ID, select one of the links in the Occurrences column.

   The Messages by Message Type page or Message by Message ID is displayed.
6. Select a file type by clicking the arrow near **Export All to File.**

   You can select one of the following:

   - **As Oracle Diagnostic Log Text (.txt)**
   - **As Oracle Diagnostic Log Text (.xml)**
   - **As Comma-Separated List (.csv)**

   An Opening dialog box is displayed.
7. Select either **Open With** or **Save to Disk.** Click **OK.**

To download the log files for a specific component using Fusion Middleware Control:

1. From the navigation pane, expand the farm. For system components, expand the installation type, such as **Web Tier** and select the component. For Java components, expand the farm, then the component type, and then select the component.
2. From the dynamic target menu, choose **Logs,** then **View Log Messages.**

   The Log Messages page is displayed.
3. Click **Target Log Files.**

   The Log Files page is displayed. On this page, you can see a list of log files related to the component or application.
4. Select a log file and click **Download.**
5. An Opening dialog box is displayed.
6. Select either **Open With** or **Save to Disk.** Click **OK.**

12.3.3.2 Downloading Log Files Using WLST

You can download log files using the WLST `displayLogs` command and redirecting the output to a file. For example:

```
displayLogs(type=['ERROR','INCIDENT_ERROR'], exportFile='/scratch/tmp/download_
log.txt')
```

The messages are written to the file download_log.txt in the specified directory. By default, they are written to standard output.

12.4 Configuring Settings for Log Files

You can change the log settings of Managed Servers and Java components using Fusion Middleware Control or WLST.

> **Note:** For many system components, which are listed in Section 3.5.2, you cannot configure settings for log files using Fusion Middleware Control. For information about how to configure options for log files for system components, see the Administrator's Guide for the component.

For Java components, you can configure the names and locations of log files, the size of the log files, the level of information written to the log files, the format, and the Locale encoding, as described in the following topics:

- Changing Log File Locations
- Configuring Log File Rotation
- Setting the Level of Information Written to Log Files
- Specifying the Log File Format
- Specifying the Log File Locale

Note the following about using the WLST commands to configure log settings:

- To use the custom WLST logging commands, you must invoke the WLST script from the Oracle Common home. See Section 3.5.1.1 for more information.
- The configuration commands, such as `setLogLevel`, only work in connected mode. That is, you must connect to a running WebLogic Server instance before you invoke the commands.

  The configuration commands are supported for Java components that run within a WebLogic Server, but are not supported for Oracle WebLogic Server. The configuration commands are not supported for system components.
- Most of the WLST logging commands require that you are running in the domainRuntime tree. For example, to connect and to run in the domainRuntime tree, use the following commands:

```
connect('username', 'password', 'localhost:port_number')
domainRuntime()
```

- The `listLoggers`, `getLogLevel`, and `setLogLevel` commands work in `config` and `runtime` mode. In `config` mode the commands work on loggers that are defined in the configuration file. In `runtime` mode, the commands work directly with loggers that are defined in the server JVM. By default, the `setLogLevel` command sets the level on the run-time logger and updates the logger definition in the configuration file. By default, the `listLoggers` and `getLogLevel` commands return run-time loggers.

> **See Also:** "Logging Custom WLST Commands" in the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*

12.4.1 Changing Log File Locations

You can change the name and location of log files by using Fusion Middleware Control or WLST commands, as described in the following topics:

- Changing Log File Locations Using Fusion Middleware Control
- Changing Log File Locations Using WLST

12.4.1.1 Changing Log File Locations Using Fusion Middleware Control

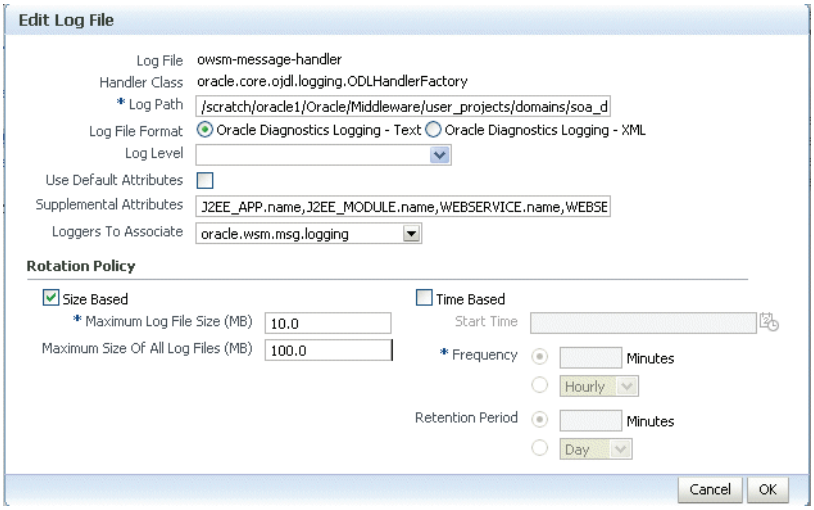
To change the name and location of a component log file using Fusion Middleware Control:

1. From the navigation pane, select the component.
2. From the dynamic target menu, choose **Logs,** then **Log Configuration.**

   The Log Configuration page is displayed.

   Note that the navigation may be different for some components. For example, for Oracle HTTP Server, you choose **Administration,** then **Log Configuration.**
3. Select the Log Files tab.
4. In the table, select the log handler and click **Edit Configuration.**

   The Edit Log File dialog box is displayed, as shown in the following figure:



5. For **Log Path,** enter a new path.
6. Click **OK.**
7. In the confirmation window, click **Close.**

Note that if you change the location of Oracle HTTP Server log files, the location of the access_log and ohs*n*.log files are changed, but the location of console~OHS~1.log is not changed.

12.4.1.2 Changing Log File Locations Using WLST

To change the log file location using WLST, use the `configureLogHandler` command. For example, to change the path of the logger named odl-handler, use the following command:

```
configureLogHandler(name='odl-handler', path='/scratch/Oracle/logs')
```

12.4.2 Configuring Log File Rotation

An **ODL log** is a set of log files that includes the current ODL log file and zero or more **ODL Archives (segment files)** that contain older messages. As the log file grows, new

information is added to the end of the log file, *server_name*-diagnostic.log. When the log file reaches the rotation point, it is renamed and a new log file, *server_name*-diagnostic.log is created. You specify the rotation point, by specifying the maximum ODL segment size or the rotation time and rotation frequency.

Segment files are created when the ODL log file *server_name*-diagnostic.log reaches the rotation point. That is, the *server_name*-diagnostic.log is renamed to *server_name*-diagnostic-*n*.log, where *n* is an integer, and a new *server_name*-diagnostic.log file is created when the component generates new diagnostic messages.

To limit the size of the ODL log, you can specify:

- The maximum size of the logging directory. Whenever the sum of the sizes of all of the files in the directory reaches the maximum, the oldest archive is deleted to keep the total size under the specified limit.

  By default, the log files are rotated when they reach 10 MB. The maximum size of all log files for a particular component is 100 MB.

- The maximum size of the log file. You specify that a new log file be created when a specific time or frequency is reached.

> **Note:** After you change the log file rotation, the configuration is reloaded dynamically. It may take 1 or 2 seconds to reload the configuration.

The following topics describe how to change the rotation:

- Specifying Log File Rotation Using Fusion Middleware Control
- Specifying Log File Rotation Using WLST

12.4.2.1 Specifying Log File Rotation Using Fusion Middleware Control

To configure log file rotation using Fusion Middleware Control:

1. From the navigation pane, select the component.
2. From the dynamic target menu, choose **Logs,** then **Log Configuration.**

   The Log Configuration page is displayed.

   Note that the navigation may be different for some components. For example, for Oracle HTTP Server, you choose **Administration,** then **Log Configuration.**
3. Select the Log Files tab.
4. In the table, select the logger and click **Edit Configuration.**

   The Edit Log File dialog box is displayed.
5. In the Rotation Policy section, you can select one of the following:
   - **Size Based:** If you select this, enter the following:
     - For **Maximum Log File Size,** enter the size in MB, for example, 15.
     - For **Maximum Size of All Log Files,** enter the size in MB, for example, 150.
   - **Time Based:** If you select this, enter the following:

- For **Start Time,** click the calendar and select the date and time when you want the rotation to start. For example, select September 8, 2010 6:00 AM.
- For **Frequency,** you can select **Minutes** and enter the number of minutes, or you can select **Hourly, Daily,** or **Weekly.**
- For **Retention Period,** you can specify how long the log files are kept. You can select **Minutes** and enter the number of minutes, or you can specify **Day, Week**, **Month,** or **Year.**

  Specifying a shorter period means that you use less disk space, but are not able to retrieve older information.

6. Click **OK.**
7. In the confirmation window, click **Close.**

12.4.2.2 Specifying Log File Rotation Using WLST

To specify log file rotation using WLST, use the `configureLogHandler` command. You can specify size-based rotation or time-based rotation.

For example, to specify that the log files rotate daily and that they are retained for a week, use the following command:

```
configureLogHandler(name='odl-handler', rotationFrequency='daily',
                    retentionPeriod='week')
```

To specify that the size of a log file does not exceed 5 MB and rotates when it reaches that size, use the following command:

```
configureLogHandler(name='odl-handler', maxFileSize='5M')
```

12.4.3 Setting the Level of Information Written to Log Files

You can configure the amount and type of information written to log files by specifying the message type and level. For each message type, possible values for the message level are from 1 (lowest severity) through 32 (highest severity). Some components support only some of the levels for each message type. See the Administration Guide for your component for more information. Generally, you need to specify only the type; you do not need to specify the level.

When you specify the type, Oracle Fusion Middleware returns all messages of that type, as well as the messages that have a higher severity. For example, if you set the message type to WARNING, Oracle Fusion Middleware also returns messages of type INCIDENT_ERROR and ERROR.

Table 12–3 describes the message types and the most common levels for each type.

Table 12–3 Diagnostic Message Types and Level

Message Type	Level	Description
INCIDENT_ERROR	1	A serious problem that may be caused by a bug in the product and that should be reported to Oracle Support. Examples are errors from which you cannot recover or serious problems.
ERROR	1	A serious problem that requires immediate attention from the administrator and is not caused by a bug in the product. An example is if Oracle Fusion Middleware cannot process a log file, but you can correct the problem by fixing the permissions on the document.

Table 12–3 (Cont.) Diagnostic Message Types and Level

Message Type	Level	Description
WARNING	1	A potential problem that should be reviewed by the administrator. Examples are invalid parameter values or a specified file does not exist.
NOTIFICATION	1	A major lifecycle event such as the activation or deactivation of a primary sub-component or feature. This is the default level for NOTIFICATION.
NOTIFICATION	16	A finer level of granularity for reporting normal events.
TRACE	1	Trace or debug information for events that are meaningful to administrators, such as public API entry or exit points.
TRACE	16	Detailed trace or debug information that can help Oracle Support diagnose problems with a particular subsystem.
TRACE	32	Very detailed trace or debug information that can help Oracle Support diagnose problems with a particular subsystem.

The default is NOTIFICATION, level 1.

The INCIDENT_ERROR, ERROR, WARNING, and NOTIFICATION with level 1 have no performance impact. For other types and levels, note the following:

- NOTIFICATION, with level 16: Minimal performance impact.
- TRACE, with level 1: Small performance impact. You can enable this level occasionally on a production environment to debug problems.
- TRACE, with level 16: High performance impact. This level should not be enabled on a production environment, except on special situations to debug problems.
- TRACE, with level 32: Very high performance impact. This level should not be enabled in a production environment. It is intended to be used to debug the product on a test or development environment.

Table 12–4 shows the log level mappings among ODL format, Oracle WebLogic Server, and Java.

Table 12–4 Mapping of Log Levels Among ODL, Oracle WebLogic Server, and Java

ODL	WebLogic Server	Java
OFF	OFF	2147483647 - OFF
INCIDENT_ERROR:1	(EMERGENCY)	1100
INCIDENT_ERROR:4	EMERGENCY	1090
INCIDENT_ERROR:14	ALERT	1060
INCIDENT_ERROR:24	CRITICAL	1030
ERROR:1	(ERROR)	1000 - SEVERE
ERROR:7	ERROR	980
WARNING:1	WARNING	900 - WARNING
WARNING:7	NOTICE	880
NOTIFICATION:1	INFO	800 - INFO

Table 12–4 (Cont.) Mapping of Log Levels Among ODL, Oracle WebLogic Server, and

ODL	WebLogic Server	Java
NOTIFICATION:16	(DEBUG)	700 - CONFIG
TRACE:1	(DEBUG)	500 - FINE
TRACE:1	DEBUG	495
TRACE:16	(TRACE)	400 - FINER
TRACE:32	(TRACE)	300 - FINEST
TRACE:32	TRACE	295

You can configure the message levels using Fusion Middleware Control or WLST commands, as described in the following topics:

- Configuring Message Levels Using Fusion Middleware Control
- Configuring Message Levels Using WLST

12.4.3.1 Configuring Message Levels Using Fusion Middleware Control

You can set the message level for a particular log file or for loggers.

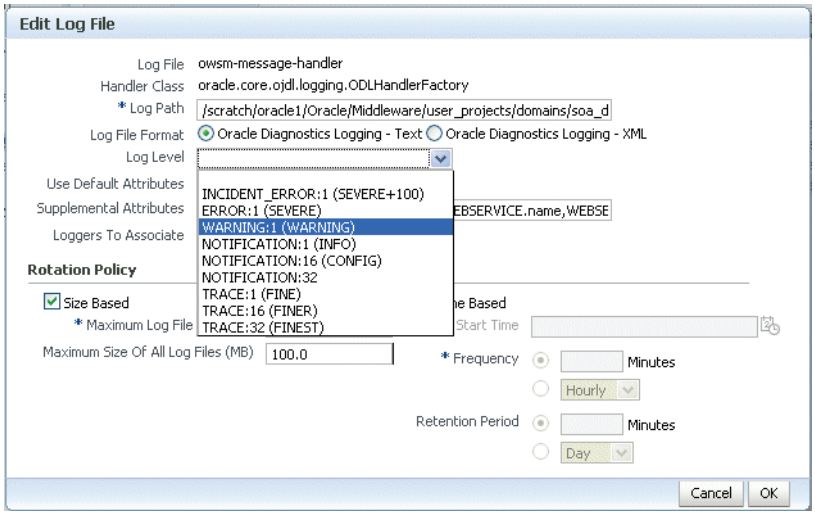
To set the message level for a component log file:

1. From the navigation pane, select the component.
2. From the dynamic target menu, choose **Logs,** then **Log Configuration.**

   The Log Configuration page is displayed.

   Note that the navigation may be different for some components. For example, for Oracle HTTP Server, you choose **Administration,** then **Log Configuration.**
3. Select the Log Files tab.
4. In the table, select the log file and click **Edit Configuration.**

   The Edit Log File dialog box is displayed, as shown in the following figure:



5. For **Log Level,** select the logging level. For example, select **WARNING:1 (WARNING).**
6. Click **OK.**

7. In the confirmation window, click **Close.**

To set the message level for one or more loggers for a component:

1. From the navigation pane, select the component.
2. From the dynamic target menu, choose **Logs,** then **Log Configuration.**

   The Log Configuration page is displayed.

   Note that the navigation may be different for some components. For example, for Oracle HTTP Server, you choose **Administration,** then **Log Configuration.**
3. Select the **Log Levels** tab.
4. For **View,** select **Runtime Loggers** or **Loggers with Persistent Log Level State.**

   Run-time loggers are loggers that are currently active. Persistent loggers are loggers that are saved in a configuration file and the log levels of these loggers are persistent across component restarts. A run-time logger can also be a persistent logger, but not all run-time loggers are persistent loggers.
5. In the table, to specify the same level for all loggers, select the logging level for the top-level logger. Then, for child loggers that do not specify that the logging level is inherited from the parent, specify **Inherited from Parent.** For most situations, that is sufficient.

   However, if you need to specify the level for a particular logger, expand the logger and then, for the logger that you want to modify, select the logging level. For example, for the logger oracle.wsm.management.logging, select **WARNING:1 (WARNING).**
6. Click **Apply.**

12.4.3.2 Configuring Message Levels Using WLST

To set the message level with WLST, you use the `setLoglevel` command. To get the current message level, you use the `getLogLevel` command. You must be connected to WebLogic Server before you use the configuration commands.

You can view the log level for a logger for an Oracle WebLogic Server. For example, to view the log level of the Oracle WebLogic Server soa_server1, use the following command:

```
getLogLevel(logger='oracle',  target='soa_server1')
NOTIFICATION:1
```

You can set the log level for a particular logger. The following example sets the message type to WARNING for the logger oracle.soa:

```
setLogLevel(target='soa_server1', logger='oracle.soa', level='WARNING')
```

To get a list of loggers for the Oracle WebLogic Server soa_server1, use the listLoggers command:

```
listLoggers(target='soa_server1')
.
.
.
oracle.soa                                | WARNING:1
oracle.soa.adapter                        | <Inherited>
orac                                      | <Inherited>
oracle.soa.b2b.apptransport               | <Inherited>
oracle.soa.b2b.engine                     | <Inherited>
```

```
oracle.soa.b2b.repository                | <Inherited>
oracle.soa.b2b.transport                 | <Inherited>
oracle.soa.b2b.ui                        | <Inherited>
.
.
.
```

You can also filter logger names using the pattern parameter and a regular expression. For example, to return all loggers that begin with `oracle` in the Oracle WebLogic Server soa_server1, use the following command:

```
listLoggers(target='soa_server1', pattern='oracle.*')
oracle                                   | NOTIFICATION:1
oracle.adapter                           | <Inherited>
oracle.adapter.jms.logger                | <Inherited>
oracle.adf                               | <Inherited>
```

12.4.4 Specifying the Log File Format

By default, information is written to log files in ODL text format. You can change the format to ODL XML format using Fusion Middleware Control or WLST commands, as described in the following topics:

- Specifying the Log File Format Using Fusion Middleware Control
- Specifying the Log File Format Using WLST

12.4.4.1 Specifying the Log File Format Using Fusion Middleware Control

To change the format using Fusion Middleware Control:

1. From the navigation pane, select the component.
2. From the dynamic target menu, choose **Logs,** then **Log Configuration.**

   The Log Configuration page is displayed.

   Note that the navigation may be different for some components. For example, for Oracle HTTP Server, you choose **Administration,** then **Log Configuration.**
3. Select the Log Files tab.
4. In the table, select the log file and click **Edit Configuration.**

   The Edit Log File dialog box is displayed.
5. For Log File Format, select **Oracle Diagnostics Logging - Text** or **Oracle Diagnostics Logging - XML.**
6. Click **OK.**
7. In the confirmation window, click **Close.**

12.4.4.2 Specifying the Log File Format Using WLST

To specify the log file format using WLST, you use the `configureLogHandler` command, with the `format` parameter and specify either ODL-Text or ODL-XML. ODL-Text is the default.

For example, to specify ODL-XML format, use the following command:

```
configureLogHandler(name='odl-handler', format='ODL-XML')
```

12.4.5 Specifying the Log File Locale

The language and data formats used in the log files are determined by the default locale of the server Java Virtual Machine (JVM). You can change them using the Language and Regional Options applet in Control Panel on Windows or the LANG and LC_ALL environment variables on a UNIX platform.

The character encoding of log files is determined by the server JVM's default character encoding or an optional configuration setting. You should choose an encoding that supports all languages used by the users, or the log file may be corrupted. By default, the log is in the server JVM's default character encoding. If you change the encoding, delete or rename old log files to prevent them from being damaged by the new logs appended in a different encoding.

For support of any language, Oracle recommends that you use Unicode UTF-8 encoding. On a UNIX operating system, setting the LANG and LC_All environment variables to a locale with the UTF-8 character set enables UTF-8 logging (for example, `en_US.UTF-8` for the US locale in UTF-8 encoding).

You can specify the log file locale using WLST commands or by editing a file, as described in the following topics:

- Specifying the Log File Encoding Using WLST
- Specifying the Log File Encoding in logging.xml

12.4.5.1 Specifying the Log File Encoding Using WLST

To specify the log file encoding using WLST, use the `configureLogHandler` command. You can use the encoding parameter to specify the character set encoding.

For example, to specify UTF-8, use the following command:

```
configureLogHandler(name="odl-handler", encoding="UTF-8")
```

12.4.5.2 Specifying the Log File Encoding in logging.xml

To specify the log file encoding in the logging.xml file, use an optional encoding property to specify the character set encoding.

The logging.xml file is located in the following directory:

DOMAIN_HOME`/config/fmwconfig/servers/`*server_name*`/`

For example, to specify UTF-8, add the following encoding property in the log_handler element:

```
<property name='encoding' value='UTF-8'/>
```

12.5 Correlating Messages Across Log Files and Components

Oracle Fusion Middleware components provide **message correlation** information for diagnostic messages. Message correlation information helps those viewing diagnostic messages to determine relationships between messages across components. Each diagnostic message contains an **Execution Context ID (ECID)** and a **Relationship ID (RID)**:

- An ECID is a globally unique identifier associated with the execution of a particular request. An ECID is generated when the request is first processed.
- A RID distinguishes the work done in one thread on one process, from work done by any other threads on this and other processes on behalf of the same request.

The ECID and RID help you to use log file entries to correlate messages from one application or across Oracle Fusion Middleware components. By searching for related messages using the message correlation information, multiple messages can be examined and the component that first generates a problem can be identified (this technique is called **first-fault component isolation**). Message correlation data can help establish a clear path for a diagnostic message across components, within which errors and related behavior can be understood.

You can use the ECID and RID to track requests as they move through Oracle Fusion Middleware.

The following shows an example of an ECID:

```
0000I3K7DCnAhKB5JZ4Eyf19wAgN000001,0
```

The RID is one or more numbers separated by a colon (:). The first RID created for a request is 0. Each time work is passed from a thread that has an ECID associated with it to another thread or process, a new RID is generated that encodes the relationship to its creator. That is, a new generation is created. Each shift in generation is represented by a colon and another number. For example, the seventh child of the third child of the creator of the request is:

```
0:3:7
```

You can view all the messages with the same ECID using the WLST `displayLogs` command. The following example searches for the ECID in the domain:

```
displayLogs(ecid='0000Hl9TwKUCslT6uBi8UH18lkWX000002')
```

You can also search for the ECID in a WebLogic Server instance, or a system component, by specifying it in the target option.

You can search for messages with a particular ECID on the Log Messages page in Fusion Middleware Control:

1. From the WebLogic Domain menu, choose **Logs,** then **View Log Messages.**

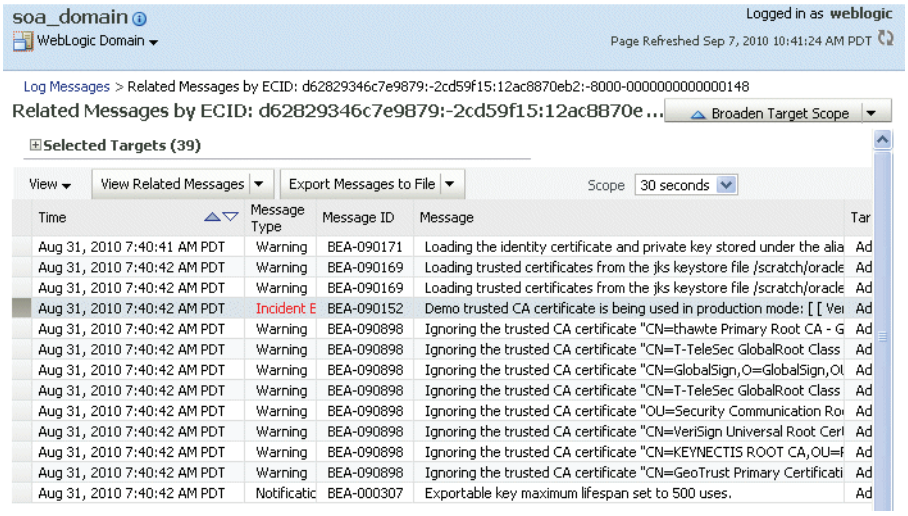
   To search for messages for a component or application, select the component or application and then choose **Logs,** then **View Log Messages** from that target's menu.

2. Specify search criteria, as described in Section 12.3.2.1.2.

3. Click **Search.**

4. Select a message, then click **View Related Messages** and select **by ECID (Execution Context ID).**

   The messages with the same ECID are displayed, as shown in the following figure:



5. Trace the ECID to the earliest message. (You may need to increase the scope to view the first message with the ECID.)

12.6 Configuring Tracing

Sometimes you need more information to troubleshoot a problem than it is usually recorded in the logs. One way to achieve that is to increase the level of messages logged by one or more components. For example, you can set the logging level to TRACE:1 or TRACE:32, as described in Section 12.4.3, which results in more detailed messages being written to the log files. This is referred to as **tracing**.

However, this can often result in a large amount of log messages being written to the log files. Oracle Fusion Middleware provides the following mechanisms to fine-tune which messages are traced:

- QuickTrace, which provides fine-grained logging to memory
- Selective Trace, which provides fine-grained logging for a specific user or other properties of a request

The following topics provide information about how to use these mechanisms:

- Configuring and Using QuickTrace
- Configuring and Using Selective Tracing

12.6.1 Configuring and Using QuickTrace

With QuickTrace, you can trace messages from specific loggers and store the messages in memory. Because QuickTrace logs the messages to memory, it avoids the cost of formatting, string manipulations, and input/output operations. As as result, you can enable fine-level application logging for specific loggers without performance overhead.

By default, QuickTrace writes the messages to one common buffer. However, you can specify that messages for particular users are written to separate buffers.

You can save the messages that are in memory to a file by invoking the QuickTrace Dump in Fusion Middleware Control as described in Section 12.6.1.1.2 or by using the WLST, as described in Section 12.6.1.2.2.

To enable QuickTrace, you create a QuickTrace handler and associate a logger with it. You can specify the buffer size, as well as other attributes, for the handler. Then, you set the level of the amount and type of information to be written by the loggers to memory.

The following topics describe how to enable and use QuickTrace:

- Configuring QuickTrace Using Fusion Middleware Control
- Configuring QuickTrace Using WLST

12.6.1.1 Configuring QuickTrace Using Fusion Middleware Control

You can configure and use QuickTrace using Fusion Middleware Control, as described in the following topics:

- Configuring QuickTrace Using Fusion Middleware Control
- Writing the Trace Messages to a File Using Fusion Middleware Control

12.6.1.1.1 Configuring QuickTrace Using Fusion Middleware Control To configure QuickTrace using Fusion Middleware Control:

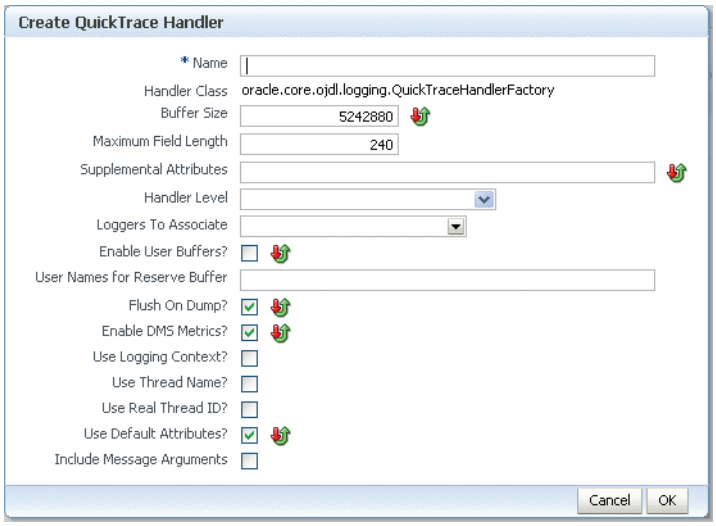
1. From the navigation pane, expand the farm, then **WebLogic Domain**, and then the domain. Right-click the Managed Server name and choose **Logs,** then **Log Configuration.**

   The Log Configuration page is displayed.

2. Select the QuickTrace tab.

3. Click **Create.**

   The Create QuickTrace Handler dialog box is displayed, as shown in the following figure:



4. For **Name,** enter a name for the handler.

5. For **Buffer Size,** enter the size, in bytes, for the buffer for storing log messages in memory. The default is 5242880.

6. For **Maximum Field Length,** enter the length, in bytes, for each field in a message. The fields can include the message text, supplemental attributes, and the thread name. The default is 240.

An excessively long field for each message can reduce the amount of log records in the buffer.

7. For **Handler Level,** select the log level for the handler. See Section 12.4.3 for information about the levels.
8. For **Loggers to Associate,** select the loggers that you want to associate with this QuickTrace handler. All messages of the specified level for these handlers will be written to memory.

   Many loggers are associated with other handlers. For example, the oracle.adf logger is associated with the handlers odl-handler, wls-domain, and console-handler. When you set the level of the logger, these handlers will use the same level, such as TRACE:1, for the logger, such as oracle.adf. As a result, much information will be written to the log files, consuming resources. To avoid consuming resources, set the level of the handlers to a lower level, such as WARNING or INFORMATION.
9. Select **Enable User Buffer?** if you want to enable a user buffer. If you enable this, the handler maintains an individual buffer for each user you specify.

   Then, for **User Names for Reserve Buffer,** enter the names of the users, separated by commas.
10. For the remaining options, accept the default values. For information about the options, see "ConfigureLogHandler" in the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference.*
11. Click **OK.**
12. When the configuration completes processing, click **OK.**

Now, messages of the specified level for the specified loggers are written to memory.

12.6.1.1.2 Writing the Trace Messages to a File Using Fusion Middleware Control

You can save the messages that are in memory to a file by invoking the QuickTrace Dump in Fusion Middleware Control:

1. From the QuickTrace tab of the Log Configuration page, select the handler and click **Invoke QuickTrace Dump.**

   The Invoke QuickTrace Dump dialog box is displayed.
2. For **Buffer Name,** if you have specified user buffers when you configured the QuickTrace handler, select the user, or select Common Buffer for users that you did not specify. If you did not specify any user buffers, the Common Buffer is the only option.
3. Click **OK.**

   When the processing is complete, the View Log Messages page is displayed.
4. You can search the messages, as described in Section 12.3.2, and you can correlate the messages as described in Section 12.5.

   In addition, you can download the messages to a file, as described in Section 12.3.3.1.

12.6.1.2 Configuring QuickTrace Using WLST

You can configure and use QuickTrace Using WLST, as described in the following topics:

- Configuring QuickTrace Using WLST

- Writing the Trace Messages to a File Using WLST
- Disabling QuickTrace Using WLST

12.6.1.2.1 Configuring QuickTrace Using WLST To configure QuickTrace using WLST, you associate a logger with the QuickTrace handler, using the configureLogHandler command.

For example, to associate the oracle.adf logger with the QuickTrace handler and write all TRACE:1 messages to memory:

1. Use the configureLogHandler command to associate the logger with the QuickTrace handler:

```
configureLogHandler(name="quicktrace-handler", addToLogger="oracle.adf")

Handler Name: quicktrace-handler
type:   oracle.core.ojdl.logging.QuickraceHandlerFactory
useLoggingContext:      false
bufferSize:     5242880
 .
 .
 .
enableUserBuffer:       false
```

   The messages for the handler are written to a common buffer.

   You can set additional properties for the QuickTrace handler. For example, to enable user buffers for the users user1 and user2:

```
configureLogHandler(name="quicktrace-handler", addToLogger="oracle.adf.faces",
      propertyName="enableUserBuffer", propertyValue="true",
      propertyName="reserveBufferUserID", propertyValue="user1, user2")
...
Handler Name: quicktrace-handler
type:   oracle.core.ojdl.logging.QuickTraceHandlerFactory
useLoggingContext:      false
bufferSize:     5242880
 .
 .
 .
reserveBufferUserID:    user1, user2
enableUserBuffer:       true
```

   Messages for user1 and user2 are written to separate buffers. In addition, messages related to other users are written to the common buffer.

   To confirm the settings for the handler, use the listLogHandlers command, as described in "listLogHandlers" in the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

2. Set the level of the logger, using the setLogLevel command:

```
setLogLevel(logger='oracle.adf', level='TRACE:1')
```

   To confirm the settings for the logger, use the listLoggers command, as described in "listLoggers" in the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

3. Many loggers are associated with other handlers. For example, the oracle.adf logger is associated with the handlers odl-handler, wls-domain, and console-handler. When you set the level of the logger, these handlers will use the

same level (TRACE:1) for the logger oracle.adf. As a result, much information will be written to the log files, consuming resources. To avoid consuming resources, set the level of the handlers to a lower level, such as WARNING or INFORMATION.

For this example, set the level of the three handlers to WARNING:1:

```
configureLogHandler(name="odl-handler", level="WARNING:1")
configureLogHandler(name="wls-domain", level="WARNING:1")
configureLogHandler(name="console-handler", level="WARNING:1")
```

Note that you should keep the level of the QuickTrace handler at ALL, which is the default.

See Also: The command "configureLogHandler" in the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*

To confirm the level for the handler, use the getLogLevel command, as described in Section 12.4.3.2.

12.6.1.2.2 Writing the Trace Messages to a File Using WLST You can save the messages to a file by using the executeDump command.

For example:

```
executeDump(name="odl.quicktrace", outputFile="/scratch/oracle1/qt1.dmp")
```

The command writes the dump to the specified file.

For more information about the executeDump command, see Section 13.4.4.3.

In addition, if an incident is created (automatically or manually), the QuickTrace messages are written to dump files in the incident directory. If you enabled user buffers, each user will have one file and the common buffer will have one file.

The file names have the following format:

```
odl_quicktraceN_iincident_number.username.dmp
```

For example:

```
odl_quicktrace6_i1.weblogic.dmp
```

See Section 13.4.5.1 for information about creating an incident.

12.6.1.2.3 Disabling QuickTrace Using WLST To disable QuickTrace, use the configureLogHandler command and specify that the level is OFF:

```
configureLogHandler(name="quicktrace-handler", level="OFF")

Handler Name: quicktrace-handler
type:   oracle.core.ojdl.logging.QuickraceHandlerFactory
 .
 .
 .
reserveBufferUserID:    user1, user2
enableUserBuffer:       true
```

To remove a specific logger from association with the QuickTrace handler, use the configureLogHandler command with the removeFromLogger parameter:

```
configureLogHandler(name="quicktrace-handler",
removeFromLogger="oracle.adf.faces")
```

```
Handler Name: quicktrace-handler
type:   oracle.core.ojdl.logging.QuickraceHandlerFactory
reserveBufferUserID:    user1, user2
enableUserBuffer:       true
```

See Also: The command "configureLogHandler" in the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*

12.6.2 Configuring and Using Selective Tracing

Selective tracing provides fine-grained logging for specified users or other attributes of a request.

For example, a user cannot perform some functions because of security permissions, but it is not clear what operations or lack of permission for those operations are posing a problem.

In this case, you can enable tracing across the entire system but this would generate a large volume of log messages for all users in the system, not only for the user having a problem. With selective tracing, you can enable tracing only for the user who is having a problem. Then, you can ask the user to retry the functions. Following that, you can look at the trace messages which apply to the specific request made by the user.

You can also specify the logger to narrow the scope of the messages being logged.

Before you use selective tracing:

1. Modify the following file:

   (UNIX) *DOMAIN_HOME*/bin/setDomainEnv.sh
   (Windows) *DOMAIN_HOME*\bin\setDomainEnv.cmd

2. Append the following lines to the file:

   - On UNIX:

     ```
     JAVA_
     OPTIONS="-Djava.util.logging.manager=oracle.core.ojdl.logging.ODLLogManager
     ${JAVA_OPTIONS}"
     export JAVA_OPTIONS
     FMWCONFIG_CLASSPATH=${FMWCONFIG_CLASSPATH}${CLASSPATHSEP}${COMMON_
     COMPONENTS_HOME}/modules/oracle.odl_11.1.1/ojdl.jar
     export FMWCONFIG_CLASSPATH
     ```

   - On Windows:

     ```
     set JAVA_
     OPTIONS=-Djava.util.logging.manager=oracle.core.ojdl.logging.ODLLogManager
     %JAVA_OPTIONS%
     set FMWCONFIG_CLASSPATH=%FMWCONFIG_CLASSPATH%;%COMMON_COMPONENTS_
     HOME%\modules\oracle.odl_11.1.1\ojdl.jar
     ```

3. Restart the Administration Server and Managed Servers, as described in Section 4.2.

If you upgraded a domain from a version previous to 11.1.1.5, you must take the following steps:

1. Back up the following file:

   (Unix) *DOMAIN_HOME*/config/fmwconfig/mbeans/odl_mbeans.xml
   (Windows) *DOMAIN_HOME*\config\fmwconfig\mbeans\odl_mbeans.xml

2. Back up the following file for all Managed Servers:

(Unix) *DOMAIN_HOME*/config/fmwconfig/servers/*server_name*/mbeans/odl_mbeans.xml

(Windows) *ORACLE_COMMON_HOME*\modules\oracle.odl_11.1.1*server_name*]\mbeans\odl_mbeans.xml

3. Copy the following file:

   (UNIX) *ORACLE_COMMON_HOME*/modules/oracle.odl_11.1.1/domain_config/mbeans/odl_mbeans.xml
   (Windows) *ORACLE_COMMON_HOME*\modules\oracle.odl_11.1.1\domain_config\mbeans\odl_mbeans.xml

   Copy it to the following location:

   (UNIX) *DOMAIN_HOME*/config/fmwconfig/mbeans/odl_mbeans.xml
   (Windows) *DOMAIN_HOME*\config\fmwconfig\mbeans\odl_mbeans.xml

4. Copy the following file:

   (UNIX) *ORACLE_COMMON_HOME*/modules/oracle.odl_11.1.1/server_config/mbeans/odl_mbeans.xml
   (Windows) *ORACLE_COMMON_HOME*\modules\oracle.odl_11.1.1\server_config\mbeans\odl_mbeans.xml

   Copy it to the following location:

   (UNIX)*DOMAIN_HOME*/config/fmwconfig/servers/*server_name*/mbeans/odl_mbeans.xml
   (Windows) *ORACLE_COMMON_HOME*\modules\oracle.odl_11.1.1*server_name*\mbeans\odl_mbeans.xml

   Note that if you have multiple servers, you must copy the file to the location for each server.

You can use Selective Tracing using Fusion Middleware Control or WLST, as described in the following topics:

- Configuring Selective Tracing Using Fusion Middleware Control
- Configuring Selective Tracing Using WLST

12.6.2.1 Configuring Selective Tracing Using Fusion Middleware Control

You can configure selective tracing, view traces, and disable selective tracing using Fusion Middleware Control, as described in the following topics:

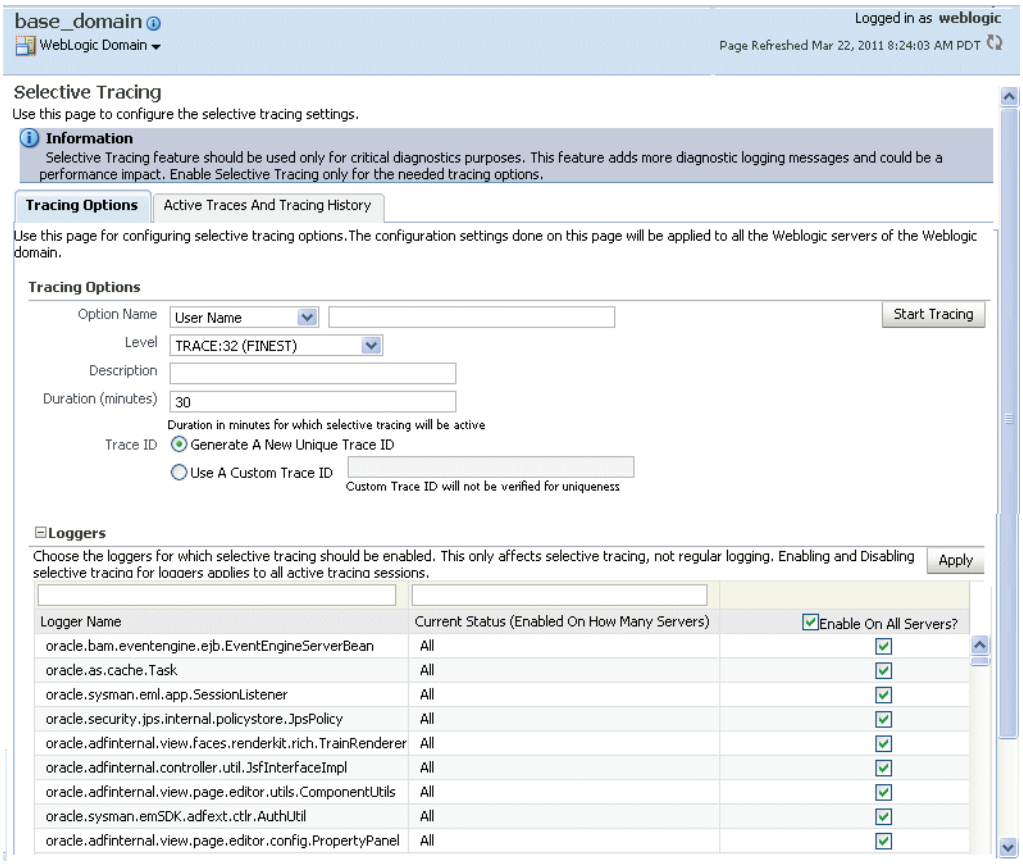
- Configuring Selective Tracing Using Fusion Middleware Control
- Viewing Selective Traces Using Fusion Middleware Control
- Disabling Selective Tracing Using Fusion Middleware Control

12.6.2.1.1 Configuring Selective Tracing Using Fusion Middleware Control

To configure selective tracing using Fusion Middleware Control:

1. From the navigation pane, expand the farm, then **WebLogic Domain**. Right-click the domain name and choose **Logs,** then **Selective Tracing.**

   The Selective Tracing page is displayed, as shown in the following figure:



2. For **Option Name,** select an option, such as User Name, Application Name, or Client Host. Then, enter the name, for example, user1.
3. For **Level,** select a logging level. Table 12–3 describes the logging levels.
4. For **Description,** enter a description.
5. For **Duration,** enter the number of minutes that you want the selective trace to run.

   The selective trace is disabled after the specified time.
6. For Trace ID, select either **Generate a New Unique Trace ID** or **Use a Custom Trace ID.** If you select Use a Custom Trace ID, enter an ID of your choosing, but make sure that it is unique. Note Fusion Middleware Control does not verify the uniqueness of the ID.
7. In the Loggers section, by default, all loggers are selected.

   You can select specific loggers that you want to trace. To find particular loggers, you can enter a string in the field above the table and click the Return key. For example, to find all loggers that begin with oracle.security, enter oracle.security.

   Then, in the table, select the loggers in the **Enable on All Servers** column.

   Note when you select loggers, those loggers apply to all current and active traces. Also note that even if you disable the loggers, you may see messages because all loggers have a general logging level, such as Notification. Those messages would still be written.
8. Click **Apply.**
9. Click **Start Tracing.**

Now that you have started the trace, you can view active traces, as well as former traces, as described in Section 12.6.2.1.2.

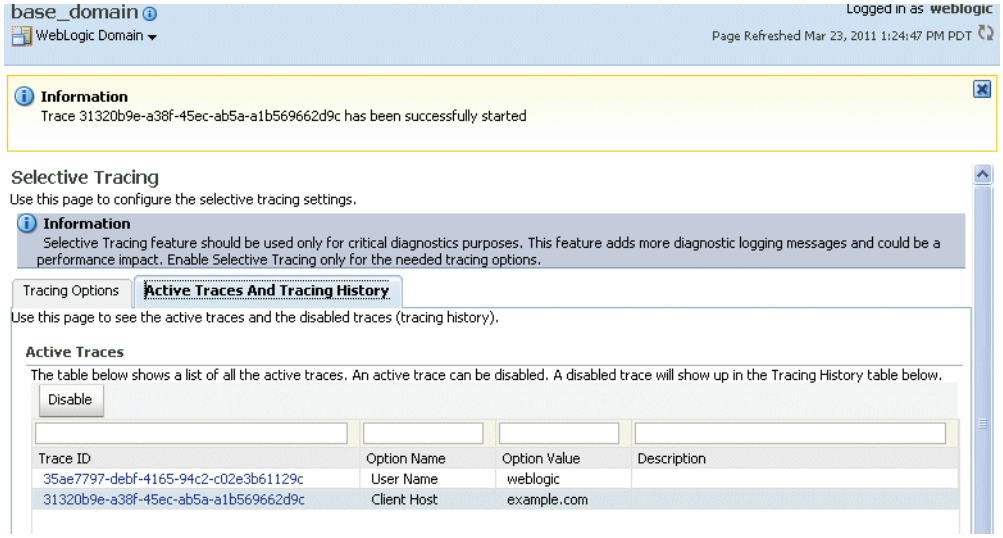
12.6.2.1.2 Viewing Selective Traces Using Fusion Middleware Control

You can view the selective traces that are currently active and the history of selective traces.

To view the selective traces:

1. From the Selective Tracing page, select the **Active Traces and Tracing History** tab.

   The tab shows a table with the active traces and a table with the tracing history, as shown in the following figure:



2. To view a trace, select it from the appropriate table.

   The Log Messages page is displayed, with the messages that were captured by Selective Tracing. You can search the messages, as described in Section 12.3.2, and you can correlate the messages as described in Section 12.5.

   In addition, you can download the messages to a file, as described in Section 12.3.3.1.

12.6.2.1.3 Disabling Selective Tracing Using Fusion Middleware Control

To disable selective tracing using Fusion Middleware Control:

1. From the navigation pane, expand the farm, then **WebLogic Domain**. Right-click the domain name and choose **Logs,** then **Selective Tracing.**
2. Select the **Active Traces and Tracing History** tab.
3. In the Active Traces table, select the trace and click **Disable.**

12.6.2.2 Configuring Selective Tracing Using WLST

You can configure selective tracing, view traces, and disable selective tracing using WLST, as described in the following topics:

- Configuring Selective Tracing Using WLST
- Viewing Selective Traces Using WLST
- Disabling Selective Traces Using WLST

12.6.2.2.1 Configuring Selective Tracing Using WLST

You can configure loggers for selective tracing and start tracing using the WLST configureTracingLoggers and startTracing commands.

For the simplest case, you can configure and start a trace using the startTracing command. When you do so, the selective tracing includes all loggers enabled for selective tracing.

For example, user1 receives errors when attempting to perform certain operations. To start a trace of messages related to user1 and to set the logging level to FINE, use the following command:

```
startTracing(user="user1",level="FINE")
Started tracing with ID: 885649f7-8efd-4a7a-9898-accbfc0bbba3
```

The startTracing command does not provide options to include or exclude particular loggers. In this case, you can use the configureTracingLoggers command. This command allows you to configure selective tracing to include only particular loggers and particular Oracle WebLogic Server instances. Note that the options you specify apply to all current and active traces.

For example, to configure selective tracing to include only security-related loggers:

1. Specify that all loggers be disabled for tracing, as shown in the following example:

   ```
   configureTracingLoggers(action="disable")
   Configured 1244 loggers
   ```

2. Enable the security-related loggers, by specifying the pattern option with a regular expression:

   ```
   configureTracingLoggers(pattern='oracle.security.*', action="enable")
   Configured 62 loggers
   ```

   To see a list of the loggers that support selective tracing, use the WLST listTracingLoggers command, as shown in the following example:

   ```
   listTracingLoggers(pattern="oracle.security.*")
   ------------------------------------------------------------------+--------
   Logger                                                            | Status
   ------------------------------------------------------------------+--------
   oracle.security                                                   | enabled
   oracle.security.audit.logger                                      | enabled
   oracle.security.jps.az.common.util.JpsLock                        | enabled
    .
    .
    .
   ```

3. Use the startTracing command, specifying the users and the level. For example:

   ```
   startTracing(user="user1",level="FINE")
   Started tracing with ID: a9580e65-13c4-420b-977e-5ba7dd88ca7f
   ```

See Also: The following commands in the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference* for complete syntax:

- configureTracingLoggers
- startTracing
- listTracingLoggers

12.6.2.2.2 Viewing Selective Traces Using WLST After you have begun a trace, you can see the active traces by using the listActiveTraces command, as shown in the following example:

```
listActiveTraces()
-------------------------------------+----------+----------+------+----------+----------
Trace ID                             |Attr. Name|Attr. Value| Level| Exp. Time |Description
-------------------------------------+----------+----------+------+----------+----------
a9580e65-13c4-420b-977e-5ba7dd88ca7f |USER_ID   |user1      | FINE |           |
a04b47f7-2830-4d80-92ee-ba160cdacf6b |USER_ID   |user2      | FINE |           |
```

You can view the contents of the trace using the displayLogs command and passing it the trace ID. You can also view traces that have stopped. For example:

```
displayLogs("a9580e65-13c4-420b-977e-5ba7dd88ca7f")
```

> **See Also:** The listActiveTraces command in the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference* for complete syntax

12.6.2.2.3 Disabling Selective Traces Using WLST To avoid excessive logging in the system, you can disable a selective trace when you have obtained the information that you need. To disable a selective trace, you use the WLST stopTracing command, passing it the trace ID or user. For example:

```
stopTracing(traceId="885649f7-8efd-4a7a-9898-accbfc0bbba3")
Stopped 1 traces
```

You can also disable all traces by using the stopAll option. For example:

```
stopTracing(stopAll=1)
```

> **See Also:** The stopTracing command in the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference* for complete syntax

13 Diagnosing Problems

This chapter describes how to use the Oracle Fusion Middleware Diagnostic Framework to collect and manage information about a problem so that you can resolve it or send it to Oracle Support for resolution.

This chapter contains the following topics:

13.1 Understanding the Diagnostic Framework

Oracle Fusion Middleware includes a Diagnostic Framework, which aids in detecting, diagnosing, and resolving problems. The problems that are targeted in particular are critical errors such as those caused by code bugs, metadata corruption, customer data corruption, deadlocked threads, and inconsistent state.

When a critical error occurs, it is assigned an incident number, and diagnostic data for the error (such as log files) are immediately captured and tagged with this number. The data is then stored in the Automatic Diagnostic Repository (ADR), where it can later be retrieved by incident number and analyzed.

The goals of the Diagnostic Framework are:

- First-failure diagnosis
- Limiting damage and interruptions after a problem is detected
- Reducing problem diagnostic time
- Reducing problem resolution time
- Simplifying customer interaction with Oracle Support

The Diagnostic Framework includes the following technologies:

- **Automatic capture of diagnostic data upon first failure:** For critical errors, the ability to capture error information at first failure greatly increases the chance of a quick problem resolution and reduced downtime. The Diagnostic Framework automatically collects diagnostics, such as thread dumps, DMS metric dumps, and WebLogic Diagnostics Framework (WLDF) server image dumps. Such diagnostic data is similar to the data collected by airplane "black box" flight recorders. When a problem is detected, alerts are generated and the fault diagnosability

infrastructure is activated to capture and store diagnostic data. The data is stored in a file-based repository and is accessible with command-line utilities.

- **Standardized log formats:** Standardized log formats (using the ODL log file format) across all Oracle Fusion Middleware components allows administrators and Oracle Support personnel to use a single set of tools for problem analysis. Problems are more easily diagnosed, and downtime is reduced.
- **Diagnostic rules:** Each component defines diagnostic rules that are used to evaluate whether a given log message should result in an incident being created and which dumps should be executed. The diagnostic rules also indicate whether an individual dump should be created synchronously or asynchronously.
- **Incident detection log filter:** The incident detection log filter implements the java.util.logging filter. It inspects each log message to see if an incident should be created, basing its decision on the diagnostic rules for components and applications.
- **Incident packaging service (IPS) and incident packages:** The IPS enables you to automatically and easily gather the diagnostic data—log files, dumps, reports, and more—pertaining to a critical error that has a corresponding incident, and package the data into a zip file for transmission to Oracle Support. All diagnostic data relating to a critical error that has been detected by the Diagnostics Framework is captured and stored as an incident in ADR. The incident packaging service identifies the required files automatically and adds them to the zip file.

  Before creating the zip file, the IPS first collects diagnostic data into an intermediate logical structure called an incident **package.** Packages are stored in the Automatic Diagnostic Repository. If you choose to, you can access this intermediate logical structure, view and modify its contents, add or remove additional diagnostic data at any time, and when you are ready, create the zip file from the package and upload it to Oracle Support.
- **Integration with WebLogic Diagnostics Framework (WLDF):** The Oracle Fusion Middleware Diagnostics Framework integrates with some features of WebLogic Diagnostics Framework (WLDF), including the capturing of WebLogic Server images on detection of critical errors. WLDF is a monitoring and diagnostic framework that defines and implements a set of services that run within WebLogic Server processes and participate in the standard server life cycle. Using WLDF, you can create, collect, analyze, archive, and access diagnostic data generated by a running server and the applications deployed within its containers. This data provides insight into the run-time performance of servers and applications and enables you to isolate and diagnose faults when they occur.

  Oracle Fusion Middleware Diagnostics Framework integrates with the following components of WLDF:

  – WLDF Watch and Notification, which watches specific logs and metrics for specified conditions and sends a notification when a condition is met. There are several types of notifications, including JMX notification and a notification to create a Diagnostic Image. Oracle Fusion Middleware Diagnostics Framework integrates with the WLDF Watch and Notification component to create incidents.

  – Diagnostic Image Capture, which gathers the most common sources of the key server state used in diagnosing problems. It packages that state into a single artifact, the Diagnostic Image. With Oracle Fusion Middleware Diagnostics Framework, it writes the artifact to ADR.

For more information about WLDF, see *Oracle Fusion Middleware Configuring and Using the Diagnostics Framework for Oracle WebLogic Server*.

13.1.1 About Incidents and Problems

To facilitate diagnosis and resolution of critical errors, the Diagnostic Framework introduces two concepts for Oracle Fusion Middleware: problems and incidents.

A **problem** is a critical error. Critical errors manifest as internal errors or other severe errors. Problems are tracked in the ADR. Each problem has a **problem key**, which is a text string that describes the problem. It includes an error code (in the format *XXX-nnnnn*) and in some cases, other error-specific values.

An **incident** is a single occurrence of a problem. When a problem (critical error) occurs multiple times, an incident is created for each occurrence. Incidents are timestamped and tracked in the ADR. Each incident is identified by a numeric incident ID, which is unique within the ADR home. When an incident occurs, the Diagnostic Framework:

- Gathers first-failure diagnostic data about the incident in the form of dump files (incident dumps).
- Stores the incident dumps in an ADR subdirectory created for that incident.
- Registers the incidents dumps with the incident in ADR.

13.1.1.1 Incident Flood Control

It is conceivable that a problem could generate dozens or perhaps hundreds of incidents in a short period of time. This would generate too much diagnostic data, which would consume too much space in the ADR and could possibly slow down your efforts to diagnose and resolve the problem. For these reasons, the Diagnostic Framework applies flood control to incident generation after certain thresholds are reached. A **flood-controlled incident** is an incident that is not recorded in the ADR. Instead, the Diagnostic Framework writes a message at the WARNING level to the log file and returns an oracle.dfw.incident.Incident object. Flood-controlled incidents provide a way of informing you that a critical error is ongoing, without overloading the system with diagnostic data.

By default, if more than 5 incidents with the same problem key occur within 60 minutes, subsequent incidents with the same problem key are flood controlled. You can change this value using MBeans, as described in Section 13.3.

13.1.2 Diagnostic Framework Components

The following topics describe the key components of the Diagnostic Framework:

- Automatic Diagnostic Repository
- Diagnostic Dumps
- Management MBeans
- WLST Commands for Diagnostic Framework
- ADRCI Command-Line Utility

13.1.2.1 Automatic Diagnostic Repository

The Automatic Diagnostic Repository (ADR) is a file-based hierarchical repository for Oracle Fusion Middleware diagnostic data, such as traces and dumps. The Oracle Fusion Middleware components store all incident data in the ADR. Each Oracle WebLogic Server stores diagnostic data in subdirectories of its own home directory

within the ADR. For example, each Managed Server and Administration Server has an ADR home directory.

The ADR root directory is known as ADR base. By default, the ADR base is located in the following directory:

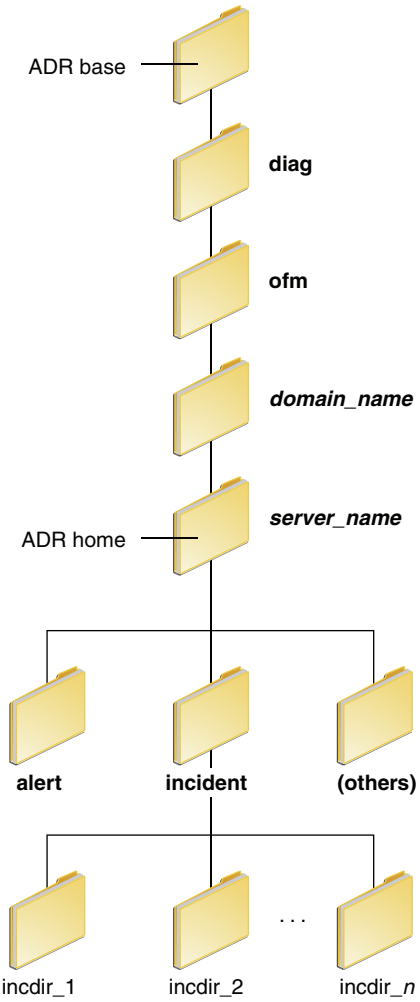
DOMAIN_HOME/servers/*server_name*/adr

Within ADR base, there can be multiple ADR homes, where each ADR home is the root directory for all incident data for a particular instance of Oracle WebLogic Server. The following path shows the location of the ADR home:

ADR_BASE/diag/ofm/*domain_name*/*server_name*

Figure 13–1 illustrates the directory hierarchy of the ADR home for an Oracle WebLogic Server instance.

Figure 13–1 ADR Directory Structure for Oracle Fusion Middleware



The subdirectories in the ADR home contain the following information:

- alert: The XML-formatted alert log.

- incident: A directory that can contain multiple subdirectories, where each subdirectory is named for a particular incident. The subdirectories are named incdir_*n*, with *n* representing the number of the incident. Each subdirectory contains information and diagnostic dumps pertaining only to that incident.
- (others): Other subdirectories of ADR home, which store incident packages and other information.

> **Note:** ADR uses the domain name as the Product ID and the server name as the Instance ID when it packages an incident. However, if either name is more than 30 characters, ADR truncates the name. In addition, dollar sign ($) and space characters are replaced with underscores.

13.1.2.2 Diagnostic Dumps

A **diagnostic dump** captures and dumps specific diagnostic information when an incident is created (automatic) or on the request of an administrator (manual). When executed as part of incident creation, the dump is included with the set of incident diagnostics data. Examples of diagnostic dumps include a JVM thread dump, JVM class histogram dump, and DMS metric dump. For a list of diagnostic dumps, see Table 13–4.

13.1.2.3 Management MBeans

The Diagnostic Framework provides MBeans that you can use to configure the Diagnostic Framework. For example, you can enable or disable flood control and you can configure how many incidents with the same problem key can occur within a specified time period. For information about using the management MBeans to configure the Diagnostic Framework, see Section 13.3.

You can also use the MBeans to query and create incidents, discover the list of available diagnostic dump types, and execute individual diagnostic dumps.

13.1.2.4 WLST Commands for Diagnostic Framework

The Diagnostic Framework provides WLST commands that you can use to view information about problems and incidents, create incidents, execute specific dumps and query the set of diagnostic dump types. For more information, see:

- Section 13.4.2.1, "Viewing Problems"
- Section 13.4.2.2, "Viewing Incidents"
- Section 13.4.4.1, "Listing Diagnostic Dumps"
- Section 13.4.4.2, "Viewing a Description of a Diagnostic Dump"
- Section 13.4.4.3, "Executing Dumps"
- Section 13.4.5.1, "Creating an Incident Manually"
- "Diagnostic Framework Custom WLST Commands" in the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*

  To use the custom WLST Diagnostic Framework commands, you must invoke the WLST script from the Oracle Common home. See Section 3.5.1.1 for more information.

13.1.2.5 ADRCI Command-Line Utility

The ADR Command Interpreter (ADRCI) is a utility that enables you to investigate problems, and package and upload first-failure diagnostic data to Oracle Support, all within a command-line environment. ADRCI also enables you to view the names of the dump files in the ADR, and to view the alert log with XML tags stripped, with and without content filtering.

ADRCI is installed in the following directory:

```
(UNIX) MW_HOME/wlserver_10.3/server/adr
(Windows) MW_HOME\wlserver_10.3\server\adr
```

See the following sections for information about using the ADRCI command-line utility:

- Packaging an Incident
- Purging Incidents

See Also:

- The chapter "ADRCI: ADR Command Interpreter" in *Oracle Database Utilities*
- The chapter "Managing Diagnostic Data" in the *Oracle Database Administrator's Guide*

13.2 How the Diagnostic Framework Works

The Diagnostic Framework is active in each server and provides automatic error detection through predefined configured rules. Oracle Fusion Middleware components and applications automatically benefit from this always-on checking.

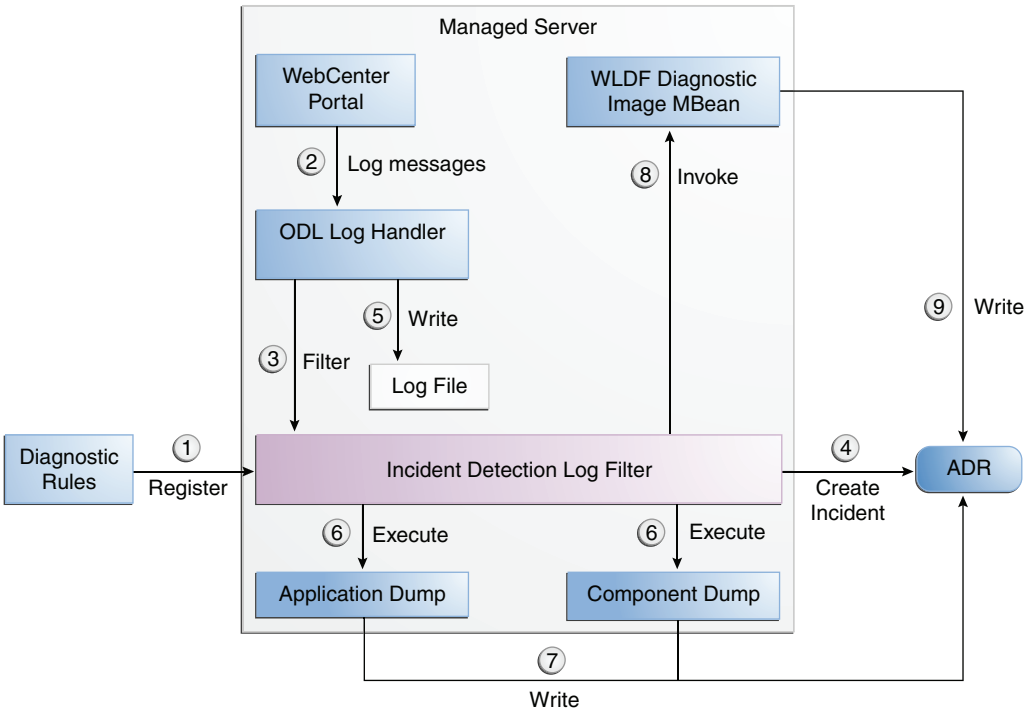
Incidents are automatically detected in two ways:

- By the incident detection log filter, which is automatically configured to detect critical errors.
- By the WLDF Watch and Notification component. The Diagnostics Framework listens for a predefined notification type and creates incidents when it receives such notifications.

  For information about configuring WLDF Watch and Notification, see Section 13.3.3.
- Programmatic incident creation. Some components create incidents directly.

Figure 13–2 shows the interaction when the incident is detected by the incident log detector. It shows the interaction among the incident log detector, the WLDF Diagnostic Image MBean, ADR, and component or application dumps when an incident is detected by the incident log detector.

Figure 13–2 Incident Creation Generated by Incident Log Detector



The steps represented in Figure 13–2 are:

1. The incident detection log filter is initialized with component and application diagnostic rules.
2. An application or component (in this case Oracle WebCenter Portal) logs a message using the java.util.logging API.
3. The ODL log handler passes the message to the incident detection log filter.
4. The incident log detection filter inspects the log message to see if an incident should be created, basing its decision on the diagnostic rules for the component. If the diagnostic rule indicates that an incident should be created, it creates an incident in the ADR.
5. The ODL log handler writes the log message to the log file, and returns control back to Oracle WebCenter Portal.

   When an incident is created, a message, similar to the following, is written to the log file:

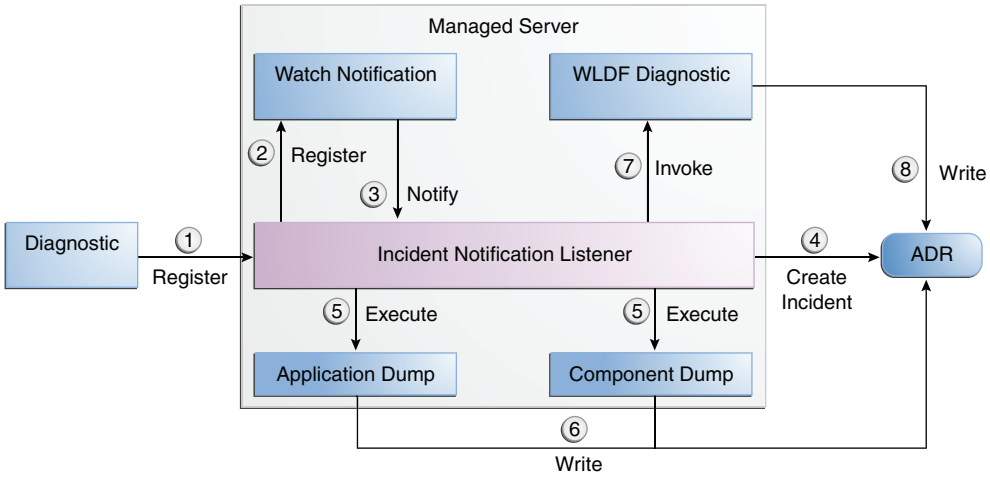
```
[2010-09-16T06:37:59.264-07:00] [dfw] [NOTIFICATION] [DFW-40104] [oracle.dfw]
[tid: 10] [ecid: 0000IF34gtMC8xT6uBf9EH1AgEck000000,0] [errid: 6]
[detailLoc: /middleware/user_projects/base_
domain/servers/AdminServer/adr/diag/ofm/base_domain/AdminServer]
[probKey: MDS-123456 [testComponent][testModule]] incident 6 created with
 problem key "MDS-123456 [testComponent][testModule]", in directory
 /middleware/user_projects/base_domain/servers/AdminServer/adr/diag/ofm/base_
domain/AdminServer/incident/incdir_6
```

6. The Diagnostic Framework executes the diagnostic dumps that are indicated by the diagnostic rules for the component.
7. The Diagnostic Framework writes the dumps to ADR, in the directory created for the incident.

8. The Diagnostic Framework invokes the WLDF Diagnostic Image MBean requesting that a Diagnostic Image be created in ADR.
9. WLDF writes the Diagnostic Image to ADR.

Figure 13–3 shows the interaction when an incident is detected by the WLDF Watch and Notification system. It shows the interaction among the incident notification listener, the WLDF Watch and Notification system, and the WLDF Diagnostic Image MBean.

Figure 13–3 Incident Creation Generated by WLDF Watch Notification



The steps represented in Figure 13–3 are:

1. The incident notification listener is initialized with component and application diagnostic rules.
2. Oracle Fusion Middleware Diagnostic Framework registers a JMX notification listener with WLDF. The listener listens for events from the WLDF Watch and Notification system. It only processes notifications of type oracle.dfw.wldfnotification.
3. Something in the system causes the configured WLDF watch to be triggered, causing a notification to be sent to the incident notification listener. The notification includes event information describing the data that caused the watch to trigger.
4. The Diagnostic Framework creates an incident in ADR.
5. The Diagnostic Framework executes the diagnostic dumps that are indicated by the diagnostic rules.
6. The Diagnostic Framework writes the dumps to ADR, in the directory created for the incident.
7. The Diagnostic Framework invokes the WLDF Diagnostic Image MBean requesting that a Diagnostic Image be created in ADR.
8. WLDF writes the Diagnostic Image to ADR.

13.3 Configuring the Diagnostic Framework

You can configure some settings for the Diagnostic Framework. In addition, you can configure an WLDF Watch and Notification to create an incident. The following topics describe how to configure the Diagnostic Framework:

- Configuring Diagnostic Framework Settings
- Configuring Problem Suppression
- Configuring WLDF Watch and Notification for the Diagnostic Framework

13.3.1 Configuring Diagnostic Framework Settings

You can configure the following settings:

- Enabling or disabling the detection of incidents through the log files
- Enabling or disabling flood control and setting parameters for flood control

You configure these settings by using the Diagnostic Framework MBean DiagnosticConfig. The following shows the MBean's ObjectName:

```
oracle.dfw:type=oracle.dfw.jmx.DiagnosticsConfigMBean,name=DiagnosticsConfig
```

Table 13–1 shows the attributes for the DiagnosticConfig MBean and a description of each parameter.

Table 13–1 DiagnosticConfig MBean Attributes for Diagnostic Framework

Attributes	Description
floodControlEnabled	Enables or disables flood control. Specify `true` for enabled or `false` for disabled. The default is `true`. Note that flood control does not apply to manually created incidents.
floodControlIncidentCount	Sets the number of incidents with the same problem key that can be created within the time period, specified by floodControlIncidentTimeoutPeriod, before they are controlled by flood control. The default is 5. When flood control is enabled, if the number of incidents with the same problem key exceeds this count, no incidents are created, but the Diagnostic Framework writes a message at the WARNING level to the log file.
floodControlIncidentTimeoutPeriod	Sets the time period in which the number of incidents, as specified by floodControlIncidentCount, with the same problem key can be created before they are controlled by flood control. The default is 60 minutes.
incidentCreationEnabled	Enables or disables incident creation. Specify `true` for enabled or `false` for disabled. The default is `true`.
logDetectionEnabled	Enables or disables the detection of incidents through the log files. Specify `true` for enabled or `false` for disabled. The default is `true`.

Table 13–1 (Cont.) DiagnosticConfig MBean Attributes for Diagnostic Framework

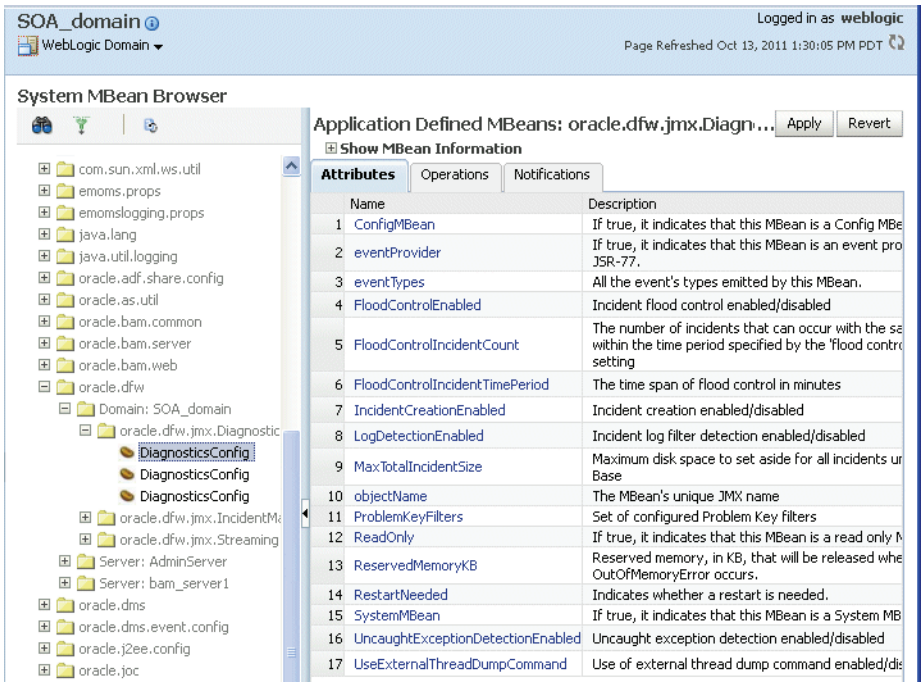
Attributes	Description
maxTotalIncidentSize	Sets the maximum total size that is allocated for all incidents. When the limit is reached, the oldest incidents are purged until the space used by all incidents is less than the amount specified by this parameter. The default is 500 MB. The limit may be exceeded during the creation of an incident, but when the incident creation completes, the oldest incidents are purged.
reservedMemoryKB	The amount of reserved memory that is released when OutOfMemoryError is detected. When the Diagnostic Framework starts, it allocates 512 KB of memory for its own private use. When the Diagnostic Framework detects that an OutOfMemoryError has occurred in the server, it frees that block of memory and proceeds to create the incident. The default is 512 KB.
uncaughtExceptionDetectionEnabled	Enables the Java-based uncaught exception handler. When enabled and an uncaught exception is detected, an incident is created. Specify `true` for enabled or `false` for disabled. The default is `true`.
useExternalCommands	Indicates whether external JVM commands should be used to perform thread dumps. Specify `true` for enabled or `false` for disabled. The default is `true`.

The following example shows how to configure these settings using the Fusion Middleware Control System MBean Browser:

1. From the target navigation pane, expand the farm, then **WebLogic Domain.**
2. Select the domain.
3. From the WebLogic Domain menu, choose **System MBean Browser.**

   The System MBean Browser page is displayed.
4. Expand **Application Defined Beans**, then **oracle.dfw,** then **domain**.*domain_name*, then **dfw.jmx.DiagnosticsConfigMBean.**
5. Select one of the **DiagnosticConfig** entries. There is one DiagnosticConfig entry for each server.
6. In the Application Defined MBean pane, expand **Show MBean Information** to see the server name.

   The following shows the System MBean Browser page:



7. To change the values for the attributes listed in Table 13–1, enter or select the value in the **Value** field.
8. Click **Apply.**

13.3.2 Configuring Problem Suppression

In certain situations, you may want to suppress the creation of incidents based on a particular problem key. For example, in a development environment, when you are developing a servlet, you may generate high number of uncaught exceptions as you refine the code. This results in the creation of unnecessary incidents.

The Diagnostic Framework allows you to configure problem suppression filters so that problems that match the filter criteria do not result in the creation of an incident.

When you configure a problem suppression filter, you use a regular expression that represents a pattern that you want to match. The regular expression is matched using the java.util.regex class. For example:

- The following regular expression matches any incident with a problem key that starts with MDS-5000.

  ```
  MDS-5000.*
  ```

- The following regular expression matches any problem with the text `OutOfMemory`. Because the regular expression is case-sensitive, it will not match problems with the text `outofmemory`.

  ```
  .*OutOfMemory.*
  ```

You can add and remove filters and get a list of filters or the detail of one filter using the DiagnosticConfig MBean.

Table 13–2 shows the operations and attribute for the configuring problem suppression filters and a description of each.

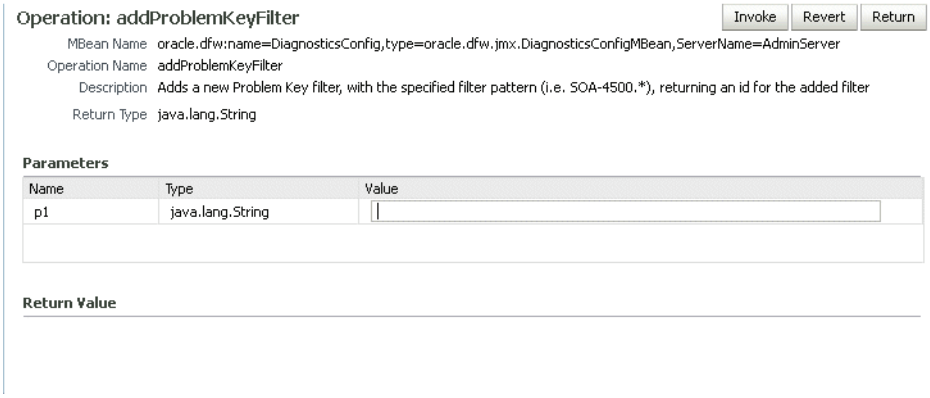
Table 13–2 DiagnosticConfig MBean Operations and Attributes for Problem Suppression Filters

Operations and Attribute	Description
Operation: addProblemKeyFilter(*filter_pattern*)	Adds a new problem suppression filter. You pass it the regular expression that represents a pattern that you want to match. For example: `addProblemKeyFilter(".*OutOfMemory.*)`
Attribute: getProblemKeyFilters()	Returns a list of the configured problem suppression filters. For example: `getProblemKeyFilters()`
Operation: getProblemKeyFilter(*filterID*)	Returns the filter pattern associated with the specified ID. For example: `getProblemKeyFilter(id)` To find the ID, use the getProblemKeyFilters() operation.
Operation: removeProblemKeyFilter(*filterID)*	Removes the filter pattern associated with the given filter ID. For example: `removeProblemKeyFilter(id)`

To configure a problem suppression filter:

1. From the target navigation pane in Fusion Middleware Control, expand the farm, then **WebLogic Domain.**
2. Select the domain.
3. From the WebLogic Domain menu, choose **System MBean Browser.**

   The System MBean Browser page is displayed.
4. Expand **Application Defined Beans**, then **oracle.dfw,** then **domain**.*domain_name*, then **dfw.jmx.DiagnosticsConfigMBean.**
5. Select one of the **DiagnosticConfig** entries. There is one DiagnosticConfig entry for each server.
6. In the Application Defined MBeans pane, select the Operations tab.
7. Click **addProblemKeyFilter.** The Operation: addProblemKeyFilter page is displayed, as shown in the following figure:



8. For **Value,** enter a regular expression that represents a pattern that you want to match pattern. For example, in a development environment, you might want to add a filter so that incidents are not created when java.lang.IllegalStateException Java Exceptions are reported. In that case, enter the following:

   ```
   ".*[java.lang.IllegalStateException].*"
   ```

9. Click **Invoke.**
10. Click **Return** to return to the Application Defined MBeans page.

You can delete the filters using the removeProblemKeyFilter operation.

You can retrieve a specific filter, passing the ID of the filter to the getProblemKeyFilter operation.

Alternatively, you can retrieve a list of the filters using the getProblemKeyFilters attribute:

1. From the target navigation pane, expand the farm, then **WebLogic Domain.**
2. Select the domain.
3. From the WebLogic Domain menu, choose **System MBean Browser.**

   The System MBean Browser page is displayed.
4. Expand **Application Defined Beans**, then **oracle.dfw,** then **domain**.*domain_name*, then **dfw.jmx.DiagnosticsConfigMBean.**
5. Select one of the **DiagnosticConfig** entries. There is one DiagnosticConfig entry for each server.
6. In the Application Defined MBeans pane, select the Attributes tab.
7. Click **ProblemKeyFilters.**

   The list of problem suppression filters is displayed.

13.3.3 Configuring WLDF Watch and Notification for the Diagnostic Framework

Fusion Middleware configures a WLDF Diagnostics Module that contains a set of Watch and Notification rules for detecting a specific set of critical errors and creating an incident for each occurrence of those errors. The module is called Module-FMWDFW and contains the following set of Watch conditions:

Name	Description
Deadlock	Two or more Java threads have circular lock chains among their Java Monitor object usage.
StuckThread	An Oracle WebLogic Server ExecuteThread, which is blocked or busy for more than the time specified by the Oracle WebLogic Server StuckThreadMaxTime parameter.
UncheckedException	This category includes all Unchecked Exception, RuntimeException, and Errors caught by the Oracle WebLogic Server ExecuteThread, such as NullPointerException, StackOverflowError, or OutOfMemoryError.

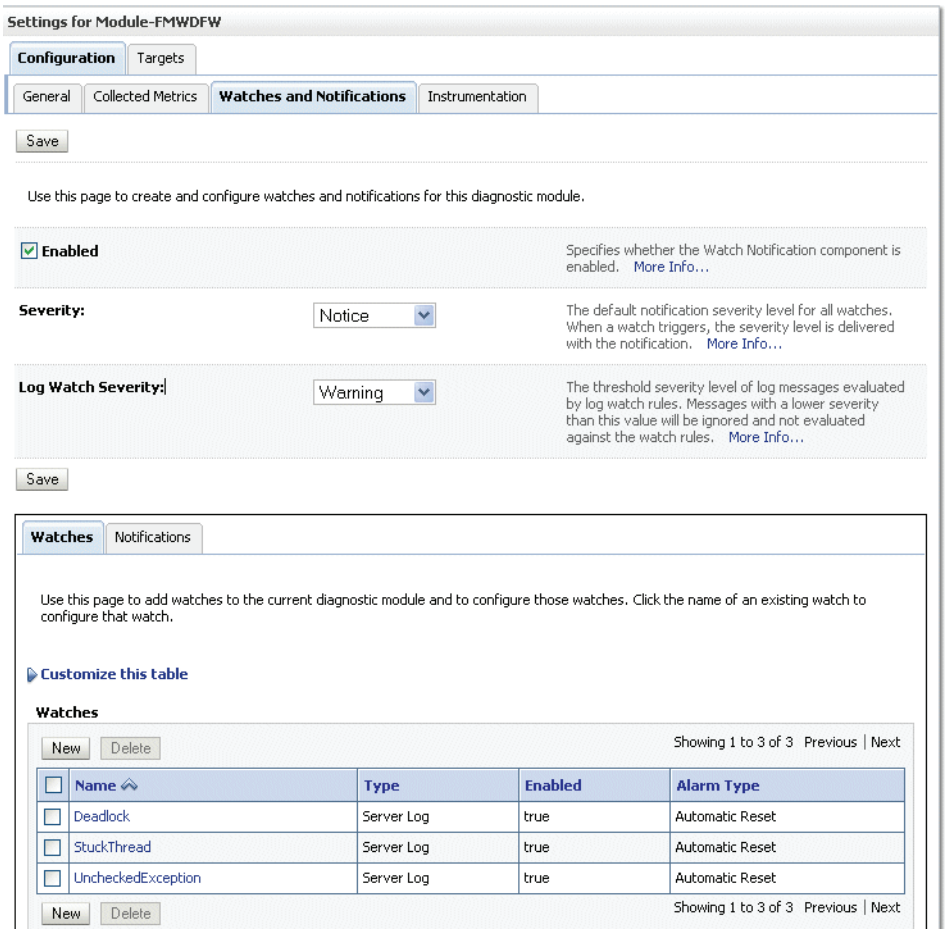
The Diagnostic Module also includes a configured WLDF JMX Notification `FMWDFW-notification` of type `oracle.dfw.wldfnotification`. You can reuse this WLDF JMX Notification for your own WLDF Watch conditions to create an incident:

1. Display the Administration Console, as described in Section 3.4.1.

2. In the Change Center, click **Lock & Edit.**
3. In the left pane, expand **Diagnostics** and select **Diagnostic Modules.**

   The Summary of Diagnostic Modules page is displayed.
4. Click **Module-FMWDFW.**

   The Settings for Module-FMWDFW page is displayed.
5. Select the Watches and Notifications tab, which is shown in the following figure:



6. Select the Watches tab and click **New.**

   The Create Watch page is displayed.
7. For **Name,** enter a name for the watch.

   You can enter any name. Alternatively, you can use the following format to force the Diagnostic Framework to use a custom message ID:

   message-id#[*application_name*]#*any_text*

   The message ID consists of a prefix that can be 1 to 6 characters, and a number, that can be 1 to 6 digits. The application name is optional. For example:

   ```
   SOA-40500#My_Watch_Name
   ```

   The following example uses the application name soa_infra:

   ```
   SOA-40501#soa-infra#My_Watch_Name
   ```

The Diagnostic Framework uses the message ID as the incident message ID in constructing the incident problem key.

8. For **Watch Type,** select a type, for example, Server log.
9. Click **Next.**
10. For **Current Watch Rule,** construct an expression. For example, to construct the expression (SEVERITY = 'Error') AND (MSGID = 'BEA-000337'):
    a. Click **Add Expressions.**
    b. For **Message Attribute,** select Severity.
    c. For **Operator,** select **=.**
    d. For **Value,** enter ERROR.
    e. Click **OK.**
    f. Click **Add Expressions.**
    g. For **Message Attribute,** select MSGID.
    h. For **Operator,** select **=.**
    i. For **Value,** enter BEA-000337.
    j. Click **OK.**
    k. In the Create Watch page, ensure that the operator selected is **AND.**
    l. Click **OK.**
11. Click **Next.**
12. Select an alarm type and click **NEXT.**
13. For **Notifications,** select **FMWDFW-notification** and move it to the Chosen box.
14. Click **Finish.**

For more information on creating watches, see "Construct watch rule expressions" in the Administration Console Online Help.

13.4 Investigating, Reporting, and Solving a Problem

This section describes how to use WLST and ADRCI commands and Remote Diagnostic Agent (RDA) to investigate and report a problem (critical error), and in some cases, resolve the problem. The section begins with a roadmap that summarizes the typical set of tasks that you must perform. It describes the following topics:

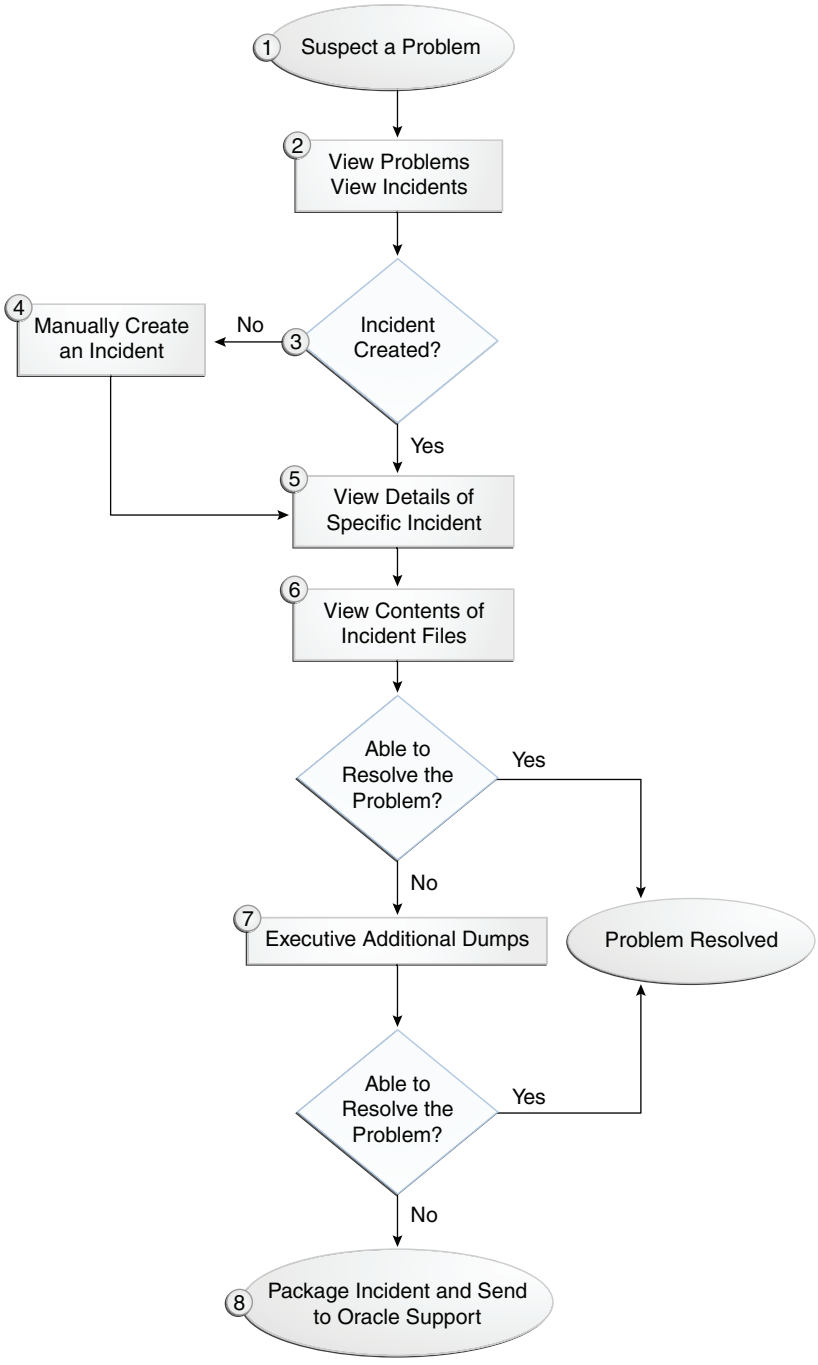
- Roadmap—Investigating, Reporting, and Resolving a Problem
- Viewing Problems and Incidents
- Analyzing Specific Problem Keys
- Working with Diagnostic Dumps
- Managing Incidents
- Generating an RDA Report

13.4.1 Roadmap—Investigating, Reporting, and Resolving a Problem

Typically, investigating, reporting, and resolving a problem begins with a critical error. This section provides an overview of that workflow.

Figure 13–4 illustrates the tasks that you complete to investigate, report, and resolve a problem.

Figure 13–4 Flow for Investigating a Problem



The following describes the workflow illustrated in Figure 13–4:

1. You notice that the system, component, or application is not functioning as expected. For example, you notice that there is a performance problem or users have reported that the application that they are trying to access is reporting errors.

2. Check to see if a problem and an incident have been created that may be related to the symptoms you are observing:
   a. View the set of problems by using the WLST `listProblems` command, as described in Section 13.4.2.1.
   b. If a problem has been created, list the incidents related to the specific problem using the `listIncidents` command, as described in Section 13.4.2.2.
3. If an incident has not been created, go to Step 4. If an incident has been created, go to Step 5.
4. If you do not see any incidents listed that are related to your problem, you can create an incident manually using the `createIncident` command to capture diagnostics for the problem.

   Consider creating an incident when you encounter an issue, such as software failure or performance problem, and you want to gather more diagnostic data. You can view the log files and the messages in the files. If there is a specific message that you believe is related to the issue you are seeing, you can use the message ID in the `createIncident` command.

   See Section 13.4.5.1 for more information about creating an incident.
5. View the details of the specific incident using the `showIncident` command, as described in Section 13.4.2.2. This command lists information about the incident, including the related message ID, the time of the incident, the ECID, and the files generated by the incident.
6. Use the `getIncidentFile` command to view the contents of files for the incident, as described in Section 13.4.2.2. The contents may provide information to guide you to the source of the problem and help in resolving it.
7. If the contents of the files for the incident do not help you to resolve the problem, you can execute additional dumps to view detailed diagnostics. For example, if you are experiencing performance problems, execute the dms.metrics dump. See Section 13.4.4 for information about the dumps available and how to execute them.
8. If you still cannot resolve the problem, package the incident, along with the RDA report, and send them to Oracle Support. See Section 13.4.5.2 and Section 13.4.6 for information about packaging incidents and generating RDA reports.

13.4.2 Viewing Problems and Incidents

You can view the set of problems, the list of incidents, and the details of a particular incident using the WLST command-line utility, as described in the following topics:

- Viewing Problems
- Viewing Incidents

13.4.2.1 Viewing Problems

You can view the set of problems by executing the WLST `listProblems` command, using the following format:

```
listProblems([adrHome] [,server])
```

The `listProblems` command lists the problems in the ADR home. Each problem has a unique ID:

```
listProblems()
```

```
Problem Id      Problem Key
        1       BEA-101020 [HTTP]
```

13.4.2.2 Viewing Incidents

You can list of all available incidents or the incidents related to a specific problem by executing the WLST `listIncidents` command, using the following format:

```
listIncidents([id], [ADRHome])
```

For example, to see the list of all incidents, use the following command:

```
listIncidents()
Incident Id     Problem Key             Incident Time
        2       BEA-101020 [HTTP]       Fri Feb 26 13:42:01 PDT 2010
        1       BEA-101020 [HTTP]       Tue Feb 23 06:17:39 PDT 2010
```

To view the incidents related to a specific problem, use the following command:

```
listIncidents(id='1')
Incident Id     Problem Key             Incident Time
        2       BEA-101020 [HTTP]       Fri Feb 26 13:42:01 PDT 2010
        1       BEA-101020 [HTTP]       Tue Feb 23 06:17:39 PDT 2010
```

To view the details of a particular incident, use the WLST `showIncident` command, using the following format:

```
showIncident(id, [adrHome] [,server])
```

For example, to see the details of incident 1, use the following command:

```
showIncident(id='1')
Incident Id: 1
Problem Id: 1
Problem Key: BEA-101020 [HTTP]
Incident Time: Tue  Feb 23 06:17:39 PDT 2010
Error Message Id:  BEA-101020
Execution Context: 0000IExqUvyAhKB5JZ4Eyf1Afdj600009i
Flood Controlled: false
Dump Files :
    dms_ecidctx1_i1.dmp
    jvm_threads2_i1.dmp
    dms_metrics3_i1.dmp
    odl_logs4_i1.dmp
    odl_logs5_i1.dmp
    diagnostic_image_AdminServer_2010_02_23_06_17_42.zip
    readme.txt
```

To view the contents of a file in the incident, use the WLST `getIncidentFile` command, using the following format:

```
getIncidentFile(id, name [,outputFile] [,adrHome] [,server])
```

For example, to view the contents for the file odl_logs4_i1.dmp use the following command:

```
getIncidentFile(id='1', name='odl_logs4_i1.dmp', outputFile='/tmp/odl_logs4_i1_
dmp.output')
```

The command writes the output to the file odl_logs4_i1_dmp.output.

13.4.3 Analyzing Specific Problem Keys

The Diagnostic Framework provides a set of well-defined problem keys for unhandled exceptions. These exceptions are either detected through the existing WLDF Watch "UncheckedException" or through the Diagnostic Framework java.lang.Thread.UncaughtExceptionHandler uncaught exception handler. Previously, the Diagnostic Framework generated problem keys with different formats for the same type of issues. Table 13–3 describes these problem keys and how to use them to investigate a problem.

Table 13–3 Uncaught Exception Problem Keys

Exception	Problem Key	Description
java.land.OutOfMemoryError	DFW-99997 [java.land.OutOfMemoryError]	Used by all java.lang.OUtOfMemoryError incidents. With each incident of this type, a jvm.classhistogram dump is executed. The dump captures statistics about the instances of classes that have been loaded and the counts of associated Objects. Review the contents of this dump for a good starting point for understanding what has been loaded into the JVM's memory. In addition, the dms.metrics dump records statistics about the overall JVM memory.
java.sql.SQLException	DFW-99996 [*ora-code* | java.sql.SQLException]] [*package.class.method*][*app-name*]	Used for all exceptions of type java.sql.SQLException, including its subclasses. The Diagnostic Framework attempts to extract the Oracle error code from the exception error message, and if it is successful, uses that in the problem key. If not, it uses the exception name. Review the text associated with the exception to get more details, such as the operation that could not be performed on the database. In addition, you can review the SQL error code details for additional information.
All others	DFW-99998 [*exception-name*][*package.class.name*][*app-name*]	Used by all other types of exceptions, such as java.lang.NullPointerException, java.io.IOException, java.lang.StringIndexOutOfBoundsException, that are not handled in a unique way. Review the text associated with the exception to get more details, such as the reason for the failure. The source line in the problem key is a best-attempt indicator of the location of the failure.

13.4.4 Working with Diagnostic Dumps

If you suspect a problem, you can make use of the built-in diagnostic dumps to report detailed diagnostics that can help diagnose the problem. Diagnostic dumps provide a means to output and record diagnostics data which serve as valuable information when diagnosing issues with Oracle Fusion Middleware components, applications, and infrastructure. The output from these dumps is intended to be used by customers and Oracle Support to diagnose issues with Oracle Fusion Middleware.

Diagnostic dumps are executed in the following ways:

- Manually, using WLST commands, as described in the following sections

  For example, if your Java EE application is hanging and you suspect a deadlock, you could use the jvm.threads dump to obtain the set of threads.

- Automatically, when the Diagnostic Framework detects a critical error and creates an incident or when the administrator creates an incident

13.4.4.1 Listing Diagnostic Dumps

You can find a list of diagnostic dumps that are available for a Managed Server by executing the WLST `listDumps` command, using the following format:

```
listDumps([appName] [,server])
```

For example, to list the available dumps for soa_server1:

```
listDumps(server='soa_server1')
Location changed to domainRuntime tree. This is a read-only tree with DomainMBean
as the root.
For more help, use help(domainRuntime)

odl.activeLogConfig
jvm.classhistogram
dms.ecidctx
wls.image
odl.logs
dms.metrics
odl.quicktrace
http.requests
jvm.threads

Use the command describeDump(name=<dumpName>) for help on a specific dump.
```

Table 13–4 lists the diagnostic dump actions that are defined by Oracle Fusion Middleware and their descriptions.

Table 13–4 Diagnostic Dump Actions

Dump Action	Description
dms.ecidctx	The data associated with a specific Execution Context ID (ECID), if specified. Otherwise, the data associated with all available ECIDs.
dms.metrics	Dynamic Monitoring Service (DMS) metrics. For information about these metrics, see "About Dynamic Monitoring Service (DMS)" in the *Oracle Fusion Middleware Performance and Tuning Guide.*
http.requests	A summary of the currently active HTTP requests.
jvm.classhistogram	A JVM class histogram, the output of which varies depending on the JVM vendor.
jvm.flightRecording	The active JRockit Flight Recorder recording.
jvm.threads	Summary statistics about the threads running in a JVM as well as performing a full thread dump.
odl.activeLogConfig	The active Java logging configuration.
odl.logs	Contents of diagnostic logs, correlated by ECID or time range.
odl.quicktrace	Quick trace messages.
wls.image	The WLDF server image dump.

In addition, Oracle SOA Suite provides diagnostic dumps, as described in "Diagnosing Problems with SOA Composite Applications" in the *Oracle Fusion Middleware Administrator's Guide for Oracle SOA Suite and Oracle Business Process Management Suite.*

13.4.4.2 Viewing a Description of a Diagnostic Dump

You can view a description of a particular dump, including the syntax for executing the dump by using the WLST `describeDump` command. You specify the name of the dump in which you are interested. For example, to view a description of the dms.metrics dump, use the following command:

```
describeDump(name='dms.metrics')
Name: dms.metrics
Description: Dumps DMS (Dynamic Monitoring Service) metrics.
Mandatory Arguments:
Optional Arguments:
    Name        Type     Description
    format      STRING   Format of the dump output; raw or xml
```

13.4.4.3 Executing Dumps

If you detect a problem and want to gather additional diagnostic data, you can invoke the `executeDump` command for a specified dump. Each dump may have mandatory or optional arguments, or both. To view the arguments for a particular dump and how to specify them, use the `describeDump` command, as described in Section 13.4.4.2.

The following example executes the dump with the name dms.metrics and the incident ID 1 and writes it to the file dumpout.txt:

```
executeDump(name='dms.metrics', outputFile='/tmp/dumpout.txt', id='1')
Dump file dms_metrics1_i1.dmp added to incident 1
```

The command writes the dump output to the information about incident 1. If you execute the showIncident command for incident 1, the output includes dms_metrics1_i1.dmp.

13.4.5 Managing Incidents

The Diagnostic Framework stores incidents, whether they are created automatically or manually, and Oracle Fusion Middleware provides tools to help you process incident reports and to package those incidents to send to Oracle Support. The following sections describe:

- Creating an Incident Manually
- Packaging an Incident
- Generating an RDA Report
- Purging Incidents

13.4.5.1 Creating an Incident Manually

System-generated problems—critical errors generated internally—are automatically added to the Automatic Diagnostic Repository (ADR). You can gather additional diagnostic data on these problems, upload diagnostic data to Oracle Support, and in some cases, resolve the problems, all with the workflow that is explained in Section 13.4.

Consider creating an incident manually when you encounter an issue, such as software failure or performance problem and you want to gather more diagnostic data, but the Diagnostic Framework has not automatically created an incident.

You use the WLST command `createIncident` to create an incident manually. You can specify an incident based on time, a message ID, an impact area, or an ECID. Then, you can inspect the content of the incident or send it to Oracle Support for further analysis.

The following describes how to manually create an incident based on a message ID:

1. Search the log files, as described in Section 12.3.2. If you find a message that you suspect is related to the issue you are seeing, you can use the message ID when you create the incident.
2. Use the following commands to invoke WLST, connect to the Managed Server and navigate to the Managed Server instance:

   ```
   java weblogic.WLST
   connect('weblogic', 'password', 'localhost:7001')
   cd('servers/server_name')
   ```

3. Create the incident, using the `createIncident` command, with the following format:

   ```
   createIncident([adrHome] [,incidentTime] [,messageId] [,ecid] [,appName]
     [,description] [,server])
   ```

   For example, to create an incident based on the error with the message ID MDS-50500, use the following command, specifying the message ID, and provide a description of the incident to help you and Oracle support track the incident:

   ```
   createIncident(messageId='MDS-50500', description='sample incident')
   Incident Id: 55
   Problem Id: 4
   Problem Key: MDS-50500 [MANUAL]
   Incident Time: 23rd February 2010 11:55:45 GMT
   Error Message Id: MDS-50500
   Flood Controlled: false
   ```

   If you do not specify a server, the incident collects information from the server to which you are connected. To specify a server, use the `server` option, as shown in the following example:

   ```
   createIncident(messageId='MDS-50500', description='sample incident',
   server='soa_server1')
   )
   ```

   If you do not specify the `adrHome` option, the incident is created in the server to which you are connected. For example, if you are connected to the Administration Server, the incident is created in the adrHome for the Administration Server.

   The Diagnostic Framework evaluates the command and invokes the appropriate diagnostic dumps. The incident and the diagnostic dumps are written to the ADR. Each diagnostic dump writes its output to the incident.

   You can view the information about the incident, as described in Section 13.4.2.2.

   You can view the information in the dumps, as described in Section 13.4.4.

13.4.5.2 Packaging an Incident

You can package the incident to facilitate sending the information to Oracle Support by using the ADR Command Interpreter (ADRCI). The ADRCI utility enables you to

investigate and report problems in a command-line environment. With ADRCI, you can package incident and problem information into a zip file for transmission to Oracle Support.

The ADRCI command-line utility is located in the following directory:

```
(UNIX) MW_HOME/wlserver_10.3/server/adr
(Windows) MW_HOME\wlserver_10.3\server\adr
```

Packaging an incident involves a three-step process:

1. Create a logical package.

   The package is denoted as logical because it exists only as metadata in the ADR. It has no content until you generate a physical package from the logical package. The logical package is assigned a package number, and you refer to it by that number in subsequent commands.

   You can create the logical package as an empty package, or as a package based on an incident number, a problem number, a problem key, or a time interval. If you create the package as an empty package, you can add diagnostic information to it in step 2.

   Creating a package based on an incident means including diagnostic data, such as dumps, for that incident. Creating a package based on a problem number or problem key means including in the package diagnostic data for incidents that reference that problem number or problem key. Creating a package based on a time interval means including diagnostic data on incidents that occurred in the time interval.

2. Add diagnostic information to the package.

   If you created a logical package based on an incident number, a problem number, a problem key, or a time interval, this step is optional. You can add additional incidents to the package or you can add any file within the ADR to the package. If you created an empty package, you must use ADRCI commands to add incidents or files to the package.

3. Generate the physical package.

   When you submit the command to generate the physical package, ADRCI gathers all required diagnostic files and adds them to a zip file in a designated directory. You can generate a complete zip file or an incremental zip file. An incremental file contains all the diagnostic files that were added or changed since the last zip file was created for the same logical package. You can create incremental files only after you create a complete file, and you can create as many incremental files as you want. Each zip file is assigned a sequence number so that the files can be analyzed in the correct order.

   Zip files are named according to the following format:

   ```
   packageName_mode_sequence.zip
   ```

   In the format:

   - `packageName` consists of a portion of the problem key followed by a timestamp.
   - `mode` is either `COM` or `INC`, for complete or incremental.
   - `sequence` is an integer.

For example, to package an incident, take the following steps:

1. Set the ORACLE_HOME and LD_LIBRARY_PATH environment variables to point to the following directory:

   MW_HOME/wlserver_10.3/server/adr

2. Invoke ADRCI. For example:

   MW_HOME/wlserver_10.3/server/adr/adrci

3. Use the SET BASE command to specify the ADR Base and the SET HOMEPATH command to specify the ADR home that contains the incident. The path for the HOMEPATH is relative to the ADR Base.

   ```
   SET BASE /scratch/oracle1/Oracle/Middleware/user_projects/domains/soa_
   domain/servers/soa_server1/adr
   SET HOMEPATH diag/ofm/soa_domain/soa_server1
   ```

4. Generate the logical package:

   ```
   IPS CREATE PACKAGE INCIDENT incident_number
   ```

   For example, the following command creates a package based on incident 1:

   ```
   IPS CREATE PACKAGE INCIDENT 1
   Created package 1 based on incident id 1, correlation level typical
   ```

   ADRCI assigns the logical package a number.

5. Optionally, you can add diagnostic information to the logical package. You can add the following types of information:

   – All diagnostic information for a particular incident. For example, you can add another incident that you think might be related to the incident you are packaging, using the following command:

     ```
     IPS ADD INCIDENT incident_number PACKAGE package_number
     ```

   – A named file within the ADR. For example, if an incident is related to an application, you can add the .ear file for the application. You can also add a readme file with notes you provide to Oracle Support. For example, to add a file to the package, use the following command:

     ```
     IPS ADD FILE filespec PACKAGE package_number
     ```

6. Generate the physical package using the following command:

   ```
   IPS GENERATE PACKAGE package_number IN path
   ```

   For example, to generate a package with the number 1, use the following command:

   ```
   IPS GENERATE PACKAGE 1 in /tmp
   Generated package 1 in file /tmp/BEA337Web_20100223132315_COM_1.zip, mode
   complete
   ```

   This generates a complete physical package (zip file) in the designated path.

   See Also: The "ADRCI: ADR Command Interpreter" chapter of the *Oracle Database Utilities*

13.4.5.3 Purging Incidents

By default, incidents are purged when the total size of all incidents exceed 500 MB. You can use the maxTotalIncidentSize MBean parameter to change this value, as described in Section 13.3.1.

You can manually purge incidents using the ADRCI command. You can purge based on an ID or range of IDs, the age of the incident, or the type of incident. For example, to purge incidents that are older than 60 minutes, use the following command:

```
purge -age 60
```

See the "ADRCI: ADR Command Interpreter" chapter of the *Oracle Database Utilities*.

13.4.6 Generating an RDA Report

You can use the Remote Diagnostic Agent (RDA), a command-line diagnostic tool, to provide a comprehensive picture of your environment. Additionally, RDA can provide recommendations on various topics, for example configuration and security. This aids you and Oracle Support in resolving issues.

RDA is designed to be as unobtrusive as possible; it does not modify systems in any way. A security filter is provided if required.

For more information about RDA, see the readme file, which is located at:

```
(UNIX) ORACLE_HOME/rda/README_Unix.txt
(Windows) ORACLE_HOME\rda\README_Windows.txt
```

Part VI

Advanced Administration

This part describes advanced administration tasks, such as managing the metadata repository and changing the network configuration, that involve reconfiguring Oracle Fusion Middleware.

Part VI contains the following chapters:

14

Managing the Metadata Repository

Many Oracle Fusion Middleware components use metadata repositories to hold configuration information about the component and metadata for applications. This chapter provides information on managing the metadata repositories used by Oracle Fusion Middleware.

It contains the following topics:

- Understanding a Metadata Repository
- Creating a Database-Based Metadata Repository
- Managing the MDS Repository
- Managing Metadata Repository Schemas
- Purging Data

> **Note:** For information about managing a metadata repository for IBM WebSphere, see "Configuring Metadata Services (MDS) on IBM WebSphere" in the *Oracle Fusion Middleware Third-Party Application Server Guide*.

14.1 Understanding a Metadata Repository

A metadata repository contains metadata for Oracle Fusion Middleware components, such as Oracle BPEL Process Manager, Oracle B2B, and Oracle WebCenter Portal. It can also contain metadata about the configuration of Oracle Fusion Middleware and metadata for your applications.

Oracle Fusion Middleware supports multiple repository types. A repository type represents a specific schema or set of schemas that belong to a specific Oracle Fusion Middleware component (for example, Oracle BPEL Process Manager or Oracle Internet Directory.) Oracle Fusion Middleware supports Edition-Based Redefinition (EBR), which enables you to upgrade the database component of an application while it is in use, thereby minimizing or eliminating down time. The schemas in a repository can be EBR-enabled schemas.

A particular type of repository, the Oracle Metadata Services (MDS) Repository, contains metadata for certain types of deployed applications. This includes custom Java EE applications developed by your organization and some Oracle Fusion Middleware component applications, such as Oracle B2B. For information related specifically to the MDS Repository type, see Section 14.3.

You can create a database-based repository or, for MDS, a database-based repository or a file-based repository. For production environments, you use a database-based

repository. Most components, such as Oracle BPEL Process Manager, require that a schema be installed in a database, necessitating the use of a database-based repository. MDS supports Edition-Based Redefinition (EBR) enabled schemas.

> **Note:** After the database for the metadata repository has been used for the Oracle Fusion Middleware installation, the database, service name or SID cannot be changed.

14.2 Creating a Database-Based Metadata Repository

You use the Oracle Fusion Middleware Metadata Repository Creation Utility (RCU) to create the metadata repository in an existing database.

You can use RCU to create multiple repositories in a single database. You can use it to create the MDS Repository or a repository for metadata for particular components, such as Oracle WebCenter Portal. RCU creates the necessary schemas for the components. See Appendix D for a list of the schemas and their tablespaces and datafiles.

With RCU, you can also drop component schemas.

For information about the supported versions of database platforms and versions, and other prerequisites for the database, see:

`http://www.oracle.com/technology/software/products/ias/files/fusion_certification.html`

> **Note:** Oracle recommends that all metadata repositories reside on a database at the same site as the components to minimize network latency issues.

For information about managing an MDS Repository, see Section 14.3.

> **See Also:** *Oracle Fusion Middleware Repository Creation Utility User's Guide* for information about how to use RCU to create a database-based metadata repository

14.3 Managing the MDS Repository

Oracle Metadata Services (MDS) Repository contains metadata for certain types of deployed applications. Those deployed applications can be custom Java EE applications developed by your organization and some Oracle Fusion Middleware component applications, such as Oracle B2B and Oracle Web Services Manager. A Metadata Archive (MAR), a compressed archive of selected metadata, is used to deploy metadata content to the MDS Repository, which contains the metadata for the application.

You should deploy your applications to MDS in the following situations, so that the metadata can be managed after deployment:

- The application contains seeded metadata packaged in a MAR.
- You want to enable user personalizations at run time.
- You have a custom Oracle WebCenter Portal application.
- You have a SOA composite application (SCA).

The following topics provide information about the MDS Repository:

- Understanding the MDS Repository
- Registering and Deregistering a Database-Based MDS Repository
- Registering and Deregistering a File-Based MDS Repository
- Changing the System Data Source
- Using System MBeans to Manage an MDS Repository
- Viewing Information About an MDS Repository
- Configuring an Application to Use a Different MDS Repository or Partition
- Moving Metadata from a Source System to a Target System
- Moving from a File-Based Repository to a Database-Based Repository
- Deleting a Metadata Partition from a Repository
- Purging Metadata Version History
- Managing Metadata Labels in the MDS Repository

> **See Also:** *Oracle Fusion Middleware High Availability Guide* for information about using an MDS Repository with Oracle Real Application Clusters (Oracle RAC)

14.3.1 Understanding the MDS Repository

The MDS framework allows you to create customizable applications. A customized application contains a base application (the base documents) and one or more layers containing customizations. MDS stores the customizations in a metadata repository and retrieves them at run time to merge the customizations with the base metadata to reveal the customized application. Since the customizations are saved separately from the base, the customizations are upgrade safe; a new patch to base can be applied without breaking customizations. When a customized application is launched, the customization content is applied over the base application.

A customizable application can have multiple customization layers. Examples of customization layers are *industry* and *site*. Each layer can have multiple customization layer values, but typically only one such layer value from each layer is applied at run time. For example, the industry layer for a customizable application can contain values for health care and financial industries; but in the deployed customized application, only one of the values from this layer is used at a time. For more information about base documents and customization layers, see "Customizing Applications with MDS" in the *Oracle Fusion Middleware Fusion Developer's Guide for Oracle Application Development Framework*.

An MDS Repository can be file-based or database-based. For production environments, you use a database-based repository. You can have more than one MDS Repository for a domain.

A database-based MDS Repository provides the following features that are not supported by a file-based MDS Repository:

- Efficient query capability: A database-based MDS Repository is optimized for set-based queries. As a result, it provides better performance on such searches with the database repository.

  The MDS Repository query API provides constructs to define the query operation and to specify conditions on metadata objects. These conditions are a set of criteria

that restrict the search results to a certain set of attribute types and values, component types, text content, and metadata paths. The API allows multiple conditions to be combined to achieve dynamic recursive composition using OR and AND constructs.

- Atomic transaction semantics: A database-based MDS Repository uses the database transaction semantics, which provides rollbacks of failed transactions, such as failed imports or deployments.
- Versioning: A database-based MDS Repository maintains versions of the documents in a database-based repository. Versioning allows changes to metadata objects to be stored as separate versions rather than simply overwriting the existing data in the metadata repository. It provides version history, as well as the ability to label versions so that you can access the set of metadata as it was at a given point in time.
- Isolate metadata changes: A database-based MDS Repository has the capability to isolate metadata changes in a running environment and test them for a subset of users before committing them for all users.
- Support for external change detection based on polling: This allows one application to detect changes that another application makes to shared metadata. For example, if you have an application deployed to Managed Servers A and B in a cluster, and you modify the customizations for the application deployed to Managed Server A, the data is written to the database-based repository. The application deployed to Managed Server B uses the updated customizations. This supports high availability (in particular, active/active scenarios.)
- Clustered updates: A database-based MDS Repository allows updates from multiple hosts to the metadata. For a file-based MDS Repository, updates can be made from only one host at a time.

Multiple applications can share metadata by configuring a shared metadata repository. When you do this, changes made by one application to the metadata in this repository are seen by other applications using the shared repository, if you configure external change detection for the applications.

In an MDS Repository, each application, including Oracle Fusion Middleware components, is deployed to its own partition. A **partition** is an independent logical repository within one physical MDS Repository, whether it is database-based or file-based.

For information about deploying applications and associating them with an MDS Repository, see Chapter 10.

Note the following points about patching the MDS Repository:

- An MDS Repository must be registered with a domain before it is patched. Otherwise, the applied patches cannot be rolled back and no additional patches can be applied.
- You can apply patches to the following:
  - The MDS metadata
  - An MDS jar file
  - An MDS shared library
  - An MDS schema in the database-based metadata repository. The patch can include additive changes such as adding a new column or increasing the size of a column. Note that you cannot rollback this type of patch.

- The MDS database PL/SQL in the database-based metadata repository. The patch can include changes to a PL/SQL package or new PL/SQL packages and procedures.
- An MDS schema or PL/SQL in the database-based metadata repository that requires a corresponding MDS JAR file patch.

14.3.1.1 Databases Supported by MDS

The MDS Repository supports Oracle databases, as well as non-Oracle databases, including SQL Server, DB2, and MySQL. For more information about the supported versions of these databases, see:

http://www.oracle.com/technology/software/products/ias/files/fusion_certification.html

For information about using these databases with the MDS Repository, see "Supported Databases for the MDS Schema" in the *Oracle Fusion Middleware System Requirements and Specifications.*

14.3.1.2 Understanding MDS Operations

You can use Fusion Middleware Control or WLST commands to perform most operations on the MDS Repository. However, for some operations that do not have a custom user interface in Fusion Middleware Control or do not have WLST commands, you must use the System MBeans.

The sections that follow describe using Fusion Middleware Control and WLST commands to perform the operations, unless only System MBeans are supported. In that case, the sections describe how to use System MBeans to perform the operation.

You can view information about the repositories, including the partitions and the applications deployed to each partition. You can also perform operations on the partitions, such as purging, deleting, importing metadata, or exporting metadata.

Note the following when you use the MDS operations described in the sections that follow:

- The export operation exports a versioned stripe (either the tip version or based on a label) of metadata documents from an MDS Repository partition to a file system directory or archive. If you export to a directory, the directory must be accessible from the host where the application is running. If you export to an archive, the archive can be located on the system on which you are executing the command.

  Because versioning of metadata is not supported for file-based repositories, the tip version (which is also the only version) is exported from a file-based repository.
- The import operation imports metadata documents from a file system directory or archive to a MDS Repository partition. If you exported to a directory, the directory must be accessible from the host where the application is running. If you exported to an archive, the archive can be located on the system on which you are executing the command.

  If the target repository is a database-based repository, the metadata documents are imported as new tip versions. If the target repository is a file-based repository, the metadata documents are overwritten.

Note:

- To use the custom WLST MDS commands, you must invoke the WLST script from the Oracle Common home. See Section 3.5.1.1 for more information.
- For more information about the custom WLST MDS commands, see "Metadata Services (MDS) Custom WLST Commands" in the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

Table 14–1 lists the logical roles needed for each operation. The roles apply whether the operations are performed through the WLST commands, Fusion Middleware Control, or MBeans.

Table 14–1 MDS Operations and Required Roles

Operation	Logical Role
Clear cache	Operator role for application
Clone metadata partition	Admin role for domain
Create metadata label	Admin role for application
Create metadata partition	Admin role for domain
Delete metadata	Admin role for application
Delete metadata label	Admin role for application
Delete metadata partition	Admin role for domain
Deregister metadata database repository	Admin role for domain
Deregister metadata file repository	Admin role for domain
Export metadata	Operator role for application
Import MAR	Admin role for application
Import metadata	Admin role for application
List metadata label	Monitor role for application
Promote metadata label	Admin role for application
Purge metadata	Admin role for application
Purge metadata labels	Admin role for application
Register metadata database repository	Admin role for domain
Register metadata file repository	Admin role for domain

For information about how these roles map to WebLogic Server roles, see "Mapping of Logical Roles to WebLogic Roles" in the *Oracle Fusion Middleware Application Security Guide*.

14.3.2 Registering and Deregistering a Database-Based MDS Repository

The following topics describe how to register and deregister a database-based MDS Repository:

- Registering a Database-Based MDS Repository Using Fusion Middleware Control

- Deregistering a Database-Based MDS Repository

14.3.2.1 Registering a Database-Based MDS Repository

Before you can deploy an application to an MDS Repository, you must register the repository with the Oracle WebLogic Server domain. You can register a database-based MDS Repository using Fusion Middleware Control or WLST, as described in the following topics:

- Registering a Database-Based MDS Repository Using Fusion Middleware Control
- Registering a Database-Based MDS Repository Using WLST

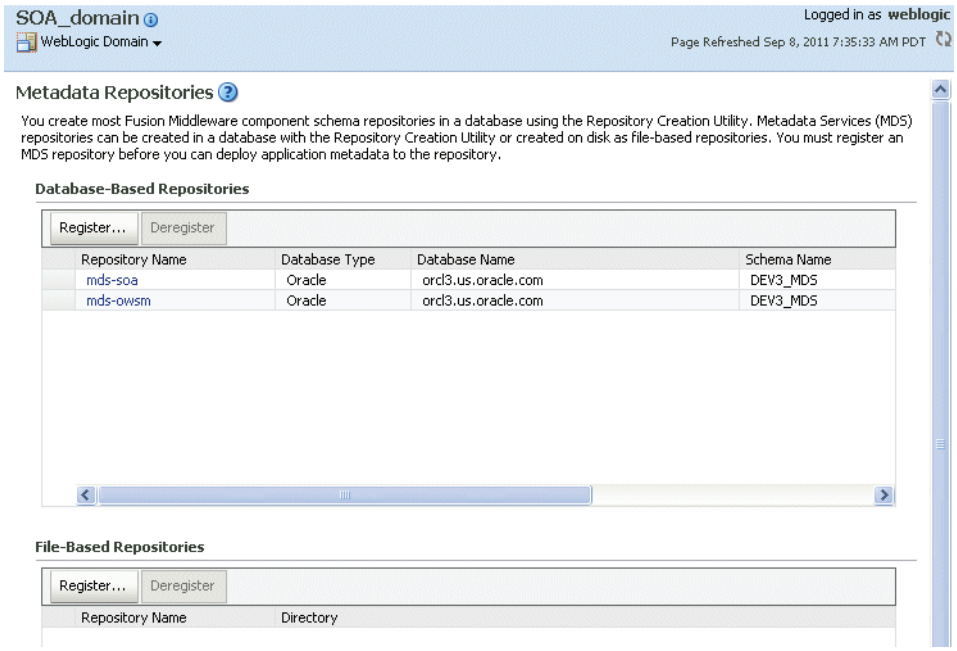
14.3.2.1.1 Registering a Database-Based MDS Repository Using Fusion Middleware Control

You create a database-based MDS Repository using RCU, as described in Section 14.2.

To register a database-based MDS Repository using Fusion Middleware Control:

1. From the navigation pane, expand the farm, then **WebLogic Domain.**
2. Select the domain.
3. From the WebLogic Domain menu, choose **Metadata Repositories.**

   The Metadata Repositories page is displayed, as shown in the following figure:



4. In the Database-Based Repositories section, click **Register.**

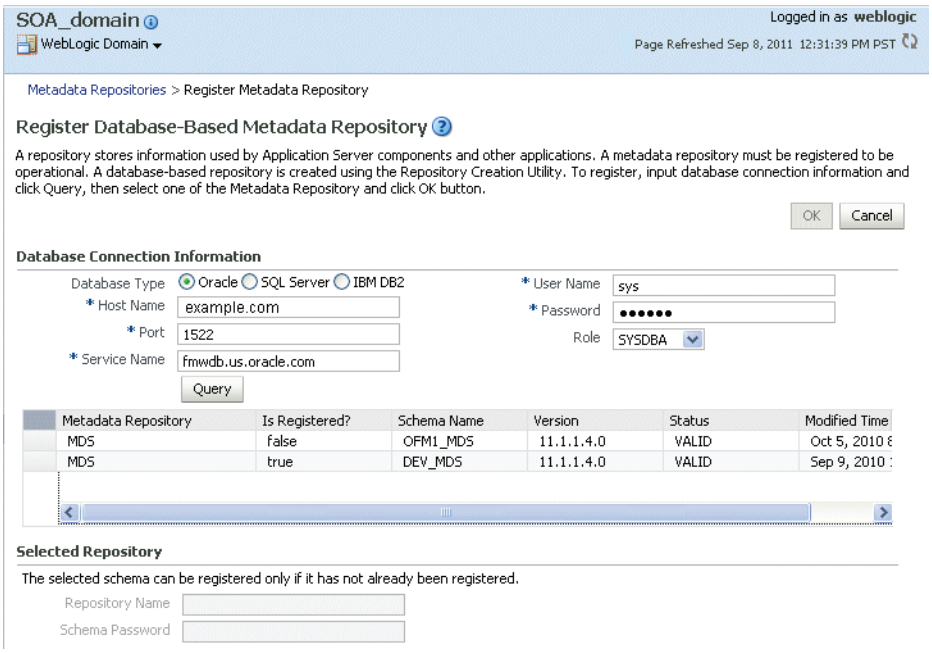
   The Register Database-Based Metadata Repository page is displayed.
5. In the Database Connection section, enter the following information:
   - For **Database Type,** select the type of database.
   - For **Host Name,** enter the name of the host.
   - For **Port,** enter the port number for the database, for example: `1521`.
   - For **Service Name,** enter the service name for the database. The default service name for a database is the global database name, comprising the database

name, such as `orcl`, and the domain name. In this case, the service name would be `orcl.domain_name.com`.

- For **User Name,** enter a user name for the database which is assigned the SYSDBA role, for example: `SYS`.
- For **Password,** enter the password for the user.
- For **Role,** select a database role, for example, **SYSDBA.**

6. Click **Query.**

   A table is displayed that the metadata repositories in the database, as shown in the following figure:



7. Select a repository, then enter the following information:

   - For **Repository Name,** enter a name.
   - For **Schema Password**, enter the password you specified when you created the schema.

8. Click **OK.**

   The repository is registered with the Oracle WebLogic Server domain and is targeted to the Administration Server. To target the repository to other servers, see Section 14.3.2.2.

   In addition, a system data source is created with the name mds-*repository_name*. Global transaction support is disabled for the data source.

14.3.2.1.2 Registering a Database-Based MDS Repository Using WLST

To register a database-based MDS Repository using the command line, you use the WLST `registerMetadataDBRepository` command. For example, to register the MDS Repository mds-repos1, use the following command:

```
registerMetadataDBRepository(name='mds-repos1', dbVendor='ORACLE',
    host='hostname', port='1521', dbName='ora11',
    user='username', password='password', targetServers='server1')
```

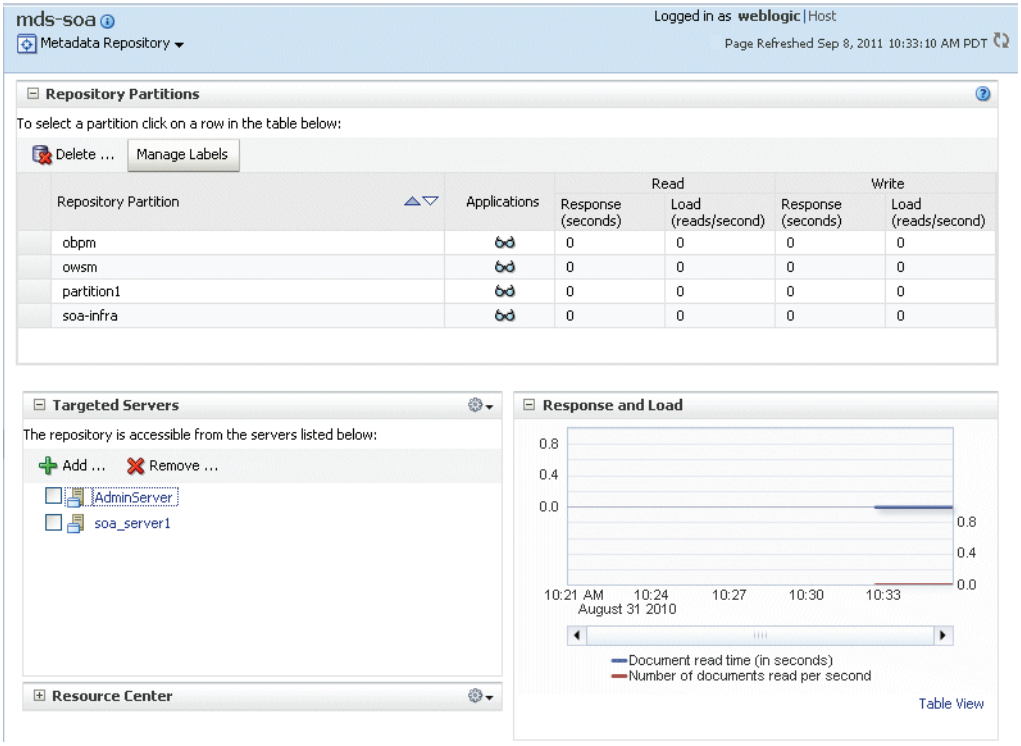
14.3.2.2 Adding or Removing Servers Targeted to the MDS Repository

When you register an MDS Repository using Fusion Middleware Control, the repository is targeted to the Administration Server. You can target the repository to additional servers or remove servers as targets.

To target the MDS Repository to additional servers:

1. From the navigation pane, expand the farm, then **Metadata Repositories.**
2. Select the repository.

   The repository home page is displayed, as shown in the following figure:



3. In the Targeted Servers section, click **Add.**

   The Target the Repository dialog box is displayed.
4. Select the server or cluster and click **Target.**

   You can expand the cluster to see the servers in the cluster. However, if you select a cluster, the repository is targeted to all servers in the cluster.
5. When the operation completes, click **Close.**

   The server is now listed in the Targeted Servers section.

To remove a server as a target for the repository:

1. From the navigation pane, expand the farm, then **Metadata Repositories.**
2. Select the repository.

   The repository home page is displayed.
3. In the Targeted Servers section, select the target server and click **Remove.**

   The Untarget the Repository dialog box is displayed.
4. Select the server or cluster and click **Untarget.**

You can expand the cluster to see the servers in the cluster. However, if you select a cluster, the repository will be untargeted from all servers in the cluster.

5. When the operation completes, click **Close.**

14.3.2.3 Deregistering a Database-Based MDS Repository

Deregistration does not result in loss of data stored in the repository. However, any applications using a deregistered repository will not function after the repository is deregistered. You must ensure that no application is using the repository before you deregister it.

You can deregister a database-based MDS Repository using Fusion Middleware Control or WLST, as described in the following topics:

- Deregistering a Database-Based MDS Repository Using Fusion Middleware Control
- Deregistering a Database-Based MDS Repository Using WLST

14.3.2.3.1 Deregistering a Database-Based MDS Repository Using Fusion Middleware Control

To deregister an MDS Repository using Fusion Middleware Control:

1. From the navigation pane, expand the farm, then **WebLogic Domain.**
2. Select the domain.
3. From the WebLogic Domain menu, choose **Metadata Repositories.**

   The Metadata Repositories page is displayed.

   Alternatively, you can navigate to the Register Metadata Repositories page by choosing **Administration,** then **Register/Deregister** from the Metadata Repository menu when you are viewing a metadata repository home page.
4. Select the repository from the table.
5. Click **Deregister.**
6. Click **Yes** in the Confirmation dialog box.

14.3.2.3.2 Deregistering a Database-Based MDS Repository Using WLST

To deregister a database-based MDS Repository using the command line, you use the WLST `deregisterMetadataDBRepository` command. For example, to deregister the MDS Repository mds-repos1, use the following command:

```
deregisterMetadataDBRepository(name='mds-repos1')
```

14.3.3 Registering and Deregistering a File-Based MDS Repository

The following topics describe how to register and deregister a file-based metadata repository:

- Creating and Registering a File-Based MDS Repository
- Deregistering a File-Based MDS Repository

14.3.3.1 Creating and Registering a File-Based MDS Repository

You can create a file-based MDS Repository and register it with an Oracle WebLogic Server domain using Fusion Middleware Control.

To create and register a file-based repository using Fusion Middleware Control:

1. From the navigation pane, expand the farm, then **WebLogic Domain.**

2. Select the domain.
3. From the WebLogic Domain menu, choose **Metadata Repositories.**

   The Metadata Repositories page is displayed.
4. In the File-Based Repository section, click **Register.**

   The Register Metadata Repository page is displayed.
5. Enter the following information:
   - For **Name,** enter a name. For example, enter repos1. The prefix `mds-` is added to the name and a repository with the name mds-repos1 is registered. If you enter a name that begins with `mds-`, a repository with the given name is registered.
   - For **Directory,** specify the directory. The Administration Server and Managed Servers that run the applications that use this repository must have write access to the directory.

     Note the following:
     - If the specified path exists on the file system, the metadata file repository is registered; all the subdirectories under this path are automatically loaded as partitions of this file-based repository.
     - If the path specified does not exist, a directory with this name is created on the file system during the registration. Because there are no partitions created yet, there are no subdirectories to load.
     - If the specified path is invalid and cannot be created for some reason, such as permission denied, an error is displayed and the registration fails.
     - If the specified path exists, but as a file not a directory, an error is not displayed and the registration succeeds.
6. Click **OK.**

The repository is created and registered and is displayed on the Metadata Repositories page.

You can now create and delete partitions. Those changes are reflected in the directory on the file system.

You can also create a file-based repository using system MBeans. For information about using the System MBean Browser, see Section 14.3.5.

14.3.3.2 Deregistering a File-Based MDS Repository

You can deregister a file-based MDS Repository using Fusion Middleware Control.

To deregister a file-based repository using Fusion Middleware Control:

1. From the navigation pane, expand the farm, then **WebLogic Domain.**
2. Select the domain.
3. From the WebLogic Domain menu, choose **Metadata Repositories.**

   The Metadata Repositories page is displayed.
4. In the File-Based Repository section, select the repository and click **Deregister.**
5. Click **OK** in the Confirmation dialog box.

   If the file-based repository is valid, it is removed from the repository list. Otherwise, an error is displayed.

You can also deregister a file-based repository using system MBeans. For information about using the System MBean Browser, see Section 14.3.5.

14.3.4 Changing the System Data Source

You can change the system data source to reassociate an application to a new repository. You can change the database or the schema that contains the data source. To do so, you can use Oracle WebLogic Server Administration Console or Fusion Middleware Control. To use Fusion Middleware Control:

1. From the navigation pane, expand the farm, then **WebLogic Domain.**
2. Select the domain.
3. From the WebLogic Domain menu, choose **JDBC Data Sources.**

   The JDBC Data Sources page is displayed.
4. Select the data source you want to change and click **Edit.**

   The Edit JDBC Data Source page is displayed.
5. Select the Connection Properties tab.
6. To change the database, modify the **Database URL** field. For example:

   ```
   jdbc:oracle:thin:@hostname.domainname.com:1522/orcl
   ```
7. For **Password,** enter the password for the database.
8. To change the schema, modify the Properties section, changing the value for **user.**
9. If the database is a DB2 database, add the property sendStreamAsBlob, with a value of true.
10. Click **Apply.**
11. Restart the servers that use this data source.

14.3.5 Using System MBeans to Manage an MDS Repository

Although most procedures in this chapter discuss using Fusion Middleware Control or WLST to manage the MDS Repository, you can also use system MBeans:

1. In Fusion Middleware Control, from the navigation pane, navigate to the domain and select it. From the WebLogic Domain menu, choose **System MBean Browser.**

   The System MBean Browser page is displayed.
2. In the page's navigation pane, expand **Application Defined MBeans,** then expand **oracle.mds.lcm.** Expand the domain, then **MDSDomainRuntime,** and then select **MDSDomainRuntime.**
3. In the Application Defined MBeans pane, select the Operations tab.
4. Click one of the operations, such as **registerMetadataFileRepository.**

   The Operations page is displayed.
5. In the Value column, enter values for the operation.
6. Click **Invoke.**

14.3.6 Viewing Information About an MDS Repository

You can view information about an MDS Repository using Fusion Middleware Control or system MBeans, as described in the following topics:

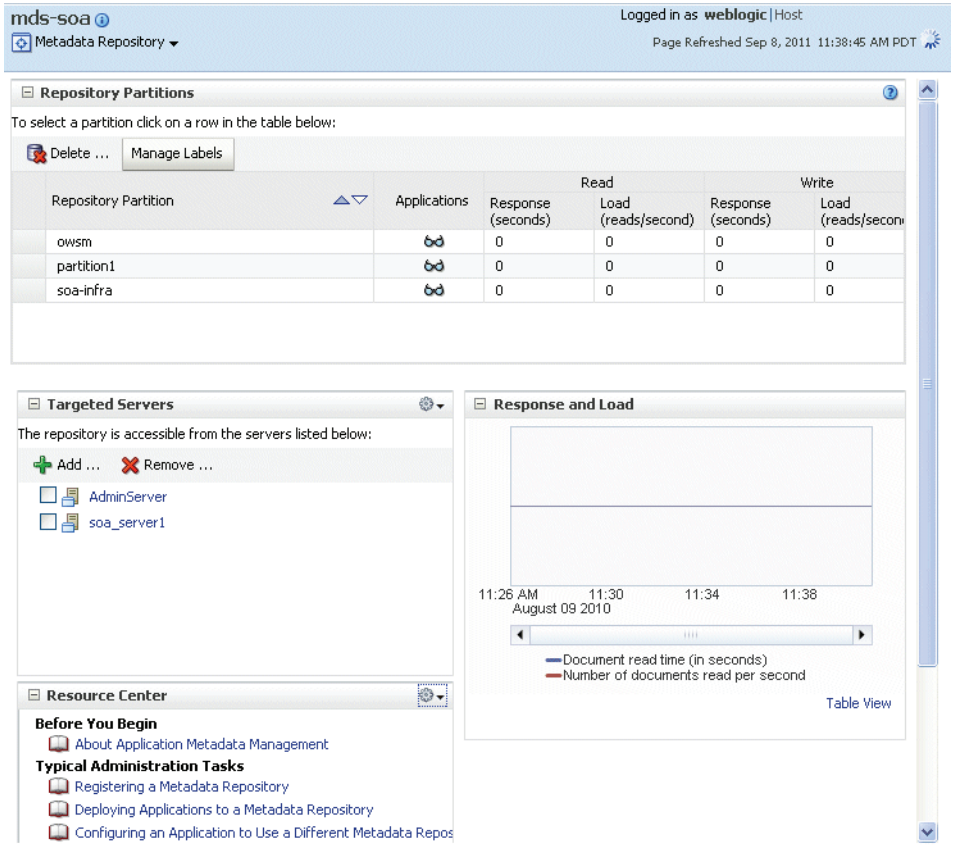
- Viewing Information About an MDS Repository Using Fusion Middleware Control
- Viewing Information About an MDS Repository Using System MBeans

14.3.6.1 Viewing Information About an MDS Repository Using Fusion Middleware Control

To view information about an MDS Repository using Fusion Middleware Control:

1. From the navigation pane, expand the farm and then expand **Metadata Repositories.**
2. Select the repository.

   The following figure shows the home page for an MDS Repository:



3. To see which applications use the repository, click the icon in the Applications column. The Applications using the partition dialog box is displayed, with tabs for Deployed Applications and Referenced by Applications:
   - The Deployed Applications tab shows the list of applications whose metadata is deployed to the repository partition.
   - The Referenced by Applications tab shows the list of applications that refer to the metadata stored in the repository partition.

From this page, you can also:

- Delete partitions, as described in Section 14.3.10.1.
- Delete labels, as described in Section 14.3.12.5.
- Add or remove targeted servers, as described in Section 14.3.2.2.

14.3.6.2 Viewing Information About an MDS Repository Using System MBeans

You can use the System MBean operations listPartitions, listRepositories, and listRepositoryDetails to get a list of partitions in the repository, a list of repositories, and details of the repository registered with the domain:

1. In Fusion Middleware Control, from the navigation pane, navigate to the domain and select it. From the WebLogic Domain menu, choose **System MBean Browser.**

   The System MBean Browser page is displayed.

2. In the page's navigation pane, expand **Application Defined MBeans,** then expand **oracle.mds.lcm.** Expand the domain, then **MDSDomainRuntime,** and then select **MDSDomainRuntime.**
3. In the Application Defined MBeans pane, select the Operations tab.
4. Click one of the operations, such as listPartitions, listRepositories, and listRepositoryDetails.

   The Operations page is displayed.

5. Click **Invoke.**

   The information is displayed in the Return Value table.

For information about changing the MDS configuration attributes for an application, see Section 10.9.

14.3.7 Configuring an Application to Use a Different MDS Repository or Partition

When you deploy an application, you can associate it with an MDS Repository. You can subsequently change the MDS Repository or partition to which an application is associated, using WLST or Fusion Middleware Control. For example, a different repository contains different metadata that needs to be used for a particular application.

To associate an application with a new MDS Repository or partition, you can either:

- Redeploy the application, specifying the new repository or partition.

  To create a new partition, you can either:

  – Clone the partition to a different repository. Cloning the partition is valid only with a database-based repository with databases of the same type and version. When you clone the partition, you preserve the metadata version history, including any customizations and labels.

    Section 14.3.7.1 describes how to clone a partition and how to redeploy the application, specifying the partition that you have cloned.

  – Create a new partition, then export the metadata from the current partition and import the metadata into the new partition.

    Section 14.3.7.2 describes how to create the partition and export and import data and how to redeploy the application, specifying the new repository or partition.

- Change the system data source. When you change the system data source, you can change the database or the schema in which it is stored.

  Section 14.3.4 describes how to change the system data source.

14.3.7.1 Cloning a Partition

You can clone a partition to the same repository or a different repository using the system MBean cloneMetadataPartition. Both the original repository and the target repository must be a database-based repository.

To clone the partition, and then redeploy the application to a new repository or to the same repository:

1. Clone the partition, using the cloneMetadataPartition operation on the system MBean. The following example clones partition1 from the old repository to the new repository:

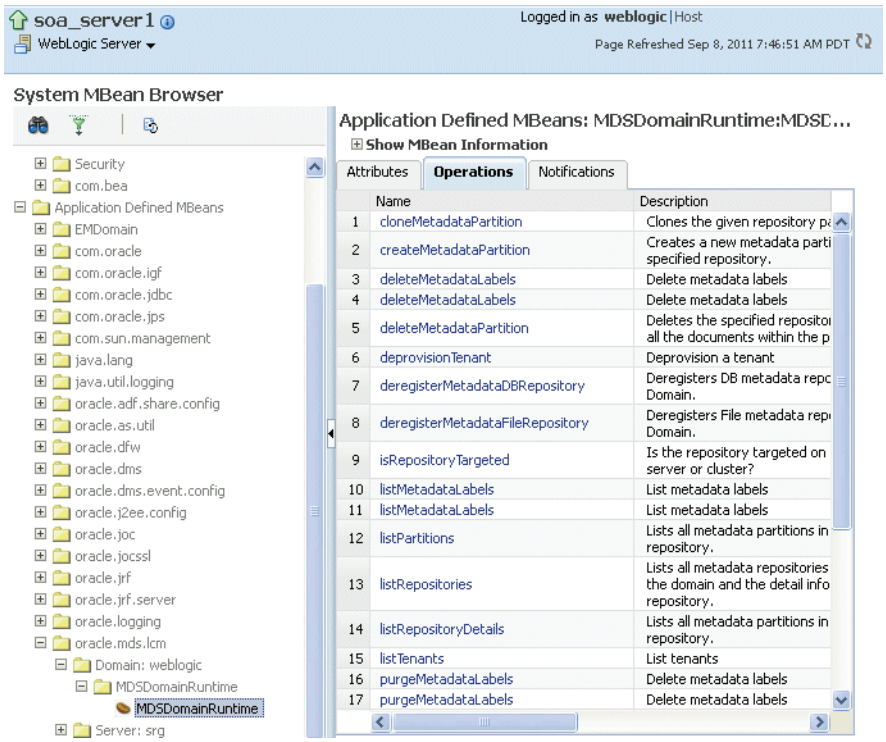
   a. In Fusion Middleware Control, from the navigation pane, navigate to the Managed Server from which the application is deployed. From the WebLogic Server menu, choose **System MBean Browser.**

      The System MBean Browser page is displayed.

   b. In the System MBean Browser's navigation pane, expand **Application Defined MBeans,** then expand **oracle.mds.lcm.** Expand the domain, and then **MDSDomainRuntime.** Select **MDSDomainRuntime.**

   c. In the Application Defined MBeans pane, select the Operations tab.

      The following figure shows the System MBeans Browser with the Application Defined MBeans pane:



   d. Select **cloneMetadataPartiton.**

      The Operation: cloneMetadataPartiton page is displayed.

e. In the Parameters table, enter the following values:

   - For **fromRepository,** enter the name of the metadata repository that contains the metadata partition from which the metadata documents are to be cloned.
   - For **fromPartition,** enter the name of the partition from which the metadata documents are to be cloned.
   - For **toRepository,** enter the name of the metadata repository to which the metadata documents from the source repository partition are to be cloned.
   - For **toPartition,** enter the name of metadata repository partition to be used for the target partition. The name must be unique within the repository. If you do not supply a value for this parameter, the name of the source partition is used for the target partition.

     If the toRepository name is the same as the original repository, you must enter a partition name and the name must be unique within the repository.

f. Click **Invoke.**

g. Verify that the partition has been created by selecting the repository in the navigation pane. The partition is listed in the Partitions table on the Metadata Repository home page.

2. Redeploy the application, as described in Section 10.4.3, Section 10.5.3, or Section 10.6.3, depending on the type of application. When you do so, you specify the new partition and repository in the Application Attributes page:

   a. To change the repository, click the icon next to the **Repository Name.** In the Metadata Repositories dialog box, select the repository and click **OK.**

   b. To change the partition, enter the partition name in **Partition Name.**

14.3.7.2 Creating a New Partition and Reassociating the Application to It

You can create a new partition in the same or a different repository by redeploying the application and specifying the new partition. Then, you transfer the metadata to the new partition using WLST.

You can use this procedure to transfer metadata between two different types of repositories (file-based to database-based or from an Oracle Database to another database.)

To create a new partition and reassociate the application to it:

1. Export the metadata from the source partition to a directory on the file system using the WLST `exportMetadata` command:

```
exportMetadata(application='sampleApp', server='server1',
       toLocation='/tmp/myrepos/mypartition', docs='/**')
```

2. Redeploy the application, as described in Section 10.4.3, Section 10.5.3, or Section 10.6.3, depending on the type of application. When you do so, you specify the new partition and repository in the Application Attributes page:

   a. To change the repository, click the icon next to the **Repository Name.** In the Metadata Repositories dialog box, select the repository and click **OK.**

   b. To change the partition, enter the partition name in **Partition Name.**

3. Import the metadata from the file system to the new partition using the WLST `importMetadata` command:

```
importMetadata(application='sampleApp', server='server1',
       fromLocation='/tmp/myrepos/mypartiton', docs='/**')
```

4. Optionally, deregister the original repository, as described in Section 14.3.3.2 or Section 14.3.2.3.

Alternatively, you can create a new partition using the WLST command `createMetadataPartition`. The partition name must be unique within the repository. If the partition parameter is missing, the name of the source partition is used for the target partition. The following example creates the partition partition1:

```
createMetadataPartition(repository='mds-repos1', partition='partition1')
```

14.3.8 Moving Metadata from a Source System to a Target System

You can transfer the metadata in MDS from one partition to another. As an example, you want to move an application from a test system to a production system. You have a test application that is deployed in a domain in the test system and a production application deployed in a domain in the production system. You want to transfer the customizations from the test system to the production system. To do that, you transfer the metadata from the partition in the test system to a partition in the production system.

To transfer the metadata from one partition to another, you export the metadata from the partition and then import it into the other partition. You can use Fusion Middleware Control or WLST to transfer the metadata, as described in the following topics:

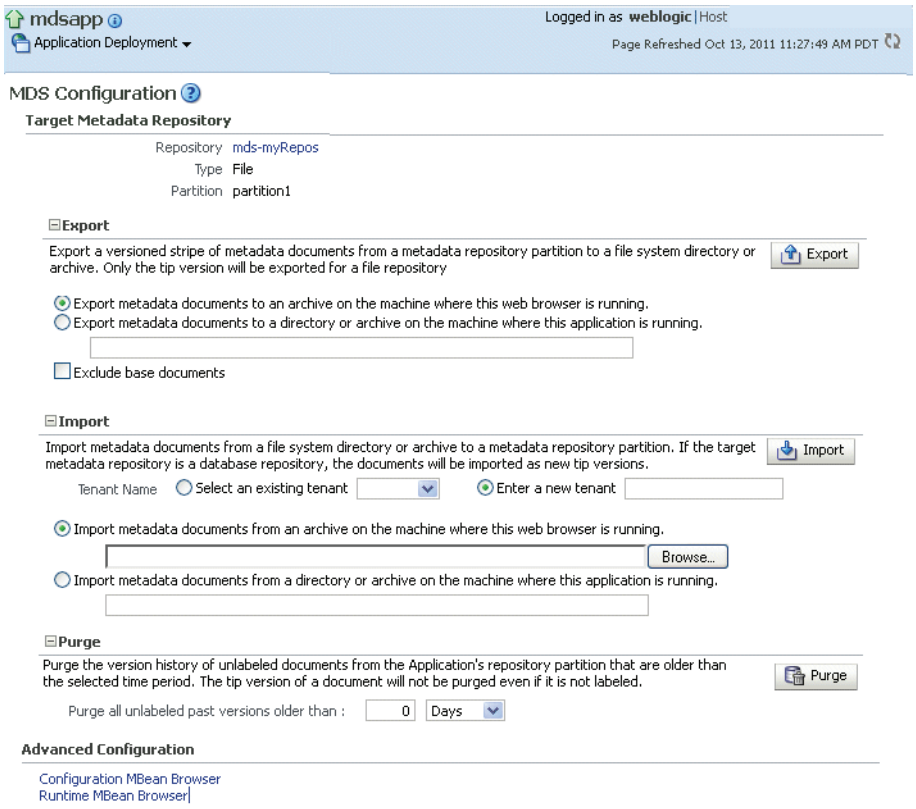
- Transferring Metadata Using Fusion Middleware Control
- Transferring Metadata using WLST

14.3.8.1 Transferring Metadata Using Fusion Middleware Control

To use Fusion Middleware Control to transfer metadata:

1. From the navigation pane, expand the farm for the source, expand **Application Deployments**, then select the application.
2. From the Application Deployment menu, choose **MDS Configuration.**

   The MDS Configuration page is displayed, as shown in the following figure:



3. In the Export section, select one of the following:

   - **Export metadata documents to an archive on the machine where this web browser is running.**

     Click **Export.**

     The export operation exports a zip file. Depending on the operating system and browser, a dialog box is displayed that asks you if you want to save or open the file.

   - **Export metadata documents to a directory or archive on the machine where this application is running.**

     Enter a directory location or archive to which the metadata can be exported.

     The target directory or archive file (.jar, .JAR, .zip or .ZIP) to which to transfer the documents selected from the source partition. If you export to a directory, the directory must be a local or network directory or file where the application is physically deployed. If you export to an archive, the archive can be located on a local or network directory or file where the application is physically deployed, or on the system on which you are executing the command.

     If the location does not exist in the file system, a directory is created except that when the names ends with .jar, .JAR, .zip or .ZIP, an archive file is created. If the archive file already exists, the exportMetadata operation overwrites the file.

     Click **Export.** Then, in the Confirmation dialog box, click **Close.**

   If you check **Exclude base documents,** this operation exports only the customizations, not the base documents. See Section 14.3.1 for information about base documents and customizations.

4. If the target application is on a different system, copy the exported metadata to that system.
5. From the navigation pane for the target system, expand the farm, expand **Application Deployments**, then select the application.
6. From the Application Deployment menu, choose **MDS Configuration.**

   The MDS Configuration page is displayed
7. In the Import section, select one of the following:
   - **Import metadata documents from an archive on the machine where this web browser is running.**
   - **Import metadata documents from a directory or archive on the machine where this application is running.**

   Enter the location of the directory or archive that contains the exported metadata. If you specify a directory, include the subdirectory with the partition name in the specification. The directory or archive file must be a local or network directory or file where the application is physically deployed.
8. Click **Import.**
9. In the Confirmation dialog box, click **Close.**

14.3.8.2 Transferring Metadata using WLST

To use WLST to transfer metadata:

1. Export the metadata from the original partition using the `exportMetadata` command:

   ```
   exportMetadata(application='sampleApp', server='server1',
          toLocation='/tmp/myrepos/mypartition', docs='/**')
   ```

   This command exports a versioned stripe of the metadata documents from the metadata partition to a file system directory. Only customization classes declared in the cust-config element of adf-config.xml are exported. If there is no cust-config element declared in adf-config.xml, all customization classes are exported.

   To export all customizations, use the option restrictCustTo="%".
2. If the production application is on a different system, copy the exported metadata to that system.
3. Import the metadata to the other partition using the WLST `importMetadata` command:

   ```
   importMetadata(application='sampleApp', server='server1',
          fromLocation='/tmp/myrepos/mypartiton', docs='/**')
   ```

   The value of the `fromLocation` parameter must be on the same system that is running WLST or on a mapped network drive or directory mount. You cannot use direct network references such as `\\mymachine\repositories\`.

   Only customization classes declared in the cust-config element of adf-config.xml are imported. If there is no cust-config element declared in adf-config.xml, all customization classes are imported.

   To import all customizations, use the option restrictCustTo="%".

14.3.9 Moving from a File-Based Repository to a Database-Based Repository

You can move from a file-based repository to a database-based repository. (You cannot move from a database-based repository to a file-based repository.)

To minimize downtime, take the following steps to move an application's metadata from a file-based repository to a database-based repository:

1. Use RCU to create schemas in the new repository, as described in Section 14.2.
2. Create a new partition using the WLST command `createMetadataPartition` with same name as source partition:

   ```
   createMetadataPartition(repository='mds-repos1', partition='partition1')
   ```

3. Export the metadata from the source partition to a directory on the file system:

   ```
   exportMetadata(application='sampleApp', server='server1',
         toLocation='/tmp/myrepos/partition1', docs='/**')
   ```

4. Import the metadata from the file system to the new partition:

   ```
   importMetadata(application='sampleApp', server='server1',
          fromLocation='/tmp/myrepos/partition1', docs='/**')
   ```

5. Redeploy the application, as described in Section 10.4.3, Section 10.5.3, or Section 10.6.3, depending on the type of application. When you do so, you specify the new partition and repository in the Application Attributes page:
   a. To change the repository, click the icon next to the **Repository Name.** In the Metadata Repositories dialog box, select the repository and click **OK.**
   b. To change the partition, enter the partition name in **Partition Name.**
6. Deregister the file-based repository, as described in Section 14.3.3.2.

14.3.10 Deleting a Metadata Partition from a Repository

You can delete metadata partitions if there are no applications either deployed to the partition or referring to the partition. You may want to delete a metadata partition from the repository in the following circumstances:

- When you undeploy an application. Oracle Fusion Middleware leaves the metadata partition because you may still want the metadata, such as user customizations, in the partition. If you do not need the metadata, you can delete the partition.
- When you have transferred metadata from one partition to another and configured the application to use the new partition.
- When you have cloned a partition and configured the application to use the new partition.

Note that deleting a partition deletes all the data contained in the partition.

You can delete a metadata partition using WLST or Fusion Middleware Control, as described in the following topics:

- Deleting a Metadata Partition Using Fusion Middleware Control
- Deleting a Metadata Partition Using WLST

14.3.10.1 Deleting a Metadata Partition Using Fusion Middleware Control

To delete a metadata partition from a repository partition using Fusion Middleware Control:

1. From the navigation pane, expand the farm and then expand **Metadata Repositories.**
2. Select the repository.

   The repository home page is displayed.
3. In the Repository Partitions section, select the partition and click **Delete.**
4. In the confirmation dialog box, click **OK.**

14.3.10.2 Deleting a Metadata Partition Using WLST

To delete a metadata partition from a repository, you can use the WLST command `deleteMetadataPartition`. For example, to delete the metadata partition from the file-based repository mds-repos1, use the following command:

```
deleteMetadataPartition(repository='mds-repos1', partition='partition1')
```

14.3.11 Purging Metadata Version History

For database-based MDS Repositories, you can purge the metadata version history from a partition. (File-based MDS Repositories do not maintain version history.) This operation purges version history of unlabeled documents from the application's repository partition. The tip version (the latest version) is not purged, even if it is unlabeled.

To purge metadata labels, you use the `purgeMetadataLabels` command, as described in Section 14.3.12.4. Then, you can purge the metadata version history.

Consider purging metadata version history on a regular basis as part of MDS Repository maintenance, when you suspect that the database is running out of space or performance is becoming slower. This operation may be performance intensive, so plan to do it in a maintenance window or when the system is not busy.

For specific recommendations for particular types of applications, see the documentation for a particular component.

You can purge metadata version history using WLST or Fusion Middleware Control, as described in the following topics:

- Purging Metadata Version History Using Fusion Middleware Control
- Purging Metadata Version History Using WLST
- Enabling Auto-Purge

14.3.11.1 Purging Metadata Version History Using Fusion Middleware Control

To use Fusion Middleware Control to purge the metadata version history:

1. From the navigation pane, expand the farm, expand **Application Deployments,** then select the application.
2. From the Application Deployment menu, choose **MDS Configuration.**

   For Oracle SOA Suite, you can expand **SOA** in the navigation tree, then select **soa-infra.** From the SOA Infrastructure menu, select **Administration,** then **MDS Configuration.**

The MDS Configuration page is displayed.

3. In the Purge section, in the **Purge all unlabeled past versions older than** field, enter a number and select the unit of time. For example, enter **3** and select **months.**
4. Click **Purge.**
5. In the Confirmation dialog box, click **Close.**

14.3.11.2 Purging Metadata Version History Using WLST

To use WLST to purge metadata version history, use the `purgeMetadata` command. You specify the documents to be purged by using the `olderThan` parameter, specifying the number of seconds. The following example purges all documents older than 100 seconds:

```
purgeMetadata(application='sampleApp', server='server1', olderThan=100)
```

14.3.11.3 Enabling Auto-Purge

You can enable automatic purging using the MDSAppConfig MBean:

1. In Fusion Middleware Control, from the navigation pane, navigate to the domain and select it. From the WebLogic Domain menu, choose **System MBean Browser.**

   The System MBean Browser page is displayed.
2. Expand **Application Defined MBeans,** then **oracle.adf.share.config,** then **Server:** ***name***, then **Application:** ***name***, then **ADFConfig,** then **ADFConfig,** and **ADFConfig.**
3. Select **MDSAppConfig.**

   The Application Defined MBeans page is displayed.
4. For **AutoPurgeTimeToLive,** enter a value, in seconds.
5. Navigate up to ADFConfig (the parent of MDSAppConfig) and select it.
6. In the Operations tab, click **Save.**

14.3.12 Managing Metadata Labels in the MDS Repository

A **metadata label** is a means of selecting a particular version of each object from a metadata repository partition. Conceptually, it is a collection of document versions, one version per document, representing a *horizontal stripe* through the various document versions. This stripe comprises the document versions which were the tip versions (latest versions) at the time the label was created.

You can use a label to view the metadata as it was at the point in time when the label was created. You can use the WLST commands to support logical backup and recovery of an application's metadata contained in the partition.

Labels are supported only in database-based repositories.

Document versions belonging to a label are not deleted by automatic purging, unless the label is explicitly deleted. In this way, creating a label guarantees that a view of the metadata as it was at the time the label was created remains available until the label is deleted.

When an application that contains a MAR is deployed, a label with the prefix `postDeployLabel_` is created. For example: postDeployLabel_mdsappdb_ mdsappdb.mar_2556916398.

Each time you patch the MAR, a new deployment label is created, but the previous deployment label is not deleted Similarly, when you undeploy an application that contains a MAR, the application is undeployed, but the label remains in the metadata repository partition.

If you delete a deployment label, when the application is restarted, the MAR is automatically redeployed, and the deployment label is also re-created.

The following topics describe how to manage labels:

- Creating Metadata Labels
- Listing Metadata Labels
- Promoting Metadata Labels
- Purging Metadata Labels
- Deleting Metadata Labels

14.3.12.1 Creating Metadata Labels

To create a label for a particular version of objects in a partition in an MDS Repository, you use the WLST command `createMetadataLabel`. For example, to create a label named prod1 for the application my_mds_app, use the following command:

```
createMetadataLabel(application='my_mds_app', server='server1', name='prod1')
Executing operation: createMetadataLabel.

Created metadata label "prod1".
```

If the application has more than one version, you must use the applicationVersion parameter to specify the version.

14.3.12.2 Listing Metadata Labels

You can list the metadata labels for a particular application. To do so, use the WLST command `listMetadataLabel`. For example, to list the labels for the application my_mds_app, use the following command:

```
listMetadataLabels(application='my_mds_app', server='server1')
Executing operation: listMetadataLabels.

Database Repository partition contains the following labels:
prod1
prod2
postDeployLabel_mdsappdb_mdsappdb.mar_2556916398
```

If the application has more than one version, you must use the applicationVersion parameter to specify the version.

14.3.12.3 Promoting Metadata Labels

You can promote documents associated with a metadata label so that they become the latest version. That is, you can promote them to the tip. Promote a label if you want to roll back to an earlier version of all of the documents captured by the label.

To promote a label to the tip, use the WLST command `promoteMetadataLabel`. For example to promote the label prod1, use the following command:

```
promoteMetadataLabel(application='my_mds_app', server='server1', name='prod1')
Location changed to domainRuntime tree. This is a read-only tree with DomainMBean
as the root.
```

```
For more help, use help(domainRuntime)

Executing operation: promoteMetadataLabel.

Promoted metadata label "prod1" to tip.
```

If the application has more than one version, you must use the applicationVersion parameter to specify the version.

14.3.12.4 Purging Metadata Labels

You can purge metadata labels that match the given name pattern or age, allowing you to purge labels that are no longer in use. This reduces the size of the database, improving performance. You must delete the labels associated with unused metadata documents before you can purge the documents and revision history from the repository.

You may want to delete a label for older applications that were undeployed, but the labels were not deleted. Each time you patch the MAR, a new label is created, but the previous label is not deleted.

You can use Fusion Middleware Control or WLST to purge metadata labels, as described in the following topics:

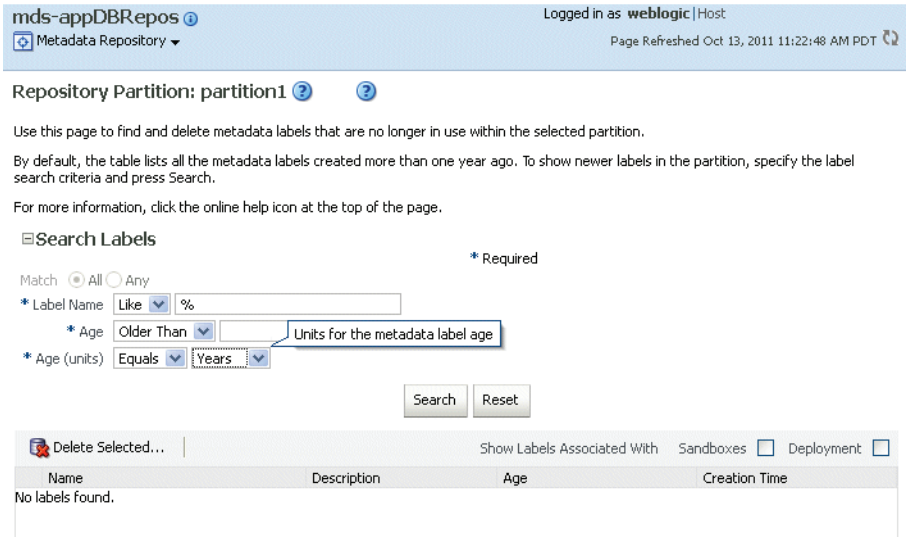
- Purging Metadata Labels Using Fusion Middleware Control
- Purging Metadata Labels Using WLST

14.3.12.4.1 Purging Metadata Labels Using Fusion Middleware Control To purge metadata labels using Fusion Middleware Control:

1. Expand the farm, then expand **Metadata Repositories.**
2. Select the repository.

   The repository home page is displayed.
3. Select a partition and click **Manage Labels.**

   The Manage Labels page is displayed, as shown in the following figure:



By default, the table lists all metadata labels created in the selected partition that are more than one year old and that are not deployed or associated with a sandbox.

4. To search for a particular label or labels, you can:
   - For **Label Name,** select an operator and enter the filter criteria. The characters are case sensitive. You can use the following wildcards:
     - Percent (%): Matches any number of characters
     - Underscore (_): Matches a single character
     - Backslash (\): Used as an escape character for the wildcards

     For example, the string `postDeployLabel%` returns any label beginning with `postDeployLabel`. As a result, it displays labels associated with a deployed MAR.
   - For **Age,** enter a number, such as 2. (The only operator available is Older Than.
   - For **Age (units),** select a unit, such as Hours, Days, Weeks, Months, Years. The only operator available is Equals.
5. Click **Search.**
6. By default, labels associated with sandboxes and deployed applications are not shown. To display those labels, select **Sandboxes** or **Deployment** or both. Note the following:
   - You cannot delete a label associated with a sandbox.
   - If you select **Deployment,** the labels that are associated with MAR deployments are displayed.
7. Select the label and click **Delete Selected.**
8. In the confirmation box, click **OK.**

If you want to purge all unused labels, for a particular deployed application:

1. Select **Deployment.**
2. Filter by name, using the string `postDeployLabel_`*`application_name`*`%`.
3. Select all but the latest (which is in use) to delete. (The most recent label---the one that is currently being used---is listed first.)
4. Click **Delete Selected.**

14.3.12.4.2 Purging Metadata Labels Using WLST

You can purge metadata labels that match the given pattern or age, using the WLST command `purgeMetadataLabels`. The command purges the labels that match the criteria specified, but it does not delete the metadata documents that were specified by the labels.

For example, to purge all metadata labels that match the specified namePattern and that are older than 30 minutes:

```
purgeMetadataLabels(repository='mds-myRepos', partition='partition1',
                    namePattern='prod*', olderThanInMin='30')
Location changed to domainRuntime tree. This is a read-only tree with DomainMBean
as the root.
For more help, use help(domainRuntime)

Executing operation: purgeMetadataLabels.
```

```
The following metadata labels were purged:
repository=mds-soa,parititon=partition1,namePattern=prod*,olderThanInMin=30:
```

14.3.12.5 Deleting Metadata Labels

To delete a specified metadata label, you use the WLST command `deleteMetadataLabel`. For example, to delete a label named prod1 for the application my_mds_app, use the following command:

```
deleteMetadataLabel(application='my_mds_app', server='server1', name='prod1')
```

If the application has more than one version, you must use the applicationVersion parameter to specify the version.

To find the labels associated with an application, use the `listMetadataLabels` command, as described in Section 14.3.12.2.

14.4 Managing Metadata Repository Schemas

The following topics describe how to manage the metadata repository schemas:

- Changing Metadata Repository Schema Passwords
- Changing the Character Set of the Metadata Repository

14.4.1 Changing Metadata Repository Schema Passwords

The schema passwords are stored in the database.

For example, to change the password of the schema OFM_MDS:

1. Connect to the database using SQL*Plus. Connect as a user with SYSDBA privileges.
2. Issue the following command:

   ```
   SQL> ALTER USER schema IDENTIFIED BY new_password;
   ```

   For example, to change the OFM_ MDS password to `abc123`:

   ```
   SQL> ALTER USER OFM_MDS IDENTIFIED BY abc123;
   ```

3. If you change the MDS Repository schema password, you must change the password for the corresponding MDS Repository data source, using Oracle WebLogic Server Administration Console:
   a. From Domain Structure, expand **Services,** then **Data Sources.**
   b. Click the data source that is related to the MDS Repository.
   c. Click the Configuration tab, then the Connection Pool tab.
   d. For **Password,** enter the new password.
   e. Click **Save.**
   f. Restart the Managed Servers that consume the data source.

14.4.2 Changing the Character Set of the Metadata Repository

For information about changing the character set of metadata repository that is stored in an Oracle Database, see *Oracle Database Globalization Support Guide*:

http://www.oracle.com/technetwork/database/enterprise-edition/documentation/index.html

Oracle recommends using Unicode for all new system deployments. Deploying your systems in Unicode offers many advantages in usability, compatibility, and extensibility. Oracle Database enables you to deploy high-performing systems faster and more easily while utilizing the advantages of Unicode. Even if you do not need to support multilingual data today, nor have any requirement for Unicode, it is still likely to be the best choice for a new system in the long run and ultimately saves time and money and gives you competitive advantages in the long term.

When storing the metadata in a SQL Server database, if the character set being considered for your locale is not case neutral, the case-sensitive collation must be selected during the creation of the database instance. Unicode support is the default when creating the MDS schema for SQL Server using RCU. You may overwrite this default to use non-unicode schema if that meets your requirements.

14.5 Purging Data

When the amount of data in Oracle Fusion Middleware databases grows very large, maintaining the databases can become difficult and can affect performance. In some cases, Oracle Fusion Middleware automatically purges data. In other cases, Oracle Fusion Middleware provides methods to manage growth, including scripts to purge data that can accumulate over time and that can affect performance.

Many of the Oracle Fusion Middleware components provide scripts written as PL/SQL procedures to purge the data. The scripts are located in:

ORACLE_HOME/common/sql/*component-name*_purge_*purgetype*.sql

For example, a script that purges logs for Oracle Business Process Management is located in:

ORACLE_HOME/common/sql/bpm_purge_logs.sql

Table 14–2 provides pointers to information about purging data for Oracle Fusion Middleware components.

Table 14–2 Purging Data Documentation

Component	Description
MDS Repository	See Section 14.3.11 for information on automatically and manually purging data.
Oracle Access Management Identity Federation	No configuration required. Automatically purges data.
Oracle Application Development Framework	See "Cleaning Up Temporary Storage Tables" in the *Oracle Fusion Middleware Fusion Developer's Guide for Oracle Application Development Framework*.
Oracle Application Development Framework Business Components	Use the following script to purge rows in the database used by Oracle ADF Business Components to store user session state and temporary persistent collections: *ORACLE_COMMON_HOME*/common/sql/adfbc_purge_statesnapshots.sql The PS_TXN table is automatically purged.
Oracle BI Enterprise Edition	No configuration required. Automatically purges data.

Table 14–2 (Cont.) Purging Data Documentation

Component	Description
Oracle Business Intelligence Publisher	Delete job history, as described in "Deleting a Job History" in the *Oracle Fusion Middleware Report Designer's Guide for Oracle Business Intelligence Publisher.*
Oracle Identity Manager	No configuration required. Automatically purges data.
Oracle Internet Directory	No configuration required. Automatically purges data.
Oracle Real-Time Decisions	No configuration required. Automatically purges data.
Oracle SOA Suite	See "Managing Database Growth" in the *Oracle Fusion Middleware Administrator's Guide for Oracle SOA Suite and Oracle Business Process Management Suite.*
Oracle Web Services Manager	No configuration required. Automatically purges data.
Oracle WebCenter Content	Export the data with deletion, as described in "Exporting Data in Archives." Then, remove the collection, as described in "Removing a Collection." Both sections are in the *Oracle WebCenter Content System Administrator's Guide for Content Server*.
Oracle WebCenter Portal Analytics	See Section 14.5.2.2.
Oracle WebCenter Portal Spaces and Lists	Purge MDS metadata, as described in Section 14.3.11.
Oracle WebCenter Portal's Activity Stream	See Section 14.5.2.1.
Oracle WebLogic Server: JAXWS Web Services	Clean up the Web service persistence store, as described in "Cleaning Up Web Service Persistence" in *Oracle Fusion Middleware Programming Advanced Features of JAX-WS Web Services for Oracle WebLogic Server*. Use the defaultMaximumObjectLifetime field of the WebServicePersistenceMBean to set the maximum lifetime of the objects. See "Understanding WebLogic Server MBeans" in *Oracle Fusion Middleware Developing Custom Management Utilities With JMX for Oracle WebLogic Server*.
Oracle WebLogic Server: JMS	See "Configuring Basic JMS System Resources" and "Managing JMS Messages" in *Oracle Fusion Middleware Configuring and Managing JMS for Oracle WebLogic Server*. Also see "Tuning WebLogic JMS" in *Oracle Fusion Middleware Performance and Tuning for Oracle WebLogic Server*.
Oracle WebLogic Server: Oracle Infrastructure Web Services	Use the following script to purge data if WS-RM uses a database store: *ORACLE_COMMON_HOME*/common/sql/ows_purge_wsrm_msgs.sql
Oracle WebLogic Server: Session persistence for JDBC or file-based data sources	No configuration required. Automatically purges data.
Oracle WebLogic Server: Stateful EJBs	No configuration required. Automatically purges data.

In certain circumstances, you can consider using Oracle Scheduler to automate the running of the scripts. For example, you may want to set up a scheduled job to purge the last 14 days for completed instances for Oracle SOA Suite.

Oracle Scheduler, an enterprise job scheduler, is part of Oracle Database. Oracle Scheduler is implemented by the procedures and functions in the DBMS_SCHEDULER PL/SQL package.

For information about Oracle Scheduler, see " Oracle Scheduler Concepts" and "Creating, Running, and Managing Jobs" in the *Oracle Database Administrator's Guide*.

14.5.1 Purging Oracle Infrastructure Web Services Data

Use the following script to purge data if WS-RM uses a database store:

ORACLE_COMMON_HOME/common/sql/ows_purge_wsrm_msgs.sql

14.5.2 Purging Oracle WebCenter Portal Data

The following topics describe purging Oracle WebCenter Portal data:

- Purging Oracle WebCenter Portal's Activity Stream Data
- Purging Oracle WebCenter Portal's Analytics Data
- Partitioning Oracle WebCenter Portal's Analytics Data

14.5.2.1 Purging Oracle WebCenter Portal's Activity Stream Data

Oracle WebCenter Portal's Activity Streaming provides a set of WLST commands for purging database records in a nonpartitioned environment. Purging is necessary when a database contains records that are not needed as an analysis in reports or when the performance of Oracle WebCenter Portal decreases because of the large volume of data.

To purge Oracle WebCenter Portal's Activity Stream data, you use the following WLST commands:

- archiveASByDate: Archives activity stream data that is older than a specified date.
- archiveASByDeletedObjects: Archives activity stream data associated with deleted objects
- archiveASByClosedSpaces: Archives activity stream data associated with Spaces that are currently closed.
- archiveASByInactiveSpaces: Archives activity stream data associated with Spaces that have been inactive since a specified date.
- restoreASByDate: Restores archived activity stream data from a specified date into production tables.

Note that you must invoke the WLST script from the Oracle home containing Oracle WebCenter Portal Activity Streaming. Do not use the WLST script in the WebLogic Server home.

For more information about these commands, see "Activity Stream" in the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

14.5.2.2 Purging Oracle WebCenter Portal's Analytics Data

Oracle WebCenter Portal's Analytics provides a script for purging database records in a nonpartitioned environment. Purging is necessary when a database contains records that are not needed for analysis in reports or when the performance of Oracle WebCenter Portal decreases because of the large volume of data.

The script, analytics_purge_facts.sql, deletes all fact tables that meet the specified criteria.

When Oracle WebCenter Portal's Analytics runs in a partitioned environment, you should use the drop partitioning feature of the database before running these scripts.

14.5.2.2.1 Loading the Oracle WebCenter Portal Purge Package Before you run the script for the first time, you must install the purge package into the database by running the analytics_purge_package script:

1. Log in to the database as the schema user for the ACTIVITIES schema.
2. Execute the analytics_purge_package script. For example, for an Oracle Database:

   ```
   @ORACLE_HOME/oracle_common/common/sql/oracle/analytics_purge_package.sql
   ```

   For a DB2 database, use the following command:

   ```
   db2 -td@ -f analytics_purge_package.sql
   ```

14.5.2.2.2 Running the Oracle WebCenter Portal Purge Script The location of the analytics_purge_facts.sql script differs depending on the type of database used:

- Oracle Database:

  ```
  ORACLE_HOME/oracle_common/common/sql/oracle/analytics_purge_facts.sql
  ```

- SQL Server:

  ```
  ORACLE_HOME/oracle_common/common/sql/sqlserver/analytics_purge_facts.sql
  ```

- DB2:

  ```
  ORACLE_HOME/oracle_common/common/sql/db2/analytics_purge_facts.sql
  ```

The analytics_purge_facts.sql script takes the following parameters:

- Month From: The script purges data that was created after the beginning of the specified month. Enter the month in the format MM. For example, 08 to specify August.
- Year From: With the Month From parameter, the script purges data that was created after the beginning of the specified month in the specified year. Enter the year in YYYY format. For example, 2010.
- Month To: The script purges data that was created through the end of the specified month. Enter the month in the format MM. For example, if you specify 09 for September, the script purges all data that was created before the end of September.
- Year To: With the Month To parameter, the script purges data that was created through the end of the specified month in the specified year. Enter the year in YYYY format. For example, 2010.
- Record Batch Size: The maximum size of records to commit at one time.
- Max Run Time: The maximum amount of time, in minutes, that the you want the process to run. When the process reaches this time, it stops, regardless of the progress of the purge.

Note: You cannot delete the current month. If you specify the current month, the script returns an error.

When you are using an Oracle Database or a DB2 database, the script prompts you for input for each parameter.

When you are using a SQL Server database, you must edit the analytics_purge_facts.sql script to specify the criteria for purging data.

The following shows an example of the script for SQL Server that deletes all Analytics fact database records from August 1, 2010 through November 30, 2010:

```
CALL ANALYTICS_PURGE
        (
        8, --from month
        2010, --from year
        11, --to month
        2010, --to_year
        1000, --commit batch size
        60 --max run time minutes
        );
```

To use the script:

1. If you are using a SQL Server database, edit the script to specify the criteria.
2. Execute the script. For example, to execute the script on an Oracle Database:

```
sqlplus analytics_user/analytics_user_pwd @analytics_purge_facts.sql
Enter value for month_from: 8
old   4: ANALYTICS_PURGE.PURGE_ANALYTICS_INSTANCES ( &month_from,
-- MM format
new   4: ANALYTICS_PURGE.PURGE_ANALYTICS_INSTANCES ( 8,     -- MM format
Enter value for year_from: 2010
old   5:                           &year_from,               -- YYYY format
new   5:                           2010,                     -- YYYY format
Enter value for month_to: 11
old   6:                           &month_to,                -- MM format
new   6:                           11,                       -- MM format
Enter value for year_to: 2010
old   7:                           &year_to,                 -- YYYY format
new   7:                           2010,                     -- YYYY format
Enter value for record_commit_batch_size: 1000
old   8:                           &record_commit_batch_size,
new   8:                           1000,
Enter value for max_minutes_run: 60
old  10:                           &max_minutes_run) ;
new  10:                             60) ;
Log (09-12-2010 08:27:49) Purge Process Started
 .
 .
 .
Log (09-12-2010 08:27:49)
Log (09-12-2010 08:27:49) Purge Process Finished

PL/SQL procedure successfully completed.
```

14.5.2.3 Partitioning Oracle WebCenter Portal's Analytics Data

When you use the Oracle Fusion Middleware Metadata Repository Creation Utility (RCU) to create schemas, you can specify that Activity Graph and Analytics tables are partitioned (see the Custom Variables screen in RCU). If you chose to partition the tables, Oracle WebCenter Portal uses the native partitioning of the database to automatically create partitions.

Oracle WebCenter Portal provides a partition manager process, which runs once every 24 hours as a separate thread. It creates partitions on each Analytics fact table (ASFACT_*) in the database. Initially, the process generates six partitions in advance, with each partition corresponding to a month in the future. Whenever a new month starts, the partition manager creates a new partition.

Partitioning the data makes it easier to purge data, because you can purge the data by dropping the older partitions that the partition manager creates. Thus, in a partitioned environment, the recommended method for purging data is simply to drop the month-based partitions that are no longer required.

Note: The WC_Utilities Managed Server must be started for the partition manager process to run.

For example, to drop older partitions for a table, use the following SQL command:

```
alter table table_name drop partition partition_name;
```

15 Changing Network Configurations

This chapter provides procedures for changing the network configuration, such as the host name, domain name, or IP address, of an Oracle Fusion Middleware host and the Oracle database that Oracle Fusion Middleware uses. It also includes information about using the IPv6 protocol with Oracle Fusion Middleware.

This chapter includes the following topics:

15.1 Changing the Network Configuration of Oracle Fusion Middleware

This section describes how to change the host name, domain name, IP address, or any combination of these, of a host that contains the following installation types:

- Oracle WebLogic Server and Java components. When you change the host name, domain name, or IP address of Oracle WebLogic Server, you also automatically change the information for Java components, such as Oracle SOA Suite and Oracle WebCenter Portal components that are deployed to Oracle WebLogic Server.
- Oracle Fusion Middleware Web Tier components, Oracle Web Cache and Oracle HTTP Server. You can change the host name or the IP address.

The following topics describe how to change the host name, domain name, or IP address:

15.1.1 Changing the Network Configuration of a Managed Server

You can change the network configuration of a Managed Server using the Oracle WebLogic Server Administration Console.

To change the host name, domain name, or IP address of a Managed Server:

1. Display the Administration Console, as described in Section 3.4.1.
2. In the Change Center, click **Lock & Edit.**

3. Create a machine, which is a logical representation of the computer that hosts one or more WebLogic Servers, and point it to the new host. (From the Home page, select **Machines**. Then, click **New.**) Follow the directions in the Administration Console help.

   You must disable Host Name Verification on Administration Servers that access Node Manager, as described in "Using Hostname Verification" in *Oracle Fusion Middleware Securing Oracle WebLogic Server*.

4. Change the Managed Server configuration to point to the new machine:

   a. From the left pane of the Console, expand **Environment** and then **Servers**. Then, select the name of the server.

   b. Select the **Configuration** tab, then the **General** tab. In the **Machine** field, select the machine to which you want to assign the server.

   c. Change **Listen Address** to the new host.

      Click **Save.**

5. Start the Managed Server. You can use the Oracle WebLogic Server Administration Console, WLST, or the following command:

   ```
   DOMAIN_NAME/bin/startManagedWeblogic.sh managed_server_name
            admin_url
   ```

   The Managed Server connects to the Administration Server and updates its configuration changes.

15.1.2 Changing the Network Configuration of Web Tier Components

If you change the host name, domain name, or IP address of a host that contains multiple Oracle instances, you must change the network configuration of each Oracle instance that resides on that host. You do not need to make changes to any system component that resides on another host.

You can change the network configuration of Oracle HTTP Server and Oracle Web Cache by using the following command:

```
(UNIX) ORACLE_HOME/chgip/scripts/chpiphost.sh
(Windows) ORACLE_HOME\chgip\scripts\chpiphost.bat
```

The format of the command is:

```
chgiphost.sh | chgiphost.bat
             [-noconfig] [-version] [-help]
             [ -oldhost old_host_name -newhost new_host_name]
             [-oldip old_IP_address -newip new_IP_address]
              -instanceHome Instance_path
```

The parameters have the following meanings:

- `noconfig`: The default for changing the network parameters.
- `version`: Displays the version of the chgiphost tool.
- `help`: Displays help for the command.
- `oldhost`: The fully qualified name of the old host. Use this parameter, with `newhost`, to change the host name or domain name, or both.
- `newhost`: The fully qualified name of the new host. Use this parameter, with `oldhost`, to change the host name or domain name, or both.

- `oldip`: The old IP address.
- `newip`: The new IP address.
- `instanceHome`: The full path of the Oracle instance.

For example, to change the host name, domain name, and IP address of a host that contains either Oracle HTTP Server or Oracle Web Cache, or both, perform the following tasks:

- Task 1, "Prepare Your Host"
- Task 2, "Change the Host Name, Domain Name, or IP Address"
- Task 3, "Run the chgiphost Command"
- Task 4, "Restart Processes"

Task 1 Prepare Your Host

Prepare your host for the change:

1. Perform a backup of your environment before you start this procedure. See Chapter 17.
2. Shutdown all Oracle Fusion Middleware processes. See Chapter 4.

Task 2 Change the Host Name, Domain Name, or IP Address

Update your operating system with the new host name, domain name, IP address, or any combination of these. Consult your operating system documentation for information on how to perform the following steps.

1. Make the updates to your operating system to properly change the host name, domain name, or IP address.
2. Restart the host, if necessary for your operating system.
3. Verify that you can ping the host from another host in your network. Be sure to ping using the new host name to ensure that everything is resolving properly.

Task 3 Run the chgiphost Command

Follow these steps for each Oracle instance that contains Oracle HTTP Server or Oracle Web Cache on your host. Be sure to complete the steps entirely for one Oracle instance before you move on to the next.

1. Log in to the host as the user that installed Oracle Fusion Middleware.
2. Run the chgiphost command.

   The following example changes the host name from host_a to host_b and the domain name from dom_1 to dom_2 for an Oracle instance named inst_a.

```
chgiphost.sh  -noconfig
              -oldhost host_a.dom_1 -newhost host_b.dom_2
              -instanceHome /scratch/Oracle/Middleware/inst_a
```

Task 4 Restart Processes

Restart all Oracle Fusion Middleware processes. See Chapter 4.

15.2 Changing the Network Configuration of a Database

This section describes how to change the host name, domain name, or IP address of a host that contains a database that contains the metadata for Oracle Fusion Middleware components:

The following tasks describe the procedure:

- Task 1, "Stop All Oracle Fusion Middleware Components"
- Task 2, "Shut Down the Database"
- Task 3, "Change the Network Configuration"
- Task 4, "Change References to the Network Configuration"
- Task 5, "Start the Database"
- Task 6, "Change the System Data Source"
- Task 7, "Restart Your Environment"

Task 1 Stop All Oracle Fusion Middleware Components

Stop all components that use the database, even if they are on other hosts. Stop the Administration Server, the Managed Servers, and all components, as described in Chapter 4.

Task 2 Shut Down the Database

Prepare your host for the change by stopping the database:

1. Set the ORACLE_HOME and ORACLE_SID environment variables.
2. Shut down the listener and database:

```
lsnrctl stop

sqlplus /nolog
SQL> connect SYS as SYSDBA
SQL> shutdown
SQL> quit
```

3. Verify that all Oracle Fusion Middleware processes have stopped.
4. To ensure that Oracle Fusion Middleware processes do not start automatically after a restart of the host, disable any automated startup scripts you may have set up, such as `/etc/init.d` scripts.

Task 3 Change the Network Configuration

If you are changing the host name, domain name, or IP address, update your operating system with the new names or IP address, restart the host, and verify that the host is functioning properly on your network. Consult your operating system documentation for information on how to perform the following steps:

1. Make the updates to your operating system to properly change the host name, domain name or IP address.
2. Restart the host, if required by your operating system.
3. Verify that you can ping the host from another host in your network. Be sure to ping using the new host name, domain name, or IP address to ensure that everything is resolving properly.

Task 4 Change References to the Network Configuration

You must modify files that contain the host name, domain name, or IP address, depending on the components that you are using. The following lists some of the files that you may need to modify to change references to the new host name, domain name or IP address:

- tnsnames.ora, which is located in:

  ORACLE_HOME/network/admin/tnsnames.ora

- listener.ora, which is located in:

  (UNIX) *ORACLE_HOME*/network/admin/listener.ora
  (Windows) *ORACLE_HOME*\network\admin\listener.ora

- For Oracle HTTP Server, edit the httpd.conf file, making the following changes:
  - Update the Listen directive with the new host name or IP address and port (if the production environment Oracle HTTP Server is using a different port).
  - Update the VirtualHost directive, if the host name, IP address, or port number is defined, with the new values for the production environment.
  - Update any other nondefault directives that were configured at the test environment and have topological (host name, IP address, port number) or other machine-specific information.
- For Oracle HTTP Server, the PlsqlDatabaseConnectString in the dads.conf file
- For Oracle HTTP Server, if you use mod_oradav, the ORACONNECTSN parameter in the mod_oradav.conf file
- For Oracle HTTP Server, if you use mod_plsql, the PlsqlDatabaseConnectString attribute in the dads.conf file
- For Oracle HTTP Server, if you use mod_wl_ohs, update the mod_wl_ohs.conf file

  Update the WebLogicHost, WebLogicPort, or WebLogicCluster directives with the host name, IP address, and port number.
- For Oracle Portal, portal_dads.conf and sqlnet.ora
- For Oracle Forms Services, sqlnet.ora
- For Oracle Business Intelligence Discoverer, module_disco.conf

This is not an exhaustive list. See Chapter 21 for additional information about files used by components. That chapter describes how to move components, including a database, from a test to a production system, in effect changing the host name.

Task 5 Start the Database

Start the database:

1. Log in to the host as the user that installed the database.
2. Set the ORACLE_HOME and ORACLE_SID environment variables.
3. On UNIX systems, set the LD_LIBRARY_PATH, LD_LIBRARY_PATH_64, LIB_PATH, or SHLIB_PATH environment variables to the proper values, as shown in Table 3–1. The actual environment variables and values that you must set depend on the type of your UNIX operating system.
4. Start the database and listener:

   ```
   sqlplus /nolog
   ```

```
SQL> connect SYS as SYSDBA
SQL> startup
SQL> quit

lsnrctl start
```

Task 6 Change the System Data Source

Change the system data source to use the new host name, domain name, or IP address for the database. To do so, you use Oracle WebLogic Server Administration Console:

1. Start the Administration Server, as described in Section 4.2.2.
2. Go to the Administration Console.
3. In the Change Center, click **Lock & Edit.**
4. In the Domain Structure section, expand **Services,** then **Data Sources.**

   The Summary of JDBC Data Sources page is displayed.
5. Select the data source you want to change.

   The Settings page is displayed.
6. Select the Connection Pool tab.
7. Change the following entries to reflect the information for the database on the production environment:
   - **URL:** The database host name and port details. For example:

     `jdbc:oracle:thin:@`*`newhostname.domainname:port/sid`*
     `jdbc:sqlserver://`*`newhostname.domainname:port`*`;database=database`
   - **Driver class:** This is specific to the type of database. For example:

     `oracle.jdbc.OracleDrivercom.microsoft.sqlserver.jdbc.SQLServerDriver`
   - **Properties:** Database user name
   - **Password:** Database password
8. Click **Save.**
9. Restart the servers that use this data source. (Click the Target tab to see the servers that use this data source.)

Task 7 Restart Your Environment

Start the components that use the database:

1. Start all components that use the database, even if they are on other hosts. Start the Administration Server, the Managed Servers, and all components, as described in Chapter 4.
2. If you disabled any processes from automatically starting Oracle Fusion Middleware at the beginning of this procedure, enable them.

15.3 Moving Between On-Network and Off-Network

This section describes how to move an Oracle Fusion Middleware host on and off the network. The following assumptions and restrictions apply:

- The host must contain an instance that does not use an Infrastructure, or both the middle-tier instance and Infrastructure must be on the same host.

- DHCP must be used in loopback mode. Refer to the *Oracle Fusion Middleware System Requirements and Specifications* document for more information.
- Only IP address change is supported; the host name must remain unchanged.
- Hosts in DHCP mode should not use the default host name (`localhost.localdomain`). The hosts should be configured to use a standard host name and the loopback IP should resolve to that host name.
- A loopback adapter is required for all off-network installations (DHCP or static IP). Refer to the *Oracle Fusion Middleware Installation Planning Guide* for more information.

15.3.1 Moving from Off-Network to On-Network (Static IP Address)

This procedure assumes you have installed Oracle Fusion Middleware on a host that is off the network, using a standard host name (not `localhost`), and would like to move on to the network and use a static IP address. The IP address may be the default loopback IP, or any standard IP address.

To move on to the network, you can simply connect the host to the network. No updates to Oracle Fusion Middleware are required.

15.3.2 Moving from Off-Network to On-Network (DHCP)

This procedure assumes you have installed on a host that is off the network, using a standard host name (not `localhost`), and would like to move on to the network and use DHCP. The IP address of the host can be any static IP address or loopback IP address, and should be configured to the host name.

To move on to the network:

1. Connect the host to the network using DHCP.
2. Configure the host name to the loopback IP address only.

15.3.3 Moving from On-Network to Off-Network (Static IP Address)

Follow this procedure if your host is on the network, using a static IP address, and you would like to move it off the network:

1. Configure the `/etc/hosts` file so the IP address and host name can be resolved locally.
2. Take the host off the network.

   There is no need to perform any steps to change the host name or IP address.

15.4 Changing Between a Static IP Address and DHCP

This section describes how to change between a static IP address and DHCP. The following assumptions and restrictions apply:

- The host must contain all Oracle Fusion Middleware components, including Identity Management components, and any database associated with those components. That is, the entire Oracle Fusion Middleware environment must be on the host.
- DHCP must be used in loopback mode. Refer to *Oracle Fusion Middleware Installation Planning Guide* for more information.

- Only IP address change is supported; the host name must remain unchanged.
- Hosts in DHCP mode should not use the default host name (`localhost.localdomain`). The hosts should be configured to use a standard host name and the loopback IP should resolve to that host name.

15.4.1 Changing from a Static IP Address to DHCP

To change a host from a static IP address to DHCP:

1. Configure the host to have a host name associated with the loopback IP address before you convert the host to DHCP.
2. Convert the host to DHCP. There is no need to update Oracle Fusion Middleware.

15.4.2 Changing from DHCP to a Static IP Address

To change a host from DHCP to a static IP address:

1. Configure the host to use a static IP address.
2. There is no need to update Oracle Fusion Middleware.

15.5 Using IPv6

Oracle Fusion Middleware supports Internet Protocol Version 4 (IPv4) and Internet Protocol Version 6 (IPv6.) Among other features, IPv6 supports a larger address space (128 bits) than IPv4 (32 bits), providing an exponential increase in the number of computers that can be addressable on the Web.

An IPv6 address is expressed as 8 groups of 4 hexadecimal digits. For example:

```
2001:0db8:85a3:08d3:1319:8a2e:0370:7334
```

Table 15–1 describes support for IPv6 by Oracle Fusion Middleware components. In the table:

- The column **IPv6 Only** shows whether a component supports using IPv6 only for all communication.
- The column **Dual Stack** shows whether a component supports using both IPv6 and IPv4 for communication. For example, some components do not support using IPv6 only, because some of the communication is with the Oracle Database, which supports IPv4, not IPv6. Those components support dual stack, allowing for IPv6 communication with other components.

Table 15–1 Support for IPv6

Component	IPv6 Only	Dual Stack	Notes
Oracle Access Management Access Manager 11*g*	No	Yes	To configure for IPv6, see Section 15.5.5.
Oracle Access Manager 10*g*	No	Yes	To configure for IPv6, see Section 15.5.6.
Oracle Access Management Identity Federation	No	Yes	Requires a dual stack because Oracle Database requires IPv4 addresses.
Oracle Application Development Framework	Yes	Yes	

Table 15–1 (Cont.) Support for IPv6

Component	IPv6 Only	Dual Stack	Notes
Oracle Business Intelligence Discoverer	No	No	Uses reverse proxy to communicate with Oracle Web Cache or Oracle HTTP Server, which can be configured for IPv6.
Oracle Data Integrator	No	Yes	Requires a dual stack because Oracle Database requires IPv4 addresses. The Agent requires IPv4 addresses. The Oracle Data Integrator server can be on a dual-stack host. The browser client can be on either IPv4 or IPv6 hosts.
Oracle Directory Integration Platform	Yes	Yes	Uses JNDI to communicate with LDAP servers and uses data sources to communicate with the database. JNDI and data sources (JDBC) support IPV6. No additional configuration is necessary.
Oracle Directory Services Manager	Yes	Yes	Uses JNDI to communicate with LDAP servers and uses data sources to communicate with the database. JNDI and data sources (JDBC) support IPV6. No additional configuration is necessary.
Oracle Forms Services	No	No	Uses reverse proxy to communicate with Oracle Web Cache or Oracle HTTP Server, which can be configured for IPv6.
Oracle HTTP Server	Yes	Yes	To configure for IPv6, see Section 15.5.2.
Oracle Identity Manager	No	Yes	Requires a dual stack because Oracle Database requires IPv4 addresses. The Design Console and Remote Manager also require IPv4 addresses. The Oracle Identity Manager server can be on a dual-stack host. The browser client can be on either IPv4 or IPv6 hosts.
Oracle Information Rights Management	No	Yes	Requires a dual stack but the client (the browser) can be on a host configured for IPv6
Oracle Internet Directory	No	Yes	Requires a dual stack because Oracle Database requires IPv4 addresses. See "Managing IP Addresses" in the *Oracle Fusion Middleware Administrator's Guide for Oracle Internet Directory*.
Oracle Platform Security Services	No	Yes	Requires a dual stack because Oracle Database requires IPv4 addresses.
Oracle Portal	No	No	Uses Oracle HTTP Server reverse proxy to communicate with Oracle Web Cache or Oracle HTTP Server, which can be configured for IPv6. See "Configuring Reverse Proxy Servers" in the *Oracle Fusion Middleware Administrator's Guide for Oracle Portal* for more information.
Oracle Reports	No	No	Uses reverse proxy to communicate with Oracle Web Cache or Oracle HTTP Server, which can be configured for IPv6.

Table 15–1 (Cont.) Support for IPv6

Component	IPv6 Only	Dual Stack	Notes
Oracle Single Sign-On Server	No	No	Uses Oracle HTTP Server proxy, which can be configured for IPv6. Oracle Single Sign-On must be Release 10.1.4.3. See Section 15.5.4.
Oracle SOA Suite	No	Yes	Requires a dual stack because Oracle Database requires IPv4 addresses.
Oracle Virtual Directory	No	Yes	Requires a dual stack because Oracle Database requires IPv4 addresses. See *Oracle Fusion Middleware Administrator's Guide for Oracle Virtual Directory*.
Oracle Web Cache	Yes	Yes	Enabled by default. To disable, see Section 15.5.3.
Oracle WebCenter Content: Imaging	No	Yes	Requires a dual stack, but the client (the browser) can be on a host configured for IPv6
Oracle WebCenter Portal	No	Yes	Requires a dual stack because Oracle Database requires IPv4 addresses.
Oracle WebLogic Server	Yes	Yes	The Oracle WebLogic Server plug-ins support IPV6, beginning with the 11*g* release.

The following topics provide more information about Oracle Fusion Middleware support for IPv6:

- Supported Topologies for IPv6 Network Protocols
- Configuring Oracle HTTP Server for IPv6
- Disabling IPv6 Support for Oracle Web Cache
- Configuring Oracle Single Sign-On to Use Oracle HTTP Server with IPv6
- Configuring Oracle Access Management Access Manager 11g for IPv6
- Configuring Oracle Access Manager 10g Support for IPv6

15.5.1 Supported Topologies for IPv6 Network Protocols

The following topologies for IPv4 and IPv6 are supported (dual-stack means that the host is configured with both IPv4 and IPv6):

- Topology A:
  - Oracle Database on IPv4 protocol host
  - Oracle WebLogic Server on dual-stack host
  - Clients on IPv4 protocol host
  - Clients on IPv6 protocol host
- Topology B:
  - Oracle Database on IPv4 protocol host
  - One or more of the following components on dual-stack hosts: Oracle WebLogic Server, Oracle SOA Suite, Oracle WebCenter Portal, Oracle Business Activity Monitoring, Fusion Middleware Control

  - Oracle HTTP Server with mod_wl_ohs on IPv6 protocol host
- Topology C:
  - Database, such as MySQL, that supports IPv6 on IPv6 protocol host
  - Oracle WebLogic Server on IPv6 protocol host
  - Clients on IPv6 protocol host
- Topology D:
  - Oracle Database on IPv4 protocol host
  - One or more of the following components on dual-stack hosts: Identity Management, Oracle SOA Suite, Oracle WebCenter Portal, Oracle Business Activity Monitoring, Fusion Middleware Control
  - Clients on IPv4 protocol host
  - Clients on IPv6 protocol host
- Topology E:
  - Oracle Database on IPv4 protocol host
  - One or more of the following components on IPv4 protocol host: Oracle Portal, Oracle Forms Services, Oracle Reports, Oracle Business Intelligence Discoverer, and Oracle Single Sign-On Release 10.1.4.3
  - Oracle HTTP Server with mod_proxy on dual-stack host
  - Clients on IPv6 protocol host
- Topology F:
  - Oracle Access Manager Release 10.1.4.3 and applications, such as SOA composite applications, on IPv4 protocol host
  - Oracle HTTP Server with mod_proxy on dual-stack host
  - Clients on IPv6 protocol host
- Topology G:
  - Oracle Database on IPv4 protocol host
  - One or more of the following components on IPv4 protocol host: Oracle SOA Suite, Oracle WebCenter Portal, Oracle Business Activity Monitoring, Fusion Middleware Control on IPv4 protocol host
  - Oracle HTTP Server with mod_wl_ohs on dual-stack host
  - Clients on IPv6 protocol host

15.5.2 Configuring Oracle HTTP Server for IPv6

To configure Oracle HTTP Server to communicate using IPv6, you modify configuration files in the following directory:

ORACLE_INSTANCE/config/OHS/*ohs_name*

For example, to configure Oracle HTTP Server to communicate with Oracle WebLogic Server on hosts that are running IPv6, you configure mod_wl_ohs. You edit the configuration files in the following directory:

ORACLE_INSTANCE/config/OHS/*ohs_name*

In the files, specify either the resolvable host name or the IPv6 address in one of the following parameters:

```
WebLogicHost hostname | [IPaddress]
WebCluster [IPaddress_1]:portnum1, [IPaddress_2]:portnum2, [IPaddress_3]:portnum3,
...
```

You must enclose the IPv6 address in brackets.

Any errors are logged in the Oracle HTTP Server logs. To generate more information, set the mod_weblogic directives Debug All and WLLogFile path. Oracle HTTP Server logs module-specific messages.

Note: In previous versions, Oracle HTTP Server contained restrictions about using dynamic clusters with IPv6 nodes. For example, the Oracle HTTP Server plug-in for Oracle WebLogic Server had limited IPv6 support in that the DSL (dynamic server list) feature of the plug-in was not supported; only the static configuration of server lists was supported (DynamicServerList=OFF). For this release, those restrictions have been lifted.

15.5.3 Disabling IPv6 Support for Oracle Web Cache

By default, IPv6 support is enabled for Oracle Web Cache. You can disable it in the webcache.xml file, which is located in the following directory:

```
(UNIX) ORACLE_INSTANCE/config/WebCache/webcache_name
(Windows) ORACLE_INSTANCE\config\WebCache\webcache_name
```

In the file, change the value of the IPV6 element to "NO". For example:

```
<IPV6 ENABLED="NO"/>
```

If the IPV6 element does not exist in the webcache.xml file, you can add the element to the file. Add it after the MULTIPORT element, as shown in the following example:

```
...
         <LISTEN IPADDR="ANY" PORT="7786" PORTTYPE="ADMINISTRATION"/>
         <LISTEN IPADDR="ANY" PORT="7788" PORTTYPE="INVALIDATION"/>
         <LISTEN IPADDR="ANY" PORT="7787" PORTTYPE="STATISTICS"/>
      </MULTIPORT>
      <IPV6 ENABLED="NO"/>
```

15.5.4 Configuring Oracle Single Sign-On to Use Oracle HTTP Server with IPv6

Oracle Single Sign-On Server supports IPv4. However, you can configure Oracle Single Sign-On Server to work with clients that support IPv6 by setting up a proxy server and a reverse proxy.

The steps in this section assume that you have installed Oracle Single Sign-On Server Release 10.1.4.3 and a proxy server such as Oracle HTTP Server that acts as a front end to the Oracle Single Sign-On Server.

Take the following steps to configure Oracle Single Sign-On to work with clients that support IPv6:

1. Enable the proxy server:
   a. Run the ssocfg script on the single sign-on middle tier. This script changes the host name stored in the single sign-on server to the proxy host name. Use the

following command syntax, entering values for the protocol, host name, and port of the proxy server:

```
(UNIX) $ORACLE_HOME/sso/bin/ssocfg.sh http proxy_server_name proxy_port
(Windows) %ORACLE_HOME%\sso\bin\ssocfg.bat http proxy_server_name proxy_
port
```

b. Update the targets.xml file on the single sign-on middle tier. The file is located in:

```
(UNIX) ORACLE_HOME/sysman/emd
(Windows) ORACLE_HOME\sysman\emd
```

Open the file and find the target type `oracle_sso_server`. Within this target type, locate and edit the three attributes that you passed to ssocfg:

- HTTPMachine: The HTTP server host name
- HTTPPort: The SSL port number of the Oracle HTTP server
- HTTPProtocol: The server protocol

c. Add the lines that follow to the httpd.conf file on the single sign-on middle tier. The file is in the directory ORACLE_HOME/Apache/Apache/conf. These lines change the directive ServerName from the name of the actual server to the name of the proxy:

```
KeepAlive off
ServerName proxy_host_name
Port proxy_port
```

Note that if you are using SSL, the port must be an SSL port such as 4443.

d. (SSL only) If you have configured SSL communication between just the browser and the proxy server, configure mod_certheaders on the middle tier. This module enables the Oracle HTTP Server to treat HTTP proxy requests that it receives as SSL requests. Add the lines that follow to httpd.conf. You can place them at the end of the file; where they appear is unimportant.

Enter this line to load the module:

```
(UNIX) LoadModule certheaders_module libexec/mod_certheaders.so
(Windows) LoadModule certheaders_module modules/ApacheModuleCertHeaders.dll
```

If you are using Oracle Web Cache as a proxy, enter this line:

```
AddCertHeader HTTPS
```

If you are using a proxy other than Oracle Web Cache, enter this line:

```
SimulateHttps on
```

e. Reregister mod_osso on the single sign-on middle tier. This step configures mod_osso to use the proxy host name instead of the actual host name. For example, on Linux:

```
$ORACLE_HOME/sso/bin/ssoreg.sh
   -oracle_home_path ORACLE_HOME
   -site_name example.mydomain.com
   -config_mod_osso TRUE
   -mod_osso_url http://example.mydomain.com
```

f. Update the Distributed Configuration Management schema:

```
ORACLE_HOME/dcm/bin/dcmctl updateconfig
```

g. Restart the single sign-on middle tier:

```
ORACLE_INSTANCE/opmn/bin/opmnctl restartproc process-type=HTTP_Server
ORACLE_INSTANCE/opmn/bin/opmnctl restartproc process-type=OC4J_SECURITY
```

h. Log in to the single sign-on server, using the single sign-on login URL:

```
http://proxy_host_name:proxy_port/sso/
```

This URL takes you to the single sign-on home page. If you are able to log in, you have configured the proxy correctly.

2. If you have not already done so, install Oracle HTTP Server 11*g* Release 2 (11.1.2) to use as a reverse proxy for IPv6.
3. Change the Oracle HTTP Server 11*g* Release 2 (11.1.2) configuration to enable reverse proxy:

a. Stop Oracle HTTP Server:

```
opmnctl stopproc ias-component=component_name
```

b. Edit the following file:

```
(UNIX) ORACLE_INSTANCE/config/OHS/ohs_name/httpd.conf
(Windows) ORACLE_INSTANCE\config\OHS\ohs_name\httpd.conf
```

Append the following to the httpd.conf file:

```
#---Added for Mod Proxy
ProxyRequests Off

<Proxy *>
Order deny,allow
Allow from all
</Proxy>

ProxyPass /sso http://OHS_host:OHS_port/sso
ProxyPass / http://OHS_host:OHS_port/
ProxyPassReverse / http://OHS_host:OHS_port/
ProxyPreserveHost On
```

In the example, *OHS_host* and *OHS_port* are the host name and port of the front-end server for Oracle Single Sign-On, discussed in Step 1.

c. Restart the Oracle HTTP Server. For example, to restart ohs1:

```
opmnctl startproc ias-component=ohs1
```

15.5.5 Configuring Oracle Access Management Access Manager 11*g* for IPv6

Access Manager supports Internet Protocol Version 4 (IPv4). You can configure Access Manager to work with clients that support IPv6 by setting up a reverse proxy server. IPv6 is enabled with Oracle HTTP Server with the mod_wl_ohs plug-in. Several scenarios are provided here. Be sure to choose the right configuration for your environment.

This section provides the following topics:

- Prerequisites
- Introduction to Access Manager and IPv6

- Configuring IPv6: Separate Proxy for Access Manager and Webgates

15.5.5.1 Prerequisites

Regardless of the manner in which you plan to use Mobile and Social with IPv6 clients, an Oracle HTTP Server instance must be installed to act as a reverse proxy to the Web server (required for 10g and 11g Webgates).

15.5.5.2 Introduction to Access Manager and IPv6

The supported topologies for Access Manager with IPv4/IPv6 are:

- Webgate 11*g* plus protected applications on an IPv4 protocol host
- OHS reverse proxy on dual-stack host
- Client on IPv6 protocol host
- OAM Server Proxy

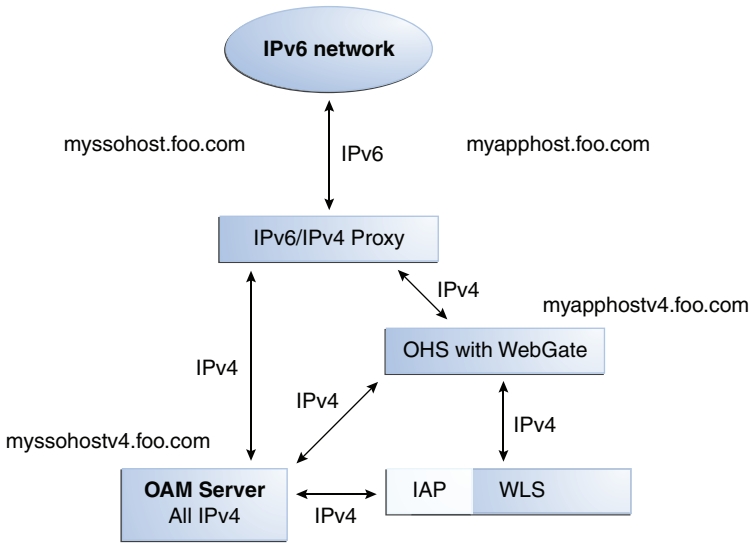
> **Note:** IPv6 for the Identity Store is not yet supported.

15.5.5.2.1 Configuring IPv6 with Access Manager and Challenge Redirect Figure 15–1 illustrates configuration with a single IPv6 to IPv4 Proxy (host configured *myssohost* and *myapphost* can use separate proxies).

With Access Manager, the virtual host name must be specified as a host name, for example, *myapphost.foo.com*, not as an IP address. The redirect host name, for example, *myssohost.foo.com* must also be specified as a host name and not an IP address. The IPv6 address cannot be specified in a Webgate registration.

> **Note:** With Access Manager, there is no concept of an authenticating Webgate or a resource Webgate. Instead, redirection always goes to OAM Server whether you have 11g Webgates or 10g Webgates.

Figure 15–1 IPv6 with Access Manager and Challenge Redirect



As illustrated in Figure 15–2, the IPv6 network communicates with the IPv6/IPv4 proxy, which in turn communicates with the Oracle HTTP Server using IPv4. Webgate, OAM Server, and Oracle WebLogic Server with the Identity Asserter all communicate with each other using IPV4.

You should be able to access the application from a browser on the IPv6 network to the IPv6 server host (*myapphost.foo*.com) and have login with redirect to IPv6 *myssohost.foo*.com.

15.5.5.2.2 Considerations The following considerations apply to each intended use scenario:

- IP validation does not work by default. To enable IP validation, you must add the IP address of the Proxy server as the Webgate's IPValidationException parameter value in the Oracle Access Management Console.
- IP address-based authorization does not work because all requests come through one IP (proxy IP) that would not serve its purpose.
- ipValidationException is required if IPValidation is `On` (parameter `ipValidation=1`). However, you cannot add this parameter using either the Oracle Access Management Console or the remote registration tool. Instead, you must add the proxy's IP as single-valued user-defined parameter for the proxy in the oam-config.xml file.

15.5.5.3 Configuring IPv6: Separate Proxy for Access Manager and Webgates

OAM 10g provided a resource Webgate configuration (that redirects) and an Authenticating Webgate configuration. The Access Manager credential collector replaces and performs the function of an 10g authenticating Webgate.

> **Note:** With Access Manager, the 10g Webgate always redirects to the Access Manager credential collector which acts like the earlier "authenticating" Webgate.

In this configuration you have multiple proxies: for example a separate proxy for the OAM Server and another proxy for the Webgate.

You can access the application from a browser on the IPv4 network directly to an IPv4 server host name with a login redirect to an IPv6 host. For example:

- Webgate is on http://*myapphostv4.foo*.com/
- OAM Server is on http://*myssohostv4.foo*.com
- Proxy used for *myapphostv4.foo*.com should be *myapphost.foo*.com
- Proxy used for *myssohostv4.foo*.com should be *myssohost*.com

> **Note:** You cannot use the IPv6 proxy name as the Preferred HTTP host in a Webgate registration.

With Access Manager, the ProxyRequests parameter must be `On` because Webgates (11g or 10g) always redirect to obrareq.cgi. This directive makes the proxy act as a forward proxy.

The Preferred HTTP host should be set to the *host:port* of the Web server hosting the Webgate (or SERVER_NAME if the Web server hosting the Webgate is configured for virtual hosting).

If IPValidation is ON, IPValidationException must be added for the proxy.

If reverse proxy is configured to perform SSL termination, then the user-defined Webgate `proxySSLHeaderVar` parameter must be defined during remote registration. This parameter is used when the Webgate is located behind a reverse proxy. The value of the `proxySSLHeaderVar` parameter defines the name of the header variable the proxy must set. The value of the header variable must be `ssl` or `nonssl`. If the header variable is not set, the SSL state is decided by the SSL state of the current Web server. Syntax is as follows:

```
<name>proxySSLHeaderVar</name>
<value>IS_SSL</value>
```

Modify the Load Balancing Router (reverse proxy Web server) settings to insert an HTTP header string that sets the IS_SSL value to `ssl`. For example, in the F5 load balancer, in Advanced Proxy Settings, you add the HTTP header string `IS_SSL:ssl`.

In the following procedure, *OHS_host* and *OHS_port* are the host name and port of the actual Oracle HTTP Server that is configured for Webgate. Be sure to use values for your own environment. Your values will be different.

To configure IPv6 with a separate proxy for Access Manager and Webgates

1. Install and configure Oracle HTTP Server for reverse proxy. Ensure that you have a separate Oracle HTTP Server instance for each proxy
2. Enable mod_proxy to the OAM Server and Webgate: Configure Oracle HTTP Server 11g Release 1 (11.1.1) or any other server for multiple proxies, as follows:
   a. Stop Oracle HTTP Server for the corresponding proxy instance with the following command:

      ```
      opmnctl stopproc ias-component=OHS instance name
      ```

   b. Edit the following file of the Oracle HTTP Server instance for the corresponding proxy:

      ```
      (UNIX) ORACLE_INSTANCE/config/OHS/ohs_name1/httpd.conf
      (Windows) ORACLE_INSTANCE\config\OHS\ohs_name1\httpd.conf
      ```

   c. To configure the proxy to OAM Server, append the following information for your environment to the httpd.conf file to enable mod_proxy. For example:

      ```
      <IfModule mod_proxy.c>
      ProxyRequests On
      ProxyPreserveHost On

      ProxyPass / http://oam_server_host:port/
      ProxyPassReverse / http://oam_server_host:port/
      </IfModule>
      ```

   d. To configure reverse proxy to 11*g* Webgate, append information for your environment to the httpd.conf file to enable mod_proxy, as follows:

      ```
      <IfModule mod_proxy.c>
      ProxyRequests On
      ProxyPreserveHost On

      ProxyPass / http://webgate_OHS_host:port/
      ```

```
ProxyPassReverse / http://webgate_OHS_host:port/
</IfModule>
```

e. Restart Oracle HTTP Server with the following command:

```
opmnctl startproc ias-component=OHS instance name
```

3. In the Authentication Scheme, change the Challenge Redirect URL to http://*oam_server_proxy_host*:*port*/oam/server.
4. Set the Preferred HTTP host for each Webgate to the *host:port* of the Web server hosting the Webgate (or SERVER_NAME if the Web server hosting Webgate is configured for virtual hosting):
   a. Log in to Oracle Access Management Console. For example:

   ```
   http://hostname:port/oamconsole
   ```

   b. Click **System Configuration**, **Access Manager Settings, SSO Agents, OAM Agents**.
   c. Find the agent and click its name in the Search Results table to display the registration page.
   d. For **Preferred HTTP Host**, enter the name of the Oracle HTTP Server Web server that is configured for this Webgate. For instance, a Webgate deployed on *myapphostv4.foo*.com must use *myapphostv4.foo*.com as the Preferred HTTP host.
   e. Click Apply.
5. Repeat for each Webgate and specify name of the Oracle HTTP Server Web server that is configured for this Webgate.
6. For **IPValidationException**, if IPValidation is `On` (parameter "ipValidation"=1), add the proxy's IP as single-valued user-defined parameter for the proxy in the oam-config.xml file.
   a. Stop all OAM Servers and the AdminServer.
   b. Locate the oam-config.xml in the following path:

   ```
   WLS_DOMAIN_HOME/config/fmwconfig/oamconfig.xm
   ```

   c. Enter the following information:

   ```
   <Setting Name="ipValidationExceptions"Type="xsd:string"> 10.1.1.1</Setting>
   ```

   d. Save the file.
   e. Restart the OAM Servers and AdminServer.
   f. If reverse proxy is configured to perform SSL termination, the Webgate user-defined "`proxySSLHeaderVar`" parameter must be set (default is `IS_SSL`). Modify the Load Balancing Router (reverse proxy Web server) settings to insert an HTTP header string that sets the `IS_SSL` value to `ssl`. For example, in the F5 load balancer, in Advanced Proxy Settings, you add the HTTP header string `IS_SSL:ssl`.

15.5.6 Configuring Oracle Access Manager 10*g* Support for IPv6

Oracle Access Manager 10*g* supports Internet Protocol Version 4 (IPv4). You can configure Oracle Access Manager to work with clients that support IPv6 by setting up

a reverse proxy server. Several scenarios are provided here. Be sure to choose the right configuration for your environment.

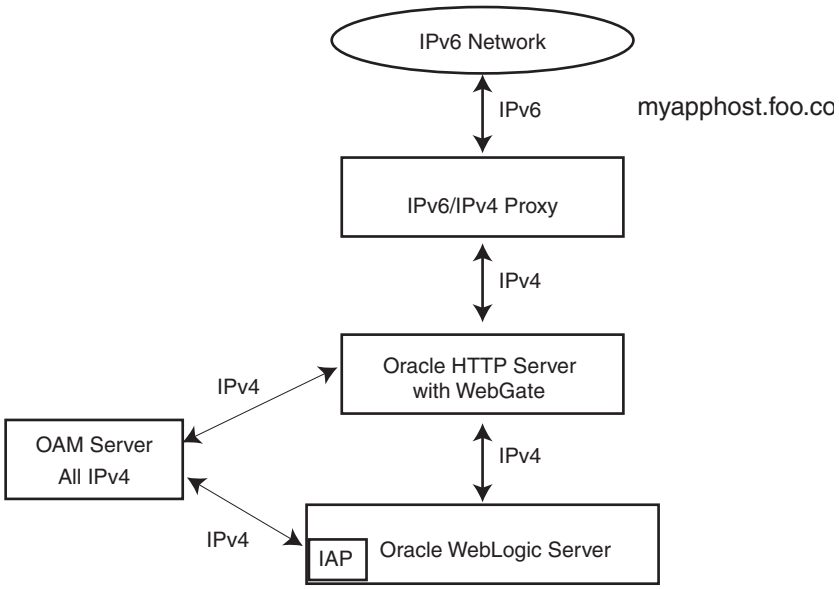
You can configure Oracle Access Manager to work with clients that support IPv6 by setting up a reverse proxy server. Several scenarios are provided here. Be sure to choose the right configuration for your environment.

15.5.6.1 Simple Authentication with IPv6

Figure 15–2 illustrates simple authentication with Oracle Access Manager configured to use the IPv6/IPv4 proxy.

> **Note:** In a WebGate profile, an IPv6 address cannot be specified. In a WebGate profile, the virtual host name must be specified as a host name, for example, *myapphost.foo.com*, not as an IP address.

Figure 15–2 Simple Authentication with the IPv6/IPv4 Proxy



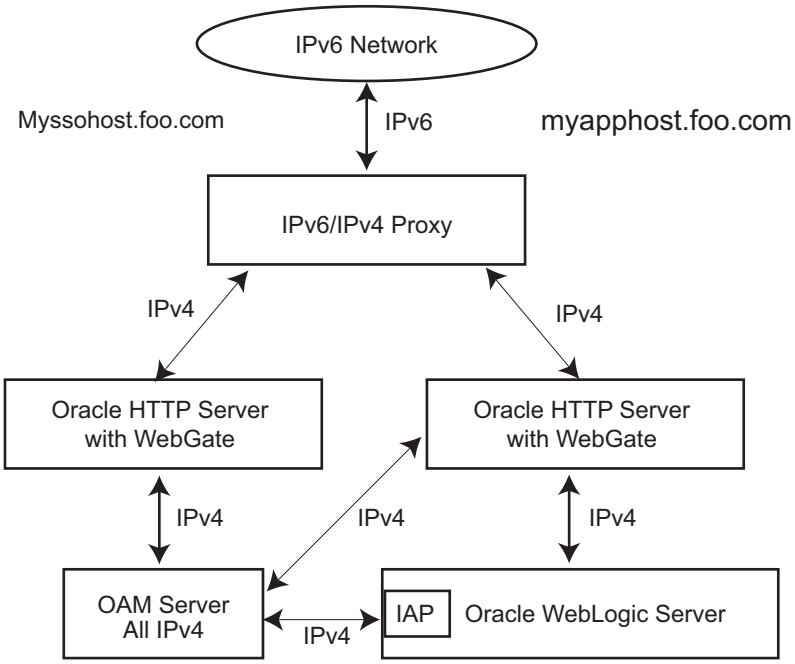
As illustrated in Figure 15–2, the IPv6 network communicates with the IPv6/IPv4 proxy, which in turn communicates with the Oracle HTTP Server and WebGate using IPv4. WebGate, Oracle Access Manager servers, and Oracle WebLogic Server with the Authentication provider all communicate with each other using IPV4.

15.5.6.2 Configuring IPv6 with an Authenticating WebGate and Challenge Redirect

Figure 15–3 illustrates configuration with a single IPv6 to IPv4 proxy (even though *myssohost* and *myapphost* could use separate proxies).

> **Note:** In a WebGate profile, the virtual host name must be specified as a host name, for example, *myapphost.foo.com*, not as an IP address. The redirect host name, for example, *myssohost.foo.com* must also be specified as a host name and not an IP address. The IPv6 address cannot be specified in a WebGate profile.

Figure 15–3 IPv6 with an Authenticating WebGate and Challenge Redirect



As illustrated in Figure 15–3, the IPv6 network communicates with the IPv6/IPv4 proxy, which in turn communicates with the Oracle HTTP Server using IPv4. WebGate, Oracle Access Manager server, and Oracle WebLogic Server with the Identity Asserter all communicate with each other using IPV4.

You should be able to access the application from a browser on the IPv4 network directly to the IPv4 server host name and have login with redirect to IPv6 *myssohost.foo*.com.

15.5.6.3 Considerations

The following considerations apply to each intended usage scenario:

- IP validation does not work by default. To enable IP validation, you must add the IP address of the Proxy server as the WebGate's IPValidationException parameter value in the Access System Console.
- IP address-based authorization does not work because all requests come through one IP (proxy IP) that would not serve its purpose.

15.5.6.4 Prerequisites

Regardless of the manner in which you plan to use Oracle Access Manager with IPv6 clients, the following tasks should be completed before you start:

- Install an Oracle HTTP Server instance to act as a reverse proxy to the Web server (required for WebGate).
- Install and complete the initial set up of Oracle Access Manager (Identity Server, WebPass, Policy Manager, Access Server, WebGate) as described in *Oracle Access Manager Access Administration Guide*.

See Also:

- *Oracle Fusion Middleware Installation Guide for Oracle Web Tier*
- *Oracle Fusion Middleware Administrator's Guide for Oracle HTTP Server*

15.5.6.5 Configuring IPv6 with Simple Authentication

Use the procedure in this section to configure your environment for simple authentication with Oracle Access Manager using the IPv6/IPv4 proxy. See Figure 15–2 for a depiction of this scenario.

The configuration in this procedure is an example only. In the example, *OHS_host* and *OHS_port* are the host name and port of the actual Oracle HTTP Server with WebGate. You must use values for your environment.

Note: For this configuration you must use the Web server on which the WebGate is deployed as the Preferred HTTP host in the WebGate profile. You cannot use the IPv6 proxy name.

To configure IPv6 with simple authentication:

1. Configure Oracle HTTP Server 11*g* Release 1 (11.1.1) or any other server to enable reverse proxy:
   a. Stop Oracle HTTP Server with the following command:

      ```
      opmnctl stopproc ias-component=component_name
      ```

   b. Edit the following file:

      ```
      (UNIX) ORACLE_INSTANCE/config/OHS/ohs_name/httpd.conf
      (Windows) ORACLE_INSTANCE\config\OHS\ohs_name\httpd.conf
      ```

   c. Append the following to the httpd.conf file:

      ```
      #---Added for Mod Proxy
      <IfModule mod_proxy.c>

      ProxyRequests Off
      ProxyPreserveHost On

      ProxyPass /http://OHS_host:OHS_port/
      ProxyPassReverse /http://OHS_host:OHS_port/

      </IfModule>
      ```

   d. Restart Oracle HTTP Server using the following command:

      ```
      opmnctl startproc ias-component=component_name
      ```

2. Log in to the Access System Console. For example:

   ```
   http://hostname:port/access/oblix
   ```

   In the example, *hostname* refers to the computer that hosts the WebPass Web server; *port* refers to the HTTP port number of the WebPass Web server instance; /access/oblix connects to the Access System Console.

   The Access System main page appears.

3. Click **Access System Configuration**, and then click **AccessGate Configuration**.

   The Search for AccessGates page appears. The Search list contains a selection of attributes that can be searched. Remaining fields allow you to specify search criteria that are appropriate for the selected attribute.

4. Select the search attribute and condition from the lists (or click **All** to find all AccessGates), and then click **Go.**

5. Click an AccessGate's name to view its details.

6. Click **Modify.**

7. For **Preferred HTTP Host**, specify the Web server name on which WebGate is deployed as it appears in all HTTP requests. The host name within the HTTP request is translated into the value entered into this field regardless of the way it was defined in a user's HTTP request.

8. To enable IP validation, add the IP address of the proxy server as the value of the **IPValidationException** parameter.

9. Click **Save.**

15.5.6.6 Configuring IPv6 with an Authenticating WebGate and Challenge Redirect

Use the procedure in this section to configure your environment to use Oracle Access Manager with the IPv6/IPv4 proxy and an authenticating WebGate and challenge redirect. Figure 15–3 shows a depiction of this scenario.

The following procedure presumes a common proxy for both form-based authentication and the resource WebGate. For example, suppose you have the following configuration:

- Resource WebGate is installed on http://*myapphostv4.foo*.com/
- Resource is on http://*myapphostv4.foo*.com/testing.html
- Authenticating WebGate is on http://*myssohostv4.foo*.com/
- Login form is http://*myssohostv4.foo*.com/oamsso/login.html
- Reverse proxy URL is http://*myapphost.foo*.com/

> **Note:** For this configuration, the Preferred HTTP host must be the name of the Oracle HTTP Server Web server that is configured for this WebGate. For example, a WebGate deployed on *myapphost4.foo*.com must use *myapphost4.foo*.com as the Preferred HTTP host. You cannot use the IPv6 proxy name.

In the following procedure, you configure the Oracle HTTP Server, configure WebGate profiles to use the corresponding Oracle HTTP Server as the Preferred HTTP host, and configure the form-based authentication scheme with a challenge redirect value of the reverse proxy server URL (`http://myapphost.foo.com/` in this example).

Be sure to use values for your own environment.

To configure IPv6 with an authenticating WebGate and challenge redirect:

1. Configure Oracle HTTP Server 11*g* Release 1 (11.1.1) or any other server, as follows:

   a. Stop Oracle HTTP Server with the following command:

```
opmnctl stopproc ias-component=component_name
```

b. Edit the following file:

```
(UNIX) ORACLE_INSTANCE/config/OHS/ohs_name/httpd.conf
(Windows) ORACLE_INSTANCE\config\OHS\ohs_name\httpd.conf
```

c. Append the following information for your environment to the httpd.conf file. For example:

```
<IfModule mod_proxy.c>
ProxyRequests On
ProxyPreserveHost On
#Redirect login form requests and redirection requests to Authentication
WebGate

ProxyPass /obrareq.cgi   http://myssohostv4.foo.com/obrareq.cgi
ProxyPassReverse /obrareq.cgi  http://myssohostv4.foo.com/obrareq.cgi

ProxyPass /oamsso/login.html http://myssohostv4.foo.com/oamsso/login.html
ProxyPassReverse /oamsso/login.html http://myssohostv4.foo.com/oamsso/login
.html

ProxyPass /access/sso   http://myssohostv4.foo.com/ /access/sso
ProxyPassReverse /access/sso http://myssohostv4.foo.com/access/sso

# Redirect resource requests to Resource WG
ProxyPass /http://myapphostv4.foo.com /
ProxyPassReverse /http://myapphostv4.foo.com /

</IfModule>
```

d. Restart Oracle HTTP Server using the following command:

```
opmnctl startproc ias-component=component_name
```

2. In the Access System Console, set the Preferred HTTP host for each WebGate as follows:

a. Log in to the Access System Console. For example:

```
http://hostname:port/access/oblix
```

In the example, *hostname* refers to the computer that hosts the WebPass Web server; *port* refers to the HTTP port number of the WebPass Web server instance; /access/oblix connects to the Access System Console.

The Access System main page appears.

b. Click **Access System Configuration**, and then click **AccessGate Configuration**.

The Search for AccessGates page appears. The Search list contains a selection of attributes that can be searched. Remaining fields allow you to specify search criteria that are appropriate for the selected attribute.

c. Select the search attribute and condition from the lists (or click **All** to find all AccessGates), and then click **Go.**

d. Click an AccessGate's name to view its details.

e. Click **Modify.**

f. For **Preferred HTTP Host,** specify the name of the Oracle HTTP Server Web server that is configured for this WebGate. For example, a WebGate deployed on *myapphostv4.foo*.com must use *myapphostv4.foo*.com as the Preferred HTTP host.

g. To enable IP validation, add the IP address of the Proxy server as the value of the **IPValidationException** parameter.

h. Click **Save.**

i. Repeat for each WebGate and specify name of the Oracle HTTP Server Web server that is configured for this WebGate.

3. From the Access System Console, modify the Form authentication scheme to include a challenge redirect to the Proxy server, as follows:

   a. Click **Access System Configuration**, and then click **Authentication Management**.

   b. Click the name of the scheme to modify, and then click **Modify.**

   c. Configure the challenge redirect value to the Proxy server URL. In this example, the Proxy server URL is `http://myapphost.foo.com/`.

   d. Click **Save.**

15.5.6.7 Configuring IPv6: Separate Proxy for Authentication and Resource WebGates

Use the procedure in this section to configure a separate proxy for authentication and resource WebGates. In this configuration, you have multiple proxies: for example a separate proxy for the authentication WebGate and another proxy for the resource WebGate. You can access the application from a browser on the IPv4 network directly to an IPv4 server host name with a login redirect to an IPv6 host. For example:

- Resource WebGate is on http://*myapphostv4.foo*.com/
- Authenticating WebGate is on http://*myssohostv4.foo*.com
- Proxy used for *myapphostv4.foo*.com should be *myapphostv4.foo*.com
- Proxy used for *myssohostv4.foo*.com should be *myssohostv4*.com

---

Note: You cannot use the IPv6 proxy name as the Preferred HTTP host in a WebGate profile.

In the example, *OHS_host* and *OHS_port* are the host name and port of the Oracle HTTP Server that is configured for WebGate. Be sure to use values for your own environment.

To configure IPv6 with a separate proxy for authentication and resource WebGates:

1. Configure Oracle HTTP Server 11*g* Release 1 (11.1.1) or any other server for multiple proxies, as follows:

   a. Stop Oracle HTTP Server with the following command:

   ```
   opmnctl stopproc ias-component=component_name
   ```

   b. Edit the following file:

   ```
   (UNIX) ORACLE_INSTANCE/config/OHS/ohs_name/httpd.conf
   (Windows) ORACLE_INSTANCE\config\OHS\ohs_name\httpd.conf
   ```

c. Append the following information for your environment to the httpd.conf file. For example:

```
<IfModule mod_proxy.c>
ProxyRequests Off
ProxyPreserveHost On

ProxyPass /http://OHS_host:OHS_port
ProxyPassReverse /http://OHS_host:OHS_port

</IfModule>
```

d. Restart Oracle HTTP Server using the following command:

```
opmnctl startproc ias-component=component_name
```

2. In the Access System Console, set the Preferred HTTP host for each WebGate as follows:

a. Log in to the Access System Console. For example:

```
http://hostname:port/access/oblix
```

In the example, *hostname* refers to the computer that hosts the WebPass Web server; *port* refers to the HTTP port number of the WebPass Web server instance; /access/oblix connects to the Access System Console.

The Access System main page appears.

b. Click **Access System Configuration**, and then click **AccessGate Configuration**.

The Search for AccessGates page appears. The Search list contains a selection of attributes that can be searched. Remaining fields allow you to specify search criteria that are appropriate for the selected attribute.

c. Select the search attribute and condition from the lists (or click **All** to find all AccessGates), and then click **Go.**

d. Click an AccessGate's name to view its details.

e. Click **Modify.**

f. For **Preferred HTTP Host,** specify the name of the Oracle HTTP Server Web server that is configured for this WebGate. For instance, a WebGate deployed on *myapphostv4.foo*.com must use *myapphostv4.foo*.com as the Preferred HTTP host.

g. To enable IP validation, add the IP address of the Proxy server as the value of the **IPValidationException** parameter.

h. Click **Save.**

i. Repeat for each WebGate and specify the name of the Oracle HTTP Server Web server that is configured for this WebGate.

3. From the Access System Console, modify the Form authentication scheme to include a challenge redirect to the Proxy server, as follows:

a. Click **Access System Configuration**, and then click **Authentication Management**.

b. Click the name of the scheme to modify, and then click **Modify.**

c. Configure the challenge redirect value to the Proxy server URL that acts as a reverse proxy for the authentication WebGate. In this example, the Proxy server URL is `http://myssohost.foo.com/`.

d. Click **Save.**

Part VII

Advanced Administration: Backup and Recovery

Backup and recovery refers to the various strategies and procedures involved in guarding against hardware failures and data loss, and reconstructing data should loss occur. This part describes how to back up and recover Oracle Fusion Middleware.

It contains the following chapters:

16

Introducing Backup and Recovery

This chapter provides an introduction to backing up and recovering Oracle Fusion Middleware, including backup and recovery recommendations for Oracle Fusion Middleware components.

This chapter includes the following topics:

- Understanding Oracle Fusion Middleware Backup and Recovery
- Oracle Fusion Middleware Directory Structure
- Overview of the Backup Strategies
- Overview of Recovery Strategies
- Backup and Recovery Recommendations for Oracle Fusion Middleware Components
- Assumptions and Restrictions

16.1 Understanding Oracle Fusion Middleware Backup and Recovery

An Oracle Fusion Middleware environment can consist of different components and configurations. A typical Oracle Fusion Middleware environment contains an Oracle WebLogic Server domain with Java components, such as Oracle SOA Suite, and a WebLogic Server domain with Identity Management components. It can also include Oracle instances containing system components such as Oracle HTTP Server, Oracle Web Cache, Oracle Internet Directory, and Oracle Virtual Directory.

The installations of an Oracle Fusion Middleware environment are interdependent in that they contain configuration information, applications, and data that are kept in synchronization. For example, when you perform a configuration change, information in configuration files is updated. When you deploy an application, you might deploy it to all Managed Servers in a domain or cluster.

It is, therefore, important to consider your entire Oracle Fusion Middleware environment when performing backup and recovery. You should back up your entire Oracle Fusion Middleware environment at once, then periodically. If a loss occurs, you can restore your environment to a consistent state.

The following topics describe concepts that are important to understanding backup and recovery:

- Impact of Administration Server Failure
- Managed Server Independence (MSI) Mode
- Configuration Changes in Managed Servers

See Also:

- Section 2.2 for conceptual information about an Oracle WebLogic Server domain
- Section 2.2.1 for conceptual information about the Administration Server
- Section 2.2.2 for conceptual information about Managed Servers and clusters
- Section 2.2.3 for conceptual information about Node Manager

16.1.1 Impact of Administration Server Failure

The failure of an Administration Server does not affect the operation of Managed Servers in the domain but it does prevent you from changing the domain's configuration. If an Administration Server fails because of a hardware or software failure on its host computer, other server instances on the same computer may be similarly affected.

If an Administration Server for a domain becomes unavailable while the server instances it manages—clustered or otherwise—are running, those Managed Servers continue to run. Periodically, these Managed Servers attempt to reconnect to the Administration Server. For clustered Managed Server instances, the load balancing and failover capabilities supported by the domain configuration continue to remain available.

When you first start a Managed Server, it must be able to connect to the Administration Server to retrieve a copy of the configuration. Subsequently, you can start a Managed Server even if the Administration Server is not running. In this case, the Managed Server uses a local copy of the domain's configuration files for its starting configuration and then periodically attempts to connect with the Administration Server. When it does connect, it synchronizes its configuration state with that of the Administration Server.

16.1.2 Managed Server Independence (MSI) Mode

A Managed Server maintains a local copy of the domain configuration. When a Managed Server starts, it contacts its Administration Server to retrieve any changes to the domain configuration that were made since the Managed Server was last shut down. If a Managed Server cannot connect to the Administration Server during startup, it can use its locally cached configuration information—this is the configuration that was current at the time of the Managed Server's most recent shutdown. A Managed Server that starts without contacting its Administration Server to check for configuration updates is running in Managed Server Independence (MSI) mode. By default, MSI mode is enabled. However a Managed Server cannot be started even in MSI mode for the first time if the Administration Server is down due to non-availability of the cached configuration.

16.1.3 Configuration Changes in Managed Servers

Configuration changes are updated in a Managed Server during the following events:

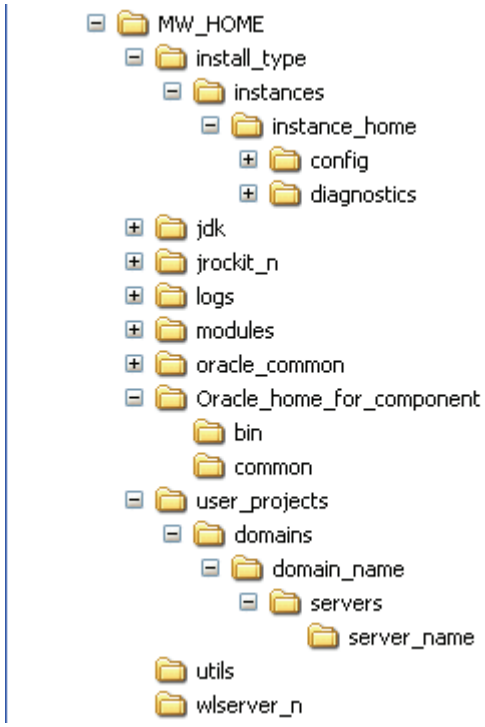
- On each Managed Server restart, the latest configuration is retrieved from the Administration Server. This happens even when Node Manager is down on the node where the Managed Server is running. If the Administration Server is unavailable during the Managed Server restart and if the MSI (Managed Server Independence) mode is enabled in the Managed Server, it starts by reading its

local copy of the configuration and synchronizes with the Administration Server when it is available. By default MSI mode is enabled.

- Upon activating every administrative change such as configuration changes, deployment or redeployment of applications, and topology changes, the Administration Server pushes the latest configuration to the Managed Server. If the Managed Server is not running, the Administration Server pushes the latest version of the configuration to the Managed Server when it does start.

16.2 Oracle Fusion Middleware Directory Structure

The following shows a simplified view of the Oracle Fusion Middleware directory structure:



16.3 Overview of the Backup Strategies

To back up your Oracle Fusion Middleware environment, you can use:

- File copy utilities such as copy, xcopy, tar, or jar. Make sure that the utilities:
  - Preserve symbolic links
  - Support long file names
  - Preserve the permissions and ownership of the files

  For example:

  - On Windows, for online backups, use copy; for offline backups, use copy, xcopy, or jar. Do not use Winzip because it does not work with long filenames or extensions.

    Note that for some versions of Windows, any file name with more than 256 characters fails. You can use the xcopy command with the following switches to work around this issue:

```
xcopy /s/e  "C:\Temp\*.*"  "C:\copy"
```

See the xcopy help for more information about syntax and restrictions.

- On Linux and UNIX, for online and offline backups, use tar.

- Oracle Recovery Manager (RMAN) to back up database-based metadata repositories and any databases used by Oracle Fusion Middleware. With RMAN, you can perform full backups or incremental backups. See *Oracle Database Backup and Recovery User's Guide* for information about using RMAN to back up a database.

If you want to retain your backups for a longer duration, you may want to back up to tape, for example using Oracle Secure Backup.

You can also configure Oracle WebLogic Server to make backup copies of the configuration files. This facilitates recovery in cases where configuration changes need to be reversed or in the unlikely case that configuration files become corrupted. When the Administration Server starts, it saves a .jar file named config-booted.jar that contains the configuration files. When you make changes to the configuration files, the old files are saved in the configArchive directory under the domain directory, in a .jar file with a sequentially numbered name such as config-1.jar. However, the configuration archive is always local to the Administration Server host. It is a best practice to back up the archives to an external location.

16.3.1 Types of Backups

You can back up your Oracle Fusion Middleware environment offline or online:

- An **offline backup** means that you must shut down the environment before backing up the files. When you perform an offline backup, the Administration Server, all Managed Servers in the domain, and all system components in the Oracle instances should be shut down.

  Back up the environment offline immediately after installation and after applying any patches or upgrades.

- An **online backup** means that you do not shut down the environment before backing up the files. To avoid an inconsistent backup, do not make any configuration changes until the backup is completed. To ensure that no changes are made in the WebLogic Server domain, lock the WebLogic Server configuration, as described in Section 3.4.2.

You can perform backups on your full Oracle Fusion Middleware environment, or on the run-time artifacts, which are those files that change frequently.

To perform a full backup, you should back up the static files and directories, as well as run-time artifacts, which are described in Section 16.3.2.

16.3.2 Backup Artifacts

Backup artifacts include static files and directories and run-time artifacts.

Static files and directories are those that do not change frequently. These include:

- The Middleware home (*MW_HOME*). A Middleware home consists of a WebLogic Server home (containing the Oracle WebLogic Server product directories), an Oracle Common home, and optionally an Oracle home. It can also contain the user_projects directories, which contains Oracle WebLogic Server domains and Oracle instance homes, which are not static files.

- OraInventory
- On Linux and UNIX, the oraInst.loc file, which is located in the following directory:

  ```
  (Linux and IBM AIX) /etc
  (Other UNIX systems) /var/opt/oracle
  ```

- On Linux and UNIX, the oratab file, which is located in the following directory:

  ```
  /etc
  ```

- The beahomelist file, which is located at:

  ```
  (UNIX) user_home/bea/beahomelist
  (Windows) C:\bea\beahomelist
  ```

- On Windows, the following registry key:

  ```
  HKEY_LOCAL_MACHINE\Software\oracle
  ```

  In addition, for system components, such as Oracle Web Cache, you must back up the following Windows Registry key:

  ```
  HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services
  ```

Run-time artifacts are those files that change frequently. Back up these files when you perform a full backup and on a regular basis. Run-time artifacts include:

- Domain directories of the Administration Server and the Managed Servers (by default, a domain directory resides in MW_HOME, but it can be configured by the user to point to a different location.)

  In most cases, you do not need to back up Managed Server directories separately because the Administration Server contains information about all of the Managed Servers in its domain.

- All Oracle instance homes, which reside, by default, in the MW_HOME but can be configured to be in a different location.
- Application artifacts, such as .ear or .war files that reside outside of the domain.

  You do not need to back up application artifacts in a Managed Server directory structure because they can be retrieved from the Administration Server during Managed Server startup.

- Database artifacts, such as the MDS Repository.
- Any database-based metadata repositories used by Oracle Fusion Middleware. You use Oracle Recovery Manager (RMAN) to back up an Oracle database.
- Persistent stores, such as JMS Providers and transaction logs, which reside by default in the user_projects directory, but can be configured in a different location.

16.3.3 Recommended Backup Strategy

This section outlines the recommended strategy for performing backups. Using this strategy ensures that you can perform the recovery procedures in this book.

- **Perform a full offline backup:** This involves backing up the entities described in Section 16.3.1. Perform a full offline backup at the following times:
  - Immediately after you install Oracle Fusion Middleware
  - Immediately before upgrading your Oracle Fusion Middleware environment

  - Immediately after an operating system software upgrade
  - Immediately after upgrading or patching Oracle Fusion Middleware
- **Perform an online backup of run-time artifacts:** This involves backing up the run-time artifacts described in Section 16.3.1. Backing up the run-time artifacts enables you to restore your environment to a consistent state as of the time of your most recent configuration and metadata backup. To avoid an inconsistent backup, do not make any configuration changes until backup completes. Perform an online backup of run-time artifacts at the following times:
  - On a regular basis. Oracle recommends that you back up run-time artifacts nightly.
  - Prior to making configuration changes to a component.
  - After making configuration changes to a component.
  - Prior to deploying a custom Java EE application to a Managed Server or cluster.
  - After a major change to the deployment architecture, such as creating servers or clusters.
- **Perform a full or incremental backup of your databases:** Use RMAN to backup your databases. See the *Oracle Database Backup and Recovery User's Guide* for information about using RMAN and for suggested methods of backing up the databases.

16.4 Overview of Recovery Strategies

Recovery strategies enable you to recover from critical failures that involve actual data loss. Depending on the type of loss, they can involve recovering any combination of the following types of files:

- Oracle software files
- Configuration files
- Oracle system files
- Windows Registry keys
- Application artifacts

You can recover your Oracle Fusion Middleware environment while Oracle Fusion Middleware is offline.

To recover your Oracle Fusion Middleware environment, you can use:

- File copy utilities such as copy, xcopy, or tar

  When you restore the files, use your preferred tool to extract the compressed files:

  - On Windows, for online recovery, use copy; for offline recovery, use copy, xcopy, or jar.

    Note that for some versions of Windows, any file name with more than 256 characters fails. You can use the xcopy command with the following switches to work around this issue:

    ```
    xcopy /s/e  "C:\Temp\*.*"  "C:\copy"
    ```

    See the xcopy help for more information about syntax and restrictions.

    Do not use Winzip because it does not work with long filenames or extensions.

  - On Linux and UNIX, use tar.

  Ensure that the tool you are using preserves the permissions and timestamps of the files.

- Oracle Recovery Manager (RMAN) to recover database-based metadata repositories

16.4.1 Types of Recovery

You can recover your Oracle Fusion Middleware environment in part or in full. You can recover the following:

- The Middleware home
- WebLogic Server domains
- The WebLogic Server Administration Server
- WebLogic Server Managed Servers
- Oracle homes
- Oracle instance homes
- A component, such as Oracle SOA Suite or Oracle HTTP Server
- WebLogic Server cluster
- Deployed applications

16.4.2 Recommended Recovery Strategies

Note the following key points about recovery:

- Your Oracle Fusion Middleware environment must be offline while you are performing recovery.
- Rename important existing files and directories before you begin restoring the files from backup so that you do not unintentionally override necessary files.
- Although, in some cases, it may appear that only one or two files are lost or corrupted, you should restore the directory structure for the entire element, such as an Oracle instance home or a domain, rather than just restoring one or two files. In this way, you are more likely to guarantee a successful recovery.
- Recover the database to the most current state, using point-in-time recovery (if the database is configured in Archive Log Mode). This is typically a time right before the database failure occurred.

16.5 Backup and Recovery Recommendations for Oracle Fusion Middleware Components

The following sections describe backup and recovery recommendations for specific Oracle Fusion Middleware components:

- Backup and Recovery Recommendations for Oracle WebLogic Server
- Backup and Recovery Recommendations for Oracle Identity Management
- Backup and Recovery Recommendations for Oracle SOA Suite
- Backup and Recovery Recommendations for Oracle WebCenter Portal
- Backup and Recovery Recommendations for Oracle JRF Installations

- Backup and Recovery Recommendations for Web Tier Installations
- Backup and Recovery Recommendations for Oracle Portal, Oracle Forms Services, Oracle Reports, and Oracle BI Discoverer Installations
- Backup and Recovery Recommendations for Oracle Business Intelligence
- Backup and Recovery Recommendations for Oracle Hyperion Enterprise Performance Management System
- Backup and Recovery Recommendations for Oracle Data Integrator
- Backup and Recovery Recommendations for Oracle WebCenter Content

These topics include information about configuration files for particular components. Note that the list of files in not an exhaustive list. You do not back up or recover the individual files. Generally, you back up or recover a Middleware home, the domain, Oracle home, or Oracle instance.

For the steps you take to back up your environment, see Section 17.3. For the steps you take to recover a component, see Chapter 18.

16.5.1 Backup and Recovery Recommendations for Oracle WebLogic Server

The following sections describe backup and recovery recommendations for Oracle WebLogic Server:

- Backup and Recovery Recommendations for Oracle WebLogic Server
- Backup and Recovery Recommendations for Oracle WebLogic Server JMS

16.5.1.1 Backup and Recovery Recommendations for Oracle WebLogic Server

This section describes the Oracle WebLogic Server data that must be backed up and restored.

Configuration Files

Configuration files and applications are stored in the domain home.

Database Repository Dependencies

Oracle WebLogic Server does not, by default, depend on any database repository. However, applications deployed on Oracle WebLogic Server may use databases as data sources. To back up a database, see the *Oracle Database Backup and Recovery User's Guide*.

Backup Recommendations

Back up the Middleware home and the domain.

Recovery Recommendations

Depending on what has failed, you may need to recover the following:

- The domain: See Section 18.2.2.
- The Administration Server configuration: See Section 18.2.5.
- A Managed Server: See Section 18.2.6.
- A cluster: See Section 18.2.8.
- Applications: See Section 18.2.9.

If you use Whole Server Migration, the leasing information is stored in a table in a database. If you recover Oracle WebLogic Server, you should discard the information in the leasing table. (For more information about Whole Server Migration, see "Whole Server Migration" in *Oracle Fusion Middleware Using Clusters for Oracle WebLogic Server*.)

After a loss of host, you may need to recover the following:

- The Administration Server host: See Section 18.3.2.
- The Managed Server host: See Section 18.3.3.

16.5.1.2 Backup and Recovery Recommendations for Oracle WebLogic Server JMS

This section describes the Oracle WebLogic Server JMS data that must be backed up and restored.

Configuration Files

DOMAIN_HOME/config/jms

If a JMS uses file-system accessible stores, the default file-system store is either in a user-configured location that is specified in config.xml, or in the following location:

DOMAIN_HOME/servers/*server_name*/data/store/default

Database Repository Dependencies

Only if JMS is database-based

Backup Recommendations

Back up the domain.

If you are using a database-based JMS, back up the database using RMAN.

If you are using file-based JMS, use storage snapshot techniques for taking consistent online backups. Alternatively, you can use a file system copy to perform an offline backup.

Recovery Recommendations

Recover the domain.

If the JMS persistent store is file-based, recover it from backup. If the JMS persistent store is database-based, recover the database to the most recent point in time, if needed. Note the following:

- Always try to keep JMS data as current as possible. This can be achieved by using the point-in-time recovery capabilities of Oracle Database, recovering to the most recent time (in the case of database-based persistence) or using a highly available RAID-backed storage device (for example, SAN/NAS).
- If you are using a file-based JMS, you can use storage snapshots to recover.
- If, for whatever reason, you need to restore JMS data to a previous point in time, there are potential implications. Restoring the system state to a previous point in time not only can cause duplicate messages, but can also cause lost messages. The lost messages are messages that were enqueued before or after the system restore point time, but never processed.

  Use the following procedure *before recovery* to drain messages in the JMS queue after persistent-store recovery to avoid processing duplicate messages:

Note: Do not drain and discard messages without first being certain that the messages contain no data that must be preserved. The recovered messages may include unprocessed messages with important application data, in addition to duplicate messages that have already been processed.

1. Log into the Oracle WebLogic Server Administration Console.
2. Before recovery, configure JMS server to pause Production, Insertion, and consumption operations at boot time to ensure that no new messages are produced or inserted into the destination or consumed from the destination before you drain stale messages. To do this:
   a. Expand **Services,** then **Messaging,** and then click **JMS Servers.**
   b. On the Summary of JMS Servers page, click the JMS server you want to configure for message pausing.
   c. On the Configuration: General page, click **Advanced** to define the message pausing options. Select **Insertion Paused At Startup, Production Paused At Startup,** and **Consumption Paused At Startup.**
   d. Click **Save.**

Use the following procedure *after recovery*:

1. After recovering the persistent store, start the Managed Servers.
2. Drain the stale messages from JMS destinations, by taking the following steps:
   a. Expand **Services,** then **Messaging,** and then **JMS Modules.**
   b. Select a JMS module, then select a target.
   c. Select **Monitoring,** then **Show Messages.**
3. Click **Delete All.**
4. Resume operations, by taking the following steps:
   a. Expand **Services,** then **Messaging,** and then **JMS Servers.**
   b. On the Summary of JMS Servers page, click the JMS server you want to configure for message pausing.
   c. On the Configuration: General page, click **Advanced.** Deselect **Insertion Paused At Startup, Production Paused At Startup,** and **Consumption Paused At Startup.**
   d. Click **Save.**

If the store is not dedicated to JMS use, use the Oracle WebLogic Server JMS message management administrative tool. This tool can perform import, export, move, and delete operations from the Administration Console, MBeans, and WLST.

For applications that use publish and subscribe in addition to queuing, you should manipulate topic subscriptions in addition to queues.

For the steps to recover the domain, see Section 18.2.2 and Section 18.3.1.

16.5.2 Backup and Recovery Recommendations for Oracle Identity Management

The following sections describe backup and recovery recommendations for Oracle Identity Management:

- Backup and Recovery Recommendations for Oracle Internet Directory
- Backup and Recovery Recommendations for Oracle Virtual Directory
- Backup and Recovery Recommendations for Oracle Directory Integration Platform
- Backup and Recovery Recommendations for Oracle Directory Services Manager
- Backup and Recovery Recommendations for Oracle Access Management Identity Federation
- Backup and Recovery Recommendations for Oracle Access Management Access Manager
- Backup and Recovery Recommendations for Oracle Adaptive Access Manager
- Backup and Recovery Recommendations for Oracle Identity Manager
- Backup and Recovery Recommendations for Oracle Identity Navigator
- Backup and Recovery Recommendations for Oracle Entitlements Server
- Backup and Recovery Recommendations for Oracle Privileged Account Manager
- Backup and Recovery Recommendations for Oracle Access Management Mobile and Social
- Backup and Recovery Recommendations for Oracle Access Management Secure Token Service

16.5.2.1 Backup and Recovery Recommendations for Oracle Internet Directory

This section describes the Oracle Internet Directory data that must be backed up and restored.

Configuration Files

ORACLE_INSTANCE/config/tnsnames.ora
ORACLE_INSTANCE/OID/admin
ORACLE_INSTANCE/OID/ldap/server/plugin
ORACLE_INSTANCE/OID/*component_name*
ORACLE_INSTANCE/config/OID/*component_name*

Database Repository Dependencies

ODS and ODSSM schemas

Backup Recommendations

Back up the Oracle Internet Directory component directory and the Oracle instance home that contains Oracle Internet Directory. Back up the database containing the ODS and ODSSM schemas.

Recovery Recommendations

Recover the Oracle instance home that contains Oracle Internet Directory.

Recover the database to the most recent point in time, if needed.

For the steps to recover the Oracle instance home that contains Oracle Internet Directory, see Section 18.2.4. For the steps specific to recovering from loss of host, see Section 18.3.4.5.1.

16.5.2.2 Backup and Recovery Recommendations for Oracle Virtual Directory

This section describes the Oracle Virtual Directory data that must be backed up and restored.

Configuration Files

ORACLE_INSTANCE/OVD/*component_name*
ORACLE_INSTANCE/config/OVD/*component_name*
ORACLE_INSTANCE/diagnostics/logs/OVD/*component_name*

Database Repository Dependencies

None

Backup Recommendations

Back up the Oracle instance home that contains Oracle Virtual Directory. Back up the database containing the ODSSM schema.

Recovery Recommendations

Restore the Oracle instance home that contains Oracle Virtual Directory.

For the steps to recover the Oracle instance home that contains Oracle Virtual Directory, see Section 18.2.4. For the steps specific to recovering from loss of host, see Section 18.3.4.5.2.

16.5.2.3 Backup and Recovery Recommendations for Oracle Directory Integration Platform

This section describes the Oracle Directory Integration Platform data that must be backed up and restored.

Configuration Files

DOMAIN_HOME/config/fmwconfig/servers/*server_name*/applications/*dip_version_number*/configuration/dip-config.xml

The file dip-config.xml is part of the Oracle Directory Integration Platform application. It is backed up when you back up the Administration Server domain.

Database Repository Dependencies

ODSSM schema, used by Oracle Internet Directory

Backup Recommendations

Back up the Administration Server domain directories, the Managed Server directories, and Oracle Internet Directory and its dependencies.

Recovery Recommendations

Recover the Managed Server where the Oracle Directory Integration Platform application is deployed.

Recover Oracle Internet Directory.

For the steps to recover the Managed Server, see Section 18.2.6. For the steps specific to recovering from loss of host, see Section 18.3.4.5.3.

16.5.2.4 Backup and Recovery Recommendations for Oracle Directory Services Manager

This section describes the Oracle Directory Services Manager data that must be backed up and restored.

Configuration Files

Oracle Directory Services Manager, which is the graphical user interface for Oracle Internet Directory and Oracle Virtual Directory, does not have configuration files, but keeps track of host and port information of Oracle Internet Directory and Oracle Virtual Directory in serverlist.txt, which is part of the application .ear file:

DOMAIN_HOME/servers/*server_name*/tmp/_WL_user/odsm_*version*/nx1i7i/war/WEB-INF/serverlist.txt

Database Repository Dependencies

None

Backup Recommendations

Back up the domain.

Recovery Recommendations

To restore Oracle Directory Services Manager, enter the user name and password to connect to Oracle Internet Directory or Oracle Virtual Directory.

For the steps to recover components, see Section 18.2.7. For the steps specific to recovering from loss of host, see Section 18.3.4.

16.5.2.5 Backup and Recovery Recommendations for Oracle Access Management Identity Federation

This section describes the Identity Federation data that must be backed up and restored.

Configuration Files

DOMAIN_HOME/servers/*server_name*/stage/OIF/*version*/OIF/configuration

Database Repository Dependencies

OIF schema

Backup Recommendations

Back up the Administration Server domain, the Managed Server, and the database containing the OIF schema.

Recovery Recommendations

Recover the Managed Server where the Identity Federation application is deployed.

Recover the database to the most recent point in time, if needed.

For the steps to recover the Managed Server, see Section 18.2.7. For the steps specific to recovering from loss of host, see Section 18.3.4.5.4.

16.5.2.6 Backup and Recovery Recommendations for Oracle Access Management Access Manager

This section describes the Access Manager data that must be backed up and restored.

Configuration Files

DOMAIN_HOME/config/fmwconfig/oam-config.xml

Database Repository Dependencies

The schema used by the Access Manager policy store.

Backup Recommendations

Back up the Middleware home and the domain home for the Access Manager server. Back up the Oracle home and the Oracle instance for the Oracle HTTP Server that contains the Webgate, and the database containing the schema used by the Access Manager policy store.

Recovery Recommendations

Recover the Middleware home and the domain home for the Access Manager server. Recover the Oracle home and the Oracle instance for the Oracle HTTP Server that contains the Webgate, as needed.

Recover the database to the most recent point in time, if needed.

For the steps to recover Access Manager, see Section 18.2.7.5. For the steps specific to recovering from loss of host, see Section 18.3.4.5.7.

16.5.2.7 Backup and Recovery Recommendations for Oracle Adaptive Access Manager

This section describes the Oracle Adaptive Access Manager data that must be backed up and restored.

Configuration Files

Configuration files are located within the domain home.

Database Repository Dependencies

OAAM, OAAM_PARTN, and OAAM_OFFLINE schemas

Backup Recommendations

Back up the domain, the Oracle home, and the database containing the schemas.

Recovery Recommendations

Recover the domain or Oracle home depending on the extent of the failure.

Recover the database to the most recent point in time, if needed.

For the steps to recover Oracle Adaptive Access Manager, see Section 18.2.7.6. For the steps specific to recovering from loss of host, see Section 18.3.4.5.8.

16.5.2.8 Backup and Recovery Recommendations for Oracle Identity Manager

This section describes the Oracle Identity Manager data that must be backed up and restored.

Configuration Files

Configuration files specific to Oracle WebLogic Server are located in the domain home. The Oracle Identity Manager configuration file, oim-config.xml, and other configurations are stored in MDS.

Because Oracle Identity Manager uses Oracle SOA Suite for workflow, see the configuration files for Oracle SOA Suite, described in Section 16.5.3.

Database Repository Dependencies

OIM, MDS, and Oracle SOA Suite schemas and, optionally, the OID schema

Backup Recommendations

Back up the domain, the Oracle home, and the database containing the schemas.

Recovery Recommendations

Recover the domain or Oracle home depending on the extent of the failure.

Recover the database to the most recent point in time, if needed.

For the steps to recover Oracle Identity Manager, see Section 18.2.7.3. For the steps specific to recovering from loss of host, see Section 18.3.4.5.5.

16.5.2.9 Backup and Recovery Recommendations for Oracle Identity Navigator

This section describes the Oracle Identity Navigator data that must be backed up and restored.

Configuration Files

Configuration files are stored in a file-based MDS repository.

Database Repository Dependencies

MDS schema

Backup Recommendations

Back up the domain and the Oracle home. Back up the file-based MDS repository using the WLST exportMetadata command. For example:

```
exportMetadata(application='oinav',server='server_name',toLocation='export_
directory')
```

Recovery Recommendations

Recover the domain, the Oracle home, and the file-based MDS repository.

For the steps to recover Oracle Identity Navigator, see Section 18.2.7.4. For the steps specific to recovering from loss of host, see Section 18.3.4.5.6.

16.5.2.10 Backup and Recovery Recommendations for Oracle Entitlements Server

This section describes the Oracle Entitlements Server data that must be backed up and restored.

Configuration Files

Configuration files are stored the domain home and the Oracle home.

Database Repository Dependencies

Oracle Platform Security Services policy store schema, MDS schema

Backup Recommendations

Back up the domain home and the Oracle home. Back up the policy store. (Alternatively, you can migrate the database or LDAP policy store into an XML file and back up that file.)

Recovery Recommendations

Recover the domain home or the Oracle home or both. Also recover the database or LDAP policy store, if needed. Alternatively, if you backed up the policy store into an XML file, you can restore it from the backup file.

For the steps to recover the domain home, see Section 18.2.2. For the steps to recover the Oracle home, see Section 18.2.3. To recover from loss of host, see Section 18.3.1 and Section 18.2.3.

16.5.2.11 Backup and Recovery Recommendations for Oracle Privileged Account Manager

This section describes the Oracle Privileged Account Manager data that must be backed up and restored.

Configuration Files

Configuration files are stored in the domain home and the Oracle home.

Database Repository Dependencies

Oracle Platform Security Services policy store schema.

Backup Recommendations

Back up the domain home and the Oracle home. Back up the policy store.

Recovery Recommendations

Recover the domain home or the Oracle home or both. Also recover the database or LDAP policy store, if needed.

For the steps to recover the domain home, see Section 18.2.2. For the steps to recover the Oracle home, see Section 18.2.3. For the steps specific to recovering from loss of host, see Section 18.3.4.5.11.

16.5.2.12 Backup and Recovery Recommendations for Oracle Access Management Mobile and Social

This section describes the Mobile and Social data that must be backed up and restored.

Configuration Files

Configuration files are stored in the domain home.

Database Repository Dependencies

None

Backup Recommendations

Back up the domain home, Oracle home, and the IdaaS.xml and OIC_RP.xml files. These files are located in the following location in the domain home containing Mobile and Social configuration:

```
DOMAIN_HOME/config/fmwconfig
```

Recovery Recommendations

Recover the domain home or the Oracle home or both, if necessary. Also, recover the image location and configuration, depending upon extent of failure

For the steps to recover the domain home, see Section 18.2.2. For the steps to recover the Oracle home, see Section 18.2.3. For the steps specific to recovering from loss of host, to the same host, Section 18.2.2 and Section 18.2.3. For the steps for recovering to a different host, see Section 18.3.4.5.9.

16.5.2.13 Backup and Recovery Recommendations for Oracle Access Management Secure Token Service

This section describes the Secure Token Service data that must be backed up and restored.

Configuration Files

DOMAIN_HOME/config/fmwconfig/oam-config.xml

Database Repository Dependencies

Database data used by Oracle Entitlements Server for the Access Manager and Secure Token Service policy store.

Backup Recommendations

Back up the Middleware home and domain home where the Access Manager and Secure Token Service are configured.

Recovery Recommendations

Recover the Middleware home and domain home where the Access Manager and Secure Token Service are configured.

For the steps to recover the Middleware home, see Section 18.2.1. For the steps to recover the domain home, see Section 18.2.2. For the steps specific to recovering from loss of host, see Section 18.3.4.5.10.

16.5.3 Backup and Recovery Recommendations for Oracle SOA Suite

The following sections describe backup and recovery recommendations for Oracle SOA Suite:

- Backup and Recovery Recommendations for Oracle BPEL Process Manager
- Backup and Recovery Recommendations for Oracle Business Activity Monitoring
- Backup and Recovery Recommendations for Oracle B2B
- Backup and Recovery Recommendations for Oracle Service Bus
- Backup and Recovery Recommendations for Oracle Mediator
- Backup and Recovery Recommendations for Oracle Business Rules
- Backup and Recovery Recommendations for Oracle Business Process Management

For the steps you need to take to recover components, see Section 18.2.7. For the steps specific to recovering from loss of host, see Section 18.3.4.6.

16.5.3.1 Backup and Recovery Recommendations for Oracle BPEL Process Manager

This section describes the Oracle BPEL Process Manager data that must be backed up and restored.

Configuration Files

Configuration files are stored in the database.

Database Repository Dependencies

Process definition and configuration files are stored in the MDS schema. The dehydration store is stored in the BPEL schema.

Backup Recommendations

Back up the Administration Server domain directories. Back up the database after any configuration changes, including changes to global fault policies, callback classes for workflows and resource bundles that can potentially be outside the suitcase. Also back up the database after deploying a new composite or redeploying a composite.

Recovery Recommendations

Recover the database to the most recent point in time, if needed. Point-in-time recovery ensures that the latest process definitions and in-flight instances are restored. However, this may result in reexecution of the process steps. Oracle recommends that you strive for idempotent Oracle BPEL Process Manager processes. If the system contains processes that are not idempotent, you must clean them up from the dehydration store before starting Oracle Fusion Middleware. See the *Oracle Fusion Middleware Administrator's Guide for Oracle SOA Suite and Oracle Business Process Management Suite* for more information.

Because instances obtain the process definition and artifacts entirely from the database, there is no configuration recovery needed after the database is recovered to the most current state; instances should continue to function correctly.

For redeployed composites, a database recovery ensures consistency between the dehydrated in-flight processes and their corresponding definition since the process definition is stored in database repository where dehydrated instances are also stored.

For the steps to recover components, see Section 18.2.7. For the steps specific to recovering from loss of host, see Section 18.3.4.6.

16.5.3.2 Backup and Recovery Recommendations for Oracle Business Activity Monitoring

This section describes the Oracle Business Activity Monitoring data that must be backed up and restored.

Configuration Files

SOA_ORACLE_HOME/bam
DOMAIN_HOME/config/fmwconfig/servers/AdminServer/adml/server-oracle_bamweb-11.0.xml
DOMAIN_HOME/config/fmwconfig/servers/AdminServer/adml/server-oracle_bamserver-11.0.xml
DOMAIN_HOME/config/fmwconfig/servers/*bam-server-name*/adml/server-oracle_bamweb-11.0.xml
DOMAIN_HOME/config/fmwconfig/servers/*bam-server-name*/adml/server-oracle_bamserver-11.0.xml

Database Repository Dependencies

ORABAM schema

Backup Recommendations

Back up the Middleware home, the Administration Server domain, the Managed Server directory, and the database containing the ORABAM schema.

Recovery Recommendations

Recover the Managed Server or the Middleware home, or both, depending on the extent of failure.

Recover the database to the most recent point in time, if needed.

For the steps to recover components, see Section 18.2.7. For the steps specific to recovering from loss of host, see Section 18.3.4.6.

16.5.3.3 Backup and Recovery Recommendations for Oracle B2B

This section describes the Oracle B2B data that must be backed up and restored.

Configuration Files

DOMAIN_HOME/config/soa-infra/configuration/b2b-config.xml

Database Repository Dependencies

MDS schema

Backup Recommendations

Back up the Administration Server domain, the Oracle home if changes are made to the Oracle B2B configuration file, and the database containing the MDS schema.

Recovery Recommendations

Recover the Managed Server where the soa-infra application is deployed.

Recover the database to the most recent point in time, if needed.

After recovery, if the file Xengine.tar.gz is not unzipped, unzip the files. For example:

```
cd B2B_ORACLE_HOME/soa/thirdparty/edifecs
tar xzvf XEngine.tar.gz
```

For the steps to recover components, see Section 18.2.7. For the steps specific to recovering from loss of host, see Section 18.3.4.6.

16.5.3.4 Backup and Recovery Recommendations for Oracle Service Bus

This section describes the Oracle Service Bus data that must be backed up and recovered.

Configuration Files

DOMAIN_HOME/osb/config/core

Database Repository Dependencies

Oracle Service Bus requires a database if its reporting feature is enabled. It creates two tables, WLI_QS_REPORT_DATA and WLI_QS_REPORT_ATTRIBUTE, in a user-specified schema.

Backup Recommendations

Back up the Administration Server domain and the database containing the Oracle Service Bus tables.

Recovery Recommendations

Recover the Managed Server.

Recover the database to the most recent point in time, if needed.

For the steps you need to take to recover components, see Section 18.2.7. For the steps specific to recovering from loss of host, see Section 18.3.4.6.

16.5.3.5 Backup and Recovery Recommendations for Oracle Mediator

This section describes the Oracle Mediator data that must be backed up and restored.

Configuration Files

DOMAIN_HOME/config/soa-infra/configuration/mediator-config.xml
DOMAIN_HOME/config/soa-infra/configuration/mediator-xpath-functions-config.xml

Database Repository Dependencies

MDS and SOAINFRA schemas.

Backup Recommendations

Back up the Administration Server domain and the database containing the MDS and SOAINFRA schemas.

Recovery Recommendations

Recover the Managed Server where the soa-infra application is deployed.

Recover the database to the most recent point in time, if needed.

For the steps you need to take to recover components, see Section 18.2.7 and Section 18.3.4.

For recommendations specific to recovering from loss of host, see Section 18.3.4.6.

16.5.3.6 Backup and Recovery Recommendations for Oracle Business Rules

This section describes the Oracle Business Rules data that must be backed up and restored.

Configuration Files

DOMAIN_HOME/config/soa-infra/configuration/businessrules-config.xml

Database Repository Dependencies

MDS schema

Backup Recommendations

Back up the Administration Server domain and the database containing the MDS schema.

Recovery Recommendations

Recover the Managed Server where the soa-infra application is deployed.

Recover the database to the most recent point in time, if needed.

For the steps to recover components, see Section 18.2.7. For the steps specific to recovering from loss of host, see Section 18.3.4.6.

16.5.3.7 Backup and Recovery Recommendations for Oracle Business Process Management

For Oracle Business Process Management, you back up and restore the same data as Oracle BPEL Process Manager, as described in Section 16.5.3.1. This section describes data specific to Oracle Business Process Management.

Configuration Files

DOMAIN_HOME/config/fmwconfig/logging/oracle.bpm-logging.xml
DOMAIN_HOME/config/jms/bpmjmsmodule-jms.xml

Database Repository Dependencies

Process definition and configuration files are stored in the MDS schema.

Backup Recommendations

In addition to the recommendations for Oracle BPEL Process Manager, described in Section 16.5.3.1, you must back up the Oracle homes, including all Oracle homes in a cluster. When you extend a SOA domain to Oracle Business Process Management and configure Oracle Business Process Management, the process adds files to the Oracle Business Process Management Oracle home. However, it does not copy the files to any other Oracle homes in the cluster. After you configured Oracle Business Process Management, you should have copied the files to the other Oracle homes in the cluster. As a result, you must back up all Oracle homes in the cluster.

Recovery Recommendations

In addition to the recommendations for Oracle BPEL Process Manager, described in Section 16.5.3.1, you must recover all of the Oracle homes in the cluster.

For the steps to recover Oracle Business Process Management, see Section 18.2.7.7.

16.5.4 Backup and Recovery Recommendations for Oracle WebCenter Portal

The following sections describe backup and recovery recommendations for Oracle WebCenter Portal:

- Backup and Recovery Recommendations for Oracle WebCenter Portal
- Backup and Recovery Recommendations for Oracle WebCenter Portal's Portlet Producer
- Backup and Recovery Recommendations for Oracle WebCenter Portal's Discussion Server
- Backup and Recovery Recommendations for Oracle WebCenter Portal's Activity Graph
- Backup and Recovery Recommendations for Oracle WebCenter Portal's Analytics
- Backup and Recovery Recommendations for Oracle Content Server

16.5.4.1 Backup and Recovery Recommendations for Oracle WebCenter Portal

This section describes the Oracle WebCenter Portal data that must be backed up and restored.

Configuration Files

All configuration files are bundled in the EAR file, which is located in the domain.

Database Repository Dependencies

WEBCENTER and MDS schemas

Backup Recommendations

Back up the Administration Server domain and the database containing the WEBCENTER and MDS schemas.

Recovery Recommendations

Recover the Oracle WebCenter Portal domain.

Recover the database containing the WEBCENTER and MDS schemas to the most recent point in time, if needed.

For the steps to recover components, see Section 18.2.7. For the steps specific to recovering from loss of host, see Section 18.3.4.

16.5.4.2 Backup and Recovery Recommendations for Oracle WebCenter Portal's Portlet Producer

This section describes the Oracle WebCenter Portal's Portlet Producer data that must be backed up and restored.

Configuration Files

All configuration files are bundled in the EAR file, which is located in the domain.

Database Repository Dependencies

PORTLET schema

Backup Recommendations

Back up the Administration Server domain and the database containing the PORTLET schema.

Recovery Recommendations

Recover the Oracle WebCenter Portal domain.

Recover the database to the most recent point in time, if needed.

For the steps to recover components, see Section 18.2.7. For the steps specific to recovering from loss of host, see Section 18.3.4.

16.5.4.3 Backup and Recovery Recommendations for Oracle WebCenter Portal's Discussion Server

This section describes the Oracle WebCenter Portal's Discussion Server data that must be backed up and restored.

Configuration Files

Some configuration files are either bundled in the EAR file, which is located in the domain, or the files are located elsewhere in the domain. Other configuration files are located in:

DOMAIN_HOME/fmwconfig/server/*server_name*/owc_discussions

Database Repository Dependencies

DISCUSSIONS schema

Backup Recommendations

Back up the Administration Server domain and the database containing the DISCUSSIONS schema.

Recovery Recommendations

Recover the Oracle WebCenter Portal domain.

Recover the database to the most recent point in time, if needed.

For the steps to recover components, see Section 18.2.7. For the steps specific to recovering from loss of host, see Section 18.3.4.

16.5.4.4 Backup and Recovery Recommendations for Oracle WebCenter Portal's Activity Graph

This section describes the Oracle WebCenter Portal's Activity Graph data that must be backed up and restored.

Configuration Files

Configuration information is stored in the ACTIVITIES schema.

Database Repository Dependencies

ACTIVITIES schema

Backup Recommendations

Back up the Oracle home, the domain home, and the database containing the ACTIVITIES schema.

Recovery Recommendations

Recover the Oracle home and the domain home.

Recover the database to the most recent point in time, if needed.

For the steps to recover Oracle WebCenter Portal's Activity Graph, see Section 18.2.7.8.

16.5.4.5 Backup and Recovery Recommendations for Oracle WebCenter Portal's Analytics

This section describes the Oracle WebCenter Portal's Analytics data that must be backed up and restored.

Configuration Files

Configuration information is stored in the Analytics schema, ACTIVITIES.

Database Repository Dependencies

ACTIVITIES and MDS schema

Backup Recommendations

Back up the Oracle home, the domain home, and the database containing the ACTIVITIES and MDS schemas.

Recovery Recommendations

Recover the Oracle home and the domain home.

Recover the database to the most recent point in time, if needed.

For the steps to recover Oracle WebCenter Portal's Analytics, see Section 18.2.7.9.

16.5.4.6 Backup and Recovery Recommendations for Oracle Content Server

For information about backing up and recovering Oracle Content Server, see *Getting Started with Content Server* which is available at:

`http://download.oracle.com/docs/cd/E10316_01/owc.htm`

For information about backing up and recovering Oracle WebCenter Content, see Section 16.5.11.3.

Database Repository Dependencies

OCS schema

16.5.5 Backup and Recovery Recommendations for Oracle JRF Installations

The following topics describe backup and recovery recommendations for components that are installed with more than one type of installation:

- Backup and Recovery Recommendations for Oracle Web Services Manager
- Backup and Recovery Recommendations for Oracle Platform Security Services

16.5.5.1 Backup and Recovery Recommendations for Oracle Web Services Manager

This section describes the Oracle Web Services Manager data that must be backed up and restored.

Configuration Files

DOMAIN_HOME`/config/fmwconfig/policy-accessor-config.xml`

Database Repository Dependencies

If a database-based MDS Repository is used, Oracle Web Services Manager uses a partition in the MDS schema.

Backup Recommendations

Back up the Oracle Web Services Manager domain.

If Oracle Web Services Manager uses a file-based MDS Repository, back it up using a file copy mechanism. If it uses a database-based MDS Repository, back up the database using RMAN.

Recovery Recommendations

Restore the Oracle Web Services Manager Managed Server.

If Oracle Web Services Manager uses a file-based MDS Repository, restore it from the backup. If it uses a database-based MDS Repository, recover the database to the most recent point in time, if needed.

For the steps to recover components, see Section 18.2.7. For the steps specific to recovering from loss of host, see Section 18.3.4.

16.5.5.2 Backup and Recovery Recommendations for Oracle Platform Security Services

This section describes the Oracle Platform Security Services data that must be backed up and restored.

Configuration Files

DOMAIN_HOME/config/fmwconfig/jps-config.xml

Database Repository Dependencies

If a database-based Oracle Platform Security Repository is used, Oracle Platform Security uses a partition in the OPSS schema.

If an Oracle Internet Directory based Oracle Platform Security repository is used, Oracle Platform Security, uses Oracle Internet Directory.

Backup Recommendations

Back up the Administration Server domain. Back up Oracle Internet Directory if Oracle Platform Security uses an Oracle Internet Directory based repository.

Backup the database containing the OPSS schema if Oracle Platform Security uses a database-based repository.

Recovery Recommendations

Restore the jps-config.xml file.

If Oracle Platform Security uses a database-based repository, restore the database to the most recent point in time.

For the steps to recover components, see Section 18.2.7. For the steps specific to recovering from loss of host, see Section 18.3.4 and Section 18.3.4.5.1.

16.5.6 Backup and Recovery Recommendations for Web Tier Installations

The following sections describe backup and recovery recommendations for Web Tier installations:

- Backup and Recovery Recommendations for Oracle HTTP Server
- Backup and Recovery Recommendations for Oracle Web Cache

16.5.6.1 Backup and Recovery Recommendations for Oracle HTTP Server

This section describes the Oracle HTTP Server data that must be backed up and restored.

Configuration Files

ORACLE_INSTANCE/config/OHS/*component_name*
ORACLE_INSTANCE/diagnostics/logs/OHS/*component_name*

Database Repository Dependencies

None

Backup Recommendations

Back up the Oracle instance that contains Oracle HTTP Server.

Recovery Recommendations

Restore the Oracle instance that contains Oracle HTTP Server.

For the steps to recover components, see Section 18.2.7. For the steps specific to recovering from loss of host, see Section 18.3.4 andSection 18.3.4.7.1.

16.5.6.2 Backup and Recovery Recommendations for Oracle Web Cache

This section describes the Oracle Web Cache data that must be backed up and restored.

Configuration Files

ORACLE_INSTANCE/config/WebCache/*component_name*
ORACLE_INSTANCE/diagnostics/logs/WebCache/*component_name*

Database Repository Dependencies

None

Backup Recommendations

Back up the Oracle instance that contains Oracle Web Cache.

Recovery Recommendations

Restore the Oracle instance that contains Oracle Web Cache.

For the steps to recover components, see Section 18.2.7. For the steps specific to recovering from loss of host, see Section 18.3.4 and Section 18.3.4.7.2.

16.5.7 Backup and Recovery Recommendations for Oracle Portal, Oracle Forms Services, Oracle Reports, and Oracle BI Discoverer Installations

The following sections describe backup and recovery recommendations for these components:

- Backup and Recovery Recommendations for Oracle Portal
- Backup and Recovery Recommendations for Oracle Forms Services
- Backup and Recovery Recommendations for Oracle Reports
- Backup and Recovery Recommendations for Oracle Business Intelligence Discoverer

16.5.7.1 Backup and Recovery Recommendations for Oracle Portal

This section describes the Oracle Portal data that must be backed up and restored.

Configuration Files

DOMAIN_HOME/config/fmwconfig/servers/WLS_PORTAL/applications/portal/configuration/appConfig.xml
DOMAIN_HOME/config/fmwconfig/servers/WLS_PORTAL/applications/portal/configuration/portal_dads.conf
DOMAIN_HOME/config/fmwconfig/servers/WLS_PORTAL/applications/portal/configuration/portal_plsql.conf
DOMAIN_HOME/config/fmwconfig/servers/WLS_PORTAL/applications/portal/configuration/portal_cache.conf

Database Repository Dependencies

PORTAL, PORTAL_DEMO, PORTAL_APP, PORTAL_PUBLIC, AND PORTAL_APPROVAL schemas

Backup Recommendations

Back up the Administration Server domain, the Managed Server directory, the Oracle instance containing Oracle Portal, and the database containing the schemas.

Recovery Recommendations

Recover the WebLogic Server domain and the Oracle instance containing Oracle Portal.

Recover the database to the most recent point in time, if needed.

For the steps to recover components, see Section 18.2.7. For the steps specific to recovering from loss of host, see Section 18.3.4 and Section 18.3.4.8.1.

16.5.7.2 Backup and Recovery Recommendations for Oracle Forms Services

This section describes the Oracle Forms Services data that must be backed up and restored.

Configuration Files

Forms Component:

ORACLE_INSTANCE/config/Forms/forms
ORACLE_INSTANCE/Forms/forms

Forms Common Component:

ORACLE_INSTANCE/config/Forms/frcommon
ORACLE_INSTANCE/Forms/frcommon

Forms EE application and its configuration files:

DOMAIN_HOME/*forms_managed_server*/tmp/_WL_user/formsapp_*version*
DOMAIN_HOME/config/fmwconfig/servers/*forms_managed_server*/applications/formsapp_*version*/config

Database Repository Dependencies

Any user-configured database for Oracle Forms Services applications.

Backup Recommendations

Back up the Administration Server domain, the Managed Server directory, and the Oracle instance home where Oracle Forms Services is located.

Recovery Recommendations

Restore the Oracle instance home where Oracle Forms Services is located.

For the steps to recover components, see Section 18.2.7. For the steps specific to recovering from loss of host, see Section 18.3.4 and Section 18.3.4.8.2.

16.5.7.3 Backup and Recovery Recommendations for Oracle Reports

This section describes the Oracle Reports data that must be backed up and restored.

Configuration Files

For Reports Server:

ORACLE_INSTANCE/config/ReportsServer/*server_name*/rwserver.conf
ORACLE_INSTANCE/config/ReportsServer/*server_name*/jdbcpds.conf
ORACLE_INSTANCE/config/ReportsServer/*server_name*/xmlpds.conf
ORACLE_INSTANCE/config/ReportsServer/*server_name*/textpds.conf

ORACLE_INSTANCE/config/ReportsServer/*server_name*/rwnetwork.conf
ORACLE_INSTANCE/config/ReportsServer/*server_name*/pcscomponent.conf
ORACLE_INSTANCE/config/ReportsServer/*server_name*/component-logs.xml
ORACLE_INSTANCE/config/ReportsServer/*server_name*/logging.xml

For Oracle Reports Servlet:

In the following paths, `server_name` is usually WLS_REPORTS or WLS_REPORTS*n* and `version` is the version of the software, for example, 11.1.1.4.0:

DOMAIN_HOME/config/fmwconfig/servers/*server_name*/applications/reports_*version*/configuration/cgicmd.dat
DOMAIN_HOME/config/fmwconfig/servers/*server_name*/applications/reports_*version*/configuration/rwservlet.properties
DOMAIN_HOME/config/fmwconfig/servers/*server_name*/applications/reports_*version*/configuration/rwserver.conf
DOMAIN_HOME/config/fmwconfig/servers/*server_name*/applications/reports_*version*/configuration/jdbcpds.conf
DOMAIN_HOME/config/fmwconfig/servers/*server_name*/applications/reports_*version*/configuration/xmlpds.conf
DOMAIN_HOME/config/fmwconfig/servers/*server_name*/applications/reports_*version*/configuration/textpds.conf
DOMAIN_HOME/config/fmwconfig/servers/*server_name*/applications/reports_*version*/configuration/rwnetwork.conf
DOMAIN_HOME/config/fmwconfig/servers/*server_name*/applications/reports_*version*/configuration/logging.xml
DOMAIN_HOME/config/fmwconfig/servers/*server_name*/applications/reports_*version*/configuration/logmetadata.xml

For Oracle Reports Bridge:

ORACLE_INSTANCE/config/ReportsBridge/*bridge_name/rwbridge.conf*
ORACLE_INSTANCE/config/ReportsBridge/*bridge_name*/rwnetwork.conf
ORACLE_INSTANCE/config/ReportsBridge/*bridge_name*/component-logs.xml
ORACLE_INSTANCE/config/ReportsBridge/*bridge_name*/logging.xml
ORACLE_INSTANCE/config/ReportsBridge/*bridge_name*/pcscomponent.xml

For Oracle Reports Tool:

ORACLE_INSTANCE/config/ReportsTools/rwbuilder.conf
ORACLE_INSTANCE/config/ReportsTools/rwnetwork.conf
ORACLE_INSTANCE/config/ReportsTools/jdbcpds.conf
ORACLE_INSTANCE/config/ReportsTools/xmlpds.conf
ORACLE_INSTANCE/config/ReportsTools/textpds.conf
ORACLE_INSTANCE/config/ReportsTools/pcscomponent.xml
ORACLE_INSTANCE/config/ReportsTools/rwservlet.properties
ORACLE_INSTANCE/config/ReportsTools/cgicmd.dat
ORACLE_INSTANCE/config/ReportsTools/component-logs.xml
ORACLE_INSTANCE/config/ReportsTools/logging.xml

Other directories and files:

ORACLE_INSTANCE/reports/server/*.dat
ORACLE_INSTANCE/reports/cache/
ORACLE_INSTANCE/reports/fonts/
ORACLE_INSTANCE/reports/plugins/resource
ORACLE_INSTANCE/diagnostics/logs/reports/ReportsServer
ORACLE_INSTANCE/diagnostics/logs/reports/ReportsBridge
ORACLE_INSTANCE/diagnostics/logs/reports/ReportsTools
(UNIX) *ORACLE_INSTANCE*/config/reports/bin/rw*.sh
(Windows) *ORACLE_INSTANCE*\config\reports\bin\rw*.bat
(UNIX) *ORACLE_INSTANCE*/config/reports/bin/reports.sh

(Windows) *ORACLE_INSTANCE*\config\reports\bin\reports.bat
(UNIX) *ORACLE_INSTANCE*/config/reports/bin/namingservice.sh
(Windows) *ORACLE_INSTANCE*\config\reports\bin\namingservice.bat

Database Repository Dependencies

You can configure Oracle Reports to store job-related information, such as scheduled job data, past job data, or job status data in a database.

Backup Recommendations

Back up the Administration Server domain, the Managed Server directory, and the Oracle instance home where Oracle Reports is located.

If a database is configured for Oracle Reports, back up the database.

Recovery Recommendations

Restore the Oracle instance home where Oracle Reports is located.

If a database is configured for Oracle Reports, recover the database to most recent point in time, if needed.

For the steps to recover components, see Section 18.2.7. For the steps specific to recovering from loss of host, see Section 18.3.4 and Section 18.3.4.8.3.

16.5.7.4 Backup and Recovery Recommendations for Oracle Business Intelligence Discoverer

This section describes the Oracle Business Intelligence Discoverer data that must be backed up and restored.

Configuration Files

ORACLE_INSTANCE/config/PreferenceServer/*disco-comp-name*/pref.txt
ORACLE_INSTANCE/config/PreferenceServer/*disco-comp-name*/.reg_key.dc
DOMAIN_HOME/config/fmwconfig/servers/WLS_DISCO/applications/discoverer_*version*/configuration/configuration.xml
DOMAIN_HOME/config/config.xml
DOMAIN_HOME/config/fmwconfig/servers/*server_name*/logging.xml

Log Files

ORACLE_INSTANCE/diagnostics/logs/PreferenceServer/Discoverer_*instance_name*/console*
ORACLE_INSTANCE/diagnostics/logs/PreferenceServer/Discoverer_*instance_name*/log*
DOMAIN_HOME/servers/*server_name*/logs/discoverer/diagnostic-*.xml
DOMAIN_HOME/servers/*server_name*/logs/discoverer/diagnostics*.xml
DOMAIN_HOME/servers/*server_name*/logs/WLS_DISCO-diagnostic-*.xml

Database Repository Dependencies

DISCOVERER and DISCOVERER_PS schemas

Backup Recommendations

Back up the Administration Server domain, the Managed Server directory, and the Oracle BI Discoverer Oracle instance home.

Back up the database containing the DISCOVERER and DISCOVERER_PS schemas.

Recovery Recommendations

Restore the Oracle instance that contains Oracle BI Discoverer.

Recover the database to the most recent point in time, if needed.

For the steps to recover components, see Section 18.2.7. For the steps specific to recovering from loss of host, see Section 18.3.4 and Section 18.3.4.8.4.

16.5.8 Backup and Recovery Recommendations for Oracle Business Intelligence

The following sections describe backup and recovery recommendations for Oracle Business Intelligence:

- Backup and Recovery Recommendations for Oracle BI Enterprise Edition
- Backup and Recovery Recommendations for Oracle Business Intelligence Publisher
- Backup and Recovery Recommendations for Oracle Real-Time Decisions

16.5.8.1 Backup and Recovery Recommendations for Oracle BI Enterprise Edition

This section describes the Oracle BI EE data that must be backed up and restored.

Configuration Files

ORACLE_INSTANCE/bifoundation/OracleBIApplication
ORACLE_INSTANCE/bifoundation/OracleBIClusterControllerComponent
ORACLE_INSTANCE/bifoundation/OracleBIJavaHostComponent
ORACLE_INSTANCE/bifoundation/OracleBIPresentationServicesComponent
ORACLE_INSTANCE/bifoundation/OracleBISchedulerComponent
ORACLE_INSTANCE/bifoundation/OracleBIServerComponent
ORACLE_INSTANCE/bifoundation/OracleBIODBCComponent
ORACLE_INSTANCE/config/OracleBIApplication
ORACLE_INSTANCE/config/OracleBIClusterControllerComponent
ORACLE_INSTANCE/config/OracleBIJavaHostComponent
ORACLE_INSTANCE/config/OracleBIPresentationServicesComponent
ORACLE_INSTANCE/config/OracleBISchedulerComponent
ORACLE_INSTANCE/config/OracleBIServerComponent
ORACLE_INSTANCE/config/OracleBIODBCComponent
ORACLE_INSTANCE/diagnostics/logs/OracleBIApplication
ORACLE_INSTANCE/diagnostics/logs/OracleBIClusterControllerComponent
ORACLE_INSTANCE/diagnostics/logs/OracleBIJavaHostComponent
ORACLE_INSTANCE/diagnostics/logs/OracleBIPresentationServicesComponent
ORACLE_INSTANCE/diagnostics/logs/OracleBISchedulerComponent
ORACLE_INSTANCE/diagnostics/logs/OracleBIServerComponent
ORACLE_INSTANCE/diagnostics/logs/OracleBIODBCComponent

In addition, the following files in a file-based repository:

ORACLE_INSTANCE/bifoundation/OracleBIServerComponent/*comp_instance name*/repository/*.rpd
ORACLE_INSTANCE/bifoundation/OracleBIPresentationServicesComponent/*comp_instance name*/catalog/*catalog-name*

The NQSConfig.INI configuration file points to the RPD name. The NQSConfig.INI file *must* exist in the following location:

ORACLE_INSTANCE/bifoundation/OracleBIServerComponent/*comp_instance name*/repository/

Database Repository Dependencies

MDS and BIPLATFORM schemas

Backup Recommendations

Back up the Middleware home, the domain home, and the Oracle instance containing the Oracle BI EE components. On Windows, export Oracle BI EE Registry entries, as described in Section 17.3.3.

Back up the database containing the Oracle BI EE schemas.

Note: Before you perform a backup, you must lock the Oracle BI Presentation Catalogs so that the catalog and RPD remain synchronized. Run the following script:

```
ORACLE_
INSTANCE/bifoundation/OracleBIPresentationServicesComponent/coreapp
lication_obips1/catalogmanager/runcat.sh
```

Use the following command:

```
./runcat.sh -cmd maintenanceMode -on -online OBIPS_URL
     -credentials credentials_properties_file
```

After the backup is complete, turn off maintenance mode using the runcat command. For information on this command, see the help:

```
./runcat.sh -cmd maintenanceMode -help
```

Recovery Recommendations

Depending on the extent of the failure, recover the Middleware home, the domain, and the Oracle instance containing the Oracle BI EE components. On Windows, import Oracle BI EE Registry entries.

Recover the database to the most recent point in time, if needed.

For the steps to recover Oracle BI EE, see Section 18.2.7.10. For the steps specific to recovering from loss of host, see Section 18.3.4.9.

16.5.8.2 Backup and Recovery Recommendations for Oracle Business Intelligence Publisher

This section describes the Oracle Business Intelligence Publisher data that must be backed up and restored.

Configuration Files

Configuration files are located in the Middleware home, the domain home, and the Oracle Business Intelligence Publisher repository.

Database Repository Dependencies

BIPLATFORM schema

Backup Recommendations

Back up the Middleware home, the domain, and the BI Publisher repository.

The BI Publisher repository can be file-based or database-based.

Recovery Recommendations

Recover the Managed Server containing the Oracle BI Publisher component.

Recover the database to the most recent point in time, if needed.

For the steps to recover Oracle BI Publisher, see Section 18.2.7.11. For the steps specific to recovering from loss of host, see Section 18.3.4.10.

16.5.8.3 Backup and Recovery Recommendations for Oracle Real-Time Decisions

This section describes the Oracle Real-Time Decisions data that must be backed up and restored.

Configuration Files

Configuration files are located in the Middleware home and the domain home.

Database Repository Dependencies

Database containing analytic models and the RTD schema

Backup Recommendations

Back up the Middleware home, the domain home, and the database containing analytic models

Recovery Recommendations

Recover the Managed Server containing the Oracle Real-Time Decisions component.

Recover the database to the most recent point in time, if needed.

For the steps to recover Oracle Real-Time Decisions, see Section 18.2.7.12. For the steps specific to recovering from loss of host, see Section 18.3.4.11.

16.5.9 Backup and Recovery Recommendations for Oracle Hyperion Enterprise Performance Management System

The following topics describe backup and recovery recommendations for Oracle Hyperion Enterprise Performance Management System:

- Backup and Recovery Recommendations for Oracle Essbase
- Backup and Recovery Recommendations for Oracle Hyperion Calculation Manager
- Backup and Recovery Recommendations for Oracle Hyperion Financial Reporting
- Backup and Recovery Recommendations for Oracle Hyperion Smart View

16.5.9.1 Backup and Recovery Recommendations for Oracle Essbase

This section describes the Oracle Essbase data that must be backed up and restored.

Configuration Files

Configuration files are located in the following directories:

ORACLE_HOME
HYPERION_HOME
ESSBASEPATH
ARBORPATH
ARBORPATH/bin/essbase.sec

Dependencies on Oracle Fusion Middleware Components

Oracle Internet Directory, Oracle Fusion Middleware Extensions for Applications, and the Credential Store

Dependencies on Third-Party Products

None

Database Repository Dependencies

Oracle Essbase schemas

Backup Recommendations

Back up Oracle Essbase using storage snapshot techniques for taking consistent online backups. Alternatively, you can use a file system copy to perform an offline backup.

Recovery Recommendations

Recover the following files:

```
ARBORPATH/app/appname/*.otl
ARBORPATH/app/appname/*.csc
ARBORPATH/app/appname/*.rul
ARBORPATH/app/appname/*.eqd
ARBORPATH/app/appname/*.sel
ARBORPATH/bin/essbase.sec
```

Recover the database to the most recent point in time, if needed.

Depending upon the extent of failure, recovery should be performed at the desired granularity. For the steps to recover Oracle Essbase, see Section 18.2.7.13. For the steps specific to recovering from loss of host, see Section 18.3.4.12.

16.5.9.2 Backup and Recovery Recommendations for Oracle Hyperion Calculation Manager

This section describes the Oracle Hyperion Calculation Manager data that must be backed up and restored.

Configuration Files

Configuration files are stored in the database.

Dependencies on Oracle Fusion Middleware Components

None.

Dependencies on Third-Party Products

None.

Database Repository Dependencies

Artifacts are stored in a database.

Backup Recommendations

Back up the Managed Server to which Oracle Hyperion Calculation Manager is deployed. Back up the artifacts in the database by using the Oracle Hyperion Calculation Manager export tool. (In Calculation Manager, select **File,** and then **Export**.)

Recovery Recommendations

Recover the Managed Server to which Oracle Hyperion Calculation Manager is deployed.

Recover the database to the most recent point in time, if needed. Import the artifacts using the Oracle Hyperion Calculation Manager import tool.

Depending upon the extent of failure, recovery should be performed at the desired granularity. For the steps to recover Oracle Hyperion Calculation Manager, see Section 18.2.7.14. For the steps specific to recovering from loss of host, see Section 18.3.4.13.

16.5.9.3 Backup and Recovery Recommendations for Oracle Hyperion Financial Reporting

This section describes the Oracle Hyperion Financial Reporting data that must be backed up and restored.

Configuration Files

ORACLE_HOME/Oracle_BI1/common (HIT common files)
ORACLE_HOME/Oracle_BI1/products/biplus
ORACLE_INSTANCE/products/biplus

Dependencies on Other Oracle Fusion Middleware Components

Oracle JRF, Oracle Enterprise Scheduler (ESS), and Oracle Platform Security Services

Dependencies on Third-Party Products

None

Database Repository Dependencies

The Hyperion Registry, which is stored in the database, EPM schemas

Backup Recommendations

Back up Oracle home and Oracle instance.

Back up the database containing the EPM schemas.

Note that the Oracle BI EE repository and the Annotations database (jndi:raframework_database) must be kept synchronized. Back up both from the same point in time.

Recovery Recommendations

Recover Oracle home and Oracle instance.

Recover the database to the most recent point in time, if needed.

Note that the Oracle BI EE repository and the Annotations database (jndi:raframework_database) must be kept synchronized. Recover both from the same point in time.

Depending upon the extent of failure, recovery should be performed at the desired granularity. For the steps to recover Oracle Hyperion Financial Reporting, see Section 18.2.7.15. For the steps specific to recovering from loss of host, see Section 18.3.4.14.

16.5.9.4 Backup and Recovery Recommendations for Oracle Hyperion Smart View

This section describes the Oracle Hyperion Smart View data that must be backed up and restored.

Configuration Files

Configuration files are located in the Oracle Hyperion Smart View install location selected by user. There is no dependency on Oracle Home or components installed on this location.

Dependencies on Oracle Fusion Middleware Components

None

Dependencies on Third-Party Products

None

Database Repository Dependencies

None

Backup Recommendations

Back up the Oracle Hyperion Smart View data, which is stored in the following types of files. Note that the location of the files is determined when you install Oracle Hyperion Smart View.

- Microsoft Excel files (XLS and XLSX)
- Microsoft Word files (DOC and DOCX)
- Microsoft Powerpoint files (PPT and PPTX)

Recovery Recommendations

Recover the Oracle Hyperion Smart View data.

Depending upon the extent of failure, recovery should be performed at the desired granularity. For the steps to recover Oracle Hyperion Smart View, see Section 18.2.7.16.

16.5.10 Backup and Recovery Recommendations for Oracle Data Integrator

This section describes the Oracle Data Integrator data that must be backed up and restored.

Configuration Files

ODI_Oracle_Home/oracledi/agent/web.xml

Database Repository Dependencies

ODI_REPO schema

Backup Recommendations

Back up the domain, the Oracle home, and the *ODI_Oracle_Home*/oracledi/agent folder for each machine where a standalone agent is installed.

Back up the database containing Oracle Data Integrator schema.

Recovery Recommendations

Depending on the extent of the failure, restore the domain or the Oracle home, or both.

Recover the database to the most recent point in time, if needed.

For the steps to recover Oracle Data Integrator, see Section 18.2.7.17. For the steps specific to recovering from loss of host, see Section 18.3.4.5.

16.5.11 Backup and Recovery Recommendations for Oracle WebCenter Content

The following sections describe backup and recovery recommendations for Oracle WebCenter Content:

- Backup and Recovery Recommendations for Oracle Information Rights Management
- Backup and Recovery Recommendations for Oracle WebCenter Content: Imaging
- Backup and Recovery Recommendations for Oracle WebCenter Content
- Backup and Recovery Recommendations for Oracle WebCenter Content: Records

16.5.11.1 Backup and Recovery Recommendations for Oracle Information Rights Management

This section describes the Oracle Information Rights Management data that must be backed up and restored.

Configuration Files

Configuration files are located within the domain home.

Database Repository Dependencies

ORAIRM schema (IRM for DB2 and SQL Server databases. You must ensure that the name is IRM.)

Backup Recommendations

Back up the domain, the Oracle home, and the database containing the ORAIRM schema. Also, back up the LDAP directory and the keystore. The keystore is usually named irm.jks or irm.jceks.

Note that the database and the keystore must be kept synchronized. Back up both from the same point in time.

Recovery Recommendations

Restore the domain, the Oracle home, and the shared file system, depending on the severity of the failure.

Recover the database containing the ORAIRM schema to the most recent point in time, if needed.

Note that the database and the keystore must be kept synchronized. If you restore one, restore the other to the same point in time.

For the steps to recover Oracle Information Rights Management, see Section 18.2.7.18. You use the same procedure to recover from loss of host.

16.5.11.2 Backup and Recovery Recommendations for Oracle WebCenter Content: Imaging

This section describes the Oracle WebCenter Content: Imaging data that must be backed up and restored.

Configuration Files

Configuration files are located within the domain home.

Database Repository Dependencies

IPM and OCS schemas

Backup Recommendations

Back up the domain, the Oracle home, and the database containing the schemas.

Recovery Recommendations

Restore the domain and the Oracle home, depending on the severity of the failure.

Recover the database containing the schemas to the most recent point in time, if needed.

For the steps to recover Oracle WebCenter Content: Imaging, see Section 18.2.7.19. You use the same procedure to recover from loss of host.

16.5.11.3 Backup and Recovery Recommendations for Oracle WebCenter Content

This section describes the Oracle WebCenter Content data that must be backed up and restored.

Configuration Files

DOMAIN_HOME/ucm/CONTEXT-ROOT/bin/intradoc.cfg
DOMAIN_HOME/ucm/CONTEXT-ROOT/config/config.cfg

Database Repository Dependencies

OCS schema

Backup Recommendations

Back up the domain, the Oracle home, and database containing the OCS schema. If the Vault and WebLayout directories are not located in the domain directory, back up their directories, which are specified in:

DOMAIN_HOME/ucm/CONTEXT-ROOT/config/config.cfg

Also, back up the following directory, which is located in a shared file system:

DOMAIN_HOME/ucm/CONTEXT-ROOT/config

Recovery Recommendations

Restore the domain and the shared file system containing the Vault and WebLayout directories, depending on the severity of the failure.

Recover the database containing the OCS schema to the most recent point in time, if needed.

For the steps to recover Oracle WebCenter Content, see Section 18.2.7.20. For the steps specific to recovering from loss of host, see Section 18.3.4.16.1.

16.5.11.4 Backup and Recovery Recommendations for Oracle WebCenter Content: Records

Because Oracle WebCenter Content: Records depends on Oracle WebCenter Content and has no additional backup and recovery artifacts, see the backup and recovery recommendations for Oracle WebCenter Content in Section 16.5.11.3.

16.6 Assumptions and Restrictions

The following assumptions and restrictions apply to the backup and recovery procedures in this book. Also see the restrictions listed in Section 17.2.

- Only the user who installs the product or a user who has access privileges to the directories where Oracle Fusion Middleware has been installed should be able to execute backup and recovery operations.
- If a single Managed Server and Administration Server run on different hosts and the Managed Server is not in a cluster, you must use the pack and unpack commands on the Managed Server to retrieve the correct configuration.

> **See Also:** If you are using Cold Failover Cluster or Disaster Recovery, refer to the *Oracle Fusion Middleware High Availability Guide* for additional information.

17

Backing Up Your Environment

This chapter describes recommended backup strategies for Oracle Fusion Middleware and the procedures for backing up Oracle Fusion Middleware.

This chapter includes the following topics:

- Overview of Backing Up Your Environment
- Limitations and Restrictions for Backing Up Data
- Performing a Backup
- Creating a Record of Your Oracle Fusion Middleware Configuration

17.1 Overview of Backing Up Your Environment

As described in Section 16.3.3, you should use the following recommended strategy for backing up your Oracle Fusion Middleware environment:

- If you are performing an online backup, do not make any configuration changes until the backup is completed. To ensure that no changes are made in the WebLogic Server domain, lock the WebLogic Server configuration, as described in Section 3.4.2.
- Perform a full offline backup immediately after you install Oracle Fusion Middleware. See Section 17.3.1 for information on performing a full backup.
- Perform backups of run-time artifacts after every administrative change and on a regular basis. Oracle recommends that you back up run-time artifacts nightly. See Section 17.3.2 for information on performing a backup of run-time artifacts.
- Perform a new full backup after a major change, such as any upgrade or patch, or if any of the following files are modified:

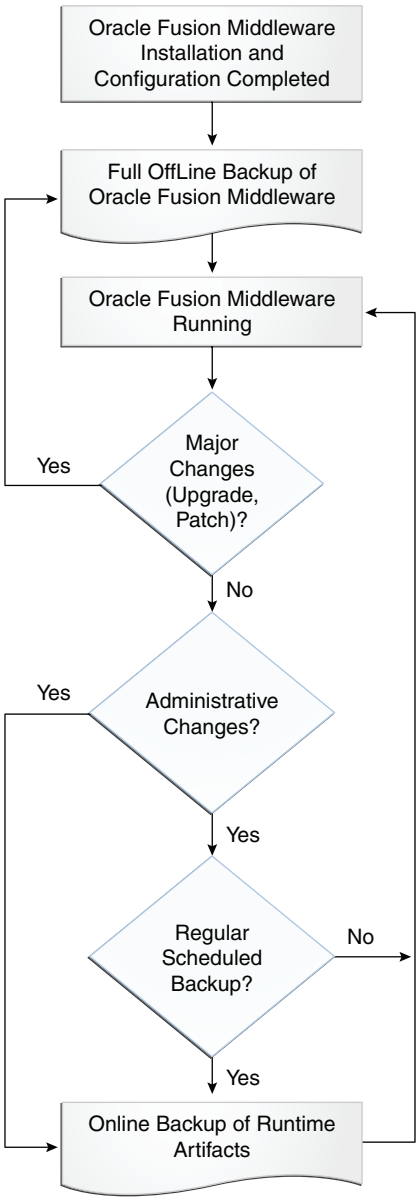
  MW_HOME/wlserver_*n*/common/nodemanager.properties
  MW_HOME/wlserver_*n*/common/bin/wlsifconfig.sh
  MW_HOME/wlserver_*n*/common/bin/setPatchEnv.sh
  MW_HOME/wlserver_*n*/common/bin/commEnv.sh

  See Section 17.3.1 for information on performing a full backup.
- Create a record of your Oracle Fusion Middleware environment. See Section 17.4.
- When you create the backup, name the archive file with a unique name. Consider appending the date and time to the name. For example, if you create a backup of the Middleware home on April 20, 2012, name the backup:

  mw_home_backup_042012.tar

The flowchart in Figure 17–1 provides an overview of how to decide which type of backup is appropriate for a given circumstance.

Figure 17–1 Decision Flow Chart for Type of Backup



17.2 Limitations and Restrictions for Backing Up Data

Note the following points:

- LDAP backups: If you use the built-in LDAP, do not update the configuration of a security provider while a backup of LDAP data is in progress. If a change is made (for example, if an administrator adds a user), while you are backing up the ldap directory tree, the backups in the ldapfiles subdirectory could become inconsistent. Refer to *WebLogic Server Managing Server Startup and Shutdown* for detailed LDAP backup procedures.

- Java Transaction API (JTA): Oracle does not recommend that you back up and restore JTA transaction logs.
- Audit Framework: If you have configured Oracle Fusion Middleware Audit Framework to write data to a database, you should not back up the local files in the bus stop. (Auditable events from each component are stored in a repository known as a bus stop; each Oracle WebLogic Server has its own bus stop. Data can be persisted in this file, or uploaded to a central repository at which point the records are available for viewing and reporting.)

  If you back up the local files, duplicate records are uploaded to the database. That is, they are uploaded to the database when the bus stop is created and then are uploaded again when you restore the files.

  The default locations for bus stop local files are:

  – For Java components:

    DOMAIN_HOME/servers/*server_name*/logs/auditlogs/*component_type*

  – For system components, such as Oracle HTTP Server or Oracle Internet Directory:

    ORACLE_INSTANCE/auditlogs/*component_type*/*component_name*

  For more information about Oracle Fusion Middleware Audit Framework and the bus stop, see "Configuring and Managing Auditing" in the *Oracle Fusion Middleware Application Security Guide*.

17.3 Performing a Backup

You can perform a full offline backup or an online or offline backup of run-time artifacts, as described in the following topics:

- Performing a Full Offline Backup
- Performing an Online Backup of Run-Time Artifacts
- Backing Up Windows Registry Entries

17.3.1 Performing a Full Offline Backup

To perform a full offline backup, you copy the directories that contain Oracle Fusion Middleware files.

Archive and compress the source Middleware home, using your preferred tool for archiving, as described in Section 16.3.

Take the following steps:

1. Shut down all processes in the Middleware home. For example, shut down the Managed Servers, the Administration Server, and any Oracle instances running in the Middleware home.
2. Back up the Middleware home (MW_HOME) on all hosts. For example:

   ```
   (UNIX) tar -cf mw_home_backup_042012.tar MW_HOME/*
   (Windows) jar cf mw_home_backup_042012.Jar MW_HOME\*
   ```

3. If the domain is not located within the Middleware home, back up the Administration Server domain separately. This backs up Java components such as Oracle SOA Suite and Oracle WebCenter Portal.

For example:

```
(UNIX) tar -cf domain_home_backup_042012.tar DOMAIN_HOME/*
(Windows) jar cf domain_home_backup_042012.tar DOMAIN_HOME\*
```

In most cases, you do not need to back up the Managed Server directories separately, because the Administration Server domain contains information about the Managed Servers in its domain. If you have customized your environment for the Managed Server, back up the Managed Server directories. See Section 16.5 for information about what you need to back up.

4. If the Oracle instance home is not located within the Middleware home, back up the Oracle instance home. The Oracle instance home contains configuration information about system components, such as Oracle HTTP Server or Oracle Internet Directory. (See Section 3.5.2 for a list of system components.)

   For example:

   ```
   (UNIX) tar -cf sc_home_backup_042012.tar ORACLE_INSTANCE/*
   (Windows) jar cf sc_home_backup_042012.Jar ORACLE_INSTANCE\*
   ```

5. If a Managed Server is not located within the domain, back up the Managed Server directory. For example:

   ```
   (UNIX) tar -cf mg1_home_backup_042012.tar server_name/*
   (Windows) jar cf mg1_home_backup_042012.jar server_name\*
   ```

6. Back up the OraInventory directory. For example:

   ```
   tar -cf Inven_home_backup_042012.tar /scratch/oracle/OraInventory
   ```

7. On Linux and UNIX, back up the oraInst.loc file, which is located in the following directory:

   ```
   (Linux and IBM AIX) /etc
   (Other UNIX systems) /var/opt/oracle
   ```

8. On Linux and UNIX, backup the oratab file, which is located in the following directory:

   ```
   /etc
   ```

9. Back up the database repositories using the Oracle Recovery Manager (RMAN). For detailed steps, see the *Oracle Database Backup and Recovery User's Guide*.
10. On Windows, you should also export the Windows Registry entries, as described in Section 17.3.3.
11. Unlock the Oracle WebLogic Server configuration by clicking **Release Configuration** on the WebLogic Server Administration Console,
12. Create a record of your Oracle Fusion Middleware environment. See Section 17.4.

17.3.2 Performing an Online Backup of Run-Time Artifacts

You should perform a backup of run-time artifacts (which are listed in Section 16.3.2) on a regular basis and at the times described in Section 16.3.3.

To back up run-time artifacts:

1. To avoid an inconsistent backup, do not make any configuration changes until the backup is completed. To ensure that no changes are made in the WebLogic Server domain, lock the WebLogic Server configuration, as described in Section 3.4.2.

2. Back up the Administration Server domain directories. This backs up Java components such as Oracle SOA Suite and Oracle WebCenter Portal. For example:

   ```
   UNIX) tar -cf domain_home_backup_042012.tar MW_HOME/user_
   projects/domains/domain_name/*
   (Windows) jar cf domain_home_backup_042012.jar MW_HOME\user_
   projects\domains\domain_name\*
   ```

   For Oracle Portal, Oracle Reports, Oracle Forms Services, and Oracle Business Intelligence Discoverer, you must back up the Managed Server directories, in addition to the Administration Server domain directories.

   ```
   (UNIX) tar -cf sc_home_backup_042012.tar ORACLE_INSTANCE/*
   (Windows) jar cf sc_home_backup_042012.jar ORACLE_INSTANCE\*
   ```

3. Back up the database repositories using the Oracle Recovery Manager (RMAN). For detailed steps, see the *Oracle Database Backup and Recovery User's Guide*.
4. Unlock the Oracle WebLogic Server configuration by clicking **Release Configuration** on the WebLogic Server Administration Console,
5. Create a record of your Oracle Fusion Middleware environment. See Section 17.4.

17.3.3 Backing Up Windows Registry Entries

On Windows, you must back up Windows Registry keys related to Oracle Fusion Middleware. Which keys you back up depends on what components you have installed.

To export a key, use the following command:

```
regedit /E  FileName Key
```

Export the following entries:

- For any component, export the following registry key:

  ```
  HKEY_LOCAL_MACHINE\Software\Oracle
  ```

- For system components, such as Oracle Web Cache, and for Oracle BI Enterprise Edition, export each node that begins **Oracle** within the following registry keys:

  ```
  HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services
  HKEY_LOCAL_MACHINE\SYSTEM\ControlSet001\Services
  HKEY_LOCAL_MACHINE\SYSTEM\ControlSet002\Services
  ```

  For example:

  ```
  regedit /E C:\oracleSMP.reg HKEY_LOCAL_
  MACHINE\SYSTEM\ControlSet001\Services\Oracleagent10gAgentSNMPPeerEncapsulator
  ```

  Use a unique file name for the each key.

- For Oracle BI EE, export the following registry key:

  ```
  HKEY_LOCAL_MACHINE\SOFTWARE\ODBC
  ```

  For example:

  ```
  regedit /E C:\oracleregistry.reg HKEY_LOCAL_MACHINE\SOFTWARE\ODBC
  ```

You can also use the Registry Editor to export the key. See the Registry Editor Help for more information.

17.4 Creating a Record of Your Oracle Fusion Middleware Configuration

In the event that you need to restore and recover your Oracle Fusion Middleware environment, it is important to have all the necessary information at your disposal. This is especially true in the event of a hardware loss that requires you to reconstruct all or part of your Oracle Fusion Middleware environment on a new disk or host.

You should maintain an up-to-date record of your Oracle Fusion Middleware environment that includes the information listed in this section. You should keep this information both in hardcopy and electronic form. The electronic form should be stored on a host or e-mail system that is completely separate from your Oracle Fusion Middleware environment.

Your Oracle Fusion Middleware hardware and software configuration record should include:

- The following information for each host in your environment:
  - Host name
  - Virtual host name (if any)
  - Domain name
  - IP address
  - Hardware platform
  - Operating system release level and patch information
- The following information for each Oracle Fusion Middleware installation in your environment:
  - Installation type (for example, Oracle SOA Suite)
  - Host on which the installation resides
  - User name, userid number, group name, groupid number, environment profile, and type of shell for the operating system user that owns the Oracle home (`/etc/passwd` and `/etc/group` entries)
  - Directory structure, mount points, and full path for the Middleware home, Oracle Common home, Oracle homes, Oracle WebLogic Server domain home (if it does not reside in the user_projects directory in the Middleware home), and the Oracle instance home
  - Amount of disk space used by the installation
  - Port numbers used by the installation
- The following information for the database containing the metadata for components:
  - Host name
  - Database version and patch level
  - Base language
  - Character set
  - Global database name
  - SID
  - Listen port

18

Recovering Your Environment

This chapter describes recommended recovery strategies and procedures for recovering Oracle Fusion Middleware from different types of failures and outages, such as media failures or loss of host.

This chapter includes the following topics:

- Overview of Recovering Your Environment
- Recovering After Data Loss, Corruption, Media Failure, or Application Malfunction
- Recovering After Loss of Host

18.1 Overview of Recovering Your Environment

This section provides an overview of recovery strategies for outages that involve actual data loss or corruption, host failure, or media failure where the host or disk cannot be restarted and they are permanently lost. This type of failure requires some type of data restoration before the Oracle Fusion Middleware environment can be restarted and continue with normal processing.

> **Note:** The procedures in this chapter assume that no administrative changes were made since the last backup. If administrative changes were made since the last backup, they must be reapplied after recovery is complete.

When you restore the files, use your preferred tool to extract the compressed files, as described in Section 16.4.

Ensure that the tool you are using preserves the permissions and timestamps of the files.

Rename existing files and directories before you begin restoring the files from backup so that you do not unintentionally override necessary files.

18.2 Recovering After Data Loss, Corruption, Media Failure, or Application Malfunction

This section describes recovery strategies for outages that involve actual data loss or corruption, or media failure where the disk cannot be restored. It also describes recovery strategies for applications that are no longer functioning properly. This type of failure requires some type of data restoration before the Oracle Fusion Middleware

environment can be restarted and continue with normal processing. It contains the following topics:

- Recovering a Middleware Home
- Recovering an Oracle WebLogic Server Domain
- Recovering an Oracle Home
- Recovering an Oracle Instance Home
- Recovering the Administration Server Configuration
- Recovering a Managed Server
- Recovering Components
- Recovering a Cluster
- Recovering Applications
- Recovering a Database

Note: You can only restore an entity to the same path as the original entity. That path can be on the same host or a different host.

18.2.1 Recovering a Middleware Home

You can recover a Middleware home that was corrupted or from which files were deleted.

To recover the Middleware home:

1. Stop all relevant processes. That is, stop all processes that are related to the domain, such as the Administration Server, Node Manager, and Managed Servers. For example, to stop the Administration Server on Linux:

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

2. Recover the Middleware home directory from backup. For example:

   ```
   cd MW_HOME
   (UNIX) tar -xf mw_home_backup_042012.tar
   (Windows) jar xtf mw_home_backup_042012.jar
   ```

3. Start all relevant processes. That is, start all processes that run in the Middleware home. For example, start the Administration Server:

   ```
   DOMAIN_HOME/bin/startWebLogic.sh -Dweblogic.management.username=username
    -Dweblogic.management.password=password
    -Dweblogic.system.StoreBootIdentity=true
   ```

18.2.2 Recovering an Oracle WebLogic Server Domain

You can recover an Oracle WebLogic Server domain that was corrupted or deleted from the file system.

Caution: Performing a domain-level recovery can impact other aspects of a running system and all of the configuration changes performed after the backup was taken will be lost.

To recover an Oracle WebLogic Server domain that was corrupted or deleted from the file system:

1. Stop all relevant processes. That is, stop all processes that are related to the domain, such as the Administration Server and Managed Servers. For example, stop the Administration Server:

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

2. Recover the domain directory from backup:

   ```
   cd DOMAIN_HOME
   (UNIX) tar -xf domain_backup_042012.tar
   (Windows) jar xtf domain_backup_042012.jar
   ```

3. Start all relevant processes. That is, start all processes that are related to the domain. For example, start the Administration Server:

   ```
   DOMAIN_HOME/bin/startWebLogic.sh -Dweblogic.management.username=username
    -Dweblogic.management.password=password
    -Dweblogic.system.StoreBootIdentity=true
   ```

4. If you cannot start the Administration Server, recover it, as described in Section 18.2.5.
5. If you cannot start a Managed Server, recover it, as described in Section 18.2.6.

18.2.3 Recovering an Oracle Home

To recover your Oracle home for a particular component:

1. Recover the Oracle home to the original directory from a backup file. For example:

   ```
   cd ORACLE_HOME
   tar -xf Oracle_home_backup_042012.tar
   ```

2. Restart the Managed Server to which applications are deployed, using the WLST `start` command. For example:

   ```
   wls:/mydomain/serverConfig> start('myserver','Server')
   ```

18.2.4 Recovering an Oracle Instance Home

An Oracle instance home contains configuration information for system components, such as Oracle HTTP Server or Oracle Internet Directory. (See Section 3.5.2 for a list of system components.) The following topics describe how to recover an Oracle instance home:

- Recovering After Oracle Instance Home Deleted from File System
- Recovering After Oracle Instance Home Deregistered

18.2.4.1 Recovering After Oracle Instance Home Deleted from File System

To recover an Oracle instance home that was corrupted or deleted from the file system:

1. Stop all relevant processes. That is, kill all processes that are related to that Oracle instance.
2. Recover the Oracle instance home directory from a backup file. For example:

   ```
   cd ORACLE_INSTANCE
   (UNIX) tar -xf Instance_home_backup_042012.tar
   (Windows) jar xtf Instance_home_backup_042012.jar
   ```

3. Start all relevant processes. That is, start all processes that are related to that Oracle instance:

```
opmnctl startall
```

18.2.4.2 Recovering After Oracle Instance Home Deregistered

To recover an Oracle instance home that was deregistered from the domain:

1. Recover the Oracle instance home directory from a backup file. For example, on Linux:

```
cd ORACLE_INSTANCE
tar -xf Instance_home_backup_042012.tar
```

2. Register the Oracle instance, along with all of its components, with the Administration Server, using the opmnctl `registerinstance` command. For example:

```
opmnctl registerinstance -adminHost admin_server_host
     -adminPort admin_server_port -adminUsername username
     -adminPassword password
     -oracleInstance ORACLE_INSTANCE_dir -oracleHome ORACLE_HOME_dir
     -instanceName Instance_name -wlserverHome Middleware_Home
```

18.2.5 Recovering the Administration Server Configuration

If the Administration Server configuration has been lost because of file deletion or file system corruption, the Administration Server console continues to function if it was already started when the problem occurred. The Administration Server directory is regenerated automatically, except for security information. As a result, whenever you start the Administration Server, it prompts for a user name and password. To prevent this, you can recover the configuration.

> **Caution:** Performing a domain-level recovery can impact other aspects of a running system and all of the configuration changes performed after the backup was taken will be lost.

To recover the Administration Server configuration:

1. Stop all processes, including the Administration Server, Managed Servers, and Node Manager, if they are started. For example, to stop the Administration Server:

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

2. Recover the Administration Server configuration by recovering the domain home backup to a temporary location. Then, restore the config directory to the following location:

```
DOMAIN_HOME/config
```

3. Start the Administration Server. For example:

```
DOMAIN_HOME/bin/startWebLogic.sh -Dweblogic.management.username=username
   -Dweblogic.management.password=password
   -Dweblogic.system.StoreBootIdentity=true
```

4. Verify that the Administration Server starts properly and is accessible.

On the next configuration change, the configuration from the Administration Server is pushed to the Managed Servers. On each Managed Server restart, the configuration is retrieved from the Administration Server.

18.2.6 Recovering a Managed Server

You can recover a Managed Server's files, including its configuration files if they are deleted or corrupted.

The following topics describe how to recover a Managed Server's files:

- Recovering a Managed Server When It Cannot Be Started
- Recovering a Managed Server When It Does Not Function Correctly
- Recovering an Oracle SOA Suite Managed Server That Has a Separate Directory

  This section pertains when Oracle SOA Suite is configured in a domain and no Managed Servers share the domain directory with the Administration Server.

18.2.6.1 Recovering a Managed Server When It Cannot Be Started

In this scenario, the Managed Server does not operate properly or cannot be started because the configuration has been deleted or corrupted or the configuration was mistakenly changed and you cannot ascertain what was changed.

To recover a Managed Server when it cannot be started:

1. If the Administration Server is not reachable, recover the Administration Server, as described in Section 18.2.5.
2. If the Managed Server fails to start, recover the Middleware home from the backup, if required. For example:

   ```
   tar -xf mw_home_backup_042012.tar
   ```

3. If the file system is lost, take the following steps:

   a. Create a domain template jar file for the Administration Server, using the pack utility. For example:

      ```
      pack.sh -domain=MW_HOME/user_projects/domains/domain_name
         -template=/scratch/temp.jar -template_name=test_install
         -template_author=myname -log=/scratch/logs/my.log -managed=true
      ```

      Specifying the -managed=true option packs up only the Managed Servers. If you want to pack the entire domain, omit this option.

   b. Unpack the domain template jar file, using the unpack utility:

      ```
      unpack.sh -template=/scratch/temp.jar
         -domain=MW_HOME/user_projects/domains/domain_name
         -log=/scratch/logs/new.log -log_priority=info
      ```

   c. Ensure that the application artifacts are accessible from the Managed Server host. That is, if the application artifacts are not on the same server as the Managed Server, they must be in a location accessible by the Managed Server.

Note:

- For stage mode applications, the Administration server copies the application bits to the staged directories on the Managed Server hosts.
- For nostage and external_stage mode applications, ensure that application files are available in the stage directories of the Managed Server.

See *Oracle Fusion Middleware Deploying Applications to Oracle WebLogic Server* for information about stage, nostage and external_stage modes.

4. Start the Managed Server. For example:

```
DOMAIN_HOME/bin/startManagedWebLogic.sh managed_server_name
         admin_url
```

The Managed Server connects to the Administration Server and updates its configuration changes.

18.2.6.2 Recovering a Managed Server When It Does Not Function Correctly

In this scenario, the Managed Server is running, but the file system for the Managed Server has been lost or corrupted.

To recover the Managed Server:

1. Stop the Managed Server. For example:

```
DOMAIN_HOME/bin/stopManagedWeblogic.sh managed_server_name admin_url
        username password
```

2. Recover the Middleware home from the backup, if required:

```
tar -xf mw_home_backup_042012.tar
```

3. Create a domain template jar file for the Administration Server, using the pack utility. For example:

```
pack.sh -domain=MW_HOME/user_projects/domains/WLS_SOAWC
  -template=/scratch/temp.jar -template_name=test_install
  -template_author=myname -log=/scratch/logs/my.log -managed=true
```

Specifying the -managed=true option packs up only the Managed Servers. If you want to pack the entire domain, omit this option.

4. Unpack the domain template jar file, using the unpack utility:

```
unpack.sh -template=/scratch/temp.jar
   -domain=MW_HOME/user_projects/domains/WLS_SOAWC
   -log=/scratch/logs/new.log -log_priority=info
```

5. Ensure that the application artifacts are accessible from the Managed Server host. That is, if the application artifacts are not on the same server as the Managed Server, they must be in a location accessible by the Managed Server.

Note:

- For stage mode applications, the Administration server copies the application bits to the staged directories on the Managed Server hosts.
- For nostage and external_stage mode applications, ensure that application files are available in the stage directories of the Managed Server.

See *Oracle Fusion Middleware Deploying Applications to Oracle WebLogic Server* for information about deploying applications.

6. Restart the Managed Server. For example:

```
DOMAIN_HOME/bin/startManagedWebLogic.sh managed_server_name admin_url
```

18.2.6.3 Recovering an Oracle SOA Suite Managed Server That Has a Separate Directory

When Oracle SOA Suite is configured in a domain and no Managed Servers share the domain directory with the Administration Server, you must restore the Managed Server directory. For example, a domain contains two Managed Servers, one of which contains Oracle SOA Suite, but neither of the Managed Server's directories are in the same directory structure as the Administration Server.

In this case, you must restore the Managed Server from backup:

1. Restore the Managed Server from backup:

```
cd ManagedServer_Home
tar -xf managed_server_backup_042012.tar
```

2. Restart the Managed Server:

```
DOMAIN_HOME/bin/startManagedWebLogic.sh managed_server_name
          admin_url
```

18.2.7 Recovering Components

For most components, the following topics describe how to recover a component:

- Recovering a Component That Is Not Functioning Properly
- Recovering a Component After Cluster Configuration Change

For some components, you must take different steps. Table 18–1 lists those components and the section that describes the procedures to recover them.

Table 18–1 Recovery Procedures for Particular Components

Component	Procedure
Oracle Access Management Access Manager	Section 18.2.7.5
Oracle Adaptive Access Manager	Section 18.2.7.6
Oracle BI Enterprise Edition	Section 18.2.7.10
Oracle Business Intelligence Publisher	Section 18.2.7.11
Oracle Business Process Management	Section 18.2.7.7
Oracle Data Integrator	Section 18.2.7.17

Table 18–1 (Cont.) Recovery Procedures for Particular Components

Component	Procedure
Oracle Essbase	Section 18.2.7.13
Oracle Hyperion Calculation Manager	Section 18.2.7.14
Oracle Hyperion Financial Reporting	Section 18.2.7.15
Oracle Hyperion Smart View	Section 18.2.7.16
Oracle Identity Manager	Section 18.2.7.3
Oracle Identity Navigator	Section 18.2.7.4
Oracle Information Rights Management	Section 18.2.7.18
Oracle Real-Time Decisions	Section 18.2.7.12
Oracle WebCenter Content	Section 18.2.7.20
Oracle WebCenter Content: Imaging	Section 18.2.7.19
Oracle WebCenter Content: Records	Section 18.2.7.21
Oracle WebCenter Portal's Activity Graph	Section 18.2.7.8
Oracle WebCenter Portal's Analytics	Section 18.2.7.9

18.2.7.1 Recovering a Component That Is Not Functioning Properly

You can recover a component if the component's files have been deleted or corrupted or if the component cannot be started or is not functioning properly because the component's configuration was changed and committed. You may not be able to ascertain what change is causing the problem and you want to revert to an earlier version.

- For Java components, such as Oracle SOA Suite, you recover the Managed Server, as described in Section 18.2.6.
- For system components, such as Oracle HTTP Server or Oracle Web Cache:

  1. Stop the component. For example, to stop Oracle HTTP Server:

     ```
     opmnctl stopproc ias-component=component_name
     ```

     For information on stopping components, see Section 4.3.

  2. Recover the component-specific files from backup. Section 16.5 lists the directories and files needed for each component. For example, to recover Oracle HTTP Server files, you recover the following directories:

     ```
     ORACLE_INSTANCE/config/OHS/component_name
     ORACLE_INSTANCE/diagnostics/logs/OHS/component_name
     ```

  3. Start the component. For example, to start Oracle HTTP Server:

     ```
     opmnctl startproc ias-component=component_name
     ```

     For information on starting components, see Section 4.3.

18.2.7.2 Recovering a Component After Cluster Configuration Change

You can recover components in a cluster when the components cannot be started or are not functioning properly because the configuration was changed and committed at the cluster level. You may not be able to ascertain what change is causing the problem and you want to revert to an earlier version.

Caution: Performing a domain-level recovery can impact other aspects of a running system and all of the configuration changes performed after the backup was taken will be lost.

To recover the components:

1. Stop the cluster:

   ```
   stop('cluster_name', 'Cluster')
   ```

2. Stop all processes, such as the Managed Servers and the Administration Server. For example, to stop the Administration Server on Linux:

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

3. Recover the Administration Server configuration by recovering the domain home backup to a temporary location. Then, restore the config directory to the following location:

   ```
   DOMAIN_HOME/config
   ```

4. Start the Administration Server. For example:

   ```
   DOMAIN_HOME/bin/startWebLogic.sh -Dweblogic.management.username=username
    -Dweblogic.management.password=password
    -Dweblogic.system.StoreBootIdentity=true
   ```

5. Start the cluster. You can use the Oracle WebLogic Server Administration Console or WLST. For example, to use the WLST start command:

   ```
   start('clusterName', 'Cluster')
   ```

The latest configuration is retrieved from the Administration Server to every member of the cluster.

18.2.7.3 Recovering Oracle Identity Manager

To recover Oracle Identity Manager:

1. Restore the domain, as described in Section 18.2.2.
2. Restore the Oracle home, as described in Section 18.2.3.
3. Restore the database containing the OIM, MDS, SOAINFRA, and the OID schemas to the same point in time. See Section 18.2.10.

   Oracle Identity Manager stores users and roles in the LDAP store. If you restore the database to a different point in time than the LDAP store, the reconciliation engine checks the change logs and reapplies all the changes that happened in the time period between the restore of the LDAP store and the database. For example, if the database is restored so that is 10 hours behind the LDAP store, the reconciliation engine checks the change logs and reapplies all the changes that happened in the last 10 hours in the LDAP store to the database.

   You do not need to explicitly trigger the reconciliation. LDAP synchronization is set up as a periodic scheduled task to submit reconciliation events periodically. You can also start the reconciliation process manually and monitor the reconciliation events from the Oracle Identity Manager console. See "Managing Reconciliation" in *Oracle Fusion Middleware Administrator's Guide for Oracle Identity Manager*.

Note: Oracle recommends that you ensure that the Oracle Identity Manager application is unavailable to the end users when a bulk reconciliation is occurring (as in the above recovery scenario). When the bulk reconciliation is complete, ensure that the Oracle Identity Manager application is again available to the end users. You can monitor the reconciliation with the Oracle Identity Manager console.

18.2.7.4 Recovering Oracle Identity Navigator

To recover Oracle Identity Navigator:

1. Restore the domain, as described in Section 18.2.2.
2. Restore the Oracle home, as described in Section 18.2.3.
3. Restore the file-based MDS repository, using the WLST importMetadata command. For example:

```
importMetadata(application='oinav', server='server_name', fromLocation='export_
directory')
```

18.2.7.5 Recovering Oracle Access Management Access Manager

To recover Access Manager:

1. Restore the Middleware home and the domain home for the Access Manager Managed Server, as described in Section 18.2.1.
2. Restore the domain, as described in Section 18.2.2.
3. Restore the Oracle home for the Oracle HTTP Server that contains the WebGate, if necessary, as described in Section 18.2.3.
4. Restore the Oracle instance for the Oracle HTTP Server that contains the WebGate, if necessary, as described in Section 18.2.4.
5. Restore the database containing the schema used by OES for the Access Manager policy store, if necessary. See Section 18.2.10.

18.2.7.6 Recovering Oracle Adaptive Access Manager

To recover Oracle Adaptive Access Manager:

1. Restore the domain, as described in Section 18.2.2.
2. Restore the Oracle home, as described in Section 18.2.3.
3. Restore the database containing the OAAM schemas, if necessary. See Section 18.2.10.

18.2.7.7 Recovering Oracle Business Process Management

To recover Oracle Business Process Management:

1. Restore the Managed Server, as described in Section 18.2.6.
2. Restore the Oracle homes, as described in Section 18.2.3.

18.2.7.8 Recovering Oracle WebCenter Portal's Activity Graph

To recover Oracle WebCenter Portal's Activity Graph:

1. Restore the domain, as described in Section 18.2.2.

2. Restore the Oracle home, as described in Section 18.2.3.
3. Restore the database containing the ACTIVITIES schema, if necessary.

18.2.7.9 Recovering Oracle WebCenter Portal's Analytics

To recover Oracle WebCenter Portal's Analytics:

1. Restore the domain, as described in Section 18.2.2.
2. Restore the Oracle home, as described in Section 18.2.3.
3. Restore the database containing the ACTIVITIES and MDS schemas, if necessary.

18.2.7.10 Recovering Oracle BI Enterprise Edition

The following topics describe how to recover Oracle BI EE:

- Recovering Oracle BI Enterprise Edition in a Non-Clustered Environment
- Recovering Oracle BI Enterprise Edition in a Clustered Environment

> **Note:** When you recover Oracle BI EE, you must ensure that the Oracle BI Presentation Catalog and the Oracle BI EE repository (RPD) are restored to the same point in time, by using the same backup set.

18.2.7.10.1 Recovering Oracle BI Enterprise Edition in a Non-Clustered Environment This scenario assumes that Oracle BI Enterprise Edition is running in a non-clustered environment.

1. Restore the following, depending on the extent of the failure:
   - If the entire Middleware home is lost, restore the Middleware home, as described in Section 18.2.1.
   - If the Oracle instance home is lost, restore the Oracle instance home, as described in Section 18.2.4.
   - If the domain home is lost on the Administration Server node, restore it, as described in Section 18.2.5.
   - If the domain home is lost on the Managed Server node, restore it, as described in Section 18.2.6.
2. Recover the Administration Server, as described in Section 18.2.5.
3. Recover the Managed Server, as described in Section 18.2.6.
4. If necessary, restore the database containing the Oracle BI EE schemas. See Section 18.2.10.
5. Reconcile the LDAP Database with the Oracle BI EE repository (RPD), as described in Section 18.2.7.10.3, and with the Oracle BI Presentation Catalog, as described in Section 18.2.7.10.4.

18.2.7.10.2 Recovering Oracle BI Enterprise Edition in a Clustered Environment This scenario assumes that Oracle BI Enterprise Edition is running in a clustered environment.

To recover Oracle BI EE in a clustered environment:

1. Restore the following, depending on the extent of the failure:
   - If the entire Middleware home is lost, restore the Middleware home, as described in Section 18.2.1.

- If the Oracle instance home is lost, restore the Oracle instance home, as described in Section 18.2.4.
- If the domain home is lost on the Administration Server node, restore it, as described in Section 18.2.5.
- If the domain home is lost on the Managed Server node, restore it, as described in Section 18.2.6.

2. Recover the Administration Server, as described in Section 18.2.5.
3. Recover all Managed Servers, as described in Section 18.2.6.
4. If necessary, restore the database containing the Oracle BI EE schemas, as described in Section 18.2.10.
5. Reconcile the LDAP Database with the Oracle BI EE repository (RPD), as described in Section 18.2.7.10.3, and with the Oracle BI Presentation Catalog, as described in Section 18.2.7.10.4.

18.2.7.10.3 Reconciling the LDAP Database with RPD You must reconcile the LDAP database with the Oracle BI EE repository (RPD).

Oracle BI Enterprise Edition provides a method to perform synchronization. You can enable automatic synchronization, at all times, or temporarily to perform the synchronization. (See "NQSConfig.INI File Configuration Settings" in the *Oracle Fusion Middleware System Administrator's Guide for Oracle Business Intelligence Enterprise Edition* for information about editing the NQSConfig.ini file.)

- To enable synchronization:
  1. Edit the following file:

     INSTANCE_HOME/config/OracleBIServerComponent/coreapplication_obis1/NQSConfig.INI

     Set the flag FMW_UPDATE_ROLE_AND_USER_REF_GUIDS to `yes`.
  2. Restart the servers. The information in the LDAP database and RPD is synchronized.
- To disable synchronization:
  1. To disable synchronization, edit the following file:

     INSTANCE_HOME/config/OracleBIServerComponent/coreapplication_obis1/NQSConfig.INI

     Set the flag FMW_UPDATE_ROLE_AND_USER_REF_GUIDS to `no`.
  2. Restart the servers.

On Windows, the Oracle BI Administration Tool provides a Consistency Check Manager that checks the validity of your repository and allows you to correct the inconsistencies. For more information, see "Checking the Consistency of Repository Objects" in the *Oracle Fusion Middleware Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition*.

18.2.7.10.4 Reconciling the LDAP database with Oracle BI Presentation Catalog If the LDAP database is restored to a previous point in time resulting in the LDAP database being behind in time to the Oracle BI Presentation Catalog, use the following command to reconcile the LDAP database with the Oracle BI Presentation Catalog:

```
runcat.cmd -cmd forgetAccounts
```

For information about the runcat command, see the help:

```
./runcat.sh -cmd maintenanceMode -help
```

18.2.7.11 Recovering Oracle Business Intelligence Publisher

To recover Oracle Business Intelligence Publisher:

1. Recover the Managed Server containing the Oracle Business Intelligence Publisher component, as described in Section 18.2.6.
2. If necessary, restore the database containing the Oracle Business Intelligence Publisher schemas, as described in Section 18.2.10.

If backup artifacts are restored from different times, then user accounts, user reports, and user permissions revert to the restored version. Restore all artifacts from the same point in time.

18.2.7.12 Recovering Oracle Real-Time Decisions

To recover Oracle Real-Time Decisions:

1. Recover the Managed Server containing the Oracle Real-Time Decisions component, as described in Section 18.2.6.

Note that if backup artifacts are restored from a different time, the analytic models miss a period of learning, but their intelligence is unaffected.

18.2.7.13 Recovering Oracle Essbase

To recover Oracle Essbase:

1. Recover the Middleware home, as described in Section 18.2.1
2. If the Oracle instance does not reside in the Middleware home, recover it, as described in Section 18.2.4

18.2.7.14 Recovering Oracle Hyperion Calculation Manager

To recover Oracle Hyperion Calculation Manager:

1. Recover the Managed Server to which Oracle Hyperion Calculation Manager is deployed, as described in Section 18.2.6.
2. If necessary, recover the database to the most recent point in time, or import the artifacts using the Oracle Hyperion Calculation Manager import tool.

18.2.7.15 Recovering Oracle Hyperion Financial Reporting

To recover Oracle Hyperion Financial Reporting:

1. Recover the Oracle Hyperion Financial Reporting Oracle home, as described in Section 18.2.3.
2. Recover the Oracle instance, as described in Section 18.2.4.
3. If necessary, recover the database to the most recent point in time.

18.2.7.16 Recovering Oracle Hyperion Smart View

To recover Oracle Hyperion Smart View:

1. Restore the Oracle Hyperion Smart View data files that you backed up, as described in Section 16.5.9.4, to their original locations.

18.2.7.17 Recovering Oracle Data Integrator

To recover Oracle Data Integrator:

1. If necessary, restore the database, as described in Section 18.3.6.
2. Recover the Oracle Data Integrator Oracle home from backup, as described in Section 18.2.3.

   ```
   cd ORACLE_HOME
   tar -xf oracle_home_backup_042012.tar
   ```

3. Recover the domain directory from backup:

   ```
   cd DOMAIN_HOME
   tar -xf domain_backup_042012.tar
   ```

18.2.7.18 Recovering Oracle Information Rights Management

To recover Oracle Information Rights Management:

1. Restore the domain, as described in Section 18.2.2.
2. Restore the shared file system.
3. If necessary, restore the database, as described in Section 18.2.10.

   Note that the database and the keystore must be kept synchronized. If you restore one, restore the other to the same point in time.
4. Restore the keystore.

18.2.7.19 Recovering Oracle WebCenter Content: Imaging

Oracle WebCenter Content: Imaging stores data in the following locations:

- A database for Imaging configuration data
- A database that functions as a document repository
- JMS persistent queues

When you recover Imaging, you should ensure that all data is restored from the same point-in-time.

To recover Imaging:

1. Restore the domain, as described in Section 18.2.2.
2. Restore the database containing the IPM and OCS schemas, if necessary. See Section 18.2.10.

18.2.7.20 Recovering Oracle WebCenter Content

To recover Oracle WebCenter Content:

1. If necessary, restore the database, as described in Section 18.3.6.
2. Restore the domain, as described in Section 18.2.2.
3. If the Vault, WebLayout, or Search directories are not located in the domain directory, restore those directories, if necessary. For example, if the Vault directory is located on a shared drive in /net/home/vault, restore it from backup:

   ```
   cd /net/home/vault
   tar -xf vault_backup_042012.tar
   ```

Note that you should restore the database and the shared file system at the same time. If you cannot do that, you can use the IDCAnalyse utility to determine if there are any inconsistencies between the database and the shared file system. If there are, you can perform a manual recovery using IDCAnalyse.

18.2.7.21 Recovering Oracle WebCenter Content: Records

Because Oracle WebCenter Content: Records depends on Oracle WebCenter Content and has no additional backup and recovery artifacts, see the recovery procedure for Oracle WebCenter Content in Section 18.2.7.20.

18.2.8 Recovering a Cluster

The following topics describe how to recover a cluster:

- Recovering a Cluster After Deletion or Cluster-Level Configuration Changes
- Recovering a Cluster After Membership Is Mistakenly Modified

18.2.8.1 Recovering a Cluster After Deletion or Cluster-Level Configuration Changes

In this scenario, the cluster has been erroneously deleted or the cluster-level configuration, such as the JMS configuration or container-level data sources, was mistakenly changed and committed. The server cannot be started or does not operate properly or the services running inside the server are not starting. You cannot ascertain what was changed.

> **Caution:** Performing a domain-level recovery can impact other aspects of a running system and all of the configuration changes performed after the backup was taken will be lost.

If the configuration changes are few, then the easiest way is to redo the configuration changes. If that is not feasible, use the following procedure to recover the configuration:

1. Stop the cluster. You can use the Oracle WebLogic Server Administration Console or WLST. For example, to use WLST:

   ```
   stop('clusterName', 'Cluster')
   ```

2. Stop the Administration Server. For example:

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

3. Recover the Administration Server configuration by recovering the domain home backup to a temporary location. Then, restore the config directory to the following location:

   ```
   DOMAIN_HOME/config
   ```

4. Start the Administration Server. For example:

   ```
   DOMAIN_HOME/bin/startWebLogic.sh -Dweblogic.management.username=username
    -Dweblogic.management.password=password
    -Dweblogic.system.StoreBootIdentity=true
   ```

5. Start the cluster. You can use the Oracle WebLogic Server Administration Console or WLST. For example, to use WLST:

```
start('clusterName', 'Cluster')
```

18.2.8.2 Recovering a Cluster After Membership Is Mistakenly Modified

You can recover a cluster when the cluster's membership has been mistakenly modified. For example, if you inadvertently delete a member from the cluster, you can restore the member to the cluster.

> **Caution:** Performing a domain-level recovery can impact other aspects of a running system and all of the configuration changes performed after the backup was taken will be lost.

To recover the cluster membership:

1. Stop all processes, such as the Managed Servers and the Administration Server. For example, to stop the Administration Server on Linux:

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

2. Recover the Administration Server configuration by recovering the domain home backup to a temporary location. Then, restore the config directory to the following location:

   ```
   DOMAIN_HOME/config
   ```

3. Start the Administration Server. For example:

   ```
   DOMAIN_HOME/bin/startWebLogic.sh -Dweblogic.management.username=username
      -Dweblogic.management.password=password
      -Dweblogic.system.StoreBootIdentity=true
   ```

   The deleted member is now back in the cluster.

4. Start all processes, such as the Managed Servers. For example, to start the Managed Server on Linux, use the following script:

   ```
   DOMAIN_HOME/bin/startManagedWebLogic.sh managed_server_name
            admin_url
   ```

5. Start the cluster. You can use the Oracle WebLogic Server Administration Console or WLST. For example, to use WLST:

   ```
   start('clusterName', 'Cluster')
   ```

   The deleted member is now part of the cluster.

6. Start all cluster members if they are not started:

   ```
   DOMAIN_HOME/bin/startManagedWebLogic.sh managed_server_name admin_url
   ```

18.2.9 Recovering Applications

The following topics describe how to recover an application:

- Recovering Application Artifacts
- Recovering a Redeployed Application That Is No Longer Functional
- Recovering an Undeployed Application
- Recovering a Composite Application

Note the following about recovering applications:

- If the application is staged, the Administration server copies the application bits to the staged directories on the Managed Server hosts.
- If the deployment mode is nostage or external_stage, ensure that additional application artifacts are available. For example, applications may reside in directories outside of the domain directory. Make your application files available to the new Administration Server by copying them from backups or by using a shared disk. Your application files should be available in the same relative location on the new file system as on the file system of the original Administration Server.

  See Also: *Oracle Fusion Middleware Deploying Applications to Oracle WebLogic Server* for information about deploying applications

18.2.9.1 Recovering Application Artifacts

If an application's artifacts, such as the .ear file, have been lost or corrupted, you can recover the application.

To recover the application:

1. Start the Managed Server to which the application was deployed. For example:

   ```
   DOMAIN_HOME/bin/startManagedWebLogic.sh managed_server_name admin_url
   ```

   This synchronizes the configuration with the Administration Server.

   On each Managed Server restart, the configuration and application artifacts are retrieved from the Administration Server.

18.2.9.2 Recovering a Redeployed Application That Is No Longer Functional

If a Java EE application was redeployed to a Managed Server (whether or not the Managed Server is part of a cluster) and the application is no longer functional, you can recover it.

To recover the application:

1. Recover the application files from backup, if needed.
2. Redeploy the old version of the application from the backup.

   You cannot just copy the original ear file. Even if the original ear file (from the backup) is copied back to the Managed Server stage directory and you restart the Managed Server, the application is still not recovered. You must redeploy the original version.

18.2.9.3 Recovering an Undeployed Application

If a deployed application was undeployed from Oracle WebLogic Server, you can recover it.

To recover the application:

1. Recover the application files from backup, if needed.
2. Redeploy the old version of the application from the backup. If the application was deployed to a cluster, redeploy the application to the same cluster.

   You cannot just copy the original ear file. Even if the original ear file (from the backup) is copied back to the Managed Server stage directory and you restart the Managed Server, the application is still not recovered. You must redeploy the original version.

18.2.9.4 Recovering a Composite Application

A new version of a composite application (such as SOA application) was redeployed to a Managed Server or cluster. The application is no longer functional.

To recover the application:

1. Recover the application files from backup, if needed.
2. Redeploy the old version of the application. If the application was deployed to a cluster, redeploy the application to the same cluster.

18.2.10 Recovering a Database

If your database that contains your metadata repository, including the MDS Repository, is corrupted, you can recover it using RMAN. You can recover the database at the desired granularity, either a full recovery or a tablespace recovery.

For best results, recover the database to the most current state, using point-in-time recovery (if the database is configured in Archive Log Mode.) This ensures that the latest data is recovered. For example:

```
rman> restore database;
rman> recover database;
```

See Appendix D for the schemas used by each component.

For detailed steps, see the *Oracle Database Backup and Recovery User's Guide*.

18.3 Recovering After Loss of Host

This section describes how to recover your Oracle Fusion Middleware environment after losing the original operating environment. For example, you could have a serious system malfunction or loss of media. The sections includes the following topics:

- Recovering After Loss of Oracle WebLogic Server Domain Host
- Recovering After Loss of Administration Server Host
- Recovering After Loss of Managed Server Host
- Recovering After Loss of Component Host
- Additional Actions for Recovering Entities After Loss of Host
- Recovering After Loss of Database Host

Note: When you are recovering in the case of loss of host, you must restore the files using the same path as on the original host.

18.3.1 Recovering After Loss of Oracle WebLogic Server Domain Host

To recover an Oracle WebLogic Server domain:

1. Stop all relevant processes. That is, stop all processes that are related to the domain, such as the Administration Server and Managed Servers. For example, stop the Administration Server:

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

2. Recover the domain directory from backup:

   ```
   cd DOMAIN_HOME
   ```

```
(UNIX) tar -xf domain_backup_042012.tar
(Windows) jar xtf domain_backup_042012.jar
```

3. Start all relevant processes. That is, start all processes that are related to the domain. For example, start the Administration Server:

```
DOMAIN_HOME/bin/startWebLogic.sh -Dweblogic.management.username=username
 -Dweblogic.management.password=password
 -Dweblogic.system.StoreBootIdentity=true
```

4. If you cannot start the Administration Server, recover it, as described in Section 18.3.2.
5. If you cannot start a Managed Server, recover it, as described in Section 18.3.3.

18.3.2 Recovering After Loss of Administration Server Host

If you lose a host that contains the Administration Server, you can recover it to the same host or a different host, as described in the following topics:

- Recovering the Administration Server to the Same Host
- Recovering the Administration Server to a Different Host

18.3.2.1 Recovering the Administration Server to the Same Host

In this scenario, you recover the Administration Server either to the same host after the operating system has been reinstalled or to a new host that has the same host name. For example, the Administration Server is running on Host A and the Managed Server is running on Host B. Host A has failed for some reason and the Administration Server must be recovered.

To recover the Administration Server to the same host:

1. Recover the file system. For example, recover the domain containing the Administration Server, as described in Section 18.3.1.
2. Attempt to start the Administration Server. For example:

```
DOMAIN_HOME/bin/startWebLogic.sh -Dweblogic.management.username=username
 -Dweblogic.management.password=password
 -Dweblogic.system.StoreBootIdentity=true
```

   If the Administration Server starts, you do not need to take any further steps.
3. If the Administration Server fails to start, take the following steps on Host A:
   a. Stop all relevant processes. That is, stop all processes that are related to the domain, such as the Administration Server and Managed Servers. For example, to stop the Administration Server on Linux:

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

   b. Recover the Middleware home, if needed:

```
tar -xf mw_home_backup_042012.tar
```

   c. If the domain directory does not reside in the Middleware home, recover the domain directory from backup:

```
cd DOMAIN_HOME
tar -xf domain_backup_042012.tar
```

   d. Start the Administration Server. For example:

```
DOMAIN_HOME/bin/startWebLogic.sh -Dweblogic.management.username=username
    -Dweblogic.management.password=password
    -Dweblogic.system.StoreBootIdentity=true
```

e. Start the Managed Servers, specifying the Administration URL for the host:

```
DOMAIN_HOME/bin/startManagedWebLogic.sh managed_server_name admin_url
```

f. Start Node Manager:

```
java weblogic.WLST
wls:/offline> startNodeManager()
```

18.3.2.2 Recovering the Administration Server to a Different Host

In this scenario, the Administration Server is running on Host A and the Managed Server is running on Host B. Host A has failed for some reason and the Administration Server must be moved to Host C.

To recover the Administration Server to a different host:

1. Recover the Middleware home to Host C (the new Host):

```
cd MW_HOME
tar -xf mw_home_backup_042012.tar
```

2. If the domain directory does not reside in the Middleware home, recover the domain directory from backup:

```
cd DOMAIN_HOME
tar -xf domain_backup_042012.tar
```

3. If the Administration Server has a Listen address, create a new machine with the new host name, as described in Section 18.3.5.5.

4. Start Node Manager on Host C if it was configured on the original host:

```
java weblogic.WLST
wls:/offline> startNodeManager()
```

5. Start the Administration Server. For example:

```
DOMAIN_HOME/bin/startWebLogic.sh -Dweblogic.management.username=username
 -Dweblogic.management.password=password
 -Dweblogic.system.StoreBootIdentity=true
```

6. Start the Managed Servers. The section "Restarting a Failed Administration Server" in the *Oracle Fusion Middleware Managing Server Startup and Shutdown for Oracle WebLogic Server* describes different ways to restart them, depending on how they were configured.

   One option is to use the following script, specifying the Administration URL of the new host:

```
DOMAIN_HOME/bin/startManagedWebLogic.sh managed_server_name admin_url
```

7. Ensure that additional application artifacts are available. For example, if the deployment mode is nostage or external_stage, applications may reside in directories outside of the domain directory. Make your application files available to the new Administration Server by copying them from backups or by using a shared disk. Your application files should be available in the same relative location on the new file system as on the file system of the original Administration Server.

If the application is staged, the Administration Server copies the application bits to the staged directories on the Managed Server hosts.

8. Update Oracle Inventory, as described in Section 18.3.5.7.
9. If your environment contains Oracle HTTP Server, modify the mod_wl_ohs.conf file, as described in Section 18.3.5.4.
10. Edit the targets.xml file for Fusion Middleware Control, as described in Section 18.3.5.2.
11. Oracle Management Service, which is part of Fusion Middleware Control, is on the original host and is recovered to the new host when you restore the Administration Server. Oracle Management Agent connects to Oracle Management Service to monitor certain components. If your environment contains components, such as Oracle Internet Directory and Oracle Virtual Directory, that use Oracle Management Agent, but they are located on a different host, you must take the following steps on each host containing the components. For example, the Administration Server was on Host A, but is restored, along with Oracle Management Service, to Host B. Oracle Internet Directory is on Host C and Oracle Virtual Directory is on Host D. You must take these steps on both Host C and Host D.

    a. Edit the following file:

    ```
    (UNIX) ORACLE_INSTANCE/EMAGENT/emagent_name/sysman/config/emd.properties
    (Windows)ORACLE_INSTANCE\EMAGENT\emagent_name\sysman\config\emd.properties
    ```

    Update the following entries, replacing the host name with the new host for the Administration Server:

    ```
    emdWalletSrcUrl=http://newhost.domain.com:port/em/wallets/emd
    REPOSITORY_URL=http://newhost.domain.com:port/em/upload/
    ```

    b. Shut down and restart the EM Agent process:

    ```
    cd ORACLE_INSTANCE/EMAGENT/emagent_dir
    ./emctl stop agent
    ./emctl start agent
    ./emctl status agent
    ```

    The status command shows the REPOSITORY_URL pointing to the new host.

Now you can start and stop the Managed Server on Host B using the Administration Console running on Host C.

If you are recovering the Administration Server for a Web Tier installation, see Section 18.3.5 for information about additional actions you must take.

18.3.3 Recovering After Loss of Managed Server Host

If you lose a host that contains a Managed Server, you can recover it to the same host or a different host, as described in the following topics:

- Recovering a Managed Server to the Same Host
- Recovering a Managed Server to a Different Host
- Recovering an Oracle SOA Suite Managed Server That Has a Separate Directory

  This section pertains when Oracle SOA Suite is configured in a domain and no Managed Servers share the domain directory with the Administration Server.

18.3.3.1 Recovering a Managed Server to the Same Host

In this scenario, you recover a Managed Server to the same host after the operating system has been reinstalled or to a new host that has the same host name. The Administration Server is running on Host A and the Managed Server is running on Host B. Host B failed for some reason and the Managed Server must be recovered to Host B.

To recover a Managed Server to the same host:

1. Start Node Manager on Host B:

```
java weblogic.WLST
wls:/offline> startNodeManager()
```

2. Start the Managed Server. For example:

```
DOMAIN_HOME/bin/startManagedWebLogic.sh managed_server_name admin_url
```

   If the Managed Server starts, it connects to the Administration Server and updates its configuration changes. You do not need to take any further steps.

3. If the Managed Server fails to start or if the file system is lost, take the following steps:

   a. Stop Node Manager:

```
java weblogic.WLST
wls:/offline> stopNodeManager()
```

   b. Recover the Middleware home to Host B from the backup, if required:

```
tar -xf mw_home_backup_042012.tar
```

   c. If the Managed Server contains Oracle Portal, Oracle Reports, Oracle Forms Services, or Oracle Business Intelligence Discoverer, and the Managed Server domain directories reside outside of the Middleware home, restore the domain, in addition to the Middleware home. For example:

```
cd Domain_Home
tar -xf domain_home_backup_042012.tar
```

      Go to Step e.

   d. If the Managed Server does not contain the components listed in Step c, take the following steps:

      – Create a domain template jar file for the Administration Server running in Host A, using the pack utility. For example:

```
pack.sh -domain=MW_HOME/user_projects/domains/domain_name
  -template=/scratch/temp.jar -template_name=test_install
  -template_author=myname -log=/scratch/logs/my.log -managed=true
```

        Specifying the -managed=true option packs up only the Managed Servers. If you want to pack the entire domain, omit this option.

      – Unpack the domain template jar file in Host B, using the unpack utility:

```
unpack.sh -template=/scratch/temp.jar
  -domain=MW_HOME/user_projects/domains/domain_name
  -log=/scratch/logs/new.log -log_priority=info
```

e. Ensure that the application artifacts are accessible from the Managed Server host. That is, if the application artifacts are not on the same server as the Managed Server, they must be in a location accessible by the Managed Server.

Note:

- For applications that are deployed in nostage and external_stage mode, copy the application artifacts from the Administration Server host directory.
- For applications that are deployed in stage mode, the Administration server copies the application bits to the staged directories on the Managed Server hosts.

See *Oracle Fusion Middleware Deploying Applications to Oracle WebLogic Server* for information about deploying applications.

---

f. If Node Manager is not started, start it:

```
java weblogic.WLST
wls:/offline> startNodeManager()
```

g. Start the Managed Server. For example:

```
DOMAIN_HOME/bin/startManagedWebLogic.sh managed_server_name admin_url
```

The Managed Server connects to the Administration Server and updates its configuration changes.

18.3.3.2 Recovering a Managed Server to a Different Host

In this scenario, the Administration Server is running on Host A and the Managed Server is running on Host B. Host B failed for some reason and the Managed Server must be recovered to Host C.

Important: Recover the Middleware home to the same location as the original.

To recover a Managed Server to a different host:

1. Recover the Middleware home for the Managed Server to Host C.

   ```
   tar -xf mw_home_backup_042012.tar
   ```

2. If the Managed Server contains Oracle Portal, Oracle Reports, Oracle Forms Services, or Oracle Business Intelligence Discoverer, and the Managed Server domain directories reside outside of the Middleware home, restore the domain, in addition to the Middleware home. For example:

   ```
   cd Domain_Home
   tar -xf domain_home_backup_042012.tar
   ```

   Go to Step 4.

3. If the Managed Server does not contain the components listed in Step 2, take the following steps:

   a. Create a domain template jar file from the Administration Server running in Host A, using the pack utility. For example:

```
pack.sh -domain=MW_HOME/user_projects/domains/domain_name
  -template=/scratch/temp.jar -template_name=test_install
  -template_author=myname -log=/scratch/logs/my.log -managed=true
```

Specifying the -managed=true option packs up only the Managed Servers. If you want to pack the entire domain, omit this option.

b. Unpack the domain template jar file on Host C, using the unpack utility:

```
unpack.sh -template=/scratch/temp.jar
   -domain=MW_HOME/user_projects/domains/domain_name
   -log=/scratch/logs/new.log -log_priority=info
```

If you are recovering to a different domain home, use the -app_dir switch in the unpack command.

4. Ensure that the application artifacts are accessible from the Managed Server host. That is, if the application artifacts are not on the same server as the Managed Server, they must be in a location accessible by the Managed Server.

Note:

- For applications that are deployed in nostage and external_stage mode, copy the application artifacts from the Administration Server host directory.
- For applications that are deployed in stage mode, the Administration server copies the application bits to the staged directories on the Managed Server hosts.

See *Oracle Fusion Middleware Deploying Applications to Oracle WebLogic Server* for information about deploying applications.

---

5. Start Node Manager on Host C, if it is not started:

```
java weblogic.WLST
wls:/offline> startNodeManager()
```

6. Using WLST, connect to the Administration Server and then enroll Node Manager running in the new host with the Administration Server:

```
connect('username','password','t3://host:port')
nmEnroll('MW_HOME/user_projects/domains/domain_name',
  'MW_HOME/wlserver_n/common/nodemanager')
```

7. Change the Managed Server configuration to point to the new host:

a. In the WebLogic Server Administration Console, create a machine, which is a logical representation of the computer that hosts one or more WebLogic Servers, and point it to the new host. (From the Home page, select **Machines**. Then, click **New.**) Follow the directions in the Administration Console help.

If you identify the Listen Address by IP address, you must disable Host Name Verification on the Administration Servers that access Node Manager. For more information and instructions, see "Using Hostname Verification" in *Oracle Fusion Middleware Securing Oracle WebLogic Server*.

b. Change the Managed Server configuration to point to the new machine. (From the left pane of the Console, expand **Environment** and then **Servers**. Then, select the name of the server. Select the **Configuration** tab, then the **General**

tab. In the **Machine** field, select the machine to which you want to assign the server.)

Change **Listen Address** to the new host. (If the listening address was set to blank, you do not need to change it.)

8. Start the Managed Server. For example:

```
DOMAIN_HOME/bin/startManagedWebLogic.sh managed_server_name admin_url
```

The Managed Server connects to the Administration Server and updates its configuration changes.

9. Update Oracle Inventory, as described in Section 18.3.5.7.
10. If your environment contains Oracle HTTP Server, modify the mod_wl_ohs.conf file, as described in Section 18.3.5.4.
11. Edit the targets.xml file for Fusion Middleware Control, as described in Section 18.3.5.2.

Now you can start and stop the Managed Server on Host C using the Administration Server running on Host A.

18.3.3.3 Recovering an Oracle SOA Suite Managed Server That Has a Separate Directory

When Oracle SOA Suite is configured in a domain and no Managed Servers share the domain directory with the Administration Server, you must restore the Managed Server directory. For example, a domain contains two Managed Servers, one of which contains Oracle SOA Suite, but neither of the Managed Server's directories are in the same directory structure as the Administration Server.

To recover to the same host or a different host, use the procedures in Section 18.2.6.3.

18.3.4 Recovering After Loss of Component Host

If you lose a host that contains a component (and its Managed Server, if applicable), you can recover most components to the same host or a different host using the procedures described in the following topics:

- Recovering a Java Component to the Same Host
- Recovering a Java Component to a Different Host
- Recovering a System Component to the Same Host
- Recovering a System Component to a Different Host

Some components require additional actions, which are described in the sections listed in Table 18–2.

Table 18–2 Recovery Procedures for Loss of Host for Particular Components

Component	Procedure
Oracle Access Management Access Manager	Section 18.3.4.5.7
Oracle Access Management Identity Federation	Section 18.3.4.5.4
Oracle Access Management Mobile and Social	No additional steps needed for loss of host when you are recovering to the same host. For recovery to a different host, follow the procedure in Section 18.3.4.5.9.

Table 18–2 (Cont.) Recovery Procedures for Loss of Host for Particular Components

Component	Procedure
Oracle Access Management Secure Token Service	Section 18.3.4.5.10
Oracle Adaptive Access Manager	Section 18.3.4.5.8
Oracle BI Discoverer	Section 18.3.4.8.4
Oracle BI Enterprise Edition	Section 18.3.4.9
Oracle BI Publisher	Section 18.3.4.10
Oracle Business Process Management	No additional steps needed for loss of host. Follow the procedure in Section 18.2.7.7.
Oracle Data Integrator	Section 18.3.4.15
Oracle Directory Integration Platform	Section 18.3.4.5.3
Oracle Entitlements Server	No additional steps needed for loss of host. Follow the procedures in Section 18.3.1 and Section 18.2.3.
Oracle Essbase	Section 18.3.4.12
Oracle Forms Services	Section 18.3.4.8.2
Oracle HTTP Server	Section 18.3.4.7.1
Oracle Hyperion Calculation Manager	Section 18.3.4.13
Oracle Hyperion Financial Reporting	Section 18.3.4.14
Oracle Hyperion Smart View	No additional steps needed for loss of host. Follow the procedure in Section 18.2.7.16.
Oracle Identity Manager	Section 18.3.4.5.5
Oracle Identity Navigator	Section 18.3.4.5.6
Oracle Information Rights Management	No additional steps needed for loss of host. Follow the procedure in Section 18.2.7.18.
Oracle Internet Directory	Section 18.3.4.5.1
Oracle Portal	Section 18.3.4.8.1
Oracle Privileged Account Manager	Section 18.3.4.5.11
Oracle Real-Time Decisions	Section 18.3.4.11
Oracle Reports	Section 18.3.4.8.3
Oracle SOA Suite	Section 18.3.4.6.
Oracle Virtual Directory	Section 18.3.4.5.2
Oracle Web Cache	Section 18.3.4.7.2
Oracle WebCenter Content	Section 18.3.4.16.1
Oracle WebCenter Content: Imaging	No additional steps needed for loss of host. Follow the procedure in Section 18.2.7.19.
Oracle WebCenter Content: Records	Section 18.3.4.16.2
Oracle WebCenter Portal's Activity Graph	No additional steps needed for loss of host. Follow the procedure in Section 18.2.7.8.
Oracle WebCenter Portal's Analytics	No additional steps needed for loss of host. Follow the procedure inSection 18.2.7.9.

18.3.4.1 Recovering a Java Component to the Same Host

To recover a Java component to the same host, such as Oracle SOA Suite:

1. Recover the Managed Server, as described in Section 18.2.6.1.
2. If the component requires additional steps, as noted in Table 18–2, take those steps.

18.3.4.2 Recovering a Java Component to a Different Host

To recover a Java component to a different host, such as Oracle SOA Suite:

1. Recover the Managed Server, as described in Section 18.3.3.2.
2. Edit the targets.xml file for Fusion Middleware Control, as described in Section 18.3.5.2.

However, note that some components require additional steps, as noted in Table 18–2.

18.3.4.3 Recovering a System Component to the Same Host

To recover a system component, such as Oracle HTTP Server, to the same host, you take the following general steps. However, note that some components require additional steps, as noted in Table 18–2.

1. Stop all relevant processes. That is, stop all processes that are related to the component. For example, to stop Oracle HTTP Server:

   ```
   opmnctl stopproc ias-component=component_name
   ```

   For information on stopping components, see Section 4.3.

2. Recover the component-specific files from backup. Section 16.5 lists the directories and files needed for each component. For example, to recover Oracle HTTP Server files, you recover the following directories:

   ```
   ORACLE_INSTANCE/config/OHS/component_name
   ORACLE_INSTANCE/diagnostics/logs/OHS/component_name
   ```

3. If the Oracle instance home has been deregistered from the Administration Server, register the Oracle instance:

   ```
   opmnctl registerinstance -adminHost admin_server_host
        -adminPort admin_server_port -adminUsername username
        -adminPassword password
        -wlserverHome wlserver_home_location
   ```

   If only the file system is being recovered, you do not need to register the Oracle instance.

4. Start all relevant processes, as described in Section 4.3.

18.3.4.4 Recovering a System Component to a Different Host

To recover a system component, such as Oracle HTTP Server, to a different host, you take the following general steps. However, note that most components require additional steps, as noted in Table 18–2.

1. Recover the Middleware home, as described in Section 18.2.1.
2. Start all relevant processes. Section 4.3 explains how to start components.
3. Update the registration of the Oracle instance with the Administration Server, using the opmnctl `updateinstanceregistration` command on the new host. For example:

```
opmnctl updateinstanceregistration -adminHost admin_server_host
```

This command updates OPMN's instance.properties file.

4. Update the registration of the component with the Administration Server, using the opmnctl `updatecomponentregistration` command on the new host. For example, to update the registration for Oracle Virtual Directory, use the following command:

```
opmnctl updatecomponentregistration -Host new_host -Port nonSSLPort
    -componentName ovd1 -componentType OVD
```

5. Edit the targets.xml file for Fusion Middleware Control, as described in Section 18.3.5.2.

18.3.4.5 Recovering Identity Management Components to a Different Host

For most Identity Management components, you recover the Managed Server, as described in Section 18.3.3.2.

Some components require additional steps to recover the components to a different host, as described in the following topics:

- Recovering Oracle Internet Directory to a Different Host
- Recovering Oracle Virtual Directory to a Different Host
- Recovering Oracle Directory Integration Platform to a Different Host
- Recovering Oracle Access Management Identity Federation to a Different Host
- Recovering Oracle Identity Manager to a Different Host
- Recovering Oracle Identity Navigator to a Different Host
- Recovering Oracle Access Management Access Manager to a Different Host
- Recovering Oracle Adaptive Access Manager to a Different Host
- Recovering Oracle Access Management Mobile and Social to a Different Host
- Recovering Oracle Access Management Secure Token Service After Loss of Host

18.3.4.5.1 Recovering Oracle Internet Directory to a Different Host To recover Oracle Internet Directory to a different host:

1. Recover the component, as described in Section 18.3.4.4.
2. On UNIX and Linux systems, before you attempt to start Oracle Internet Directory, set the following file to have root permission:

   ORACLE_HOME/bin/oidldapd

   For example:

```
chown root oidldapd
chmod 4710 oidldapd
```

3. Recover Oracle Management Agent, as described in Section 18.3.5.3.
4. If the Managed Server on which Oracle Directory Services Manager is deployed is moved to different host and if SSL is enabled, you must delete the following file on the new host:

   DOMAIN_HOME/servers/wls_ods1/tmp/_WL_user/odsm_
   11.1.1.1.0/*randomid*/war/conf/odsm.cer

Oracle Directory Services Manager uses this file as its keystore and trust store and the password is stored in JKS. However, when Oracle Directory Services Manager is copied to another host and is started, it generates a different password. If you delete the file, Oracle Directory Services Manager creates a new file when it starts, with the new password.

18.3.4.5.2 Recovering Oracle Virtual Directory to a Different Host To recover Oracle Virtual Directory to a different host:

1. Recover the component, as described in Section 18.3.4.4.
2. Recover Oracle Management Agent, as described in Section 18.3.5.3.

18.3.4.5.3 Recovering Oracle Directory Integration Platform to a Different Host To recover Oracle Directory Integration Platform to a different host:

1. Recover the Managed Server, as described in Section 18.3.3.2.
2. Before starting the Managed Server, restore the files in the following directory:

   ```
   DOMAIN_HOME/servers/wls_ods1/stage/DIP/11.1.1.1.0/
   ```

3. Start the Managed Servers and Oracle instances.
4. If Oracle Internet Directory is also moved to a different host, execute the following commands immediately after the Managed Server and the Oracle instance are started:

   ```
   set ORACLE_HOME Oracle_home_path
   set WLS_HOME WLS_Home_path
   cd ORACLE_HOME/bin
   ./manageDIPServerConfig set -h dip_server_host -p dip_server_port
       -D weblogic_user -attribute oidhostport -value oid_host:oid_ssl_port
   ```

   The manageDIPServerConfig command prompts you for a password.

   For example:

   ```
   ./manageDIPServerConfig set -h hostname -p 19523 -D weblogic
              -attribute oidhostport -value hostname.domain.com:24163
   ```

5. Register the Oracle instance, along with all of its components, with the Administration Server, using the opmnctl `registerinstance` command on the new host. For example:

   ```
   opmnctl registerinstance -adminHost admin_server_host
        -adminPort admin_server_port -adminUsername username
        -adminPassword password
        -wlserverHome wlserver_home_location
   ```

18.3.4.5.4 Recovering Oracle Access Management Identity Federation to a Different Host
Because Identity Federation provides SSO functionality, if the host name on which Identity Federation runs is changed as part of loss of host recovery, it impacts remote partners. In that case, remote partners must make changes regarding the host name to continue to operate. It may take many days for remote partners to update their data and this may cause production delays that are unacceptable. Oracle strongly recommends that you do not change the host name of a standalone Identity Federation server.

If a load balancer is part of the environment and the host where Identity Federation is being recovered is in the list of VIPs, then no host name changes are required.

In the case of a standalone installation of Identity Federation, Oracle recommends using a new host with the same name to minimize the impact. However, if, for whatever reason, you must use a different host name for recovering Identity Federation, then the host name must be updated manually for Identity Federation and remote partners.

To recover Identity Federation to a different host:

1. Recover the Managed Server, as described in Section 18.3.3.2.
2. Recover Oracle Management Agent, as described in Section 18.3.5.3.
3. Register the Oracle instance, along with all of its components, with the Administration Server, using the opmnctl `registerinstance` command on the new host. For example:

```
opmnctl registerinstance -adminHost admin_server_host
     -adminPort admin_server_port -adminUsername username
     -adminPassword password
     -wlserverHome wlserver_home_location
```

4. Provide the updated data to remote partners.
5. Modify the host name using Fusion Middleware Control:
   a. In the navigation pane, expand the farm and then **Identity and Access.**
   b. Select the Identity Federation instance.
   c. From the Identity Federation menu, choose **Administration,** then **Server Properties.**

      The Server Properties page is displayed.
   d. For **Host,** replace the old host name with the new host name.
   e. For **Port,** replace the port number if it has changed.
   f. For **SOAP Port,** replace the port number if it has changed.
   g. Click **Apply.**
   h. Restart the Managed Server to which Identity Federation is deployed:

```
DOMAIN_HOME/bin/startManagedWebLogic.sh managed_server_name
          admin_url
```

6. If Identity Federation is acting as an SSL server, you must replace the SSL certificate presented by Identity Federation to clients with a new one that has the new host name. Otherwise, host name verification by clients may fail.

18.3.4.5.5 Recovering Oracle Identity Manager to a Different Host To recover Oracle Identity Manager to a different host:

1. Restore the domain, as described in Section 18.3.2.
2. Restore the Oracle home, as described in Section 18.2.3.
3. Restore the database containing the OIM, OID, MDS, and SOAINFRA schemas, if necessary. See Section 18.2.10.
4. Synchronize the Oracle Identity Manager database and the LDAP provider. See the *Oracle Fusion Middleware Command Reference for Oracle WebLogic Server* for more information.

5. Export the oim-config.xml file, using the weblogicExportMetadata.sh script. Then, edit the file, changing the host name or IP address for the SOA URL. Import the file into MDS, using the weblogicImportMetadata.sh script.
6. Create a new machine with the new host name, as described in Section 18.3.5.5.
7. Reassociate the weblogic user with any groups, as described in Section 18.3.5.6.

18.3.4.5.6 Recovering Oracle Identity Navigator to a Different Host To recover Oracle Identity Navigator to a different host:

1. Create a new machine with the new host name, as described in Section 18.3.5.5.
2. Reassociate the weblogic user with any groups, as described in Section 18.3.5.6.

18.3.4.5.7 Recovering Oracle Access Management Access Manager to a Different Host To recover Access Manager to a different host:

1. Follow the instructions in Section 18.2.7.5.
2. To restore the WLS Agent, restore the Managed Server, as described in Section 18.3.3.2.
3. Log into the Access Manager console.
4. Modify the host name, specifying the new host name for the Access Manager proxy server. See "Viewing or Editing Individual OAM Server and Proxy Settings" in the *Oracle Fusion Middleware Administrator's Guide for Oracle Access Management*.
5. Optionally, if you have a load balancer, modify the host name. See "Managing Load Balancing" in the *Oracle Fusion Middleware Administrator's Guide for Oracle Access Management*.
6. If you have protected pages, you must reregister Oracle Single Sign-On or WebGate as partners with Access Manager:
   a. Log in to the Oracle Access Manager console.
   b. Select the System Configuration tab and select **Access Manager.**
   c. For all configured SSO agents, update all host names with the name of the new host:
      – The Server List, if set, should refer to the new host name.
      – The User Defined Parameters, if set, should refer to new host name.
      – The Logout Redirect URL, if set, should refer to the new host name.

   Alternatively, you can use the oamreg tool, described in "Registering Agents and Applications" in the *Oracle Fusion Middleware Administrator's Guide for Oracle Access Management*. Also see "Acquiring and Setting Up the Remote Registration Tool" in the same manual.
7. Create a new machine with the new host name, as described in Section 18.3.5.5.
8. Edit the WebGate configuration file, ObAccessClient.xml, to update the host name for the Access Manager server. The file is located in the following directory:

   `DOMAIN_HOME/output/agentName/`

9. Reassociate the weblogic user with any groups, as described in Section 18.3.5.6.

18.3.4.5.8 Recovering Oracle Adaptive Access Manager to a Different Host To recover Oracle Adaptive Access Manager to a different host:

1. Follow the instructions in Section 18.2.7.6.
2. Create a new machine with the new host name, as described in Section 18.3.5.5.
3. Reassociate the weblogic user with any groups, as described in Section 18.3.5.6.

18.3.4.5.9 Recovering Oracle Access Management Mobile and Social to a Different Host

To recover Mobile and Social to a different host:

1. Recover the domain home, as described in Section 18.3.1.
2. Recover the Oracle home, as described in Section 18.2.3.
3. Update the host name for the Access Manager hosts for the following server providers:
   - OAMAuthorization
   - OAMAuthentication

   You can use the following WLST command to change the host names:

```
updateServiceProvider('oracle.security.idaas.rest.provider.authorization.OAMS
     DKAuthZServiceProvider', 'Authorization', '[]','
     [{OAM_VERSION: OAM_11G},{WEBGATE_ID: accessgate-oic},
     {ENCRYPTED_PASSWORD: aaaa},{DEBUG_VALUE: 0},{TRANSPORT_SECURITY: OPEN},
     {OAM_SERVER_1: "new_server1:5575"},{OAM_SERVER_1_MAX_CONN: 4},
     {OAM_SERVER_2: "new_server2:5575"},{OAM_SERVER_2_MAX_CONN: 4}]',
     'OAMAuthorization', 'Out Of The Box Oracle Access Manager (OAM)
        Authorization Service Provider')
```

   For more information about the updateServiceProvider command, see "updateServiceProvider" in the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

   Alternatively, you can use the Access Manager console:

   a. Log into the Access Manager console.
   b. Click System Configuration, then select **Identity Connect.**
   c. In the Mobile Services section, select **Service Providers.**
   d. For the following, update the host names with the new host name:
      – OAMAuthentication (located under Authentication Service Providers)
      – OAMAuthorization (located under Authorization Service Providers)
   e. Click **Save.**

18.3.4.5.10 Recovering Oracle Access Management Secure Token Service After Loss of Host

To recover Secure Token Service after loss of host:

1. Recover the Middleware home:

```
tar -xf mw_home_backup_042012.tar
```

2. If the domain directory does not reside in the Middleware home, recover the domain directory from backup:

```
cd DOMAIN_HOME
tar -xf domain_backup_042012.tar
```

3. Start the Administration Server. For example:

```
DOMAIN_HOME/bin/startWebLogic.sh -Dweblogic.management.username=username
```

```
-Dweblogic.management.password=password
-Dweblogic.system.StoreBootIdentity=true
```

4. Start the Managed Servers, specifying the Administration URL for the host:

   ```
   DOMAIN_HOME/bin/startManagedWebLogic.sh managed_server_name admin_url
   ```

5. Start Node Manager:

   ```
   java weblogic.WLST
   wls:/offline> startNodeManager()
   ```

18.3.4.5.11 Recovering Oracle Privileged Account Manager to a Different Host To recover Oracle Privileged Account Manager to a different host:

1. Restore the domain, as described in Section 18.3.2.
2. Restore the Oracle home, as described in Section 18.2.3.
3. Restore the database containing the Oracle Privileged Account Manager and Oracle Platform Security Services schemas, if necessary. See Section 18.2.10.
4. Create a new machine with the new host name, as described in Section 18.3.5.5.
5. Reassociate the weblogic user with any groups, as described in Section 18.3.5.6.
6. Log in to the Oracle Privileged Account Manager Console, using the following URL:

   ```
   http://adminserver-host:adminserver-port/oinav/opam
   ```

7. Click **Configure OPAM** and configure the host name and port.

18.3.4.6 Recovering Oracle SOA Suite After Loss of Host

Note that when Oracle SOA Suite is configured in a domain and no Managed Servers share the domain directory with the Administration Server, take the steps described in Section 18.3.3.3. Otherwise, follow the steps in this section.

To recover the Oracle SOA Suite Managed Server to the same host, recover the Managed Server, as described in Section 18.3.3.1.

To recover the Oracle SOA Suite Managed Server to a different host after loss of host:

1. Before you recover, update the WSDL file to point to the new host name and port.
2. Recover the Managed Server, as described in Section 18.3.3.2.
3. After you recover the Oracle SOA Suite Managed Server, take the following actions:
   - If the ant command is used to deploy composites, edit the deploy-sar.xml file, which is located in:

     ```
     (UNIX) ORACLE_HOME/bin
     (Windows) ORACLE_HOME\bin
     ```

     In the following line, replace the previous host name with the new host name:

     ```
     <property name="wlsHost" value="newhostname"/>
     ```

     If a Load Balancer is used, do not modify this property. Instead, register the new host with the Load Balancer.
   - Change the host name in the soa-infra MBean:

a. In Fusion Middleware Control, navigate to the Managed Server.

b. From the WebLogic Server menu, choose **System MBean Browser.**

c. Expand **Application Defined MBeans,** then **oracle.as.soainfra.config**, then **Server:** ***server_name*** and then **SoaInfraConfig.** Select **soa-infra.**

d. In the Attributes tab, click **ServerURL.** If the ServerURL attribute contains a value, change the host name to the new host name.

e. Click **Apply.**

– Redeploy all applications which have the WSDL files updated to the new host name.

Note: If there is no Load Balancer configured with the environment and Oracle SOA Suite must be recovered to a different host, then in-flight instances that are pending a response from task flow and asynchronous responses are not recovered. Oracle recommends that you use a Load Balancer to ensure that you can recover to a different host.

If a Load Balancer is configured with the environment, take the following additional steps:

1. Log in to the Oracle WebLogic Server Administration Server.
2. In Domain Structure, navigate to Servers. For each Managed Server, select the Protocol tab, then the HTTP tab.
3. For **Frontend Host,** enter the new host name.
4. For **Frontend HTTP Port** and **Frontend HTTPs Port,** if applicable, enter the new port number.
5. Restart each Managed Server.

18.3.4.7 Recovering Web Tier Components to a Different Host

The Web tier consists of Oracle HTTP Server and Oracle Web Cache. The following topics describe how to recover these components to a different host:

- Recovering Oracle HTTP Server to a Different Host
- Recovering Oracle Web Cache to a Different Host

18.3.4.7.1 Recovering Oracle HTTP Server to a Different Host To recover Oracle HTTP Server to a different host:

1. Recover the component, as described in Section 18.3.4.4.
2. Recover Oracle Management Agent, as described in Section 18.3.5.3.
3. Modify the ServerName entry in the following file to have the new host name:

```
(UNIX) ORACLE_INSTANCE/config/OHS/ohs_name/httpd.conf
(Windows) ORACLE_INSTANCE\config\OHS\ohs_name\httpd.conf
```

18.3.4.7.2 Recovering Oracle Web Cache to a Different Host To recover Oracle Web Cache to a different host:

1. Recover the component, as described in Section 18.3.4.4.
2. Recover Oracle Management Agent, as described in Section 18.3.5.3.

3. Edit the webcache.xml file, replacing the previous host name with the new host name. The file is located in:

```
(UNIX) ORACLE_INSTANCE/config/WebCache/webcache_name
(Windows) ORACLE_INSTANCE\config\WebCache\webcache_name
```

18.3.4.8 Recovering Oracle Portal, Oracle Reports, Oracle Forms Services, and Oracle Business Intelligence Discoverer to a Different Host

The following topics describe how to recover these components to a different host:

- Recovering Oracle Portal to a Different Host
- Recovering Oracle Forms Services to a Different Host
- Recovering Oracle Reports to a Different Host
- Recovering Oracle Business Intelligence Discoverer to a Different Host

18.3.4.8.1 Recovering Oracle Portal to a Different Host To recover Oracle Portal to a different host:

1. Restore the Middleware home, the domain directory, and the Oracle instance directory to the new host. See Section 18.3.3.2 for more information.
2. Recover Oracle Management Agent, as described in Section 18.3.5.3.
3. If the instance has been deregistered, register the Oracle instance, along with all of its components, with the Administration Server, using the opmnctl `registerinstance` command on the new host. For example:

```
opmnctl registerinstance -adminHost admin_server_host
     -adminPort admin_server_port -adminUsername username
     -adminPassword password
     -wlserverHome wlserver_home_location
```

4. Update the registration of the Oracle instance with the Administration Server, using the opmnctl `updateinstanceregistration` command on the new host. For example:

```
opmnctl updateinstanceregistration -adminHost admin_server_host
```

   This command updates OPMN's instance.properties file.

5. Modify the following files, replacing the old host name with the new host name:

```
ORACLE_INSTANCE/config/OHS/ohs_name/httpd.conf
ORACLE_INSTANCE/config/OHS/ohs_name/moduleconf/portal.conf
```

6. Run the ssoreg script, which is located in:

```
Identity_Management_ORACLE_HOME/sso/bin
```

   Use the following command:

```
ssoreg.sh -site_name newhost:http_listen_port
    -mod_osso_url http://newhost:http_listen_port -config_mod_osso TRUE
    -oracle_home_path $ORACLE_HOME -config_file any_new_file_path
    -admin_info cn=orcladmin -virtualhost -remote_midtier
```

   For example:

```
ssoreg.sh -site_name example.com:8090
   -mod_osso_url http://example.com:8090 -config_mod_osso TRUE
   -oracle_home_path $ORACLE_HOME -config_file /tmp/loh_osso.conf
```

```
-admin_info cn=orcladmin -virtualhost -remote_midtier
```

7. Copy the file from the previous step to the new host.
8. In the new host, modify the OssoConfigFile section in the following file to include the path of the file in step 6:

   ORACLE_INSTANCE/config/OHS/ohs1/moduleconf/mod_osso.conf

   For example:

   ```
   <IfModule mod_osso.c>
       OssoIpCheck off
       OssoSecureCookies off
       OssoIdleTimeout off
       OssoConfigFile /tmp/path_of_file_created
   ```

9. Edit the following files, replacing the previous host name with the new host name:
   - webcache.xml. This file is located in:

     (UNIX) *ORACLE_INSTANCE*/config/WebCache/*webcache_name*
     (Windows) *ORACLE_INSTANCE*\config\WebCache*webcache_name*

     Replace all occurrences of the previous host name with the new host name.
   - instance.properties. The file is located in:

     (UNIX) *ORACLE_INSTANCE*/config/OPMN/opmn
     (Windows) *ORACLE_INSTANCE*\config\OPMN\opmn

     In the following line, replace the previous host name with the new host name if the Administration Server host name has changed.

     adminHost=*host_name*

10. If the published host used to access Oracle Portal is changing, take the following steps. This could happen if you have a single node install which contains both Oracle Web Cache and WLS_PORTAL, and those processes must move to a different host. Another scenario is when you have Oracle Web Cache running on a node remotely from WLS_PORTAL, and Oracle Web Cache must move to a different host. In both these cases, take the following steps to update the Published Host information within Oracle Portal. (Note: If you have a load balancer or reverse proxy configuration, the steps are not needed.)
    a. Recursively delete all content from the following directory, but do not delete the directory itself:

       ORACLE_INSTANCE/portal/cache

    b. Log in to Fusion Middleware Control. Expand the farm and right-click **Portal.** Then, choose **Settings,** then **Wire Configuration.**
    c. In the Portal Midtier section, update **Published Host** with the new host name.
    d. In the Oracle Web Cache section, update **Host** with the new host name.
11. Restart the WLS_PORTAL instance.

18.3.4.8.2 Recovering Oracle Forms Services to a Different Host To recover Oracle Forms Services to a different host:

1. Recover the Managed Server, as described in Section 18.3.3.2.

2. Recover Oracle Management Agent, as described in Section 18.3.5.3.
3. Register the Oracle instance, along with all of its components, with the Administration Server, using the opmnctl `registerinstance` command on the new host. For example:

```
opmnctl registerinstance -adminHost admin_server_host
     -adminPort admin_server_port -adminUsername username
     -adminPassword password
     -wlserverHome wlserver_home_location
```

4. Edit the following files, replacing the previous host name with the new host name:
   - webcache.xml. This file is located in:

     ```
     (UNIX) ORACLE_INSTANCE/config/WebCache/webcache_name
     (Windows) ORACLE_INSTANCE\config\WebCache\webcache_name
     ```

     Replace all occurrences of the previous host name with the new host name.
   - instance.properties. The file is located in:

     ```
     (UNIX) ORACLE_INSTANCE/config/OPMN/opmn
     (Windows) ORACLE_INSTANCE\config\OPMN\opmn
     ```

     In the following line, replace the previous host name with the new host name if the Administration Server host name has changed.

     ```
     adminHost=host_name
     ```

   - forms.conf. The file is located in:

     ```
     (UNIX) ORACLE_INSTANCE/config/OHS/ohs_name/moduleconf
     (Windows) ORACLE_INSTANCE\config\OHS\ohs_name\moduleconf
     ```

     Replace the host name in the parameter WebLogicHost with the name of the new host.
5. On the Administration Server host, edit the following file:

   ```
   DOMAIN_HOME/opmn/topology.xml
   ```

   Add properties for the <ias-component id> element for Oracle Forms Services. The following example shows the element after you modify it:

```
</ias-component>
  <ias-component id="forms" type="FormsComponent" >
  <em-properties>
      <property name="OracleHome" value="/path_to_oracle_home" />
      <property name="instName" value="instance_name" />
      <property name="EMTargetType" value="oracle_forms" />
      <property name="version" value="11.1.1" />
  </em-properties>
 </ias-component>
```

6. On the host where the Oracle instance has been recovered, update the registration of the component with the Administration Server, using the opmnctl `updatecomponentregistration` command on the new host.

   For example:

```
opmnctl updatecomponentregistration -Host new_host -Port nonSSLPort
    -componentName forms -componentType FormsComponent
```

7. Run the ssoreg script, which is located in:

   Identity_Management_ORACLE_HOME/sso/bin

   Use the following command:

   ```
   ssoreg.sh -site_name newhost:http_listen_port
       -mod_osso_url http://newhost:http_listen_port -config_mod_osso TRUE
       -oracle_home_path $ORACLE_HOME -config_file any_new_file_path
       -admin_info cn=orcladmin -virtualhost -remote_midtier
   ```

   For example:

   ```
   ssoreg.sh -site_name example.com:8090
      -mod_osso_url http://example.com:8090 -config_mod_osso TRUE
      -oracle_home_path $ORACLE_HOME -config_file /tmp/loh_osso.conf
      -admin_info cn=orcladmin -virtualhost -remote_midtier
   ```

8. Copy the file from the previous step to the new host.
9. In the new host, modify the OssoConfigFile section in the following file to include the path of the file in step 7:

   ORACLE_INSTANCE/config/OHS/ohs1/moduleconf/mod_osso.conf

   For example:

   ```
   <IfModule mod_osso.c>
       OssoIpCheck off
       OssoSecureCookies off
       OssoIdleTimeout off
       OssoConfigFile /tmp/path_of_file_created
   ```

18.3.4.8.3 Recovering Oracle Reports to a Different Host To recover Oracle Reports to a different host:

1. Recover the Managed Server, as described in Section 18.3.3.2.
2. Recover Oracle Management Agent, as described in Section 18.3.5.3.
3. Register the Oracle instance, along with all of its components, with the Administration Server, using the opmnctl `registerinstance` command on the new host. For example:

   ```
   opmnctl registerinstance -adminHost admin_server_host
        -adminPort admin_server_port -adminUsername username
        -adminPassword password
        -wlserverHome wlserver_home_location
   ```

4. Edit the following files, replacing the previous host name with the new host name:
   - reports_install.properties. The file is located in:

     ```
     (UNIX) ORACLE_INSTANCE/reports
     (Windows) ORACLE_INSTANCE\reports
     ```

     Edit the parameters SERVER_NAME, OHS_HOST and REPORTS_MANAGED_WLS_HOST.

   - webcache.xml. This file is located in:

     ```
     (UNIX) ORACLE_INSTANCE/config/WebCache/webcache_name
     (Windows) ORACLE_INSTANCE\config\WebCache\webcache_name
     ```

     Replace all occurrences of the previous host name with the new host name.

– instance.properties. The file is located in:

```
(UNIX) ORACLE_INSTANCE/config/OPMN/opmn
(Windows) ORACLE_INSTANCE\config\OPMN\opmn
```

In the following line, replace the previous host name with the new host name if the Administration Server host name has changed.

```
adminHost=host_name
```

– reports_ohs.conf. The file is located in:

```
(UNIX) ORACLE_INSTANCE/config/OHS/ohs_name/moduleconf
(Windows) ORACLE_INSTANCE\config\OHS\ohs_name\moduleconf
```

– rwservlet.properties. The file is located in:

```
(UNIX) DOMAIN_HOME/config/fmwconfig/servers/server_
name/applications/reports_version/configuration
(Windows) DOMAIN_HOME\config\fmwconfig\servers\server_
name\applications\reports_version\configuration
```

In the file, modify the <server> element to use the new host name.

5. In the following directory, rename the subdirectory to have the new host name:

```
(UNIX) ORACLE_INSTANCE/diagnostics/logs/ReportsServer
(Windows) ORACLE_INSTANCE\diagnostics\logs\ReportsServer
```

6. In the following directory, rename the *old_host_name*.dat file to the new host name:

```
(UNIX) ORACLE_INSTANCE/reports/server
(Windows) ORACLE_INSTANCE\reports\server
```

7. In the following directory, rename the subdirectory to have the new host name:

```
(UNIX) ORACLE_INSTANCE/config/ReportsServer
(Windows) ORACLE_INSTANCE\config\ReportsServer
```

8. Run the ssoreg script, which is located in:

```
Identity_Management_ORACLE_HOME/sso/bin
```

Use the following command:

```
ssoreg.sh -site_name newhost:http_listen_port
    -mod_osso_url http://newhost:http_listen_port -config_mod_osso TRUE
    -oracle_home_path $ORACLE_HOME -config_file any_new_file_path
    -admin_info cn=orcladmin -virtualhost -remote_midtier
```

For example:

```
ssoreg.sh -site_name example.com:8090
   -mod_osso_url http://example.com:8090 -config_mod_osso TRUE
   -oracle_home_path $ORACLE_HOME -config_file /tmp/loh_osso.conf
   -admin_info cn=orcladmin -virtualhost -remote_midtier
```

9. Copy the file from the previous step to the new host.

10. In the new host, modify the OssoConfigFile section in the following file to include the path of the file in step 8:

```
ORACLE_INSTANCE/config/OHS/ohs1/moduleconf/mod_osso.conf
```

For example:

```
<IfModule mod_osso.c>
    OssoIpCheck off
    OssoSecureCookies off
    OssoIdleTimeout off
    OssoConfigFile /tmp/path_of_file_created
```

11. In the following file, replace occurrences of the host name with the new host name:

    (UNIX) *DOMAIN_HOME*/servers/*server_name*/tmp/_WL_user/reports_*version*/*random_string*/META-INF/mbeans.xml
    (Windows) *DOMAIN_HOME*\servers*server_name*\tmp_WL_user\reports_*version**random_string*\META-INF\mbeans.xml

12. In the following file, replace occurrences of the host name with the new host name:

    ORACLE_INSTANCE/EMAGENT/emagent_<instanceName>/sysman/emd/targets.xml

    You change the host name in the elements beginning with the following:

    ```
    <Target TYPE="oracle_repapp" ..>
    <Target TYPE="oracle_repserv" ..>
    ```

18.3.4.8.4 Recovering Oracle Business Intelligence Discoverer to a Different Host To recover Oracle Business Intelligence Discoverer to a different host:

1. Recover the Managed Server, as described in Section 18.3.3.2.
2. Recover Oracle Management Agent, as described in Section 18.3.5.3.
3. Register the Oracle instance, along with all of its components, with the Administration Server, using the opmnctl `registerinstance` command on the new host. For example:

    ```
    opmnctl registerinstance -adminHost admin_server_host
         -adminPort admin_server_port -adminUsername username
         -adminPassword password
         -wlserverHome wlserver_home_location
    ```

4. Edit the following files, replacing the previous host name with the new host name:
   - module_disco.conf. This file is located in:

     (UNIX) *ORACLE_INSTANCE*/config/OHS/*ohs_name*//moduleconf
     (Windows) *ORACLE_INSTANCE*\config\OHS*ohs_name**moduleconf*

   - webcache.xml. This file is located in:

     (UNIX) *ORACLE_INSTANCE*/config/WebCache/*webcache_name*
     (Windows) *ORACLE_INSTANCE*\config\WebCache*webcache_name*

     Replace all occurrences of the previous host name with the new host name.

5. Run the ssoreg script, which is located in:

    Identity_Management_ORACLE_HOME/sso/bin

    Use the following command:

    ```
    ssoreg.sh -site_name newhost:http_listen_port
        -mod_osso_url http://newhost:http_listen_port -config_mod_osso TRUE
        -oracle_home_path $ORACLE_HOME -config_file any_new_file_path
        -admin_info cn=orcladmin -virtualhost -remote_midtier
    ```

    For example:

```
ssoreg.sh -site_name example.com:8090
   -mod_osso_url http://example.com:8090 -config_mod_osso TRUE
   -oracle_home_path $ORACLE_HOME -config_file /tmp/loh_osso.conf
   -admin_info cn=orcladmin -virtualhost -remote_midtier
```

6. Copy the file from the previous step to the new host.
7. In the new host, modify the OssoConfigFile section in the following file to include the path of the file in step 5:

   ORACLE_INSTANCE/config/OHS/ohs1/moduleconf/mod_osso.conf

   For example:

```
<IfModule mod_osso.c>
    OssoIpCheck off
    OssoSecureCookies off
    OssoIdleTimeout off
    OssoConfigFile /tmp/path_of_file_created
```

18.3.4.9 Recovering Oracle BI Enterprise Edition to a Different Host

You can recover Oracle BI EE to a different host.

The following topics describe how to move Oracle BI EE to a different host with the same name:

- Recovering Oracle BI EE to a Different Host in a Non-Clustered Environment
- Recovering Oracle BI EE to a Different Host in a Clustered Environment

18.3.4.9.1 Recovering Oracle BI EE to a Different Host in a Non-Clustered Environment The steps you take to recover Oracle BI EE to a different host depend on the operating system. Note that the host must have the same name as the original host.

- On UNIX, take the following steps:
  1. Restore the Middleware home from backup, as described in Section 18.2.1.
  2. Restore the database containing the Oracle BI EE schemas, if necessary. See Section 18.3.6.
- On Windows, take the following steps:
  1. Restore the Middleware home from backup, as described in Section 18.2.1, overwriting the Middleware home that you created with the new installation.
  2. Restore the database containing the Oracle BI EE schemas, if necessary. See Section 18.3.6.
  3. Install the C++ libraries from Microsoft, by executing the following file:

     Oracle_BI\bifoundation\install\vc80\vcredist_x86.exe

  4. Import the Registry entries that you exported into the new host, as described in Section 18.3.4.9.4.

18.3.4.9.2 Recovering Oracle BI EE to a Different Host in a Clustered Environment In this scenario, you have an Oracle BI EE cluster on two hosts, Host A and Host B. Host A contains instance1 and Host B contains instance2. Host A must be replaced for some reason, such as a host crash, and you must recover to Host C and scale out the system so that Host C contains instance3.

Take the following steps:

1. Restore the Middleware home from backup to Host C, as described in Section 18.2.1.
2. Restore the database containing the Oracle BI EE schemas, if necessary. See Section 18.3.6.
3. On Windows, install the C++ libraries from Microsoft, by executing the following file:

   ```
   Oracle_BI\bifoundation\install\vc80\vcredist_x86.exe
   ```

4. On Windows, import the Registry entries that you exported into the new host, as described in Section 18.3.4.9.4.
5. If the failed node contained the Administration Server, recover it, as described in steps 1 through 5 in Section 18.3.2.2.
6. Scale out the Oracle BI EE system, as described in "Scaling Out the BI System on APPHOST2" in the *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle Business Intelligence*.

   Note the following:

   - When you enter the directory specifications for the Domain Home and Applications Home, enter specifications for directories that do not yet exist or that are empty.
   - If the Domain Home field is empty, update the following file with the domain directory:

     ```
     MW_HOME/wlserver_10.3/common/nodemanager/nodemanager.domains
     ```

     Before you start Node Manager, take the following steps:

     a. Stop Node Manager, if it is running.

     b. Run the setNMProps.sh script, which is located in the ORACLE_COMMON_HOME/common/bin directory, to set the StartScriptEnabled property to `true` before starting Node Manager:

     ```
     cd ORACLE_COMMON_HOME/common/bin
     ./setNMProps.sh
     ```

     c. Restart Node Manager and enable dynamic registration using the following commands:

     ```
     cd WL_HOME/server/bin
     export JAVA_OPTIONS=-DDomainRegistrationEnabled=true
     ./startNodeManager.sh
     ```

     Note: It is important that you set -DDomainRegistrationEnabled=true whenever you start a Node Manager which must manage the Administration Server. If there is no Administration Server on this computer, and if this computer is not an Administration Server failover node, then Node Manager can be started as follows:

     ```
     ./startNodeManager.sh
     ```

7. Scale out the system components, as described in "Scaling Out the System Components" in the *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle Business Intelligence*. Fusion Middleware Control prompts you to restart the instances after you have changed their configuration. Restart the instances.

Because instance1 on Host A is no longer available, you must modify its count of BI Servers, Presentation Services, and JavaHosts to be 0. Fusion Middleware Control prompts you to restart the instances after you have changed their configuration. Restart the instances.

8. Make instance2 the primary instance and instance3 the secondary instance using Fusion Middleware Control:
   a. Make instance 2 the primary instance and specify the secondary instance as none. Activate and restart the instance as prompted by Fusion Middleware Control.
   b. Make instance3 the secondary instance. Activate and restart the instance as prompted by Fusion Middleware Control.

   See "Configuring Secondary Instances of Singleton System Components" in the *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle Business Intelligence* for more information.
9. Set the listen address of the bi_server*n* Managed Server, as described in "Setting the Listen Address for the bi_server2 Managed Server" in the *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle Business Intelligence*.
10. Disable host name verification for the bi_server*n* Managed Server, as described in "Disabling Host Name Verification for the bi_server2 Managed Server" in the *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle Business Intelligence*.
11. Depending on your configuration, perform additional configuration, as described in "Performing Additional Configuration for Oracle Business Intelligence Availability" in the *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle Business Intelligence*.
12. If Oracle HTTP Server is installed, set the frontend HTTP host and port for the Oracle WebLogic Server cluster to ensure that Oracle BI Search URLs are set correctly, as described in "Setting the Frontend URL for the Administration Console" in the *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle Business Intelligence*.
13. Configure Node Manager for the Managed Servers as described in "Configuring Node Manager for the Managed Servers" in the *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle Business Intelligence*.
14. Start the Oracle BI EE Managed Server and all of the OPMN-managed components.

18.3.4.9.3 Additional Steps for Recovering Oracle BI EE Depending on your environment, you may need to take additional steps after you perform the steps in Section 18.3.4.9.2:

- If the failed host contained the master BI Server, primary cluster controller, and primary Oracle BI Scheduler and you want the new instance to be the master BI Server, take the following steps as appropriate. Note that if you want to leave instance2 as the master BI server, you do not need to take these additional steps.
  - If the master BI Server is lost:
    a. Stop Oracle WebLogic Server and OPMN processes on all nodes.
    b. Update the following configuration file to designate a new master BI Server:

       INSTANCE_HOME/config/OracleBIApplication/coreapplication/biee-domain.xml

In the section <AvailabilityOptions>, edit the following:

```
masterBIServerOracleInstanceId="instance_name"
masterBIServerComponentId="component_id"
```

c. Copy the file to the other host.

d. Restart the Administration Server and the Managed Servers.

– If the primary cluster controller or scheduler is lost, it fails over to the standby cluster controller or scheduler. You must determine whether you want to reconfigure it to be the primary cluster controller or scheduler or leave it as secondary that has been activated because the primary components have failed. For more information, see "Configuring Secondary Instances of Singleton System Components" in the *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle Business Intelligence*.

- If the failed host contained the BI Server, the secondary cluster controller, and the secondary Oracle BI Scheduler, designate the new host as the secondary cluster controller or scheduler.
- If the failed host contained the BI Server and system components such as BI Presentation Services and BI Java hosts, no additional steps are needed.
- If the failed host contained the following related components, recover them:
  - Oracle Business Intelligence Publisher: See Section 18.3.4.10.
  - Oracle Real-Time Decisions: See Section 18.3.4.11.
  - Oracle Essbase: See Section 18.3.4.12.
  - Oracle Hyperion Calculation Manager: See Section 18.3.4.13.
  - Oracle Hyperion Financial Reporting: See Section 18.3.4.14.
  - Oracle Hyperion Smart View: See Section 18.2.7.16.

18.3.4.9.4 Importing Oracle BI EE Registry Entries On Windows, you must import the Oracle BI EE Registry entries to the new host. Section 17.3.3 describes how to export them from the original host.

1. Copy all the files that you exported from the original host to the new host.
2. Double-click each file you copied from the original host. Click **Yes** when prompted, to import the file into the Registry.

18.3.4.10 Recovering Oracle Business Intelligence Publisher to a Different Host

To recover Oracle Business Intelligence Publisher to a different host:

1. Recover the Managed Server containing the Oracle Business Intelligence Publisher component, as described in Section 18.3.3.
2. Restore the database containing the Oracle Business Intelligence Publisher schemas, if necessary. See Section 18.2.10.

If backup artifacts are restored from different time, then user accounts, user reports, and user permissions revert to the restored version. Restore all artifacts from the same point in time.

18.3.4.11 Recovering Oracle Real-Time Decisions to a Different Host

To recover Oracle Real-Time Decisions to a different host:

1. Recover the Managed Server containing the Oracle Real-Time Decisions component, as described in Section 18.3.3.

Note that if backup artifacts are restored from different time, the analytic models miss a period of learning, but their intelligence is unaffected.

18.3.4.12 Recovering Oracle Essbase After Loss of Host

If Oracle Essbase is in a clustered environment, and the failed host contained Essbase system component clustering using OPMN, take the following additional steps to recover Oracle Essbase. In this scenario, Oracle Essbase clustering is set up on Node A and B, and you lose Node A. You create a new Essbase component on Node C and a new cluster with Essbase components on Node C and Node B. The old cluster is gone and should not be recovered at any time.

1. Scale out the Oracle BI EE system and the system components on the new host, Node C, as described in Steps 6 and 7 in Section 18.3.4.9.
2. Mount the shared ARBORPATH directory to the same path on both Node B and Node C. For example, on Windows, map the directory to Drive Y on both Node B and Node C.
3. Edit the following file on Node B and Node C, to update the adminHost property:

   ORACLE_INSTANCE/config/OPMN/opmn/instance.properties

   For example:

   ```
   adminHost=ADMINVHN
   ```

4. On Node C, copy the wallet and push it to the Administration Server:
   a. Copy the wallet from Node B to Node C. The wallet is located in:

      ORACLE_INSTANCE/config/OPMN/opmn/wallet

   b. Ensure that the opmn.xml file has ssl enabled="true".
   c. Push the wallet to the Administration Server, using the following command:

      ```
      ./opmnctl updateinstanceregistration
      ```

      This command prompts for an Oracle WebLogic Server Administrator password. The `updateinstanceregistration` command updates information registered on the Administration Server for the Oracle instances. Specifically, the `updateinstanceregistration` command updates the registered OPMN remote port, remote host, and wallet from the current OPMN settings.

   d. Restart OPMN on all nodes:

      ```
      opmnctl startall
      ```

5. Shut down the Oracle Essbase instance on Node B because it will be made part of the new cluster:

   ```
   opmnctl stopproc ias-component=essbaseserver1 process-type=Essbase
   ```

6. Log in to Fusion Middleware Control, and expand Business Intelligence. Then, select the Business Intelligence instance.
7. Select the Availability tab, then the Failover tab. In the Essbase Agents section, specify the value of **Shared Folder Path.**

8. Click **Apply,** then **Activate Changes.**
9. In the same tab, in the Primary/Secondary Configuration section, and in the Secondary Host/Instance, choose a secondary Oracle Essbase instance on Node C.
10. Click **Apply,** then **Activate Changes.**

    The Oracle Essbase instance will be created and both servers will be part of a cluster.
11. Stop all OPMN components on Nodes B and C, and then restart them:

```
opmnctl stopall
opmnctl startall
```

18.3.4.13 Recovering Oracle Hyperion Calculation Manager After Loss of Host

To recover Oracle Hyperion Calculation Manager after loss of host, follow the procedure in Section 18.2.7.14.

If the database must be restored, restore the database and import Calculation Manager rules and rule sets from the file you exported.

In Calculation Manager, select **File,** and then **Import**.

18.3.4.14 Recovering Oracle Hyperion Financial Reporting After Loss of Host

If you lose a host that contains Oracle Hyperion Financial Reporting, you can recover it to the same host or a different host. To recover Oracle Hyperion Financial Reporting, recover the Oracle home, as described in Section 18.2.3 and the Oracle instance, as described in Section 18.2.4.

If the database host has changed, update the host name using the following commands:

```
Epmsys_registry updateproperty financial_reporting_product/@host new_host
Epmsys_registry updateproperty financial_reporting_product/logical_web_app/@host
new_host
Epmsys_registry updateproperty financial_reporting_product/logical_web_
app/financial_reporting_web_app@host new_host
```

18.3.4.15 Recovering Oracle Data Integrator to a Different Host

To recover Oracle Data Integrator to a different host:

1. If the database must be restored to a different host, restore it, as described in Section 18.3.6.
2. Recover the Oracle Data Integrator Oracle home from backup, as described in Section 18.2.3
3. Restore the domain, as described in Section 18.3.2.
4. Stop each standalone agent, and stop the Oracle Data Integrator applications deployed in Oracle WebLogic Server.
5. Modify the repository connection information in the topology, if the database is on a different host:
   a. Connect to the restored Oracle Data Integrator repository using ODI Studio. Create a new connection for the master repository to the new database host, as described in "Connecting to the Master Repository" in the *Oracle Fusion Middleware Developer's Guide for Oracle Data Integrator*.

b. Edit each of the Work Repositories. Click **Connection** and edit the connection information so that the JDBC URL points to the new database host containing the work repository.

c. Edit each physical agent's configuration and provide the updated Host Name value and, if changed, the Port value.

d. If there are standalone agent scripts generated and they contain the -PORT property, change the -PORT value to the new port value. The scripts are named *agentName*_agent.sh or *agentName*_agent.bat.

6. For each standalone agent, edit the following files and change the ODI_MASTER_ URL parameter to match the new database host location, if the database is on a different host:

```
oracledi/agent/bin/odiparams.*
```

7. Edit the following file to change the database connection information and the port number:

```
oracledi/agent/bin/odi_opmn_standaloneagent_template.xml
```

8. In the Oracle WebLogic Server configuration, edit the Data Sources to match the new database host location.

9. Restart the standalone agents and the Oracle Data Integrator applications deployed in Oracle WebLogic Server.

18.3.4.16 Recovering Oracle WebCenter Content to a Different Host

The following sections describe how to recover Oracle WebCenter Content and Oracle WebCenter Content: Records to a different host:

- Recovering Oracle WebCenter Content to a Different Host
- Recovering Oracle WebCenter Content: Records After Loss of Host

18.3.4.16.1 Recovering Oracle WebCenter Content to a Different Host

To recover Oracle WebCenter Content to a different host:

1. If the database must be restored to a different host, restore it, as described in Section 18.3.6.

2. Restore the domain, as described in Section 18.3.2.

3. If the Vault, WebLayout, or Search directories are not located in the domain directory, restore those directories, if necessary. For example, if the Vault directory is located on a shared drive in /net/home/vault, restore it from backup:

```
cd /net/home/vault
tar -xf vault_backup_042012.tar
```

4. Edit the following file:

```
DOMAIN_HOME/ucm_domain/ucm/cs/config/config.cfg
```

In the file, change the HttpServerAddress setting to specify the new host. For example:

```
HttpServerAddress=hostname:port_number
```

Note that you should restore the database and the shared file system at the same time. If you cannot do that, you can use the IDCAnalyse utility to determine if there are any

inconsistencies between the database and the shared file system. If there are, you can perform a manual recovery using IDCAnalyse.

18.3.4.16.2 Recovering Oracle WebCenter Content: Records After Loss of Host Because Oracle WebCenter Content: Records depends on Oracle WebCenter Content and has no additional backup and recovery artifacts, see the recovery procedure for Oracle WebCenter Content in Section 18.3.4.16.1.

18.3.5 Additional Actions for Recovering Entities After Loss of Host

Depending on the entity that you are recovering, you may need to take additional actions after loss of host. The sections about each entity may require you to follow one or more of the following procedures. If so, that is noted in the section describing how to recover the entity.

The following topics describe the actions you may need to take:

- Recovering Fusion Middleware Control to a Different Host
- Changing the Host Name in the targets.xml File for Fusion Middleware Control
- Recovering Oracle Management Agent When Components Are Recovered to a Different Host
- Modifying the mod_wl_ohs.conf File
- Creating a New Machine for Certain Components
- Reassociating Users to Groups for Certain Identity Management Components
- Updating Oracle Inventory
- Recovering the Windows Registry

18.3.5.1 Recovering Fusion Middleware Control to a Different Host

To recover Fusion Middleware Control to a different host, take the following steps:

1. Update the host name in the following file:

   DOMAIN_HOME/servers
   /AdminServer/tmp/_WL_user/em/hsz5x1/META-INF/emoms.properties

   In the file, change the host name for the following properties:

   mas.conn.url
   oracle.sysman.emSDK.svlt.ConsoleServerHost

2. Edit the following file:

   (UNIX) *ORACLE_INSTANCE*/EMAGENT/emagent_*name*/sysman/config/emd.properties
   (Windows) *ORACLE_INSTANCE*\EMAGENT\emagent_*name*\sysman\config\emd.properties

   In the file, edit the following entry for each component monitored by Oracle Management Agent, replacing the host name:

   REPOSITORY_URL=http://*newhost*.*domain*.com:*port*/em/upload/

18.3.5.2 Changing the Host Name in the targets.xml File for Fusion Middleware Control

When you recover a component to a different host, you must update the targets.xml file for Fusion Middleware Control. The file is located at:

DOMAIN_HOME/sysman/state/targets.xml

In the file, change the host name to the new host name for components that are recovered to a different host.

18.3.5.3 Recovering Oracle Management Agent When Components Are Recovered to a Different Host

For many components, when you recover to a different host, as in the case of loss of host, you must take actions to recover Oracle Management Agent so that Fusion Middleware Control can manage the components. This pertains to the following installation types and components:

- Identity Management components
- Identity Federation
- Oracle Portal
- Oracle Business Intelligence Discoverer
- Oracle Forms Services
- Oracle Reports

To recover Oracle Management Agent, take the following actions:

1. Edit the following file:

   (UNIX) *ORACLE_INSTANCE*/EMAGENT/emagent_*name*/sysman/emd/targets.xml
   (Windows) *ORACLE_INSTANCE*\EMAGENT\emagent_*name*\sysman\emd\targets.xml

   In the file, edit the following element, replacing the host name:

   ```
   <Target TYPE="host" NAME="newhost.domain.com"
        DISPLAY_NAME="newhost.domain.com"/>
   ```

2. Edit the following file:

   (UNIX) *ORACLE_INSTANCE*/EMAGENT/emagent_*name*/sysman/config/emd.properties
   (Windows) *ORACLE_INSTANCE*\EMAGENT\emagent_*name*\sysman\config\emd.properties

   Update the following entry, replacing the host name:

   EMD_URL=http://*newhost.domain*.com:*port*/emd/main

3. Start Oracle Management Agent, using the following command:

   ```
   opmnctl startproc ias-component=EMAGENT
   ```

4. Start the Administration Server:

   ```
   DOMAIN_HOME/bin/startWebLogic.sh -Dweblogic.management.username=username
    -Dweblogic.management.password=password
    -Dweblogic.system.StoreBootIdentity=true
   ```

   Starting the Administration Server also starts Fusion Middleware Control.

18.3.5.4 Modifying the mod_wl_ohs.conf File

When you recover an Administration Server or a Managed Server to a different host and your environment includes Oracle HTTP Server, you must modify the following file on the new host:

(UNIX) *ORACLE_INSTANCE*/config/OHS/*ohs_name*/mod_wl_ohs.conf

```
(Windows) ORACLE_INSTANCE\config\OHS\ohs_name\mod_wl_ohs.conf
```

Modify all of the instances of the host name, port, and clusters (elements such as WebLogicHost, WebLogicPort, and WebLogicCluster) entries in that file. For example:

```
<Location /console>
    SetHandler weblogic-handler
    WebLogicHost Admin_Host
    WeblogicPort Admin_Port
    WLProxySSL ON
    WLProxySSLPassThrough ON
</Location>
 .
 .
 .
<Location /soa-infra>
   SetHandler weblogic-handler
   WebLogicCluster SOAHOST1VHN2:8001,*SOAHOST2VHN1*:*8001*
   WLProxySSL ON
   WLProxySSLPassThrough ON
</Location>
```

18.3.5.5 Creating a New Machine for Certain Components

For the following Identity Management components (and for the Administration Server if it has a Listen address,) you must create a new machine with the new host name before you start the Administration Server:

- Oracle Access Management Access Manager
- Oracle Adaptive Access Manager
- Oracle Identity Manager
- Oracle Identity Navigator
- Oracle Privileged Account Manager

Take the following steps:

1. Create a new machine with the new host name. Use the following WLST commands, in offline mode:

   ```
   wls:/offline> readDomain('DOMAIN_HOME')
   wls:/offline/sampledomain> machine = create('newhostname', 'Machine')
   wls:/offline/sampledomain> cd('/Machine/newhostname')
   wls:/offline/sampledomain> nm = create('newhostname', 'NodeManager')
   wls:/offline/sampledomain> cd('/Machine/newhostname/NodeManager/newhostname')
   wls:/offline/sampledomain> set('ListenAddress', 'newhostname')
   wls:/offline/sampledomain> updateDomain()
   wls:/offline/sampledomain> exit()
   ```

2. For the Administration Server, set the machine with the new host name, using the following WLST command, in offline mode:

   ```
   wls:/offline> readDomain('DOMAIN_HOME')
   wls:/offline/sampledomain> cd ('/Machine/newhostname')
   wls:/offline/sampledomain> machine = cmo
   wls:/offline/sampledomain> cd ('/Server/AdminServer')
   wls:/offline/sampledomain> set('Machine', machine)
   wls:/offline/sampledomain> updateDomain()
   wls:/offline/sampledomain> exit()
   ```

3. Set the listen port for the Administration Server:

```
wls:/offline/sampledomain> readDomain('DOMAIN_HOME')
wls:/offline/sampledomain> cd('/Server/AdminServer')
wls:/offline/sampledomain> cmo.setListenPort(8001)
wls:/offline/sampledomain> updateDomain()
wls:/offline/sampledomain> exit()
```

18.3.5.6 Reassociating Users to Groups for Certain Identity Management Components

When you restore a backup of the following Identity Management components, the weblogic user is no longer associated with groups to which it had previously been associated:

- Oracle Access Management Access Manager
- Oracle Adaptive Access Manager
- Oracle Identity Manager
- Oracle Identity Navigator

You must reassociate the weblogic user with the groups.

For information about associating a user with a group, see the section "Add Users to Groups" in the Oracle Fusion Middleware Oracle WebLogic Server Administration Console Help.

18.3.5.7 Updating Oracle Inventory

For many components, when you recover to a different host, as in the case of loss of host, you must update the Oracle inventory. To do so, execute the following script:

ORACLE_HOME/oui/bin/attachHome.sh

In addition, you must update beahomelist to edit the location of a Middleware home. Edit the following file to update the Middleware home information:

(UNIX) *user_home*/bea/beahomelist
(Windows) C:\bea\beahomelist

18.3.5.8 Recovering the Windows Registry

When you recover any component to a different host on Windows, as in the case of loss of host, you must import any Windows Registry keys related to Oracle Fusion Middleware to the new host. (You exported the Registry keys in Section 17.3.3.

Recover the following Registry key.

HKEY_LOCAL_MACHINE\Software\Oracle

In addition, recover each node that begins with **Oracle** within the following registry keys:

HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services
HKEY_LOCAL_MACHINE\SYSTEM\ControlSet001\Services
HKEY_LOCAL_MACHINE\SYSTEM\ControlSet002\Services

To import a key that you have previously exported, use the following command:

regedit /I *FileName*

For example:

regedit /I C:\oracleregistry.reg

You can also use the Registry Editor to import the key. See the Registry Editor Help for more information.

18.3.6 Recovering After Loss of Database Host

If the host that contained your database is lost, you can recover the database using RMAN.

For example:

```
rman> restore database;
rman> recover database;
```

For best results, recover the database to the most current state, using point-in-time recovery (if the database is configured in Archive Log Mode.) This ensures that the latest data is recovered. Also, use the same name for the database. Note the following:

- See Appendix D for the schemas used by each component.
- For Oracle BPEL Process Manager, point-in-time recovery ensures that the latest process definitions and in-flight instances are restored. However, this may result in reexecution of the process steps. Oracle recommends that you strive for idempotent Oracle BPEL Process Manager processes. If the system contains processes that are not idempotent, you must clean them up from the dehydration store before starting Oracle Fusion Middleware. See *Oracle Fusion Middleware Administrator's Guide for Oracle SOA Suite and Oracle Business Process Management Suite* for more information.

For detailed steps about recovering a database and using RMAN, see the *Oracle Database Backup and Recovery User's Guide*.

Part VIII

Advanced Administration: Expanding Your Environment

This part describes how to expand your Oracle Fusion Middleware environment.

It contains the following chapters:

19

Scaling Your Environment

You can expand your environment by adding Managed Servers, expanding your domain to include other products, creating a cluster of Managed Servers, copying existing Middleware homes or existing Oracle Fusion Middleware components such as Oracle SOA Suite or Oracle HTTP Server, as described by the following topics:

19.1 Overview of Scaling Your Environment

Scalability is the ability of a system to provide throughput in proportion to, and limited only by, available hardware resources. A scalable system is one that can handle increasing numbers of requests without adversely affecting response time and throughput.

The growth of computational power within one operating environment is called vertical scaling. Horizontal scaling is leveraging multiple systems to work together on a common problem in parallel.

Oracle Fusion Middleware scales both vertically and horizontally. Horizontally, Oracle Fusion Middleware can increase its throughput with several Managed Servers grouped together to share a workload. Also, Oracle Fusion Middleware provides great vertical scalability, allowing you to add more Managed Servers or components to the same host.

High availability refers to the ability of users to access a system. Deploying a high availability system minimizes the time when the system is down (unavailable) and maximizes the time when it is running (available). Oracle Fusion Middleware is designed to provide a wide variety of high availability solutions, ranging from load balancing and basic clustering to providing maximum system availability during catastrophic hardware and software failures.

High availability solutions can be divided into two basic categories: local high availability and disaster recovery.

See Also:

- *Oracle Fusion Middleware High Availability Guide* for more information about high availability
- *Oracle Fusion Middleware Disaster Recovery Guide*

19.2 Extending a Domain to Support Additional Components

When you create an Oracle WebLogic Server domain, you create it using a particular domain template. That template supports a particular component or group of components, such as the Oracle SOA Suite. If you want to add other components, such as Oracle WebCenter Portal, to that domain, you can extend the domain by creating additional Managed Servers in the domain, using a domain template for the component which you want to add.

When you extend a domain, the domain must be offline.

To extend a domain, you use the Oracle WebLogic Server Configuration Wizard from an Oracle home into which the desired component has been installed. Then, you select the domain that you want to extend and the component you want to add.

Table 19–1 shows some of the components you can add to an existing domain and the domain templates needed.

Table 19–1 Supported Domain Extensions

Existing Domain Template	Components That Can Be Added
Oracle SOA Suite	Any Oracle SOA Suite component. Any Oracle WebCenter Portal component. Extend with Oracle WebCenter Portal domain template. Any Web Tier component. Extend with Web Tier domain template.
Oracle Identity Management	Any Identity Management component. Any Web Tier component. Extend with Web Tier domain template.
Oracle Portal, Oracle Reports, Oracle Forms Services, Oracle Business Intelligence Discoverer	Any of these components. Any Web Tier component. Extend with Web Tier domain template.

Note: For Identity Management components, Oracle Portal, Oracle Forms Services, Oracle Reports, and Oracle Business Intelligence Discoverer, if the component that you want to include in the domain is on a remote host, the WebLogic Server home must have the same full path as the WebLogic Server home for the original component.

For example, you are extending a domain that initially was created to support Identity Management components so that it can now also support Web Tier components and you install Web Tier components on a remote host. In this case, if the WebLogic Server home for Identity Management is located in /scratch/oracle/Middleware/wlserver_10.3, the WebLogic Server home for the Web Tier must also be located in /scratch/oracle/Middleware/wlserver_10.3.

For example, to extend a domain that initially was created to support Oracle SOA Suite so that it can now also support Oracle WebCenter Portal:

1. Use RCU to add any required schemas for the component, as described in the *Oracle Fusion Middleware Repository Creation Utility User's Guide*.
2. Install Oracle WebCenter Portal, as described in the *Oracle Fusion Middleware Installation Guide for Oracle WebCenter Portal*.
3. From an Oracle home that was installed for the component you want to add (for example, for Oracle WebCenter Portal), invoke the Configuration Wizard, using the following command:

   ```
   (UNIX) ORACLE_HOME/common/bin/config.sh
   (Windows) ORACLE_HOME\common\bin\config.cmd
   ```

   The Configuration Wizard's Welcome screen is displayed.
4. Select **Extend an existing WebLogic Domain.**
5. Click **Next.**

   The Select a WebLogic Domain Directory screen is displayed.
6. Select the directory for the domain to which you want to add the components.
7. Click **Next.**

   The Select Extension Source screen is displayed.
8. Select **Extend my domain automatically to support the following added products,** Then, select the source from which this domain is to be extended. For example, select **Oracle WebCenter Spaces.**
9. Click **Next.**

   The Configure JDBC Data Sources screen is displayed.
10. Select the schemas for the new component you added, entering the following information:
    - For **Vendor,** select **Oracle.**
    - For **Driver,** select **Oracle's Driver (Thin) for Service connections; Versions:9.0.1,9.2.0,10,11.**
    - For **Schema Owner,** do not enter anything. Each data source uses the user name specified in the table.
    - If you used the same password when you created the schemas, select all of the schemas and enter the password in **Schema Password.**

      Alternatively, you can specify different passwords for each data source by selecting each schema individually and entering the password.
    - With all of the schemas selected, for **DBMS/Service**, enter the SID of the database.
    - With all of the schemas selected, for **Host Name,** enter the host name of the database.
    - With all of the schemas selected, for **Port,** enter the listening port of the database.
11. Click **Next.**

    The Test Component Schema screen is displayed.

12. If the test succeeds, click **Next.**

    The Select Optional Configuration screen is displayed.

13. In this and the following customization screens, you can choose to customize. To do so, select the type of customization. If you do not want to customize the settings, click **Next.**

    The Configuration Summary screen is displayed.

14. Review the information on the screen and if it is correct, click **Extend.**

15. When the operation completes, click **Done.**

19.3 Adding Additional Managed Servers to a Domain

You can add Managed Servers to a domain to increase the capacity of your system. The Managed Servers can be added to a cluster.

When a Managed Server is added to a cluster, it inherits the applications and services that are targeted to the cluster. When a Managed Server is not added as a part of a cluster, it does not automatically inherit the applications and services from the template.

To add a Managed Server to a domain, you can use the Oracle WebLogic Server Administration Console or WLST.

> **See:** Administration Console Online Help and *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference* for complete information about adding Managed Servers.

To add a Managed Server to a domain using the Administration Console:

1. Display the Administration Console, as described in Section 3.4.1.
2. Lock the Oracle WebLogic Server configuration, as described in Section 3.4.2.
3. In the left pane, expand **Environment,** then select **Servers.**

   The Summary of Servers page is displayed.

4. In the Servers table, click **New.**

   The Create a New Server: Server Properties page is displayed.

5. Enter the following information:

   - For **Name,** enter a name for the server.

     Each server within a domain must have a name that is unique for all configuration objects in the domain. Within a domain, each server, computer, cluster, JDBC connection pool, virtual host, and any other resource type must be named uniquely and must not use the same name as the domain.

   - For **Listen Address,** to limit the valid addresses for a server instance, enter an IP address or DNS name. Otherwise, URLs to the server can specify the host computer's IP address, any DNS name that maps to one of the IP addresses, or the localhost string.

   - For **Listen Port,** enter the port number from which you want to access the server instance.

     If you run multiple server instances on a single computer, each server must use its own listen port.

- Specify whether this server is to be a standalone server or a member of an existing cluster or a new cluster.
  - If this server is to be a standalone server, select **No, this is a stand-alone server.**
  - If this server is to be part of an existing cluster, select **Yes, make this server a member an existing cluster.** Then, select the cluster.

    This option is not shown if there are no existing clusters.
  - If this server is to be part of a new cluster, select **Yes, create a new cluster for this server.**

6. Click **Next.**

   The Review Choices page is displayed.
7. Review the information. If it is correct, click **Finish.**
8. Apply JRF to the Managed Server or cluster as described in Section 19.3.1.

Note that you can also use Fusion Middleware Control to add a Managed Server to a domain. From the Farm menu, choose **Create/Delete Components.** Then, in the Fusion Middleware Components page, select **Create,** then **WebLogic Server.**

19.3.1 Applying Oracle JRF Template to a Managed Server or Cluster

Oracle JRF (Java Required Files) consists of those components not included in the Oracle WebLogic Server installation and that provide common functionality for Oracle business applications and application frameworks.

JRF consists of several independently developed libraries and applications that are deployed into a common location. The components that are considered part of Java Required Files include Oracle Application Development Framework shared libraries and ODL logging handlers.

You must apply the JRF template to a Managed Server or cluster in certain circumstances. You can only apply JRF to Managed Servers that are in a domain in which JRF was configured. That is, you must have selected Oracle JRF in the Configuration Wizard when you created or extended the domain.

Note the following points about applying JRF:

- When you add a Managed Server to an existing cluster that is already configured with JRF, you do not need to apply JRF to the Managed Server.
- When you add a Managed Server to a domain and the Managed Server requires JRF services, but the Managed Server is not part of a cluster, you must apply JRF to the Managed Server.
- When you create a new cluster and the cluster requires JRF, you must apply JRF to the cluster.
- You do not need to apply JRF to Managed Servers that are added by product templates during the template extension process (though you must select JRF in the Configuration Wizard).
- You must restart the server or cluster after you apply JRF.
- If you create a server using Fusion Middleware Control, the JRF template is automatically applied.

You use the custom WLST command `applyJRF` to configure the Managed Servers or cluster with JRF. To use the custom WLST commands, you must invoke the WLST script from the Oracle Common home. See Section 3.5.1.1 for more information.

The format of the applyJRF command is:

```
applyJRF(target={server_name | cluster_name | *}, domainDir=domain_path,
        [shouldUpdateDomain= {true | false}])
```

You can use the `applyJRF` command online or offline:

- In online mode, the JRF changes are implicitly activated if you use the `shouldUpdateDomain` option with the value `true` (which is the default.) In online mode, this option calls the online WLST save() and activate() commands.
- In offline mode, you must restart the Administration Server and the Managed Servers or cluster. (In offline mode, if you specify the `shouldUpdateDomain` option with the value `true`, this option calls the WLST updateDomain() command.)

For example, to configure the Managed Server server1 with JRF, use the following command:

```
applyJRF(target='server1', domainDir='/scratch/Oracle/Middleware/user_
projects/domains/domain1')
```

To configure all Managed servers in the domain with JRF, specify an asterisk (*) as the value of the `target` option.

To configure a cluster with JRF, use the following command:

```
applyJRF(target='cluster1', domainDir='/scratch/Oracle/Middleware/user_
projects/domains/domain1')
```

See Also:

- "Java Required Files Custom WLST Commands" in the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*
- Section I.2.2 to use a different version of Spring than that which is supplied with JRF

19.4 Creating Additional Oracle Instances and System Components

When you install and configure an Oracle Fusion Middleware environment that contains system components, Oracle instances are created for those system components. If you require additional Oracle instances, you can create an Oracle instance using the OPMN createinstance command and you can create a system component using the OPMN createcomponent command, as described in the following topics:

- Creating an Oracle Instance Using a Non-Secure Port
- Creating an Oracle Instance Using a Secure Port

19.4.1 Creating an Oracle Instance Using a Non-Secure Port

You can create an Oracle instance that does not use a secure port to communicate with the Administration Server. You use the OPMN createinstance command. For example, to create an Oracle instance and an Oracle HTTP Server component instance:

1. Create the Oracle instances and the Oracle Web Cache instances:

a. From the command line, go the following directory:

```
(UNIX) ORACLE_HOME/opmn/bin
(Windows) ORACLE_HOME\opmn\bin
```

b. Create the Oracle instance, using the opmnctl createinstance command. For example:

```
opmnctl createinstance -oracleInstance /scratch/Oracle/Middleware/inst1
   -adminHost hostname -adminPort 7001
```

This command creates the Oracle instance and, by default, registers the instance with the Oracle WebLogic Server Administration Server.

c. Create the Oracle HTTP Server instance, using the opmnctl `createcomponent` command. For example:

```
opmnctl createcomponent -componentType OHS
    -oracleInstance /scratch/Oracle/Middleware/inst1
    -componentName webcache1
```

2. Register the Oracle instances, along with all of its components, with the Administration Server, using the opmnctl registerinstance command. For example:

```
opmnctl registerinstance -adminHost admin_server_host
     -adminPort admin_server_port -adminUsername username
     -adminPassword password
     -oracleInstance ORACLE_INSTANCE_dir -oracleHome ORACLE_HOME_dir
     -instanceName Instance_name -wlserverHome Middleware_Home
```

19.4.2 Creating an Oracle Instance Using a Secure Port

You can create an Oracle instance that uses a secure port to communicate with the Administration Server. You use the OPMN createinstance command. For example, to create an Oracle instance and an Oracle HTTP Server component instance that uses a secure port:

1. Create the Oracle instance as a instance that is not registered with the Administration Server. (You will register it in a subsequent step.)

```
opmnctl createinstance -oracleInstance /scratch/Oracle/Middleware/inst1
                       -adminRegistration OFF
```

2. To access the Administration Server using a secure port, you must supply the relevant security settings, such as passphrases and trust locations. The list of system properties required for a connection depends on the type of SSL connection being made. To supply the settings, create a file named wls.connect.properties. The following shows an example wls.connect.properties file:

```
# Example wls.connect.properties file for using the default DemoTrust store

# These are jsse system property settings used for accessing mbeans
javax.net.ssl.trustStore=/scratch/aime1/middleware/wlserver_
10.3/server/lib/DemoTrust.jks
javax.net.ssl.trustStorePassword=DemoTrustKeyStorePassPhrase

# These are weblogic system property settings used for app deployment
weblogic.home=/scratch/aime1/middleware/wlserver_10.3/server
weblogic.security.TrustKeyStore=DemoTrust
weblogic.security.JavaStandardTrustKeystorePassPhrase=DemoTrustKeyStorePassPhra
se
```

```
# Other commonly used weblogic property settings for app deployment
# Comment next line for a more verbose deployment
weblogic.log.StdoutSeverity=off
# Comment next line to enable host name verification
weblogic.security.SSL.ignoreHostnameVerification=true
```

3. Copy the file to the following location:

   ORACLE_INSTANCE/config/OPMN/opmn/wls.connect.properties

   The properties are loaded as system properties each time a secure connection is created.

   Note:

   - Because OPMN conveys these properties into the JVM without validating them, you must ensure that the settings are valid and complete and that the syntax is valid.
   - On Windows, you must ensure that any backslashes (\) in the wls.connect.properties file is correctly escaped. For example:

     ```
     javax.net.ssl.trustStore=c:\\oracle\\mw2238\\wlserver_
     10.3\\server\\lib\\DemoTrust.jks
     ```

4. Create the Oracle HTTP Server instance, using the opmnctl `createcomponent` command. For example:

   ```
   opmnctl createcomponent -componentType OHS
       -oracleInstance /scratch/Oracle/Middleware/inst1
       -componentName webcache1
   ```

5. Register the Oracle instance using the secure protocol:

   ```
   ORACLE_INSTANCE/bin/opmnctl registerinstance -adminProtocol t3s
         -adminHost admin_server_host
        -adminPort admin_server_port -adminUsername username
        -adminPassword password
        -oracleInstance ORACLE_INSTANCE_dir -oracleHome ORACLE_HOME_dir
        -instanceName Instance_name -wlserverHome Middleware_Home
   ```

19.5 Creating Clusters

A WebLogic Server **cluster** consists of multiple WebLogic Server server instances running simultaneously and working together to provide increased scalability and reliability. A cluster appears to clients to be a single WebLogic Server instance. The server instances that constitute a cluster can run on the same computer, or be located on different computers. You can increase a cluster's capacity by adding additional server instances to the cluster on an existing computer, or you can add computers to the cluster to host the incremental server instances. Each server instance in a cluster must run the same version of WebLogic Server.

You can create a cluster of Managed Servers using WLST, the Oracle WebLogic Server Administration Console, or Fusion Middleware Control. This section describes how to create a cluster using Fusion Middleware Control.

Note: For Identity Management components, Oracle Portal, Oracle Forms Services, Oracle Reports, and Oracle Business Intelligence Discoverer, if one or more Managed Servers that you want to include in a cluster is on a remote host, the WebLogic Server home must have the same full path as the WebLogic Server home of the other Managed Servers.

For example, you have a Managed Server on Host A and a Managed Server on Host B, and you want to include them in a cluster. If the WebLogic Server home on Host A is located in /scratch/oracle/Middleware/wlserver_10.3, the WebLogic Server home on Host B must also be located in /scratch/oracle/Middleware/wlserver_10.3.

To create a cluster of two Managed Servers, soa_server1 and soa_server2, take the following steps:

1. From the Farm menu, choose **Create/Delete Components.**

   The Fusion Middleware Components page is displayed.

2. Choose **Create,** then **WebLogic Cluster.**

   The Create WebLogic Cluster page is displayed.

3. For **Name,** enter a name for the cluster.

4. In the Cluster Messaging Mode section, select one of the following:

   - **Unicast.** Then, for **Unicast Broadcast Channel,** enter a channel. This channel is used to transmit messages within the cluster.
   - **Multicast.** Then, for **Multicast Broadcast Channel,** enter a channel. A multicast address is an IP address in the range from 224.0.0.0 to 239.255.255.255. For **Multicast Port,** enter a port number.

   Note: You must ensure that the multicast address is not in use.

5. In the Servers section, select one or more servers to be added to the cluster. In this scenario, select soa_server1 and soa_server2.

6. Click **Create.**

Now, you have a cluster with two members, soa_server1 and soa_server2.

See Also: *Oracle Fusion Middleware Using Clusters for Oracle WebLogic Server* for more information about clusters

19.6 Copying a Middleware Home or Component

You can copy a Middleware home or many Oracle Fusion Middleware components to a different location while preserving its configuration. Some situations in which copying Oracle Fusion Middleware is useful are:

- Creating a Middleware home that is a copy of a production, test, or development environment, enabling you to create a new Middleware home or component with all patches applied to it in a single step. This is in contrast to separately installing,

configuring and applying any patches to separate Middleware homes or components.

- Preparing a "gold" image of a patched home and deploying it to many hosts.

For information about the scripts you use to copy a Middleware home or component, see Chapter 20.

20

Using the Movement Scripts

Oracle Fusion Middleware provides a series of scripts that you can use to move your environment, for example replicating (cloning) a test environment to a production environment. The scripts enable you to copy a Middleware home and the Oracle homes and Oracle WebLogic Server domains, as well as the configuration of certain Oracle Fusion Middleware components, such as Oracle SOA Suite, Oracle HTTP Server, Oracle Internet Directory, and Oracle Virtual Directory. This chapter explains the scripts you can use to move these entities.

This chapter includes the following topics:

> **Note:** For detailed procedures for moving Oracle Fusion Middleware from one environment to another, see Chapter 21. That chapter describes using these scripts, and other steps for moving from a source environment to a target environment.

20.1 Introduction to the Movement Scripts

The movement scripts minimize the amount of work that would otherwise be required to reapply all the customization and configuration changes made in one environment to another. You can use these scripts to:

- Create a Middleware home that is a copy of a production, test, or development environment. The scripts create a new Middleware home with all patches applied to all of the Oracle homes and the WebLogic Server home in a single step. This is in contrast to separately installing and applying any patches to the WebLogic Server home and separate Oracle homes.
- Prepare a "gold" image of a patched Middleware home and deploying it to many hosts.
- Move the configuration of a domain or Oracle instance, including the components in the domain or Oracle instance, from one environment to another.

You can move the following, to the same host or a different host. The source and target environments must share the same operating system and the same platform architecture (in terms of number of bits).

- Middleware home: You can copy the Middleware home, the Oracle WebLogic Server home, and all of the Oracle homes within the Middleware home. (You can copy a Middleware home that contains no Oracle homes, but you must use the cloningclient.jar file and movement scripts that are compatible with the version of the Middleware home that you want to copy.)
- Java components: You can copy the configuration of a domain containing Java components, such as Oracle SOA Suite and Oracle Business Activity Monitoring, to the same or a different Middleware home.
- An Oracle instance: You can copy the configuration of an Oracle instance to the same or a different Middleware home. When you move an Oracle instance, you move all configuration files in that instance for all system components in that Oracle instance.

  Alternatively, you can move one of the system components, such as Oracle HTTP Server, within an Oracle instance. In that case, the configuration of the Oracle instance and the specified component are moved. You can move the following system components in this way:

  – Oracle HTTP Server
  – Oracle Virtual Directory
  – Oracle Internet Directory
  – Oracle BI Enterprise Edition

  Note: The movement scripts support moving most of the Oracle Fusion Middleware components, but for some components, you must take manual steps in addition to, or instead of, using the scripts. See Chapter 21 for the procedures for moving Oracle Fusion Middleware components from a source to a target environment, including when to use the scripts.

Table 20–1 shows which Oracle Fusion Middleware components support the movement scripts and provides pointers to the procedures for moving each component.

Table 20–1 Support for Movement Scripts

Component	Supported?	Procedure Documented
Oracle Access Management Access Manager	Yes	Section 21.4.1.1
Oracle Access Management Identity Federation	Yes	Section 21.4.1.1
Oracle Access Management Mobile and Social	Yes	Section 21.4.1.1
Oracle Access Management Secure Token Service	Yes	Section 21.4.1.1
Oracle Adaptive Access Manager	Yes	Section 21.4.1.1
Oracle B2B	Yes	Section 21.4.2.1
Oracle Business Activity Monitoring	No	Section 21.4.2.1, Task 5
Oracle Business Intelligence	Yes	Section 21.4.7.1

Table 20–1 (Cont.) Support for Movement Scripts

Component	Supported?	Procedure Documented
Oracle Business Intelligence Discoverer	No	Section 21.4.9.1
Oracle Business Process Management	Yes	Section 21.4.2.1
Oracle Data Integrator	Yes	Section 21.4.10.1
Oracle Directory Integration Platform	Yes	Section 21.4.1.1
Oracle Enterprise Performance Management Workspace	No	Section 21.4.5
Oracle Entitlements Server	Yes	Section 21.4.1.1
Oracle Essbase	No	Section 21.4.5
Oracle Forms Services	No	Section 21.4.9.1
Oracle HTTP Server	Yes	Section 21.4.6.1.1
Oracle Human Workflow	Yes	Section 21.4.2.1
Oracle Hyperion Calculation Manager	No	Section 21.4.5
Oracle Hyperion Financial Reporting	No	Section 21.4.5
Oracle Hyperion Provider Service	No	Section 21.4.5
Oracle Hyperion Smart View for Office	No	Section 21.4.5
Oracle Identity Manager	No	Section 21.4.1.1
Oracle Identity Navigator	Yes	Section 21.4.1.1
Oracle Information Rights Management	No	Section 21.4.3.1, Task 3
Oracle Internet Directory	Yes	Section 21.4.1.1
Oracle Platform Security Services	Yes	Section 21.4.1.1
Oracle Portal	No	Section 21.4.9.1
Oracle Privileged Account Manager	Yes	Section 21.4.1.1
Oracle Real-Time Decisions	Yes	Section 21.4.8.1
Oracle Reports	No	Section 21.4.9.1
Oracle Service Bus	No	"Customization" in the *Oracle Fusion Middleware Administrator's Guide for Oracle Service Bus*.
Oracle SOA Suite	Yes	Section 21.4.2.1
Oracle Unified Directory	No	"Moving From a Test to a Production Environment" in the *Oracle Fusion Middleware Administrator's Guide for Oracle Unified Directory*.
Oracle User Messaging Service	No	Section 21.4.2.1, Task 7
Oracle Virtual Directory	Yes	Section 21.4.1.1

Table 20–1 (Cont.) Support for Movement Scripts

Component	Supported?	Procedure Documented
Oracle Web Cache	No	Section 21.4.6.1.2
Oracle Web Services Manager	Yes	Section 21.4.1.1
Oracle WebCenter Content	Yes	Section 21.4.4.1
Oracle WebCenter Content: Imaging	Yes	Section 21.4.3.1
Oracle WebCenter Content: Inbound Refinery	Yes	Section 21.4.3.1
Oracle WebCenter Content: Records	Yes	Section 21.4.3.1
Oracle WebCenter Portal	Yes	Section 21.4.3.1

20.2 Understanding the Movement Process

When you move an entity of Oracle Fusion Middleware, the scripts take a snapshot of the information required for the movement. The following topics describe the movement process:

- Understanding the Movement of a Middleware Home
- Understanding the Movement of Components

20.2.1 Understanding the Movement of a Middleware Home

When you move a Middleware home, you create an archive of the source Middleware home and use the archive to create the copy of the Middleware home:

1. At the source, you run the copyBinary script, specifying the Middleware home that you want to copy. The script prepares the source and creates an archive. It also records the file permissions of the Middleware home and the Oracle homes within the Middleware home.

   The archive contains the Oracle WebLogic Server home and all of the Oracle homes in the Middleware home.

2. At the destination, you run the pasteBinary script, specifying a destination for the Middleware home. The script checks to see that the prerequisites are met at the destination. It extracts the files from the archive file, registers the Oracle homes with the Oracle inventory and registers the WebLogic Server home with the Middleware home.

   The script then restores the file permissions and relinks any files if that is necessary.

Note the following:

- The copyBinary and pasteBinary scripts do not carry over all the dependencies of the source Middleware home, WebLogic Server home, and Oracle homes, such as loadable modules or application-specific libraries to the target home, because the scripts proceed by copying the Middleware home and the entire source WebLogic Server home and Oracle homes to the destination Middleware home. Any files outside the source WebLogic Server or Oracle home are not automatically copied. Hence, any applications that refer to files outside the source WebLogic Server or Oracle home may not work properly in the target home.

The Oracle home that is copied as a part of the Middleware home contains only the binary files.

- When you copy a Middleware home, only the read-only portions of the Middleware home are copied. Any user configuration files, such as the user_projects directory, are excluded from the archive. The WebLogic Server domain is not copied. (Use the copyConfig and pasteConfig scripts to copy the domain.)
- You cannot move a Middleware Home if its path is a symbolic link.

See Section 21.3.4 for detailed information about these steps.

20.2.2 Understanding the Movement of Components

When you move Oracle Fusion Middleware components, you create an archive of the source component's configuration and use the archive to create the component at the target. You use the following:

- For Node Manager, you use the copyConfig, extractMovePlan, and pasteConfig scripts to copy the configuration.
- For Java components, such as Oracle SOA Suite, you use the copyConfig, extractMovePlan, and pasteConfig scripts to copy the configuration, including the domain, the Administration Server, and the Managed Servers.
- For Oracle instances, you use the copyConfig, extractMovePlan, and pasteConfig scripts to copy the configuration of the Oracle instance, including all system components in the Oracle instance.

  Alternatively, you can specify that only one of the components, such as Oracle HTTP Server, within an Oracle instance be copied. In that case, the configuration of the Oracle instance and the specified component are moved.

> **Note:** The scripts replicate the topology of the source. For example, if the source domain contains Managed Servers server_1 and server_2 on Host A and Managed Servers server_3 and server_4 on Host B, you must specify a similar relationship between Managed Servers and hosts at the target. (You specify the hosts for each Managed Server in the move plan.)

To move components, you take the following general steps:

1. You move the Middleware home, as described in Section 20.2.1.
2. At the source, make sure that the Administration Server and all Managed Servers are started.
3. At the source, run the copyConfig script, specifying the source entity, such as a domain, Node Manager, or Oracle instance, that you want to copy. The script creates a configuration archive file that contains a snapshot of the configuration of an Oracle WebLogic Server domain, Node Manager, or system component instance.
4. At the source, extract a move plan using the extractMovePlan script. A **move plan** contains configuration settings of the source environment.
5. Edit the move plan specifying properties for the target environment.
6. At the target, run the pasteConfig script, specifying the destination for the domain, Node Manager, or Oracle instance, and the move plan location. The script checks

to see that the prerequisites are met at the target. It extracts the files from the archive file and uses the information in the move plan to modify the configuration on the target. Then, it restores the file permissions.

In addition, the pasteConfig scripts starts the Administration Server.

Note that you must ensure that components, such as Oracle WebLogic Server and Oracle Coherence, are installed in the directory structure of the source Middleware home.

See Section 21.3.6 and Section 21.3.7 for detailed information about these steps.

20.3 Movement Scripts

Oracle Fusion Middleware uses the following jar file to execute the scripts necessary to move the binary and configuration files:

```
(UNIX) ORACLE_COMMON_HOME/jlib/cloningclient.jar
(Windows) ORACLE_COMMON_HOME\jlib\cloningclient.jar
```

Table 20–2 shows the scripts you use to move a Middleware home or component.

Table 20–2 Movement Scripts

TO:	Script	See:
Copy the binary files of the source Middleware home	(UNIX) *ORACLE_COMMON_HOME*/bin/copyBinary.sh (Windows) *ORACLE_COMMON_HOME*\bin\copyBinary.cmd	Section 20.3.1.1
Apply the copied Middleware home to the target	(UNIX) *ORACLE_COMMON_HOME*/bin/pasteBinary.sh (Windows) *ORACLE_COMMON_HOME*\bin\pasteBinary.cmd	Section 20.3.1.2
Copy the domain and Java component configuration	(UNIX) *ORACLE_COMMON_HOME*/bin/copyConfig.sh (Windows) *ORACLE_COMMON_HOME*\bin\copyConfig.cmd	Section 20.3.1.3
Copy the Oracle instance configuration	(UNIX) *ORACLE_COMMON_HOME*/bin/copyConfig.sh (Windows) *ORACLE_COMMON_HOME*\bin\copyConfig.cmd	Section 20.3.1.4
Copy the system component configuration	(UNIX) *ORACLE_COMMON_HOME*/bin/copyConfig.sh (Windows) *ORACLE_COMMON_HOME*\bin\copyConfig.cmd	Section 20.3.1.5
Copy the Node Manager configuration	(UNIX) *ORACLE_COMMON_HOME*/bin/copyConfig.sh (Windows) *ORACLE_COMMON_HOME*\bin\copyConfig.cmd	Section 20.3.1.6
Extract a move plan from the domain or component	(UNIX) *ORACLE_COMMON_HOME*/bin/extractMovePlan.sh (Windows) *ORACLE_COMMON_HOME*\bin\extractMovePlan.cmd	Section 20.3.1.7
Apply the copied configuration for the domain and Java components to the target	(UNIX) *ORACLE_COMMON_HOME*/bin/pasteConfig.sh (Windows) *ORACLE_COMMON_HOME*\bin\pasteConfig.cmd	Section 20.3.1.8
Apply the copied configuration for the Oracle instance to the target	(UNIX) *ORACLE_COMMON_HOME*/bin/pasteConfig.sh (Windows) *ORACLE_COMMON_HOME*\bin\pasteConfig.cmd	Section 20.3.1.9
Apply the copied configuration for a system component to the target	(UNIX) *ORACLE_COMMON_HOME*/bin/pasteConfig.sh (Windows) *ORACLE_COMMON_HOME*\bin\pasteConfig.cmd	Section 20.3.1.10
Apply the copied configuration for the Node Manager to the target	(UNIX) *ORACLE_COMMON_HOME*/bin/pasteConfig.sh (Windows) *ORACLE_COMMON_HOME*\bin\pasteConfig.cmd	Section 20.3.1.11
Generate a file containing an obfuscated password	(UNIX) *ORACLE_COMMON_HOME*/bin/obfuscatePassword.sh (Windows) *ORACLE_COMMON_HOME*\bin\obfuscatePassword.cmd	Section 20.3.1.12

To view the help on any of these scripts, use the -help option. For example:

```
./pasteConfig.sh -javaHome /scratch/Oracle/Middleware/jdk160_21 -help
```

Note that the help shows the UNIX version of the parameter values. For other platforms, such as Windows, change the parameter values for the platform.

To specify additional Java options, define the T2P_JAVA_OPTIONS environment variable and specify the options in the variable definition. The following examples set the value for the Java temp directory:

- On Linux or UNIX:

```
setenv T2P_JAVA_OPTIONS "-Djava.io.tmpdir=/home/t2p/temp"
export T2P_JAVA_OPTIONS
```

- On Windows:

```
set T2P_JAVA_OPTIONS="-Djava.io.tmpdir=c:\home\t2p\temp"
```

Note: A Universal Uniform Naming Convention (UNC) path is not supported on Windows. For example, the following is not supported:

```
\\host_name\oracle\java\win64\jdk6\jre\bin\java
```

Note: If you are applying the archive of a Middleware home on a host that does not yet have Oracle Fusion Middleware installed, note the following:

- The host must have JDK 1.6.04 or higher installed. In addition, ensure that the PATH, CLASSPATH, and JAVA_HOME environment variables point to the JDK.
- Copy the pasteBinary script from the following location in the source host to the target host:

  (UNIX) *ORACLE_COMMON_HOME*/bin/pasteBinary.sh
  (Windows) *ORACLE_COMMON_HOME*\bin\pasteBinary.cmd

- Copy the following file from the following location in the source host to the target host:

  (UNIX) *ORACLE_COMMON_HOME*/jlib/cloningclient.jar
  (Windows) *ORACLE_COMMON_HOME*\jlib\cloningclient.jar

- If you run the pasteBinary script from a different location than *ORACLE_COMMON_HOME*/bin, then the pasteBinary script and the cloningclient.jar file must be in the same directory.

  If you are running pasteBinary on a host that has no prior Oracle Fusion Middleware installations, ORACLE_COMMON_home/bin will not exist prior to running pasteBinary, and therefore the pasteBinary script and cloningclient.jar must be in the same directory.

- Ensure that the files have execute permission.

20.3.1 Movement Scripts Syntax

The following topics describe the syntax of the movement scripts. The options are described in the tables that follow the syntax.

- copyBinary Script
- pasteBinary Script
- copyConfig Script for Java Components
- copyConfig Script for Oracle Instances
- copyConfig Script for System Components
- copyConfig Script for Node Manager
- extractMovePlan Script
- pasteConfig Script for Java Components
- pasteConfig Script for Oracle Instances
- pasteConfig Script for System Components
- pasteConfig Script for Node Manager
- obfuscatePassword Script

Note:

- All movement scripts ask if you want to continue whenever you do not specify the `-silent true` option. To continue, you must type `yes`, which is not case sensitive. Any words other than `yes` causes the script to return an error. Also note that, in silent mode, the scripts generate an error if you do not provide passwords where they are needed.
- Most options have shortcut names, as described in the tables later in the following sections.
- The value of options must not contain a space. For example, on Windows, you cannot pass the following as a value to the -javaHome option:

  ```
  C:\Program Files\jdk
  ```

- If any argument passed to a movement script contains an equals sign (=), the value must be enclosed in double quotation marks ("). For example:

  ```
  -additionalParam "search.encrypt.key=C:\T2P\encrypt.txt"
  ```

- The value of the javaHome option must use the Java home that is defined in the following file (note the period (.) before the filename):

  ```
  MW_HOME/wlserver_n/.product.properties
  ```

20.3.1.1 copyBinary Script

Creates an archive file of the source Middleware home, by copying the binary files of that Middleware home, including all of its Oracle homes and its WebLogic Server home, into the archive file.

The copyBinary script is located in:

```
(UNIX) ORACLE_COMMON_HOME/bin/copyBinary.sh
(Windows) ORACLE_COMMON_HOME\bin\copyBinary.cmd
```

The syntax is:

```
copyBinary -javaHome path_of_jdk
           -archiveLoc archive_location
           -sourceMWHomeLoc MW_HOME
            [-invPtrLoc Oracle_InventoryLocation]
            [-logDirLoc log_dir_path]
            [-silent {true | false}]
            [-ignoreDiskWarning {true | false}]
```

The following example shows how to create an archive of a Middleware home on Linux:

```
copyBinary.sh -javaHome /scratch/Oracle/Middleware1/jrockit_160_20_D1.1.0-18
              -archiveLoc /tmp/mw_copy.jar
              -sourceMWHomeLoc /scratch/Oracle/Middleware1
              -invPtrLoc /scratch/oracle/oraInst.loc
```

Note: Before you execute the copyBinary script, ensure that all Oracle homes in the Middleware home are either all 32 bit or all 64 bit. The operation does not support a mix of 32-bit and 64-bit Oracle homes.

When you execute the script, you must specify a matching Java home. That is, if the Oracle homes are 64 bit, you must specify a 64-bit Java home. If the Oracle homes are 32 bit, you must specify a 32-bit Java home.

Table 20–3 describes the options for the copyBinary script.

Table 20–3 Options for the copyBinary Script

Options	Shortcut	Description	Mandatory or Optional
-javaHome	NA	The absolute path of the Java Developer's Kit. If the operating system is SunOS, HP-UX, or Linux 64 bit, pass the -d64 option to the scripts in the command line. To set the runtime property, you can specify the -d64 option in the T2P_JAVA_OPTIONS environment variable. For example: `setenv T2P_JAVA_OPTIONS "-d64 -Djava.io.tmpdir=/home/t2p/temp"`	Mandatory
-archiveLoc	-al	The absolute path of the archive location. Use this option to specify the location of the archive file to be created with the copyBinary script. The archive location must not exist, but its parent directory must exist and have write permission. Ensure that the archive location is not within the Middleware home structure.	Mandatory
-sourceMWHomeLoc	-smw	The absolute path of the Middleware home to be archived. You can only specify one Middleware home.	Mandatory

Table 20–3 (Cont.) Options for the copyBinary Script

Options	Shortcut	Description	Mandatory or Optional
-invPtrLoc	-ipl	On UNIX and Linux, the absolute path to the Oracle Inventory pointer. Use this option if the inventory location is not in the default location, so that the operation can read the Oracle homes specified in the inventory. The file, oraInst.loc, must exist. If it does not exist, create it as a root user or user with normal privileges. The following shows an example of the contents of the file: `inventory_loc=/scratch/oraInventory` You must have write permission to the inventory location. On UNIX and Linux, the default location is /etc/oraInst.loc. On Windows, if you specify this parameter, the script ignores it. In previous releases, the shortcut was -invLoc, but that shortcut is now deprecated.	Optional, if the inventory is in the default location. Otherwise, it is mandatory on Linux.
-logDirLoc	-ldl	The location of an existing directory. A new log file is created in the directory. The default is the system Temp location.	Optional
-silent	NA	Specifies whether the operation operates silently. That is, it does not prompt for confirmation. The default is that the operation prompts for confirmation. To continue, you must type `yes`, which is not case sensitive. Typing anything other than yes causes the script to return an error. To specify that it does not prompt for confirmation, specify this option with the value of `true`.	Optional
-ignoreDiskWarning	-idw	Specifies whether the operation ignores a warning that there is not enough free space available. The default is false. You may need to use this flag if the target is NFS mounted or is on a different file system, such as Data ONTAP.	Optional

20.3.1.2 pasteBinary Script

Applies the archive to the target destination, by pasting the binary files of the source Middleware home to the target environment. You can apply the archive to the same host or a different host.

The pasteBinary script is located in:

```
(UNIX) ORACLE_COMMON_HOME/bin/pasteBinary.sh
(Windows) ORACLE_COMMON_HOME\bin\pasteBinary.cmd
```

The syntax is:

```
pasteBinary -javaHome path_of_jdk
            -archiveLoc archive_location
            -targetMWHomeLoc target_MW_Home_location
            [-executeSysPrereqs {true | false}]
            [-invPtrLoc Oracle_InventoryLocation]
            [-logDirLoc log_dir_path]
            [-silent {true | false}]
            [-ignoreDiskWarning {true | false}]
```

The following example shows how to apply the archive to the directory /scratch/oracle/MW_Home_prod, on Linux:

```
pasteBinary.sh -javaHome /scratch/Oracle/Middleware1/jrockit_160_20_D1.1.0-18
               -archiveLoc  /tmp/mw_copy.jar
               -targetMWHomeLoc /scratch/oracle/MW_Home_prod
```

Table 20–4 describes the options for the pasteBinary script.

Table 20–4 Options for the pasteBinary Script

Options	Shortcut	Description	Mandatory or Optional
-javaHome	NA	The absolute path of the Java Developer's Kit. If the source Middleware home was installed with the JDK and Oracle JRockit outside of the Middleware home, the path you specify is used to configure the Middleware home. If the operating system is SunOS, HP-UX, or Linux 64 bit, pass the -d64 option to the scripts in the command line. To set the runtime property, you can specify the -d64 option in the T2P_JAVA_OPTIONS environment variable. For example: `setenv T2P_JAVA_OPTIONS "-d64 -Djava.io.tmpdir=/home/t2p/temp"`	Mandatory
-archiveLoc	-al	The absolute path of the archive location. Use this option to specify the location of the archive file created with the copyBinary script. The location must exist. In previous releases, this option was named archiveLocation, but that name is now deprecated.	Mandatory
-targetMWHomeLoc	-tmw	The absolute path of the target Middleware home. Ensure that the Middleware home directory does not exist at that location. If it does exist, the script returns an error message. The targetMWHomeLoc cannot be inside another Middleware home. In previous releases, this option was named targetLocation and the shortcut was -tl, but those are now deprecated.	Mandatory
-executeSysPrereqs	-esp	Specifies whether the pasteBinary operation checks the prerequisites of the Oracle homes. The default is that it checks the prerequisites. To specify that it does not check the prerequisites, specify this option with the value `false`. In previous releases, the shortcut was -exsysprereqs, but that shortcut is now deprecated.	Optional

Table 20–4 (Cont.) Options for the pasteBinary Script

Options	Shortcut	Description	Mandatory or Optional
-invPtrLoc	-ipl	On UNIX and Linux, the absolute path to the Oracle Inventory pointer. Use this option if the inventory location is not in the default location, so that the operation can read the Oracle homes specified in the inventory. The file, oraInst.loc, must exist. If it does not exist, create it as a root user or user with normal privileges. The following shows an example of the contents of the file: `inventory_loc=/scratch/oraInventory` You must have write permission to the inventory location. On UNIX and Linux, the default location is /etc/oraInst.loc. On Windows, if you specify this parameter, the script ignores it. In previous releases, the shortcut was -invLoc, but that shortcut is now deprecated.	Optional, if the inventory is in the default location. Otherwise, it is mandatory on Linux.
-logDirLoc	-ldl	The location of an existing directory. A new log file is created in the directory. The default is the system Temp location.	Optional
-silent	NA	Specifies whether the operation operates silently. That is, it does not prompt for confirmation. The default is that the operation prompts for confirmation. To continue, you must type `yes`, which is not case sensitive. Typing anything other than yes causes the script to return an error. To specify that it does not prompt for confirmation, specify this option with the value of `true`.	Optional
-ignoreDiskWarning	-idw	Specifies whether the operation ignores a warning that there is not enough free space available. The default is false. You may need to use this flag if the target is NFS mounted or is on a different file system, such as Data ONTAP.	Optional

20.3.1.3 copyConfig Script for Java Components

Creates a configuration archive that contains the snapshot of the configuration of an Oracle WebLogic Server domain. The underlying components of an Oracle WebLogic Server domain retain their configuration information in different data stores, such as a file system, Oracle Metadata Services (MDS), LDAP, or a database.

You must run the copyConfig script for each Oracle WebLogic Server domain in the source environment. A configuration archive is created for each source domain.

The Administration Server and all Managed Servers in the domain must be started when you run the script.

The copyConfig script is located in:

```
(UNIX) ORACLE_COMMON_HOME/bin/copyConfig.sh
(Windows) ORACLE_COMMON_HOME\bin\copyConfig.cmd
```

The syntax is:

```
copyConfig  -javaHome path_of_jdk
            -archiveLoc archive_location
            -sourceDomainLoc domain_location
            -sourceMWHomeLoc Middleware_home_location
            -domainHostName domain_host_name
```

```
-domainPortNum  domain_port_number
-domainAdminUserName domain_admin_username
-domainAdminPassword domain_admin_password_file
[-mdsDataImport {true | false}]
[-additionalParams property1=value1[, property2=value2]
[-logDirLoc log_dir_path]
[-silent {true | false}]
```

The following example copies the configuration of a domain containing Java components:

```
copyConfig.sh -javaHome /scratch/jrockit_160_20_D1.1.0-18
              -archiveLoc /tmp/a.jar
              -sourceDomainLoc /scratch/mw_home1/user_projects/domains/WLS_SOAWC
              -sourceMWHomeLoc /scratch/work/mw_home1/
              -domainHostName myhost.example.com
              -domainPortNum 7001
              -domainAdminUserName weblogic
              -domainAdminPassword /home/oracle/password
              -silent true
```

Table 20–5 describes the options for the copyConfig script for Java components.

Table 20–5 Options for the copyConfig Script for Java Components

Options	Shortcut	Description	Mandatory or Optional
-javaHome	NA	The absolute path of the Java Developer's Kit. If the operating system is SunOS, HP-UX, or Linux 64 bit, pass the -d64 option to the scripts in the command line. To set the runtime property, you can specify the -d64 option in the T2P_JAVA_OPTIONS environment variable. For example: `setenv T2P_JAVA_OPTIONS "-d64 -Djava.io.tmpdir=/home/t2p/temp"`	Mandatory
-archiveLoc	-al	The absolute path of the archive location. Use this option to specify the location of the archive file to be created by the copyConfig script.	Mandatory
-sourceDomainLoc	-sdl	The absolute path of the source domain containing the Java component. Note that on Windows, you should not include a backslash at the end of the path.	Mandatory
-sourceMWHomeLoc	-smw	The absolute path of the source Middleware home.	Mandatory
-domainHostName	-dhn	The name of the host on which the domain is configured.	Mandatory
-domainPortNum	-dpn	The port number of the Administration Server for the domain. In previous releases, this option was named domainPortNo, and the shortcut was -domainport, but those are now deprecated.	Mandatory
-domainAdminUserName	-dau	The name of the administrative user for the domain. In previous releases, the shortcut was -domainuser, but that shortcut is now deprecated.	Mandatory

Table 20–5 (Cont.) Options for the copyConfig Script for Java Components

Options	Shortcut	Description	Mandatory or Optional
-domainAdminPassword	-dap	The absolute path of a secure file containing the password for the administrative user for the domain on the source environment. You must provide a password file, even if you are not changing the configuration. In previous releases, the shortcut was -domainpass, but that shortcut is now deprecated.	Mandatory
-mdsDataImport	-mdi	Specifies whether to export the application MDS metadata to the archive so that it can be imported into the target. The default is true. Specify false if you do not want to export the application MDS metadata. If this option is set to true, the subsequent pasteConfig script that copies the component to the target imports the application MDS metadata to the target.	Optional
-logDirLoc	-ldl	The location of an existing directory. A new log file is created in the directory. The default is the system Temp location.	Optional
-additionalParams	-ap	An additional parameter and its value to be passed to the script.	Optional
-silent	NA	Specifies whether the operation operates silently. That is, it does not prompt for confirmation. The default is that the operation prompts for confirmation. To continue, you must type `yes`, which is not case sensitive. Typing anything other than yes causes the script to return an error. To specify that it does not prompt for confirmation, specify this option with the value of `true`.	Optional

20.3.1.4 copyConfig Script for Oracle Instances

Creates a configuration archive that contains the snapshot of the configuration of an Oracle instance. The copyConfig script moves the Oracle instance and the configuration of all the system components in the Oracle instance.

You must run the copyConfig script for each Oracle instance in the source environment. A configuration archive is created for each Oracle instance.

The Administration Server and all Managed Servers in the domain must be started when you run the script.

The copyConfig script is located in:

```
(UNIX) ORACLE_COMMON_HOME/bin/copyConfig.sh
(Windows) ORACLE_COMMON_HOME\bin\copyConfig.cmd
```

The syntax is:

```
copyConfig  -javaHome path_of_jdk
            -archiveLoc archive_location
            -sourceInstanceHomeLoc src_instance_path
            [-logDirLoc log_dir_path]
            [-silent {true | false}]
```

The following example shows how to create an archive of the Oracle instance located in /scratch/Oracle/Middleware/im_1 on Linux:

```
copyConfig.sh -javaHome /scratch/Oracle/Middleware1/jrockit_160_20_D1.1.0-18
              -archiveLoc /tmp/ovd1.jar
              -sourceInstanceHomeLoc /scratch/Oracle/Middleware1/im_1
```

Table 20–6 describes the options for the copyConfig script for Oracle instances. It also describes the options for the copyConfig Script for individual system components. the only difference is that you use the -sourceComponentName option to move individual system components.

Table 20–6 Options for the copyConfig Script for Oracle Instances and System Components

Options	Shortcut	Description	Mandatory or Optional
-javaHome	NA	The absolute path of the Java Developer's Kit. If the operating system is SunOS, HP-UX, or Linux 64 bit, pass the -d64 option to the scripts in the command line. To set the runtime property, you can specify the -d64 option in the T2P_JAVA_OPTIONS environment variable. For example: `setenv T2P_JAVA_OPTIONS "-d64 -Djava.io.tmpdir=/home/t2p/temp"`	Mandatory
-archiveLoc	-al	The absolute path of the archive location. Use this option to specify the location of the archive file to be created by the copyConfig script. In previous releases, this option was named archiveLocation, but that name is now deprecated.	Mandatory
-sourceInstanceHomeLoc	-sih	The absolute path of the Oracle instance for the source component.	Mandatory
-logDirLoc	-ldl	The location of an existing directory. A new log file is created in the directory. The default is the system Temp location.	Optional
-silent	NA	Specifies whether the operation operates silently. That is, it does not prompt for confirmation. The default is that the operation prompts for confirmation. To continue, you must type `yes`, which is not case sensitive. Typing anything other than yes causes the script to return an error. To specify that it does not prompt for confirmation, specify this option with the value of `true`.	Optional
-sourceComponentName	-scn	The name of the component to be copied. For example, if your Oracle Internet Directory component is named oid1, specify oid1.	Optional. Use if you want to move only one system component, as described in Section 20.3.1.5.

20.3.1.5 copyConfig Script for System Components

Creates a configuration archive that contains the snapshot of the configuration of an Oracle instance and the specified individual system component, which retains its configuration information in different data stores, such as a file system, Oracle Metadata Services (MDS), LDAP, or a database.

Use this script instead of copyConfig for Oracle instances, if you want to move only one system component, along with the Oracle instance to the target environment.

The copyConfig script supports moving the following system components:

- Oracle HTTP Server
- Oracle Internet Directory
- Oracle Virtual Directory
- Oracle BI EE

The Administration Server and all Managed Servers in the domain must be started when you run the script.

The copyConfig script is located in:

```
(UNIX) ORACLE_COMMON_HOME/bin/copyConfig.sh
(Windows) ORACLE_COMMON_HOME\bin\copyConfig.cmd
```

The syntax is:

```
copyConfig  -javaHome path_of_jdk
            -archiveLoc archive_location
            -sourceInstanceHomeLoc src_instance_path
            -sourceComponentName src_component_name
            [-logDirLoc log_dir_path]
            [-silent {true | false}]
```

The following example shows how to create an archive of the Oracle Virtual Directory instance named ovd1 in the Oracle instance located in /scratch/Oracle/Middleware/im_1 on Linux:

```
copyConfig.sh -javaHome /scratch/Oracle/Middleware1/jrockit_160_20_D1.1.0-18
              -archiveLoc /tmp/ovd1.jar
              -sourceInstanceHomeLoc /scratch/Oracle/Middleware1/im_1
              -sourceComponentName ovd1
```

See Table 20–6 for description of the options for the copyConfig script for system components.

20.3.1.6 copyConfig Script for Node Manager

Creates a configuration archive that contains the snapshot of the configuration of Node Manager.

You must run the copyConfig script for each Node Manager in the source environment. A configuration archive is created for each source Node Manager.

The copyConfig script is located in:

```
(UNIX) ORACLE_COMMON_HOME/bin/copyConfig.sh
(Windows) ORACLE_COMMON_HOME\bin\copyConfig.cmd
```

The syntax is:

```
copyConfig  -javaHome path_of_jdk
            -archiveLoc archive_location
            -sourceNMHomeLoc source_Node_Manager_Home_location
            [-logDirLoc log_dir_path]
            [-silent {true | false}]
```

The following example shows how to create a copy of the source Node Manager configuration located in /scratch/Oracle/Middleware/wlserver_10.3/common/nodemanager:

```
copyConfig.sh -javaHome USER_HOME/jrockit_160_17_R28.0.0-679/
```

```
                -archiveLoc /tmp/nm.jar
                -sourceNMHomeLoc /scratch/Oracle/Middleware/wlserver_
10.3/common/nodemanager
                -silent true
```

Table 20–7 describes the options for the copyConfig script for Node Manager.

Table 20–7 Options for the copyConfig Script for Node Manager

Options	Shortcut	Description	Mandatory or Optional
`-javaHome`	None	The absolute path of the Java Developer's Kit. If the operating system is SunOS, HP-UX, or Linux 64 bit, pass the -d64 option to the scripts in the command line. To set the runtime property, you can specify the -d64 option in the T2P_JAVA_OPTIONS environment variable. For example: `setenv T2P_JAVA_OPTIONS "-d64 -Djava.io.tmpdir=/home/t2p/temp"`	Mandatory
`-archiveLoc`	`-al`	The absolute path of the archive location. Use this option to specify the location of the archive file to be created by the copyConfig script.	Mandatory
`-sourceNMHomeLoc`	`-snh`	The absolute path of the source Node Manager home.	Mandatory
`-logDirLoc`	`-ldl`	The location of an existing directory. A new log file is created in the directory. The default is the system Temp location.	Optional
`-silent`	None	Specifies whether the operation operates silently. That is, it does not prompt for confirmation. The default is that the operation prompts for confirmation. To continue, you must type `yes`, which is not case sensitive. Typing anything other than yes causes the script to return an error. To specify that it does not prompt for confirmation, specify this option with the value of `true`.	Optional

20.3.1.7 extractMovePlan Script

Extracts configuration information from the archive into a move plan. It also extracts any needed configuration plans. Then, you edit the move plan, specifying properties for the target environment.

The extractMovePlan script is located in:

(UNIX) *ORACLE_COMMON_HOME*/bin/extractMovePlan.sh
(Windows) *ORACLE_COMMON_HOME*\bin\extractMovePlan.cmd

The syntax is:

```
extractMovePlan -javaHome path_of_jdk
                -archiveLoc archive_location
                -planDirLoc move_plan_directory
                [-logDirLoc log_dir_path]
```

The following example extracts the plans from the archive j2ee.jar:

```
extractMovePlan.sh -javaHome /scratch/Oracle/Middleware1/jrockit_160_20_D1.1.0-18
                   -archiveLoc /tmp/j2ee.jar
                   -planDirLoc /scratch/Oracle/t2p_plans
```

The extractMovePlan script extracts the move plan to the specified directory. Depending on the type of component that you are moving, the extractMovePlan script may also extract other configuration plans.

- For Java components, such as Oracle SOA Suite, it may extract the following:

```
/scratch/Oracle/t2p_plans/moveplan.xml
/scratch/Oracle/t2p_plans/composites/configplan1.xml
/scratch/Oracle/t2p_plans/composites/configplan2.xml
/scratch/Oracle/t2p_plans/adapters/deploymentplan1.xml
/scratch/Oracle/t2p_plans/adapters/deploymentplan2.xml
```

- For system components, such as Oracle Internet Directory and Oracle Virtual Directory, it may extract the following:

```
/scratch/Oracle/t2p_plans/moveplan.xml
```

Table 20–8 describes the options for the extractMovePlan script:

Table 20–8 Options for the extractMovePlan Script

Options	Shortcut	Description	Mandatory or Optional
-javaHome	NA	The absolute path of the Java Developer's Kit. If the operating system is SunOS, HP-UX, or Linux 64 bit, pass the -d64 option to the scripts in the command line. To set the runtime property, you can specify the -d64 option in the T2P_JAVA_OPTIONS environment variable. For example: `setenv T2P_JAVA_OPTIONS "-d64 -Djava.io.tmpdir=/home/t2p/temp"`	Mandatory
-archiveLoc	-al	The absolute path of the archive location. Use this option to specify the location of the archive file created by the copyConfig script.	Mandatory
-planDirLoc	-pdl	The absolute path to a directory to which the move plan, along with any needed configuration plans, is to be extracted. The directory must not exist.	Mandatory
-logDirLoc	-ldl	The location of an existing directory. A new log file is created in the directory. The default is the system Temp location.	Optional

For information about the properties in the move plans, and which properties you should edit, see Section 20.4.

20.3.1.8 pasteConfig Script for Java Components

Applies the copied configurations from the source environment to the target environment. Inputs for the script include the location of the configuration archive created with the copyConfig script for the Java components and the location of the modified move plan. The pasteConfig script re-creates the configuration information for the Oracle WebLogic Server domain in the target environment. It also merges the move plan property values for the target environment.

The pasteConfig script is located in:

```
(UNIX) ORACLE_COMMON_HOME/bin/pasteConfig.sh
(Windows) ORACLE_COMMON_HOME\bin\pasteConfig.cmd
```

The syntax is:

```
pasteConfig  -javaHome path_of_jdk
             -archiveLoc archive_location
             -targetDomainLoc trgt_domain_path
             -targetMWHomeLoc trgt_Middleware_Home_path
             -movePlanLoc move_plan_path
             -domainAdminPassword domain_admin_password_file
             [-appDir WLS_application_directory]
             [-logDirLoc log_dir_path]
             [-silent {true | false}]
```

The following example shows how to apply the archive of the domain to the Middleware home MW_home1:

```
pasteConfig.sh -javaHome /scratch/Oracle/Middleware1/jrockit_160_20_D1.1.0-18
           -archiveLoc /tmp/java_ee_c1.jar
           -targetDomainLoc /scratch/oracle/MW_home1/user_projects/domains/dom_c1
           -targetMWHomeLoc /scratch/oracle/MW_home1
           -movePlanLoc /scratch/oracle/java_ee/move_plan.xml
           -domainAdminPassword /scratch/pwd_dir/pass
           -logDirLoc /tmp/log
```

Table 20–9 describes the options for the pasteConfig script for Java components.

Table 20–9 Options for the pasteConfig Script for Java Components

Options	Shortcut	Description	Mandatory or Optional
-javaHome	NA	The absolute path of the Java Developer's Kit. If the operating system is SunOS, HP-UX, or Linux 64 bit, pass the -d64 option to the scripts in the command line. To set the runtime property, you can specify the -d64 option in the T2P_JAVA_OPTIONS environment variable. For example: `setenv T2P_JAVA_OPTIONS "-d64 -Djava.io.tmpdir=/home/t2p/temp"`	Mandatory
-archiveLoc	-al	The absolute path of the archive location. Use this option to specify the location of the archive file created by the copyConfig script.	Mandatory
-targetDomainLoc	-tdl	The absolute path of the target domain. The domain location must not exist for the specified Middleware home. The domain directory may be located outside of the directory structure of the Middleware home.	Mandatory
-targetMWHomeLoc	-tmw	The absolute path of the target Middleware home in which the domain is to be copied.	Mandatory
-movePlanLoc	-mpl	The absolute path of the move plan extracted from the source.	Mandatory
-domainAdminPassword	-dap	The absolute path of a secure file containing the password for the administrative user for the domain on target environment. You must provide a password file, even if you are not changing the configuration. Note that the password is based on the authentication provider for the domain. For example, the authenticator can be an embedded LDAP or an external LDAP. In previous releases, the shortcut was -domainpass, but that shortcut is now deprecated.	Mandatory.

Table 20–9 (Cont.) Options for the pasteConfig Script for Java Components

Options	Shortcut	Description	Mandatory or Optional
-appDir	-ad	The absolute path of the Oracle WebLogic Server application directory on the target.	Optional
-logDirLoc	-ldl	The location of an existing directory. A new log file is created in the directory. The default is the system Temp location.	Optional
-silent	NA	Specifies whether the operation operates silently. That is, it does not prompt for confirmation. The default is that the operation prompts for confirmation. To continue, you must type `yes`, which is not case sensitive. Typing anything other than yes causes the script to return an error. To specify that it does not prompt for confirmation, specify this option with the value of `true`.	Optional

20.3.1.9 pasteConfig Script for Oracle Instances

Applies the copied configurations from the source environment to the target environment. Inputs for the script include the location of the configuration archive created with the copyConfig script for the Oracle instance and the location of the modified move plan. The pasteConfig script iterates and re-creates the configuration information for the Oracle instance and all of its system components in the target environment. It also merges the move plan property values for the target environment.

The syntax is:

```
pasteConfig  -javaHome path_of_jdk
             -archiveLoc archive_location
             -movePlanLoc move_plan_path
             -targetInstanceHomeLoc trgt_Instance_path
             [-targetInstanceName trgt_Instance_name]
             [-targetOracleHomeLoc trgt_ORACLE_HOME_path]
             [-logDirLoc log_dir_path]
             [-silent {true | false}]
             [ <Domain Detail> ]

<Domain Detail> =
         -domainHostName domain_host_name
         -domainPortNum  domain_port_number
         -domainAdminUserName domain_admin_username
         -domainAdminPassword domain_admin_password_file
```

The following example shows how to apply the archive to the target and name the Oracle instance im_2:

```
pasteConfig.sh -javaHome /scratch/Oracle/Middleware/jrockit_160_20_D1.1.0-18
            -archiveLoc /tmp/ovd1.jar
            -movePlanLoc /scratch/oracle/ovd/move_plan.xml
            -targetOracleHomeLoc /scratch/Oracle/Middleware/Oracle_IM2
            -targetInstanceHomeLoc /scratch/Oracle/Middleware/im_2
            -targetInstanceName im_2
            -domainHostName myhost
            -domainPortNum 7001
            -domainAdminUserName domain_admin_username
            -domainAdminPassword domain_admin_password_file
```

Table 20–10 describes the options for the pasteConfig script for system components. It also describes the options for the pasteConfig Script for individual system

components. The only difference is that you use the -sourceComponentName option to move individual system components.

Table 20–10 Options for the pasteConfig Script for Oracle Instances

Options	Shortcut	Description	Mandatory or Optional
-javaHome	NA	The absolute path of the Java Developer's Kit. If the operating system is SunOS, HP-UX, or Linux 64 bit, pass the -d64 option to the scripts in the command line. To set the runtime property, you can specify the -d64 option in the T2P_JAVA_OPTIONS environment variable. For example: `setenv T2P_JAVA_OPTIONS "-d64 -Djava.io.tmpdir=/home/t2p/temp"`	Mandatory
-archiveLoc	-al	The absolute path of the archive location. Use this option to specify the location of the archive file created by the copyConfig script.	Mandatory
-movePlanLoc	-mpl	The absolute path of the move plan extracted from the source.	Mandatory
-targetInstanceHomeLoc	-tih	The absolute path of the target Oracle instance. If the Oracle instance directory does not exist at that location, the script creates the directory.	Mandatory
-targetInstanceName	-tin	The name of the target Oracle instance, which is used to register the instance with the domain. The name must be unique in the domain.	Optional, if the targetInstanceHomeLoc directory exists. In this case, the operation retrieves the name from the configuration.
-targetComponentName	-tcn	The name of the target component to be copied. The name must be unique in the instance.	Optional. Use if you want to move only one system component, as described in Section 20.3.1.5.
-targetOracleHomeLoc	-toh	The absolute path of the target Oracle home. The target Oracle home must exist and it must contain the binaries for the component you are copying.	Optional, if the targetInstanceHomeLoc exists. In this case, the operation retrieves the value from the configuration.
-logDirLoc	-ldl	The location of an existing directory. A new log file is created in the directory. The default is the system Temp location.	Optional
-silent	NA	Specifies whether the operation operates silently. That is, it does not prompt for confirmation. The default is that the operation prompts for confirmation. To continue, you must type `yes`, which is not case sensitive. Typing anything other than yes causes the script to return an error. To specify that it does not prompt for confirmation, specify this option with the value of `true`.	Optional
Domain-Detail Options			

Table 20–10 (Cont.) Options for the pasteConfig Script for Oracle Instances

Options	Shortcut	Description	Mandatory or Optional
-domainHostName	-dhn	The name of the host on which the domain is configured. Use this option if you want to register the component with the domain. In previous releases, the shortcut was -domainhost, but that shortcut is now deprecated.	Optional, if you do not want to register the component with the domain.
-domainPortNum	-dpn	The port number of the domain. Use this option if you want to register the component with the domain. The domain port number is listed in the following file as the adminPort: *ORACLE_INSTANCE*/config/OPMN/opmn/instance.properties For example: adminPort=7001	Optional, if you do not want to register the component with the domain.
-domainAdminUserName	-dau	The name of the administrative user for the domain. Use this option if you want to register the component with the domain.	Optional, if you do not want to register the component with the domain.
-domainAdminPassword	-dap	The absolute path of a secure file containing the password for the administrative user for the domain. You must provide a password file, even if you are not changing the configuration. Use this option if you want to register the component with the domain.	Optional, if you do not want to register the component with the domain.

20.3.1.10 pasteConfig Script for System Components

Applies the copied configurations for individual system components from the source environment to the target environment. Inputs for the script include the location of the configuration archive created with the copyConfig script for the Oracle instance and the location of the modified move plan. The pasteConfig script re-creates the configuration information for the Oracle instance and the specified system component in the target environment. It also merges the move plan property values for the target environment.

The copyConfig script supports moving the following system components:

- Oracle HTTP Server
- Oracle Internet Directory
- Oracle Virtual Directory
- Oracle BI EE

The syntax is:

```
pasteConfig  -javaHome path_of_jdk
             -archiveLoc archive_location
             -movePlanLoc move_plan_path
             -targetComponentName trgt_component_name
             -targetInstanceHomeLoc trgt_Instance_path
             [-targetInstanceName trgt_Instance_name]
             [-targetOracleHomeLoc trgt_ORACLE_HOME_path]
             [-logDirLoc log_dir_path]
```

```
            [-silent {true | false}]
            [ <Domain Detail> ]

<Domain Detail> =
         -domainHostName domain_host_name
         -domainPortNum  domain_port_number
         -domainAdminUserName domain_admin_username
         -domainAdminPassword domain_admin_password_file
```

The following example shows how to apply the archive to the Oracle instance im_2 and to name the target Oracle Virtual Directory instance ovd_cl:

```
pasteConfig.sh -javaHome /scratch/Oracle/Middleware/jrockit_160_20_D1.1.0-18
            -archiveLoc /tmp/ovd1.jar
            -movePlanLoc /scratch/oracle/ovd/move_plan.xml
            -targetOracleHomeLoc /scratch/Oracle/Middleware/Oracle_IM2
            -targetInstanceHomeLoc /scratch/Oracle/Middleware/im_2
            -targetInstanceName im_2
            -targetComponentName ovd_cl
            -domainHostName myhost
            -domainPortNum 7001
            -domainAdminUserName domain_admin_username
            -domainAdminPassword domain_admin_password_file
```

See Table 20–10 for descriptions of the options for the pasteConfig script for system components.

20.3.1.11 pasteConfig Script for Node Manager

Applies the copied configurations of Node Manager from the source environment to the target environment. Inputs for the script include the location of the configuration archive created with the copyConfig script for the Oracle WebLogic Server Node Manger and the location of the modified move plan. The pasteConfig script re-creates the configuration information for Node Manager in the target environment. It also merges the move plan property values for the target environment.

> **Note:** All the domains that are to be managed by Node Manager should be moved before applying the copy of Node Manager to the target environment. In addition, the Administration Server must be running.
>
> Even if the source Node Manager connection between the Administration Server and Node Manager is configured with SSL, they will both change to plain socket connection type after the copy of Node Manager is applied to the target environment.

You must run the pasteConfig script for each Node Manager in the source environment.

The syntax is:

```
pasteConfig  -javaHome path_of_jdk
             -archiveLoc archive_location
             -targetNMHomeLoc trgt_Node_Manager_Home_path
             -targetMWHomeLoc trgt_Middleware_Home_path
             -movePlanLoc move_plan_path
             [-logDirLoc log_dir_path]
             [-silent {true | false}]
```

The following example shows how to apply the copy of Node Manager to the Node Manager home located in /scratch/Oracle/Middleware1/wlserver_10.3/common/nodemanager:

```
pasteConfig -javaHome USER_HOME/jrockit_160_17_R28.0.0-679/
            -archiveLoc /tmp/nm.jar
            -targetNMHomeLoc /scratch/Oracle/Middleware1/wlserver_
10.3/common/nodemanager
            -targetMWHomeLoc /scratch/Oracle/Middleware1
            -movePlanLoc /scratch/oracle/t2pplans/nm/moveplan.xml
            -silent true
```

Table 20–11 describes the options for the pasteConfig script for Node Manager.

Table 20–11 Options for the pasteConfig Script for Node Manager

Options	Shortcut	Description	Mandatory or Optional
`-javaHome`	None	The absolute path of the Java Developer's Kit. If the operating system is SunOS, HP-UX, or Linux 64 bit, pass the -d64 option to the scripts in the command line. To set the runtime property, you can specify the -d64 option in the T2P_JAVA_OPTIONS environment variable. For example: `setenv T2P_JAVA_OPTIONS "-d64 -Djava.io.tmpdir=/home/t2p/temp"`	Mandatory
`-archiveLoc`	`-al`	The absolute path of the archive location. Use this option to specify the location of the archive file created by the copyConfig script.	Mandatory
`-targetNMHomeLoc`	`-tnh`	The absolute path of the target Node Manager.	Mandatory
`-targetMWHomeLoc`	`-tmw`	The absolute path of the target Middleware home in which the copy of Node Manager is to be applied.	Mandatory
`-movePlanLoc`	`-mpl`	The absolute path of the modified move plan in the target environment.	Mandatory
`-logDirLoc`	`-ldl`	The location of an existing directory. A new log file is created in the directory. The default is the system Temp location.	Optional
`-silent`	None	Specifies whether the operation operates silently. That is, it does not prompt for confirmation. The default is that the operation prompts for confirmation. To continue, you must type `yes`, which is not case sensitive. Typing anything other than yes causes the script to return an error. To specify that it does not prompt for confirmation, specify this option with the value of `true`.	Optional

20.3.1.12 obfuscatePassword Script

Generates a file that contains the obfuscated password. In the scripts and in the move plans, you often need to provide files containing passwords. The script prompts you to enter the password and the location where the password file is to be written.

The syntax is:

```
(UNIX) ORACLE_COMMON_HOME/bin/obfuscatePassword.sh
(Windows) ORACLE_COMMON_HOME\bin\obfuscatePassword.cmd
```

20.4 Modifying Move Plans

When you move Oracle Fusion Middleware components, you run the extractMovePlan script to create a move plan for the entity that you are moving. The extractMovePlan script extracts configuration information from the archive into a move plan. It also extracts any needed configuration plans. Before you apply the archive to the target, you must edit the move plan to reflect the values of the target environment.

You can modify properties with the scope of READ_WRITE. Do not modify the properties with the scope of READ_ONLY.

Note: Do not add, comment, or remove any section of a move plan.

This section provides the following topics:

- Locating configGroup Elements
- Move Plan Properties

20.4.1 Locating configGroup Elements

Most move plans contain multiple configGroup elements. When a property is associated with a particular configGroup element, the tables listing the properties group the properties by configGroup element. For example, Table 20–14, which shows the properties for the move plan for Java components, shows multiple configGroup elements, such as SERVER_CONFIG and MACHINE_CONFIG.

The following example shows a portion of the move plan for Java components, with portions of the SERVER_CONFIG and MACHINE_CONFIG configGroup elements:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<movePlan>
    <movableComponent>
        <componentType>J2EEDomain</componentType>
        <moveDescriptor>
            <StartupMode>PRODUCTION</StartupMode>
            <configGroup>
                <type>SERVER_CONFIG</type>
                <configProperty id="Server1">
                    <configProperty>
                        <name>Server Name</name>
                        <value>AdminServer</value>
                        <itemMetadata>
                            <dataType>STRING</dataType>
                            <scope>READ_ONLY</scope>
                        </itemMetadata>
                    </configProperty>
            .
            .
            .
             </configGroup>
             <configGroup>
                <type>MACHINE_CONFIG</type>
                   <configProperty id="Machine1">
```

```
                <configProperty>
                   <name>Machine Name</name>
                   <value>LocalMachine</value>
                   <itemMetadata>
                      <dataType>STRING</dataType>
                      <scope>READ_WRITE</scope>
                   </itemMetadata>
                </configProperty>
                <configProperty>
                   <name>Node Manager Listen Address</name>
                   <value>example.com</value>
                   <itemMetadata>
                      <dataType>STRING</dataType>
                      <scope>READ_WRITE</scope>
                   </itemMetadata>
                 </configProperty>
          .
          .
          .
       </configGroup>
```

20.4.2 Move Plan Properties

The tables in this section describe the move plan properties you can customize for Oracle Fusion Middleware entities and components.

> **Note:** Many move plan properties require that you provide the location of a file containing a password. If you want to use obfuscated passwords, you can use the obfuscatePassword script, as described in Section 20.3.1.12.

The properties that you edit differ depending on the type of component. Table 20–12 provides pointers to the appropriate list of properties for each component.

Table 20–12 Move Plan Properties for Components

Component	Where to find the list of properties:
Node Manager	Table 20–13
All Java components	Table 20–14
Oracle Access Management Access Manager	Table 20–14, Table 20–30
Oracle Access Management Identity Federation	Table 20–14, Table 20–21
Oracle Access Management Mobile and Social	Table 20–14, Table 20–30
Oracle Access Management Secure Token Service	Table 20–14, Table 20–30
Oracle Adaptive Access Manager	Table 20–14, Table 20–31
Oracle ADF connections	Table 20–15
Oracle B2B	Table 20–17
Oracle BI EE	Table 20–14, Table 20–15, Table 20–22, and optionally Table 20–23, Table 20–24, Table 20–25, Table 20–26

Table 20–12 (Cont.) Move Plan Properties for Components

Component	Where to find the list of properties:
Oracle BI Publisher	Table 20–14, Table 20–15, Table 20–22
Oracle Data Integrator	Table 20–29
Oracle HTTP Server	Table 20–18
Oracle Internet Directory	Table 20–19
Oracle SOA Suite	Table 20–14, Table 20–15, Table 20–16
Oracle Virtual Directory	Table 20–20
Oracle WebCenter Content Server	Table 20–14, Table 20–15, Table 20–27
Oracle WebCenter Content: Imaging	Table 20–14, Table 20–15, Table 20–28
Oracle WebCenter Content: Inbound Refinery:	Table 20–14, Table 20–15, Table 20–27
Oracle WebCenter Content: Records	Table 20–14, Table 20–15, Table 20–27

Table 20–13 describes the move plan properties that you can change for Node Manager. Edit all properties, such as host names, port numbers, listen addresses, that have different values in the target environment.

Table 20–13 Move Plan Properties for Node Manager

Property	Description	Sample Value
Properties in the NODE_MANAGER_PROPERTIES configGroup:	Node Manager configuration	
Listen Address	The Listen address of Node Manager.	`example.com`
Listen Port	The number of the Listen port of Node Manage.	`5557`
Custom Identity KeyStore File	The absolute path of the custom identity keystore file location. This property is present in the move plan only if the source environment is configured with SSL.	`/scratch/Oracle/Middleware/wlserver_10.3/server/lib/example_identity.jks`
Custom Identity Private Key Alias	The value of the identity key store alias. This property is present in the move plan only if the source environment is configured with SSL.	`mykey`
Custom Identity Private Key Passphrase File	The absolute path to the secure file containing the private key used when creating a certificate. This property is present in the move plan only if the source environment is configured with SSL.	`/scratch/oracle/key_passwd`
Properties in the DOMAINS configGroup:	Oracle WebLogic Server domain configuration	
Domain Name	The name of the domain.	`SOA_domain`
Domain Location	The absolute path of the domain location.	`/scratch/Oracle/Middleware/user_projects/domains/SOA_domain`

Table 20–13 (Cont.) Move Plan Properties for Node Manager

Property	Description	Sample Value
AdminServer Listen Address	The Listen address of the Administration Server.	`example.com`
AdminServer Listen Port	The number of the Listen port of the Administration Server.	`7001`
AdminServer User Name	The administration user name.	`weblogic`
AdminServer Password	The absolute path to the secure file containing the administration user's password.	`/scratch/oracle/admin_passwd`
Node Manager User Name	The Node Manager user name.	`weblogic`
Node Manager Password	The absolute path to the secure file containing the Node Manager user's password.	`/scratch/oracle/nm_passwd`

Table 20–14 describes the move plan properties that you can change for Java components. Edit all properties, such as host names, port numbers, listen addresses, that have different values in the target environment.

Table 20–14 Common Move Plan Properties for Java Components

Property	Description	Sample Value
Startup Mode	The startup mode of an Oracle WebLogic Server domain. Valid values are: ▪ DEVELOPMENT. Use this mode while you are developing your applications. Development mode uses a relaxed security configuration and enables you to auto-deploy applications. ▪ PRODUCTION. Use this mode when your application is running in its final form. A production domain uses full security and may use clusters or other advanced features. The default is PRODUCTION.	`PRODUCTION`
Properties in the SERVER_ CONFIG configGroup:	Common Java properties	
Listen Address	The Listen address of the WebLogic Server. Set it to the host name or set it to `All Local Addresses` to listen on all addresses on the host.	`All Local Addresses`
Listen Port	The number of the Listen port. If you do not provide a port number or if the port number you provide is not available, the operation returns an error.	`8001`
SSL Listen Port	The number of the SSL Listen port. This property is present in the move plan if SSL is enabled.	`7002`

Table 20–14 (Cont.) Common Move Plan Properties for Java Components

Property	Description	Sample Value
Frontend Host	The host name of the HTTP Server. This property is present in the move plan only if the HTTP Server is set as the frontend to the server.	`example.com`
Frontend HTTP Port	The number of the HTTP Server port. This property is present in the move plan only if the HTTP Server is set as the frontend to the server.	`10605`
Custom Identity Keystore File	The absolute path of custom identity keystore file location.	`/scratch/Oracle/Middleware/wlserver_10.3/server/lib/example_identity.jks`
Custom Identity Keystore Passphrase File	The absolute path to the secure file containing the custom identity keystore password.	`/scratch/oracle/i_passwd`
Custom Trust Keystore File	The absolute path of custom trust keystore file location.	`/scratch/Oracle/Middleware/wlserver_10.3/server/lib/example_trust.jks`
Custom Trust Keystore Passphrase File	The absolute path to the secure file containing the custom trust keystore password.	`/scratch/oracle/key_passwd`
Custom Identity Private Key Alias	The private key alias.	`fmw_key`
Custom Identity Private Key Passphrase File	The absolute path to the secure file containing the custom identity private key password.	`/scratch/oracle/i_passwd`
Properties in the CLUSTER_CONFIG configGroup:	Oracle WebLogic Server Cluster configuration properties	
Messaging Mode	The cluster messaging mode. Acceptable values are `unicast` and `multicast`.	`multicast`
Cluster Address	The cluster address.	`localhost`
Unicast Channel	The name of the unicast channel.	`MyMulticastChannel`
Multicast Address	The multicast address.	`239.192.0.0`
Multicast Port	The port number of the multicast address.	`8899`
Frontend Host	The name or IP address of the front-end host for the cluster.	`example.com`
Frontend HTTP Port	The HTTP port number for the front-end host for the cluster.	`7008`
Properties in the MACHINE_CONFIG configGroup:	Machine configuration properties	
Machine Name	The name of the machine.	`example.com`
Node Manager Listen Address	The Listen address of the machine running Node Manager.	`example.com`
Node Manager Listen Port	The port number of the Listen address of the machine running Node Manager.	`5556`
Property in the DEPLOYMENT_PLAN_CONFIG configGroup:	Deployment plans	

Table 20–14 (Cont.) Common Move Plan Properties for Java Components

Property	Description	Sample Value
Deployment Plan	The location where an application's deployment plan is to be extracted. The location is relative to the location of the move plan.	`deploy_plans/helloWorldEar_plan.xml`
Properties in the AUTHENTICATORS configGroup:	Authenticator configuration	
Host Name	The LDAP server host name.	`example.com`
Port	The LDAP server port number.	`3060`
Principal	The administration user for the LDAP server.	`cn=orcladmin`
Password File	The absolute path of a secure file containing the password for the LDAP user. You must provide a password file, even if you are not changing the configuration.	`/scratch/oracle/ldap_passwd`
User Base DN	The user base distinguished name (DN).	`cn=users,dc=us,dc=oracle,dc=com`
User Object Class	The user object class.	`person`
Group Base DN	The group base distinguished name (DN).	`cn=groups,dc=us,dc=oracle,dc=com`
GUID Attribute	The global unique identifier.	`orclguid`
Properties in the DATASOURCE configGroup:	Data source configuration	
Driver Class	The driver class of the data source. Refer to "Using JDBC Drivers with WebLogic Server" in the *Oracle Fusion Middleware Configuring and Managing JDBC Data Sources for Oracle WebLogic Server* to choose the appropriate class.	`oracle.jdbc.OracleDriver`
Url	The URL of the database for the data source. It contains the host name, database service name or SID, and the database port number.	`jdbc:oracle:thin:@orcl.example com:1521/orcl.example.com`
User	The schema name of the data source.	`OFM_MDS`
Password File	The absolute path to the secure file containing the password of the database schema. You must provide a password file, even if you are not changing the configuration of the data source.	`/scratch/oracle/ds_passwd`

Table 20–14 (Cont.) Common Move Plan Properties for Java Components

Property	Description	Sample Value
Properties in the OPSS_ SECURITY configGroup, in the configProperty with the ID of LDAP.	LDAP-based policy and credential store configuration. If the source is a file-based store, these properties, as well as the LDAP-based and Database-Based Policy and Credential Store properties are present in the move plan. When you configure the move plan, you can change from a file-based to an LDAP or database-based store. If the source is LDAP-based, only the LDAP properties are present in the move plan. You cannot change it to a different type, but you can change the LDAP endpoints. If the source is database-based, only the database properties are present in the move plan. You cannot change it to a different type, but you can change the database-based endpoints. You can only use one type of store. To use one, uncomment the section in the move plan and ensure the other is commented.	
Password File	The absolute path to the secure file containing the password of the LDAP Server Administrative user. You must provide a password file, even if you are not changing the configuration of the LDAP Server.	`/scratch/oracle/ldap_passwd`
LDAP User	The LDAP Server administrative user name.	`cn=orcladmin`
Jps Root	The LDAP Server context root.	`cn=jpsRoot`
Domain	The name of the domain.	`SOA_domain`
LDAP Url	The URL of the LDAP connection. It contains the host name and port number of the LDAP store.	`ldap://example.com:3060`
Properties in the OPSS_ SECURITY configGroup, in the configProperty with the ID of DB:	Database-based policy and credential store configuration. If the source is a database-based store, these properties are present in the move plan. (The LDAP-based store is not present and you cannot move from a database-based to an LDAP-based store.)	
Password File	The absolute path to the secure file containing the password of the OPSS schema owner. You must provide a password file, even if you are not changing the configuration.	`/scratch/oracle/ldap_passwd`
DataSource Name	The name of the data source.	`opssds`
DataSource Jndi Name	The JNDI name of the data source.	`jdbc/opss`
Jps Root	The LDAP Server context root.	`cn=jpsRoot`
Domain	The name of the domain.	`SOA_domain`
Driver Class	The driver class of the data source. Refer to "Selecting a JDBC Driver" in the *Oracle Fusion Middleware Configuring and Managing JDBC Data Sources for Oracle WebLogic Server* to choose the appropriate class.	`oracle.jdbc.OracleDriver`

Table 20–14 (Cont.) Common Move Plan Properties for Java Components

Property	Description	Sample Value
Url	The database URL of the data source. It contains the host name, the database port number, and the database service name or SID.	`jdbc:oracle:thin:@hostname.com:1521:orcl`
User	The name of the OPSS schema owner of the data source.	`DEV_OPSS`
Properties in the RDBMS Security Store configGroup:	Database-based security store configuration	
URL	The database URL of the security store connection. It contains the host name, the database port number, and the database service name or SID.	`jdbc:oracle:thin:@hostname.com:1521/example.com`
Driver Class	The driver class of the RDBMS Security Store connection. Refer to "Using JDBC Drivers with WebLogic Server" in the *Oracle Fusion Middleware Configuring and Managing JDBC Data Sources for Oracle WebLogic Server* to choose the appropriate class.	`oracle.jdbc.OracleDriver`
User	The name of the schema owner.	`admin`
Password File	The absolute path to the secure file containing the password of the security store schema owner. You must provide a password file, even if you are not changing the configuration.	`/scratch/oracle/rbms_passwd`
Property in the ADAPTER configGroup:	Resource adapter configuration	
Deployment Plan	The path to the deployment plan to be used during movement to the target. The path can be absolute, or relative to the location of the move plan. The deployment plan is extracted by the extractMovePlan script.	`/scratch/adapters/adapters.xml`

Table 20–15 describes the move plan properties that you can change if you are using Oracle ADF connections. Edit all properties, such as host names, port numbers, listen addresses, that have different values in the target environment. The table is divided by component. For some components, the description column lists the OBJECT_NAME_PROPERTY type. You can search for the type to locate the relevant section.

Table 20–15 Move Plan Properties for Oracle ADF Connections

Property	Description	Sample Value
Oracle ADF URL Connection	**OBJECT_NAME_PROPERTY type is URLConnProvider**	
Port	The port number used for the URL connections.	`7000`
URL	The URL used for the connection.	`example.com`
Oracle ADF Business Components Service Connection	**OBJECT_NAME_PROPERTY type is ADFBCServiceConnection**	
ServiceEndpointProvider	The Business Components service endpoint provider.	`ADFBC`

Table 20–15 (Cont.) Move Plan Properties for Oracle ADF Connections

Property	Description	Sample Value
JndiFactoryInitial	The JNDI initial factory class.	`com.sun.java.jndi.InitialFactory`
JndiProviderUrl	The URL of the JNDI provider.	`t3://example.com:7101`
JndiSecurityPrincipal	The JNDI security principal name.	`weblogic`
WebServiceConnectionName	The Web service connection name.	`test`
Oracle Enterprise Scheduler	**OBJECT_NAME_PROPERTY type is EssConnection**	
NotificationServiceURL	The Oracle Enterprise Scheduler notification service URL.	`http://localhost:8001`
RequestFileDirectory	The path of the directory where request logs for jobs from OES ConcurrentProcessor (CP) extension is to be created.	`/tmp/ess/requestFileDirectory`
SAMLTokenPolicyURI	The SAML Policy URI to be used by CP extension.	`oracle/wss11_saml_token_with_message_protection_service_policy`
EssCallbackClientSecurityPolicyURI	The security policy to be used in the WS-Security headers for Web service invocations from Oracle Enterprise Scheduler for Web service callbacks.	`oracle/wss11_saml_token_with_message_protection_client_policy`
Oracle Business Activity Monitoring		
WEBTIER_SERVER	The Oracle BAM Web server host.	`example.com`
USER_NAME	The Oracle BAM user name.	`user`
PASSWORD	The password for the Oracle BAM user.	*`bam_password`*
WEBTIER_SERVER_PORT	The port number for the Web server.	`9001`
BAM_SERVER_PORT	The JNDI port number.	`8001`
BAM_WEBTIER_PROTOCOL	The network protocol. The valid values are HTTP and HTTPS.	`HTTP`
BI Presentation Services connection	**OBJECT_NAME_PROPERTY type is BISoapConnection**	
StaticResourcesLocation	The location where the browser should fetch Oracle BI EE static resources.	`http://example.com:7001/analytics`
WSDLContext	The Oracle BI EE context to use when making a Web services call.	`analytics-ws`
Host	The host where Oracle BI EE is located.	`example.com`
Port	The port that hosts the BI Presentation Services server.	`10621`
ShouldPerformImpersonation	Whether Oracle BI EE should perform impersonation. This should always be set to true.	`true`
Context	The Oracle BI EE context to use when fetching content.	`analytics`
Protocol	The protocol to use, depending on whether the Web server is configured with SSL.	`http or https`

Table 20–15 (Cont.) Move Plan Properties for Oracle ADF Connections

Property	Description	Sample Value
IsStaticResourcesLocationAutomatic	The flag indicating whether to auto-generate the StaticResourcesLocation from the Host, Port, and Context fields.	`true or false`
Oracle Essbase	**OBJECT_NAME_PROPERTY type is EssbaseConnProvider**	
Host	The host name for the Oracle Essbase server.	`example.com`
Cluster	The name of the cluster of which the Oracle Essbase server is a member.	`esbCluster`
Port	The Listen port number of the Oracle Essbase server.	`1423`
Username	The user name.	`user3`
Oracle Secure Enterprise Search in Oracle WebCenter Portal	**The OBJECT_NAME_PROPERTY type is SesConnectionProvider**	
SoapURL	The Web services URL that Oracle SES exposes to enable search requests.	`http:/example.com:`*`port`*`/search/query/OracleSearch`
Oracle WebCenter Portal Content Repository	**OBJECT_NAME_PROPERTY type is JCR**	
ServerHost	The host name of the machine where the Content Server is running.	`example.com`
ServerPort	The port number on which the Content Server listens.	`4444`
ServerWebUrl	The Web server URL for the Content Server.	`http://example.com/cms/idcplg`
Oracle WebCenter Portal Announcements and Discussions	**OBJECT_NAME_PROPERTY type is ForumConnectionProvider**	
AdminUser	The name of the discussions server administrator. This account is used by the Discussions and Announcements services to perform administrative operations on behalf of WebCenter Portal users.	`admin`
Url	The URL of the discussions server hosting discussion forums and announcements.	`http://example.com:8890/owc_discussions`
Oracle WebCenter Portal External Applications	**OBJECT_NAME_PROPERTY type is ExtAppConnectionProvider**	
Url	The login URL for the external application.	`https://example.com/config/login?`
Oracle WebCenter Portal Instant Messaging and Presence	**OBJECT_NAME_PROPERTY type is RtcConnectionProvider**	
BaseConnectionURL	The URL of the server hosting instant messaging and presence services.	`http://example.com:8888`
Domain	The network domain.	`example.com`

Table 20–15 (Cont.) Move Plan Properties for Oracle ADF Connections

Property	Description	Sample Value
ExternalAppId	The external application ID associated with the Presence server connection. If present, external application credential information is used to authenticate users against the instant messaging and presence server.	`extApp`
Oracle WebCenter Portal Mail Server	**OBJECT_NAME_PROPERTY is MailConnectionProvider**	
ExternalAppId	The external application ID associated with the mail server.	`extApp_Mail`
ImapHost	The host name of the IMAP server.	`example.com`
ImapPort	The port number of the IMAP server.	`993`
ImapSecured	The flag indicating whether the mail server connection to the IMAP server is SSL-enabled. Valid values are true and false. The default is false.	`true`
SmtpHost	The host name of the computer where the SMTP (Simple Mail Transfer Protocol) service is running.	`example.com`
SmtpPort	The port number of the SMTP host.	`587`
SmtpSecured	The flag indicating whether the SMTP server is secured. Valid values are true and false. The default is false.	`true`
Oracle WebCenter Portal Personal Events	**OBJECT_NAME_PROPERTY type is WebCenterPersonalEventConnectionProvider**	
ExternalAppId	The external application associated with the Microsoft Exchange Server providing personal events services. If specified, external application credential information is used to authenticate users against the Microsoft Exchange Server.	`ExtPEApp`
WebServiceURL	The URL of the Web service exposing the event application.	`http://example.com:80/ExchangeWS/PersonalEventsWebService.asmx`
Oracle WebCenter Portal WSRP Producers	**OBJECT_NAME_PROPERTY type is WSRPProducerConnection**	
ProxyHost	The host name or IP address of the proxy server.	`example.com`
ProxyPort	The port number of the proxy server.	`80`
Oracle WebCenter Portal PDK-Java Producers	**OBJECT_NAME_PROPERTY type is WebProducerConnection**	
Host	The host name of the proxy server to be used for the PDK Java Producer connection.	`example.com`
Port	The port number to be used for the PDK Java Producer connection.	`80`
URL	The URL for the PDK Java Producer connection.	`http:/example.com:`*`port`*

Table 20–15 (Cont.) Move Plan Properties for Oracle ADF Connections

Property	Description	Sample Value
Oracle WebCenter Portal Worklists	**OBJECT_NAME_PROPERTY type is BPEL**	
URL	The URL required to access the BPEL server. The BPEL server URL must be unique within the WebCenter application.	*protocol*://example:*port*
Oracle Web Services	**OBJECT_NAME_PROPERTY type is WebServiceConnection**	
WsdlUrl	The URL for the WSDL.	http://example.com:*port*/MyWebService1?WSDL
Oracle Web Services	**OBJECT_NAME_PROPERTY type is Port**	
AddressUrl	The service endpoint URL.	http://example.com:*port*/MyWebService1
ProxyHost	The name of the host on which the proxy server is running.	example.com
ProxyPort	The port number to which the proxy server is listening.	80

Table 20–16 describes the move plan properties that you can change for Oracle SOA Suite. Edit all properties, such as host names, port numbers, listen addresses, that have different values in the target environment.

Table 20–16 Move Plan Properties for Oracle SOA Suite

Property	Description	Sample Value
Property in the Composite configGroup:	SOA Composites configuration	
Config Plan Location	The location of the configuration plan to be used during movement to the target to redeploy the composite application. The path can be absolute, or relative to the location of the move plan. The plan is extracted during the extractMovePlan script.	/scratch/app/config_plan.xml

Table 20–17 describes the move plan properties that you can change for Oracle B2B. Edit all properties, such as host names, port numbers, listen addresses, that have different values in the target environment.

Table 20–17 Move Plan Properties for Oracle B2B

Property	Description	Sample Value
Property in the File.DeliveryChannel configGroup:	File Delivery Channel configuration.	
file-param-folder	The absolute path of the folder.	/tmp/file_deliv
Property in the File.ListeningChannel configGroup:	File Listening Channel configuration.	
file-param-folder	The absolute path of the folder.	/tmp/file_listen

Table 20–17 (Cont.) Move Plan Properties for Oracle B2B

Property	Description	Sample Value
Properties in the JMS configGroup:	JMS configuration. Each channel has its own set of property values.	
jms-param-is_topic	A flag specifying whether or not this is a configured destination topic. Valid values are `true` and `false`.	`false`
jms-param-queue_name	The JNDI name of the queue or topic.	`jms/b2b/B2B_IN_QUEUE`
jms-param-DestinationProviderProperties	The JMS destination provider properties. Use a semicolon (;) as the separator for each key/value pair.	`java.naming.provider.url=t3://example.com:7001;java.naming.factory.initial=weblogic.jndi.WLInitialContextFactory;java.naming.security.principal=weblogic;java.naming.security.credentials=weblogic`
jms-param-user	The JMS user name.	`user1`
Properties in the FTP configGroup:	FTP configuration. Each channel has its own set of property values.	
ftp-param-folder	The absolute path of the folder.	`/tmp/test1`
ftp-param-host	The FTP host name.	`example`
ftp-param-preserve_filename	A flag that specifies whether the file name will be preserved. Valid values are `true` and `false`.	`false`
ftp-param-user	The FTP user name.	`User`
Properties in the HTTP configGroup:	HTTP configuration. Each channel has its own set of property values.	
http-param-use_proxy	A flag that specifies whether to use a proxy server. Valid values are `true` and `false`.	`false`
http-param-additional_headers	Additional transport headers, for example, headers for digest authentication.	
http-param-url	The fully qualified HTTP URL.	`http://example:8001/b2b/httpReceiver`
Properties for the SFTP transport protocol:	The SFTP configuration.	
sftp-param-host	The SFTP host name.	`example`
sftp-param-port	The SFTP port number.	`22`
sftp-param-folder	The absolute path of the folder.	`/scratch/b2b/sftp`
sftp-param-user	The name of the SFTP user.	`user1`
Properties for the Email transport protocol:	The email configuration.	
email-param-host	The email host name.	`example`
email-param-user	The email user name.	`user1`
email-param-email-id	The email address to which messages are delivered (similar to specifying the path for a file channel or queues in AQ or JMS).	`user1@exampleb2b.com`
Properties for the AQ transport protocol:	The AQ configuration.	

Table 20–17 (Cont.) Move Plan Properties for Oracle B2B

Property	Description	Sample Value
aq-param-datasource	The JNDI name of the JDBC data source to access AQ queues.	`jdbc/SOADataSource`
aq-param-recipient	The value used when delivering a message to the AQ queue.	`testuser`
aq-param-queue_name	The AQ queue name.	`IP_OUT_QUEUE`
aq-param-consumer	The client that receives the message.	`b2buser`
Properties for the TCP transport protocol:	The TCP configuration.	
tcp-param-host	The TCP host name.	`example`
tcp-param-port	The TCP port number.	`23456`
tcp-param-PermanentConnectionType	A flag indicating whether or not a cached connection is used to exchange all the messages. Valid values are `true` and `false`.	`false`
tcp-param-timeout	The TCP timeout, in seconds.	`300`

Table 20–18 describes the move plan properties that you can change for Oracle HTTP Server. Edit all properties, such as host names, port numbers, listen addresses, that have different values in the target environment.

For Oracle HTTP Server, there are many configGroup elements in the move plan. Each configGroup element is associated with one Oracle HTTP Server configuration file. As a result, there may be more than one instance of a particular property, such as User.

Table 20–18 Move Plan Properties for Oracle HTTP Server

Property	Description	Sample Value
Listen	The Listen address. It can include the host name and port or just the port.	`orcl3.example.com:8888 or 8888`
User	The Oracle HTTP Server administration user.	*`admin_user`*
Group	The group for the user.	*`admin_group1`*
ServerAdmin	The administrator's email address.	`Webmaster@example.com`
ServerName	The name of the server for Oracle HTTP Server. If the host does not have a registered DNS name, use the IP address.	`orcl1.example.com`
WebLogicHost	The name of the host on which Oracle WebLogic Server is listening for requests.	`orcl2.example.com`
WebLogicPort	The port number that Oracle WebLogic Server uses to listen for requests.	`9002`
WebLogicCluster	The name of the host on which an Oracle WebLogic Server cluster is running and its port number.	`orcl3.example.com:9003`
VirtualHost	The name of the virtual host. The port number listed should also be listed in the Listen directive.	`*.8888`

Table 20–18 (Cont.) Move Plan Properties for Oracle HTTP Server

Property	Description	Sample Value
PlsqlDatabasePassword	Specific to the PLSQL module, the name of a secure file containing the password. You must provide a password file, even if you are not changing the configuration.	`/scratch/orcl/plsql_passwd`
PlsqlDatabaseConnectString	Specific to the PLSQL module, the service name of the database.	`orcl.example.com:1521:orcl1`
PlsqlNLSLanguage	Specific to the PLSQL module, the NLS_LANG variable for the database access descriptor (DAD).	`America_America.UTF8`
ORAConnectSN	Specific to the oradav module, the Oracle database to which to connect.	*`db_host:db_port:db_service_name`*
ORAUser	Specific to the oradav module, the database user (schema) to use when connecting to the service specified by the ORAConnectSN property.	`db6175_PORTAL`
ORACRYPTPASSWORD	Specific to the oradav module, the absolute path to the secure file containing the password for oradav. You must provide a password file, even if you are not changing the configuration.	`/scratch/oracle/dav_passwd`
SSLWallet	The location of the SSL wallet, if the wallet is not in the default location.	`/scratch/oracle/mw_home/`*`ORACLE_INSTANCE`*`/config/OHS/ohs1/keystores/mywallets`
DocumentRoot	The directory that stores the main content for the Web site.	`/scratch/oracle/mw_home/`*`ORACLE_INSTANCE`*`/config/ohs/ohs1/htdocs`
Alias	The location of the alias, if it is not in the default location. Note that you change the value within the double quotation marks.	`/icons/"/scratch/orcl/icons/"`
ScriptAlias	The location of the script alias, if it is not in the default location. Note that you change the value within the double quotation marks.	`/cgi-bin/"/scratch/oraclcgi-bin/"`
WebGateInstalldir	The location of the WebGate installation directory, as specified in the webgate.conf file.	`/scratch/oracle/mw_home/Oracle_OAMWebGate1/webgate/ohs`
primaryOAMServerHost	The primary Access Manager server host. Note that the configuration for the secondary Access Manager server host is updated automatically the first time that WebGate communicates with the primary server.	*`primary_oam_server_host`*`.example.com`
primaryOAMServerPort	The port number for the Access Manager primary host.	`5575`

Table 20–19 describes the move plan properties that you can change for Oracle Internet Directory. Edit all properties, such as host names, port numbers, listen addresses, that have different values in the target environment.

Table 20–19 Move Plan Properties for Oracle Internet Directory

Property	Description	Sample Value
OID Non SSL Port	The non-SSL port for Oracle Internet Directory. If you do not provide a port number or if the port number you provide is not available, the operation uses an available port.	`3060`
OID SSL Port	The SSL port for Oracle Internet Directory. If you do not provide a port number or if the port number you provide is not available, the operation uses an available port.	`3131`
Namespace	The Oracle Internet Directory namespace.	`dc=us,dc=oracle,dc=com`
OID Admin Password	The absolute path of a secure file containing the password for the Oracle Internet Directory administrator. You must provide a password file, even if you are not changing the configuration.	`/scratch/oracle/oid_passwd`
ODS Schema Password	The absolute path of a secure file containing the password for the ODS schema, which is the schema that contains metadata for Oracle Internet Directory. You must provide a password file, even if you are not changing the configuration.	`/scratch/oracle/ods_passwd`
ODSSM Schema Password	The absolute path of a secure file containing the password for the ODSSM schema, which is used to access server manageability information for Oracle Internet Directory from the database. You must provide a password file, even if you are not changing the configuration.	`/scratch/oracle/odssm_passwd`
DB Host Name	The name of the host on which the database is running, which can be found in the tnsnames.ora file.	`example.com`
DB Port	The port number of the database listener, which can be found in the tnsnames.ora file.	`1521`
DB Service Name	The service name for the database, which can be found in the tnsnames.ora file.	`orcl.example.com`

Table 20–20 describes the move plan properties that you can change for Oracle Virtual Directory. Edit all properties, such as host names, port numbers, listen addresses, that have different values in the target environment.

Table 20–20 Move Plan Properties for Oracle Virtual Directory

Property	Description	Sample Value
OVD Non SSL Port	The LDAP non-SSL port number for Oracle Virtual Directory. If you do not provide a port number or if the port number you provide is not available, the operation uses the next available port.	`6501`
OVD SSL Port	The LDAP SSL port number for Oracle Virtual Directory. If you do not provide a port number or if the port number you provide is not available, the operation uses the next available port.	`7501`

Table 20–20 (Cont.) Move Plan Properties for Oracle Virtual Directory

Property	Description	Sample Value
OVD Admin Port	The administration port number for Oracle Virtual Directory. If you do not provide a port number or if the port number you provide is not available, the operation uses the next available port.	`8899`
OVD Http Port	The HTTP listener port number for Oracle Virtual Directory.	`8080`
host.port	The host name and port for the Oracle Virtual Directory adapter.	`example.com:3060`
username	The user name for the Oracle Virtual Directory adapter.	`cn=orcladmin`
root	The root for the Oracle Virtual Directory adapter.	`dc=us,dc=oracle,dc=com`
remotebase	The remote base for the Oracle Virtual Directory adapter.	`dc=us,dc=oracle,dc=com`
password	The absolute path of a secure file containing the password for the Oracle Virtual Directory adapter user.	`/scratch/oracle/ovd_passwd`

Table 20–21 describes the move plan properties that you can change for Identity Federation. Edit all properties, such as host names, port numbers, listen addresses, that have different values in the target environment.

Table 20–21 Move Plan Properties for Identity Federation

Property	Description	Sample Value
Properties in the ServerConfig configProperty:	Server configuration	
Load Balancer Hostname	The name of the host of the Load Balancer.	`example.com`
Load Balancer Port	The port number of the Load Balancer.	`7500`
SOAP Port	The SOAP port number.	`7500`
SSL Enabled	The flag indicating that SSL is enabled. Valid values are `true` and `false`.	`true`
SOAP SSL Enabled	The flag indicating that SSL is enabled for SOAP. Valid values are `true` and `false`.	`true`
Properties in the User Data Store configProperty:	The user data store configuration. These properties are present in the move plan if the user data store uses LDAP.	
LDAP Server URL	The URL of the LDAP connection. The property contains the host name and port number of the LDAP store.	`ldap://example.com:389`
LDAP Username	The LDAP Server administrative user name.	`cn=orcladmin`
LDAP Password File	The absolute path of a secure file containing the password for the administrative user.	`/scratch/oracle/oif_ds_passwd`
Properties in the Federation Data Store configProperty:	Federation data store configuration. These properties are in the move plan if the federation data store uses LDAP.	
LDAP Server URL	The URL of the Federation LDAP connection. It contains the host name and port number of the Federation LDAP store.	`ldap://example.com:389`

Table 20–21 (Cont.) Move Plan Properties for Identity Federation

Property	Description	Sample Value
LDAP Username	The Federation LDAP Server administrative user name.	`cn=orcladmin`
LDAP Password File	The absolute path of a secure file containing the password for the Federation LDAP administrative user.	`/scratch/oracle/oif_fed_passwd`
Properties in the AuthnEngine configProperty:	Authentication engine configuration. There may be multiple Authentication engines, which are independent of each other.	
LDAP Connection URL	The URL of the LDAP connection. It contains the host name and port number of the LDAP store. This property is present in the move plan if LDAP is enabled as an Authentication Engine.	`ldap://example.com:389`
LDAP Username	The LDAP administrative user name. This property is present in the move plan if LDAP is enabled as an Authentication Engine.	`cn=orcladmin`
LDAP Password File	The absolute path of a secure file containing the password for the administrative user. This property is present in the move plan if LDAP is enabled as an Authentication Engine.	`/scratch/oracle/oif_ae_pwd_passwd`
OAM11g Logout URL	The URL for logging out of Access Manager 11*g*. It contains the host name and port number of Access Manager 11*g*. This property is present in the move plan if Access Manager 11*g* is enabled as an Authentication Engine.	`http://oam11g_host:`*`oam11g_port`*`/logout.jsp`
HTTP Header Logout Redirect URL	The URL for logging out. It contains the host name and port number of the Oracle HTTP Server. This property is present in the move plan if HTTP Header is enabled as an Authentication Engine.	`http://example.com:`*`port`*`/logout.jsp`
Properties in the SPEngine configGroup:	The SP engine configuration. There may be multiple SP engines, which are independent of each other.	
OAM11g Login URL	The URL for logging in to Access Manager 11*g*. It contains the host name and port number of Access Manager 11*g*. This property is present in the move plan if an SP Engine uses Access Manager 11*g*.	`http://oam11g_hostname:`*`oam11g_port`*`/login`
OAM11g Logout URL	The URL for logging out of Access Manager 11*g*. It contains the host name and port number of Access Manager 11*g*. This property is present in the move plan if an SP Engine uses Access Manager 11*g*.	`http://oam11g_host:`*`oam11g_port`*`/logout.jsp`

Table 20–22 describes the move plan properties that you can change for Oracle BI EE and Oracle BI Publisher. Edit all properties, such as host names, port numbers, listen addresses, that have different values in the target environment.

Table 20–22 Move Plan Properties for Oracle BI EE and Oracle BI Publisher

Property	Description	Sample Value
Properties in the XMLP-SERVER-CONFIG configGroup:	Oracle BI Publisher configuration	
SAW_SERVER	The name of the host that is running the Oracle BI Presentation Services to which you must connect.	`example_host`
SAW_PORT	The port number for connecting to Oracle BI Presentation Services.	`10217`
SAW_PASSWORD	The absolute path of a secure file that contains the password for Oracle BI Presentation Services.	`/scratch/oracle/bip_passwd`
SAW_USERNAME	The user name for Oracle BI Presentation Services.	`user1`
Properties in the XMLP-DATASOURCES configGroup:	Data source configuration	
Properties in the connection configProperty:	A sub-property of dataSource. Specifies that the data source is a Connection type. Each data source can be either a Connection or File type.	
url	The URL of the connection.	`jdbc:oracle:thin:@`*`host:port:sid`*
driver	The driver to use for the connection	`oracle.jdbc.OracleDriver`
username	The user name for the connection.	`user1`
password	The absolute path of a secure file that contains the password for the connection.	`/scratch/oracle/ds_conn_passwd`
Property in the file configProperty:	A sub-property of dataSource. Specifies that the data source is a File type.	
path	The file system path that points to the relevant data source file.	`/scratch/oracle/Middleware/user_projects/domains/BIDomain/config/bipublisher/repository/DemoFiles`
Properties in the XDO-CLIENT_CONFIG configGroup:	Oracle BI Publisher client	
XMLPClientDirPath	The absolute path to the Oracle BI Publisher client directory.	`/scratch/instance/domains/example.com/CommonDomain/config`
* (any name)	The Oracle BI Publisher configured endpoint connecting URL. The move plan may have more than one endpoint.	`http://example.com:10621/xmlpserver`
Properties in the XMLP-SCHEDULER-JMS-CONFIG configGroup:	The Scheduler configuration	
JMS_WEBLOGIC_JNDI_URL	The Oracle WebLogic Server JNDI URL for the Oracle BI EE Managed Server.	`cluster:t3://bi_cluster`
JMS_Shared_Temp_Directory	The JMS shared temp directory used in an Oracle BI EE cluster environment.	`/scratch/oracle/instance/instance1/BIPublisher/biptemp`

Table 20–22 (Cont.) Move Plan Properties for Oracle BI EE and Oracle BI Publisher

Property	Description	Sample Value
Properties in the XMLP-PROVIDER-CONFIG configGroup, in the provider configProperty:	Oracle BI Publisher provider There is a separate configProperty for each BI Publisher provider configured.	
uri	The URI for the Oracle BI Publisher provider.	`http://example.com:10603/bip`
nonSSOUri	The non-SSO URI for the Oracle BI Publisher provider.	`bip`
Property in the XDO-SERVER_CONFIG configGroup:	Oracle BI Publisher Server	
XMLPServerConfigDirPath	The absolute path to the Oracle BI Publisher server configuration directory.	`/scratch/oracle/Middleware/user_projects/domains/BIDomain/config/bipublisher`
Property in the RepositoryDirPath configProperty:	The directory path of the Oracle BI Publisher server repository.	
path	The directory of the Oracle BI Publisher server configuration repository. (It can be located outside of the BIDomain that is specified as the server `xdo.server.config.dir`' system property.) The default value is ${xdo.server.config.dir}/repository.	`/scratch/oracle/instance/instance1/BIPublisher/repository`
Property in the SawSourcePath configProperty:	The Oracle BI Publisher server connection resource details for the Oracle BI EE Presentation Server	
source	The path for the Oracle BI Publisher repository directory.	`/scratch/oracle/Middleware/user_projects/domains/BIDomain/config/bipublisher/repository`
Properties in the OracleInstances configGroup:	Oracle BI EE domain configuration	
instanceHome	The path of the Oracle instance in which Oracle BI EE is deployed.	`scratch/oracle/Middleware/instances/instance1`
host	The host where the Oracle BI EE Oracle instance is configured.	`example.com`
Properties in the BIInstanceDeployment configProperty:	Instance configuration details.	
listenAddress	The listen address for the host. It can be set to a virtual IP address or a subset on a multi-homed computer. You can specify an asterisk to specify multiple network addresses for the host.	`example.com` `*`
portRangeStart	The start of the range of ports used by the Oracle BI EE system components.	`10206`
portRangeEnd	The end of the range of ports used by the Oracle BI EE system components.	`10214`

Table 20–22 (Cont.) Move Plan Properties for Oracle BI EE and Oracle BI Publisher

Property	Description	Sample Value
Properties in the BIInstance configGroup, in the EmailOptions configProperty:	Oracle BI EE instance configuration	
smtpServerName	The host name of the SMTP server.	`example.com`
port	The port number of the SMTP server.	`25`
fromDisplayName	The sender's name that is used as the display name by the Oracle BI EE system when it sends email.	`Oracle Business Intelligence`
emailAddressOfSender	The email address used by the Oracle BI EE system when it sends email.	`defaultuser@defaultmailserver.com`
Property in the BIInstance configGroup, in the MarketingOptions configProperty:	Oracle BI EE instance configuration	
url	The base URL used by the Oracle BI EE system when the emails have embedded URLs.	`http://example.com:7012/_dav/cs/idcplg`
Property in the BIInstance configGroup, in the PresentationServerOptions configProperty:	Presentation Server configuration.	
webCatalogLocation	The absolute path of the location of the Oracle BI Presentation Catalog.	`/scratch/oracle/instance/instance1/OracleBIPresentationServicesComponent/coreapplication_obips1/catalog/OracleBIApps`
Property in the BIInstance configGroup, in the Scheduler configProperty:	Scheduler configuration	
dataSource	The connection details for the Oracle BI Scheduler data source.	`(DESCRIPTION=(ADDRESS_LIST=(ADDRESS=(PROTOCOL=TCP)(HOST=example.com)(PORT=1565)))(CONNECT_DATA=(SERVICE_NAME=d8b4lfc1))`
Properties in the BIInstance configGroup, in the ServerOptions configProperty:	Server options configuration	
repositoryDataSourceName	The name of the data source for the Oracle BI repository (RPD) file.	`Star`
repositoryName	The name of the RPD.	`OracleBI_BI0002`
repositorySharedLocation	The shared location for the RPD.	`/scratch/oracle/instance/BIShared/OracleBIServerComponent/coreapplication_obis1/repository`
Property in the BIInstance configGroup, in the PerformanceOptions configProperty:	Performance options configuration	

Table 20–22 (Cont.) Move Plan Properties for Oracle BI EE and Oracle BI Publisher

Property	Description	Sample Value
globalCacheStoragePath	The global location of the Oracle BI EE server cache.	`/scratch/oracle/instance/instance1/OracleBIServerComponent/coreapplication_obis1/cache`
Properties in the DEPLOY_USER_CREDENTIALS configGroup:	Oracle RTD Inline Services (BI_RTD_SPE_ILS_DEPLOY_CONFIG)	
username	The user name used to deploy the RTD SPE inline service.	`weblogic`
password	The absolute path of a secure file that contains the password for the connection.	`/scratch/oracle/rtd_passwd`
Properties in the CONNECTIONPOOLS configGroup:	RPD configuration	
user	Connection pool user name (the database schema name). The name may be a variable, in the format `VALUE_OF(`*`varname`*`)` which would then appear in the VARIABLES configGroup.	`VALUEOF(ORACLE_INITBLOCK_USER)`
datasource	RPD connection pool data source name or definition. The name may be a variable, in the format `VALUE_OF(`*`varname`*`)` which would then appear in the VARIABLES configGroup.	`VALUEOF(ORACLE_INITBLOCK_DSN)`
appServerName	If this is an ADF connection, the Business Component URL.	`http://example.com:10603/fscmAnalytics/obieebroker`
password	The absolute path of a secure file that contains the password for the connection to the RPD data source.	`/scratch/oracle/rpd_ds_conn_passwd`
Properties in the VARIABLES configGroup:	Definition of variables	
name	The name of the variable that is used in the RPD connection pool definitions. There can be multiple name/value pairs.	`ORACLE_INITBLOCK_USER`
value	The value of the variable that is used in the RPD connection pool definitions. There are multiple name/value pairs.	`'ORA_INIT_USER'`

Table 20–23 describes the move plan properties that you can change for Oracle BI EE Data Warehouse Administration Console. Edit all properties, such as host names, port numbers, listen addresses, that have different values in the target environment.

Table 20–23 Move Plan Properties for Oracle BI EE Data Warehouse Administration Console (DAC)

Property	Description	Example
Properties in the DAC-SERVER-CONFIGURATION configGroup:	DAC configuration	
jdbc.url	The URL to connect to the DAC repository.	`jdbc:oracle:thin:@example.com:1521/example.com`
jdbc.driver	The name of the JDBC driver.	`oracle.jdbc.driver.OracleDriver`

Table 20–23 (Cont.) Move Plan Properties for Oracle BI EE Data Warehouse Administration Console (DAC)

Property	Description	Example
Username	The user name used to connect to the DAC repository.	`IMPORT_DAC`
Password File	The absolute path of a secure file containing the password for the user to connect to the DAC repository. You must provide a password file, even if you are not changing the configuration.	`/scratch/biplatform/cloning/dac_passwd`
Properties in the EMAIL-CONFIGURATION configGroup:	Email configuration	
email_host	The host name of the email server.	`example`
email_protocol	The protocol for the email server.	`smtp`
email_address	The email address of the user.	`test@test.te`
needs_authentication	The flag indicating whether the corporate email server requires authentication. Valid values are `true` or `false`.	`true`
needs_ssl	The flag indicating whether an SSL connection is required. Valid values are `true` or `false`.	`false`
email_host_port	The port where the email server listens.	`5555`
email_user	User name for the email account.	`test`
email_password	The absolute path of a secure file containing the password for the user of the email server. (Only required if needs_authentication is true.)	`/scratch/biplatform/cloning/email_passwd`
Properties in the DATAWAREHOUSE-CONFIGURATION configGroup:	Data Warehouse configuration	
jdbc.url	The URL to connect to the Data Warehouse.	`jdbc:oracle:thin:@example.com:1521/example.com`
jdbc.driver	The name of the JDBC driver.	`oracle.jdbc.driver.OracleDriver`
Username	The user name used to connect to the Data Warehouse.	`IMPORT_DW`
Password File	The absolute path of a secure file containing the password for the user to connect to the Data Warehouse. You must provide a password file, even if you are not changing the configuration.	`/scratch/biplatform/cloning/DW_passwd`
Properties in the INFORMATICA-CONFIGURATION configGroup:	Informatica configuration	
Informatica server home	The Informatica server home.	`/scratch/infahome/`
Domains infa file location	The domain's infa file location.	`/scratch/infahome/domains.info`
InformaticaParameterFileLocation	The directory where the Informatica parameter files are stored (or DEFAULT).	`DEFAULT`

Table 20–23 (Cont.) Move Plan Properties for Oracle BI EE Data Warehouse Administration Console (DAC)

Property	Description	Example
Properties in the DATASOURCES-CONNECTION-DETAILS configGroup:	Data source connection information	
type	The physical data source type. Possible values are: Source, Warehouse, Informatica Repository, DAC Repository, Other.	`Source`
Connection Type	The type of database connection. Possible values are: BI Server, Oracle (OCI8), Oracle (Thin), DB2, DB2-390, MSSQL, Teradata, Flat File.	`Oracle (Thin)`
Connection String	The data source connection string. If you are using: ▪ Oracle (OCI8): Use the tnsnames entry. ▪ Oracle (Thin): Use the instance name. SQL Server: Use the database name. ▪ DB2-UDB/DB2-390: Use the connect string as defined in the DB2 configuration. ▪ Teradata: Use the database name.	`orcl.example.com`
Table Owner	The user name of the table owner.	`DB_USER`
Host	The host name of the server where the database resides.	`example.com`
Port	The port number where the database receives requests.	`1521`
JDBC Driver (Optional)	The JDBC driver for the data source connection. The value in this field must conform to the database specifications.	`oracle.jdbc.driver.OracleDriver`
URL (Optional)	The JDBC URL for the data source connection. The value in this field must conform to the database specifications.	`jdbc:oracle:thin:@example.com:1521/orcl.example.com`
Password File	The absolute path of a secure file containing the password for the user to connect to the data source. You must provide a password file, even if you are not changing the configuration.	`/scratch/biplatform/cloning/ds_passwd`
Connection Pool Name (BIPool)	The connection pool name.	`FSCM_OLTP."Connection Pool"`
Database Type (BIPool)	Database type of the transactional data source.	`Oracle`
Properties in the EXTERNAL-EXECUTORS configGroup:	External executors configuration	

Table 20–23 (Cont.) Move Plan Properties for Oracle BI EE Data Warehouse Administration Console (DAC)

Property	Description	Example
Execution type	The execution type for the tasks that will be executed by the external executor.	`ODI 11g Embedded Agent`
name	The name of the property that must be configured to integrate DAC with other Extract, Transform, and LoadExtract, Transform, and Load (ETL) tools. There are multiple properties for the external executors. Name is the name of the property. Value is the value that defines the property.	`<name>ODIUser</name>` `<value>TestUser</value>`

Table 20–24 describes the move plan properties that you can change for Oracle Essbase. Edit all properties, such as host names, port numbers, listen addresses, that have different values in the target environment.

Table 20–24 Move Plan Properties for Oracle Essbase

Property	Description	Example
Properties in the EssbaseAgentConfig configGroup:	Oracle Essbase configuration	
ARBORPATH	The absolute path for ARBORPATH.	`/scratch/oracle/shared_essbase`
PortRange	The port range for Oracle Essbase.	`9000-9499`
agent-port	The port number of the Oracle Essbase agent.	`9799`
EssbaseAdminUserName	The administration user name for Oracle Essbase.	`weblogic`
EssbaseAdminPassword	The absolute path of a secure file containing the password for the Oracle Essbase administration user.	`/scratch/oracle/essbase_passwd`
Properties in the ASOAppsTableSpaceCustomizations configGroup:	Aggregate Storage (ASO) application configuration. Each aggregate storage (ASO) application has a name and Table Space property, followed by the properties in the configGroup.	
file_location	The absolute path of the application file.	`/scratch/oracle/aso`
max_file_size	The maximum size of the file, in Bytes.	`1.34217727E8`
max_disk_size	The maximum size of the disk, in Bytes.	`4.294967295E9`
Properties in the BSOAppsDiskVolumeCustomizations configGroup:	Block Storage (BSO) application configuration. Each Essbase block storage (BSO) application has a name and database name property, followed by the properties in the configGroup.	
volume	The location of the disk volume.	`/scratch/biplatform`
file_type	The file type of the disk volume, such as index, data, or index_data.	`index_data`
file_size	The size of the volume, in Bytes.	`2.147483648E9`
partition_size	The size of the partition, in Bytes.	`9.007199254739968E15`

Table 20–25 describes the move plan properties that you can change for the EPM registry. Edit all properties, such as host names, port numbers, listen addresses, that have different values in the target environment.

Table 20–25 Move Plan Properties for the EPM Registry

Property	Description	Example
Properties in the reg.properties configGroup:	EPM Registry	
jdbc.url	The URL to connect to the EPM Registry database.	`jdbc:oracle:thin:@example.com:17328/example.com`
jdbc.driver	The name of the JDBC driver.	`oracle.jdbc.OracleDriver`
username	The user name used to connect to the EPM database.	`USER_BIPLATFORM`
password	The absolute path of a secure file containing the password for the user to connect to the EPM database. You must provide a password file, even if you are not changing the configuration.	`/scratch/biplatform/epm_jdbc_passwd`
Properties in the EPM_COMPONENTS configGroup, in the DATABASE_CONN configProperty:	EPM components configuration	
host	The database server host name.	`example.com`
dbUserName	The database user name.	`FUSION_BIPLATFORM`
dbJdbcUrl	The JDBC URL to connect to the database.	`jdbc:oracle:thin:@example.com:1570/db20258`
dbName	The database name. For Oracle Database, use the service name or SID.	`db20258`
dbPort	The database port number.	`1570`
dbPassword	The absolute path of a secure file containing the password for the user to connect to the database. You must provide a password file, even if you are not changing the configuration.	`/scratch/biplatform/epm_db_passwd`
Properties in the EPM_COMPONENTS configGroup, in the Default configProperty:	EPM components configuration	
host	The host name of the front-end Web server or load balancer.	`example2.com`
port	The port number of the front-end Web server or load balancer. This is a subentry to the Default property.	`10621`
isSSL	The flag indicating whether the front end is in SSL mode. Valid values are `true` and `false`.	`false`
SSLPort	The SSL port number of the front-end Web server or load balancer	`10218`
Properties in the WORKSPACE_APP configProperty:	Workspace configuration.	

Table 20–25 (Cont.) Move Plan Properties for the EPM Registry

Property	Description	Example
host	The host name of the server hosting the Workspace Web application.	`example.com`
port	The port where the Workspace Web application is running.	`10217`
SSLPort	The SSL port (if configured for SSL) where the Workspace Web application is running.	`10218`
Properties in the EPM_ COMPONENTS configGroup, in the WEB_ SERVER configProperty:	Web server configuration	
host	The host name of the Web server that the Web application is configured to use.	`example.com`
port	The port where the Web application is running.	`10217`
isSSL	The flag indicating whether the front end is in SSL mode. Valid values are `true` and `false`.	`false`
Properties in the EPM_ COMPONENTS configGroup, in the CALC_ WEBAPP configProperty:	EPM components configuration	
host	The host name of the server hosting the Oracle Calculation Manager Web application.	`example.com`
port	The port where the Oracle Calculation Manager Web application is running.	`10217`
SSL_Port	The SSL port (if configured for SSL) where the Oracle Calculation Manager Web application is running.	`10218`
name	The name of the Oracle Essbase cluster. There may be more than one cluster.	`EssbaseCluster-1`
Properties in the essbaseserver*n* configProperty:	Oracle Essbase server	
host	The host name of the Oracle Essbase server.	`example.com`
arborPath	The ARBORPATH of the Oracle Essbase server.	`/scratch/oracle/shared_essbase`
ess_AppLocation	The location of the Oracle Essbase application location.	`/scratch/biplatform/instances/instance1/Essbase/essbaseserver1`
agent_PortNumber	The agent port number of the Oracle Essbase server.	`9511`
agent_StartPort	The start of the range of ports used by the agent for Oracle Essbase server.	`9000`
agent_StopPort	The end of the range of ports used by the agent for Oracle Essbase server.	`9499`
Properties in the BIEE_ WEBAPP configProperty:	Oracle BI EE Web application configuration	
host	The host name of the server hosting the Oracle BI EE Web application.	`example.com`

Table 20–25 (Cont.) Move Plan Properties for the EPM Registry

Property	Description	Example
port	The port where the Oracle BI EE Web application is running.	`10217`
SSL_Port	The SSL port (if configured for SSL) where the Oracle BI EE Web application is running.	`10218`
Properties in the PROVIDER_SERVICES_WEB_APP configProperty:	The Oracle Essbase APS Web application configuration	
host	The host name of the server hosting the Oracle Essbase APS Web application.	`example.com`
port	The port where the Oracle Essbase APS Web application is running.	`10217`
SSL_Port	The SSL port (if configured for SSL) where the Oracle Essbase APS Web application is running.	`10218`
Properties in the PFINANCIAL_REPORTING_WEB_APP configProperty:	The Financial Reporting Web application configuration	
host	The host name of the server hosting the Financial Reporting Web application.	`example.com`
port	The port where the Financial Reporting Web application is running.	`10217`
SSL_Port	The SSL port (if configured for SSL) where the Web application is running.	`10218`

Table 20–26 describes the move plan properties that you can change for Oracle BI Action Framework. Edit all properties, such as host names, port numbers, listen addresses, that have different values in the target environment.

Table 20–26 Move Plan Properties for Oracle BI Action Framework

Property	Description	Sample Value
Property in the location-alias configGroup:	Action Framework configuration	
alias_name	The URL corresponding to the action name. Note that there may be more than one name/value pair.	`http://example.com:9704/analytics`

Table 20–27 describes the move plan properties that you can change for Oracle WebCenter Content Server, Oracle WebCenter Content: Records, and Oracle WebCenter Content: Inbound Refinery. Edit all properties, such as host names, port numbers, listen addresses, that have different values in the target environment.

You must edit the properties for each component under the appropriate componentType.

Table 20–27 Move Plan Properties for WebCenter Content Server, Records, and Inbound Refinery

Property	Description	Sample Value
Properties in componentType	The componentType is Webcenter Content - Records, Content Server, or Inbound Refinery.	
MoveType	The flag indicating whether to copy the entire test system instance, including configuration and data, or to create a new content server instance, based on the test system configuration. Valid values are `copy` and `init`. This property is not applicable to Inbound Refinery.	`copy`
Properties in the copy configGroup:	Copy the configuration and data	

IntradocDir	The path to the Intradoc directory. The directory value can begin with the string `{domainHome}` which will be substituted with the path of the domain home directory on the target system.	`/scratch/mw_home1/user_projects/domains/`*`domain_name`*`/ucm/cs`
		or
		`{domainHome}/ucm/cs`

WeblayoutDir	The path to the Weblayout directory. The directory value can begin with the string `{domainHome}` which will be substituted with the path of the domain home directory on the target system.
	This property may not be present if the WebLayoutDir is located in the default location, under the Intradoc directory.
		`/scratch/mw_home1/user_projects/domains/`*`domain_name`*`/ucm/cs/weblayout`
		or
		`{domainHome}/ucm/cs/weblayout`

VaultDir	The absolute path to the Vault directory. The directory value can begin with the string `{domainHome}` which will be substituted with the path of the domain home directory on the target system.
	This property may not be present if the VaultDir is located in the default location, under the Intradoc directory.
		`/scratch/mw_home1/user_projects/domains/`*`domain_name`*`/ucm/cs/vault`
		or
		`{domainHome}/ucm/cs/vault`

UserProfilesDir	The absolute path to the user profiles directory. The directory value can begin with the string `{domainHome}` which will be substituted with the path of the domain home directory on the target system.
	This property may not be present if it is located in the default location, under the Intradoc directory.
		`/scratch/mw_home1/user_projects/domains/`*`domain_name`*`/ucm/cs/data/users/profiles`
		or
		`{domainHome}/ucm/cs/users/profiles`

SocketHostAddressSecurityFilter	The security filter which lists the hosts that are allowed to directly access the server port. You can specify multiple values by separating them with a vertical bar (`|`).	`127.0.0.1|0.0.0.0.0.0.0.1`
Properties in the init configGroup:	Create a new instance with the configuration of the source.	
SocketHostAddressSecurityFilter	The security filter which lists the hosts that are allowed to directly access the server port. You can specify multiple values by separating them with a vertical bar (`|`).	`127.0.0.1|0.0.0.0.0.0.0.1`

Table 20–28 describes the move plan properties that you can change for Oracle WebCenter Content: Imaging. Edit all properties, such as host names, port numbers, listen addresses, that have different values in the target environment.

Table 20–28 Move Plan Properties for Oracle WebCenter Content: Imaging

Property	Description	Sample Value
AdminUser	Administrative user ID used during the `pasteConfig` operation to seed system security. If left blank, the domain administrator user provided on the `pasteConfig` command line is used. This property should be used for situations where the Imaging administrative user must be a user other than the domain administrator.	`Admin2`
Properties in the MBean Settings configGroup:	MBeans configuration	
InputAgentInputDirectories	A comma-separated list of directories where input sources look for work.	`IPM/InputAgent/Input`
InputSampleDirectory	The directory that holds the sample data for the input UI.	`IPM/InputAgent/Input/Sample`
RenderGDFontPath	Location of the TrueType (TTF) font files used by the OIT rendering package.	`/usr/share/X11/fonts/TTF`
Properties in the UCM Connection configGroup:	The WebCenter Content connection configuration	
repository.machine	The location of the repository. The value must be `localhost` if the connection is configured for "Use Local Content Server."	`localhost`
repository.port	The WebCenter Content server port used when the local content server is used. If not using local content server connection, remove the configuration property.	`4444`
repository.useSSL	A flag that specifies whether the connection to WebCenter Content systems use SSL. Valid values are `true` or `false`.	`false`
Property in the WORKFLOW Connection configGroup:	The workflow connection configuration	
bpel.front.address	The HTTP front-end address used in the Imaging SOA: Connection Settings UI.	`http://example.com:8001`

Table 20–29 describes the move plan properties that you can change for Oracle Data Integrator. Edit all properties, such as host names, port numbers, listen addresses, that have different values in the target environment.

Table 20–29 Move Plan Properties for Oracle Data Integrator

Property	Description	Sample Value
JPS configuration file	The path of the JPS configuration file for JSE components.	`/private/t2p/jps-config.xml`
Properties in the Master Repository configGroup:	Master repository configuration	

Table 20–29 (Cont.) Move Plan Properties for Oracle Data Integrator

Property	Description	Sample Value
Master Repository Id	Oracle Data Integrator master repository ID. It should be different than the ID for the source master repository.	`502`
SUPERVISOR password file	The absolute path of a secure file that contains the password for the ODI user SUPERVISOR.	`/scratch/oracle/odi_passwd`
Schema name	The name of the schema in the target database where the target ODI repository will be created.	`odi_master_11g`
Schema password file	The absolute path of a secure file that contains the password for the schema.	`/scratch/oracle/odi_schema_passwd`
Properties in the Physical Data Servers configProperty:	Data servers configuration	
Schema name	The name of the schema for the database data servers or the directory location for file type data servers.	`FG_Dir_Schema`
Work Schema	The name of the Work schema for the database data servers or the directory location for file type data servers.	`/tmp/FG_Dir_Schema`
Url	JDBC URL for connecting to the data server.	`jdbc:oracle:thin:@localhost:1521/example.com`
User	User name for the physical data servers connection.	`username`
Password File	The absolute path of a secure file that contains the password for the user for the physical data servers connection.	`/scratch/oracle/rpd_ds_conn_passwd`
Properties in the Agents configProperty:	Agents configuration	
Password File	The absolute path of a secure file containing the password for the physical data servers connection.	`/scratch/oracle/odi_ds_passwd`
Host name	The Agent host name.	`localhost`
Host port	The Agent host port number.	`12311`
Properties in the Work repositories configProperty:	Work repositories configuration	
Work Repository Id	The ID for the work repository. It should be different than the ID for the source work repository.	`1`
Url	JDBC URL for connecting to the work repository.	`jdbc:oracle:thin:@localhost:1521/example.com`
User	User name for connecting to the work repository.	`username`
Password File	The absolute path of a secure file that contains the password for the user for the physical data servers connection.	`/scratch/oracle/odi_pds_passwd`

Table 20–29 (Cont.) Move Plan Properties for Oracle Data Integrator

Property	Description	Sample Value
Properties in the ServiceProviders configProperty:	Service provider configuration if an external Access Manager is used as an SP engine.	
OAM_SERVER_*n*	The URL of the Access Manager server.	`http://example.com:14101`
Properties in the Applications configProperty:	This section is present if an external Access Manager is used as an application.	
ReturnURL	The return URL.	`example.com:5575`

Table 20–30 describes the move plan properties that you can change for Access Manager, Secure Token Service, and Mobile and Social. Edit all properties, such as host names, port numbers, listen addresses, that have different values in the target environment.

Table 20–30 Move Plan Properties for Access Manager, Secure Token Service, and Mobile and Social

Property	Description	Sample Value
Properties in the Load Balancer configProperty:	Load Balancer-related configuration properties.	
hostname	The name of the host for the Load Balancer.	`example.com`
port	The port number for the Load Balancer.	`368`
Properties in the ManagedServer configGroup:	The Managed Server-related configuration properties. Update the following property for all Managed Servers in the move plan.	
ProxyPort	The port number for the proxy server.	`5555`
Properties in the OIM Connection Information configProperty:	Oracle Identity Manager-related configuration property. This property is present in the move plan when Access Manager and Oracle Identity Manager are integrated.	
host/Port	The URL, with the name of the host and the port number for Oracle Identity Manager.	`http://example.com:7010`
Properties in the LDAP configProperty:	The LDAP configuration. These properties are present in the move plan only if an embedded LDAP is not being used.	
LDAP Url	The URL of the LDAP connection. It contains the host name and port number of the LDAP store.	`ldap://example.com:3060`
LDAP User	The Administrative user for the LDAP server.	`cn=orcladmin`
Password File	The absolute path of a secure file containing the password for the LDAP Admin user. You must provide a password file location, even if you are not changing the configuration.	`/scratch/oam/ldap_passwd`
Properties in the ServiceProviders configProperty:	Service Provider properties. This section of the move plan is present if Access Manager acts as one of the Service Providers for Mobile and Social.	
OAM_server_name	The URL, with the name of the host and the port number for the Access Manager server.	`http://example.com:14101`

Table 20–30 (Cont.) Move Plan Properties for Access Manager, Secure Token Service, and Mobile and

Property	Description	Sample Value
Properties in the Applications configProperty:	Applications-related properties. This section of the move plan is present if Access Manager is present as one of the applications for Mobile and Social.	
ReturnURL	The return URL, reflecting the URL for the target Access Manager server.	`example.com:5575`
Properties in the MulitDataCenter configProperty:	Multiple data center properties.	
ClusterID	The ID of the cluster. This must be a unique value. It cannot be the same ID as the source cluster. The value in the move plan is blank.	`MyProdCluster`

Table 20–31 describes the move plan properties that you can change for Oracle Adaptive Access Manager. Edit all properties, such as host names, port numbers, listen addresses, that have different values in the target environment.

Table 20–31 Move Plan Properties for Oracle Adaptive Access Manager

Property	Description	Sample Value
Properties in the ServerConfig configProperty:	The Oracle Adaptive Access Manager configuration properties.	
OAM PrimaryServerHost/Port	The host name and port number of the primary server for Access Manager. This property is present if Oracle Adaptive Access Manager is integrated with an Access Manager instance.	`example.com:5575`
OAM Secondary Server Host/Port	The host name and port number of the secondary server for Access Manager. This property is present if Oracle Adaptive Access Manager is integrated with an Access Manager instance.	`example.com:5560`
OIM Managed Server Host/Port	The host name and port number of the Managed Server for Oracle Identity Manager. This property is present if Oracle Adaptive Access Manager is integrated with an Oracle Identity Manager instance.	`example.com:5000`
OAAM Login URL for OIM	The host name and port number of Oracle HTTP Server. This property is present if Oracle Adaptive Access Manager is integrated with an Oracle Identity Manager instance.	`example.com:`*`ohsport`*
Webservice URL for OTP UMS Service	The Web services URL. This property is present in the move plan if OTP User Messaging Service is enabled and use of ParlayX is disabled.	`http://`*`WebserviceURL/endpoint`*
Parlayx URL for OTP UMS Service	The ParlayX URL. This property is present in the move plan if both OTP User Messaging Service and use of ParlayX is enabled.	`http://`*`ParlayxURL/endpoint`*

Table 20–31 (Cont.) Move Plan Properties for Oracle Adaptive Access Manager

Property	Description	Sample Value
SOAP KeyStore File Location	The location of the SOAP keystore. This property is present if SOAP Authentication is enabled and the configured SOAP Keystore is not located under the Domain home. Note: If the SOAP Keystore is located under the Domain Home, the SOAP Keystore is migrated to the corresponding location under the Domain Home on the target.	`/scratch/oaam/soapks.ks`
Location for OAAM images folder(Native Integration)	The location of the Oracle Adaptive Access Manager images folder. This property is present in the move plan if SOAP Tracker is enabled and the configured OAAM image folder is not located under the Domain home. Note: If the OAAM image folder is located under the Domain Home, the images folder is migrated to the corresponding location under the Domain Home on the target.	`/scratch/oaam/oaam_images`
System Config Keystore File Location	The location of the System Config Keystore. This property is present in the move plan if the System Config Keystore location is configured, but is not located under the Domain home. Note: If the System Config Keystore is located under the Domain Home, the System Config Keystore is migrated to the corresponding location under the Domain Home on the target.	`scratch/KS_loc2`
System DB Keystore File Location	The location for the keystore having keys for securing configuration values in the database. This property is present in the move plan if the System DB Keystore location is configured, but is not located under the Domain home. Note: If the System DB Keystore is located under the Domain Home, the System DB Keystore is migrated to the corresponding location under the Domain Home on the target.	`/scratch/KS_loc3`

21

Moving from a Test to a Production Environment

This chapter describes how to move Oracle Fusion Middleware from a source environment, such as a test environment, to a target environment, such as a production environment. You can develop and test applications in a source environment, and then eventually roll out the test applications and, optionally, test data to your target environment. You can also use this approach for testing and rolling out upgrades.

This chapter includes the following topics:

21.1 Introduction to Moving Oracle Fusion Middleware Components

You can move Oracle Fusion Middleware components from a source environment to a target environment.

Moving Oracle Fusion Middleware components minimizes the amount of work that would otherwise be required to reapply all the customization and configuration changes made in one environment to another. You can install, configure, customize, and validate Oracle Fusion Middleware in a source environment. Once the system is stable and performs as desired, you can create the target environment by moving a copy of the components and their configurations from the source environment, instead of redoing all the changes that were incorporated into the source environment.

If you have an existing target environment, you can move any modifications of the source environment, such as customizations, to the target environment.

21.2 Overview of Procedures for Moving from a Source to a Target Environment

This section describes the general steps in moving installations from a source environment to a target environment.

The general steps are:

1. Prepare your source environment. See Section 21.3.1.
2. Prepare your target environment. See Section 21.3.2.
3. If your environment uses a database, create a new database or copy the database from the source environment to the target environment. See Section 21.3.3.
4. Move Oracle Identity Management to the target environment. See Section 21.4.1.
5. Move a copy of the Middleware home for the component or suite from the source environment to the target environment using the copyBinary and pasteBinary scripts. See Section 21.3.4.
6. Move a copy of the configuration of components, as described in Section 21.3.6 or Section 21.3.7. In most cases, you use the copyConfig, extractMovePlan, and pasteConfig scripts.
7. Move other data, such as UMS user messaging preferences, data for Oracle WebCenter Portal applications, or Oracle Web Cache configuration files. Modify any information that is specific to the new environment such as host name or ports. See Section 21.4 for information specific to each component.

21.3 Common Procedures for Moving to a Target Environment

Many of the Oracle Fusion Middleware components use some of the same procedures to move from a source environment to a target environment. Note, however that not all components use all or some these procedures. You **must** follow the procedures in Section 21.4 for your particular component.

This section describes the common procedures and contains the following topics:

- Preparing the Source Environment
- Preparing the Target Environment
- Installing the Database on the Target Environment
- Moving the Middleware Home and the Binary Files
- Moving Oracle Platform Security Services Data
- Moving the Configuration of Java Components
- Moving the Configuration of Oracle Instances and System Components
- Configuring Users and Groups

Note: In the scripts used in these procedures and in the move plans, you often need to provide files containing passwords. To generate a file that contains an obfuscated password, use the obfuscatePassword script, which is described in Section 20.3.1.12.

21.3.1 Preparing the Source Environment

The procedures in this chapter assume that you have installed and configured Oracle Fusion Middleware on the source environment, including some or all of the following:

- Installed one or more databases to be used by Oracle Fusion Middleware components such as Identity Management, Oracle SOA Suite, or Oracle WebCenter Portal.
- Created the needed schemas in the source environment using RCU. See the *Oracle Fusion Middleware Repository Creation Utility User's Guide*.
- Installed Oracle WebLogic Server and created the Middleware home.
- Installed and configured Identity Management.

  This can include creating the desired LDAP trees and entries, in particular, users and groups for Oracle Internet Directory, creating adapters to data sources for Oracle Virtual Directory, creating policies for Oracle Web Services Manager. In addition, it can include configuring self-signed certificates for SSL. (In the target environment, you use trusted CA-signed certificates.)
- Installed and configured Oracle Fusion Middleware components such as Oracle SOA Suite or Oracle WebCenter Portal.
- Configured security policies.
- Deployed one or more applications or SOA Composite applications. The applications may have internal and external references.

Note that all Oracle homes in the Middleware home on the source environment must be registered in the same Oracle inventory. If you have installed multiple components under the same Middleware home, but used different Oracle inventory locations, the scripts are not able to detect all of the Oracle homes.

21.3.2 Preparing the Target Environment

To use the procedures in this chapter, your target environment must meet the following prerequisites:

- You must use the cloningclient.jar file and movement scripts that are compatible with the version of the Middleware home and components that you want to copy. The procedures in this chapter presume that you are using the current version of the cloningclient.jar file and movement scripts.
- The target environment must be on the same operating system as the source environment. Also, the operating system architecture must be the same in both environments. For example, both environments must be running 32-bit operating systems or 64-bit operating systems.

  All Oracle homes in the Middleware home must be either all 32 bit or all 64 bit. The operation does not support a mix of 32-bit and 64-bit Oracle homes.

  When you execute the scripts, you must specify a matching Java home. That is, if the Oracle homes are 64 bit, you must specify a 64-bit Java home. If the Oracle homes are 32 bit, you must specify a 32-bit Java home.
- The target environment must have the same superuser or administrative user as the user at the source environment. The user's password can be different if you are using an external LDAP; you specify it on the command line when you use the pasteConfig script. After you complete the movement of the installation, you can modify the user on the target environment.

Note, however, that you cannot change the password if you are using an embedded LDAP.

- The database in the target environment must be the same type of database as in the source environment. For example, if the database in the source environment is an Oracle Database, the database in the target environment must be an Oracle Database. The database on the target environment should be the same version as on the source environment.
- If the database is not tuned correctly, the copyConfig and pasteConfig operations can incur performance issues. To avoid these performance issues, in addition to following standard database performance tuning guidelines, ensure that you have sufficient RAM allocated for your database for the import of the MDS tables. Also run statistics against the target database by executing the following procedure:

```
BEGIN
dbms_stats.gather_schema_stats(ownname => 'prefix_MDS',
          METHOD_OPT => 'FOR ALL COLUMNS SIZE AUTO',
          CASCADE => TRUE, ESTIMATE_PERCENT => NULL);
END;
```

In the procedure, *prefix*_MDS is the MDS schema name for your installation.

21.3.3 Installing the Database on the Target Environment

Many components, such as Oracle Internet Directory, Oracle SOA Suite, and Oracle WebCenter Portal, require a database.

You can install a new database or you can copy the database from the source environment:

- Install a new database:
  1. Install and configure the database software.
  2. Create the required schemas in the target database using RCU. See the *Oracle Fusion Middleware Repository Creation Utility User's Guide*.
  3. Create any custom schemas used by your applications. For example, if your application uses a custom schema in the source environment, create the schema in the target environment.
- Create a duplicate database using the Oracle Database RMAN duplicate command. The duplicate database must be created with a different DBID than the source database, so that it functions entirely independently.

  To create a duplicate Oracle Database, Release 11*g*, in the target environment:

  1. On the target environment, install the Oracle Database software, but do not create a database. To do this, select **Install Database Software only** in the Select Configuration Option screen.
  2. On the source environment, edit the tnsnames.ora file, adding an entry for the database on the target environment.

     The following shows an example of the tnsnames.ora file. In the example, testDB is the database on the source environment and prodDB is the database on the target environment.

```
testDB =
   (DESCRIPTION =
     (ADDRESS =
       (PROTOCOL = TCP)
```

```
       (HOST = 192.168.1.1)
       (PORT = 1521))
         (CONNECT_DATA =
       (SERVER = DEDICATED)
       (SID = testDB)
       )
     )
prodDB=
    (DESCRIPTION =
      (ADDRESS =
        (PROTOCOL = TCP)
        (HOST = 192.168.2.4)
        (PORT = 1521))
          (CONNECT_DATA =
        (SERVER = DEDICATED)
        (SID = prodDB)
      )
  )
```

3. On the source environment, edit the listener.ora file, adding an entry for the database on the target environment.

   The following shows the added entry:

```
LISTENER_mts =
  (DESCRIPTION_LIST =
    (DESCRIPTION =
      (ADDRESS = (PROTOCOL = TCP)
      (HOST = 192.168.2.4)
      (PORT = 1521)(IP = FIRST))
    )
  )
SID_LIST_LISTENER_mts =
  (SID_LIST =
    (SID_DESC =
      (SID_NAME = prodDB)
      (ORACLE_HOME = /scratch/oracle/test)
    )
  )
```

4. In the target environment, create a password file in the *ORACLE_HOME*/dbs directory. The sys password must be the same as the password for the sys account in the database in the source environment. The following command creates the password file:

```
orapwd password=password file=ORACLE_HOME/dbs/orapwproddb
```

5. In the target environment, create a parameter file (pfile) in the *ORACLE_HOME*/dbs directory. The file should contain only the DB_NAME parameter. For example:

```
DB_NAME=prodDB
```

6. In the target environment, set the ORACLE_SID environment variable to point to the target database if it is not already set. Then, start the database in NOMOUNT mode. For example:

```
SQL> STARTUP NOMOUNT PFILE='ORACLE_HOME/dbs/pfile'
```

7. To move the database from the source environment to the target environment, use RMAN on the target environment.

The following shows an example of using RMAN to duplicate the database.

```
RMAN
DUPLICATE TARGET DATABASE
  TO prodDB
  FROM ACTIVE DATABASE
  SPFILE
  NOFILENAMECHECK;
```

RMAN automatically copies the server parameter file to the destination host, starts the auxiliary instance with the server parameter file, copies all necessary database files and archived redo logs over the network to the destination host, and recovers the database. Finally, RMAN opens the database with the RESETLOGS option to create the online redo logs.

For detailed steps, see the *Oracle Database Backup and Recovery User's Guide*.

21.3.4 Moving the Middleware Home and the Binary Files

You can move a copy of the Middleware home to the target environment using the copyBinary and pasteBinary scripts. The Oracle WebLogic Server home, the Oracle homes, and the binary files in the Middleware home are also moved.

To move the Middleware home:

1. On Windows, at the source, stop the Administration Server and any Managed Servers running in the Middleware home. In addition, stop any Java or WebLogic processes. (On UNIX, you do not need to stop the servers.)
2. At the source, execute the copyBinary script, which copies the Middleware home and the WebLogic Server home and the Oracle homes contained within the Middleware home. If there are no Oracle homes in the source Middleware home, no Oracle homes are present in the archive.

   The copyBinary script is located in:

   ```
   (UNIX) ORACLE_COMMON_HOME/bin/copyBinary.sh
   (Windows) ORACLE_COMMON_HOME\bin\copyBinary.cmd
   ```

   See Section 20.3.1.1 for the syntax of the copyBinary script.

   For example, to copy a Middleware home that is located at /scratch/Oracle /Middleware1, use the following command:

   ```
   copyBinary.sh -javaHome /scratch/Oracle/Middleware1/jrockit_160_20_D1.1.0-18
                 -archiveLoc /tmp/mw_copy.jar
                 -sourceMWHomeLoc /scratch/Oracle/Middleware1
                 -invPtrLoc /scratch/oracle/oraInst.loc
   ```

3. If you are copying the Middleware home to a different host, copy the archive file to that system.
4. Copy the pasteBinary script and the cloningclient.jar file to the target system and ensure that they have execute permission.

   The pasteBinary script is located in:

   ```
   (UNIX) ORACLE_COMMON_HOME/bin/pasteBinary.sh
   (Windows) ORACLE_COMMON_HOME\bin\pasteBinary.cmd
   ```

   The cloningclient.jar file is located in:

   ```
   (UNIX) ORACLE_COMMON_HOME/jlib/cloningclient.jar
   ```

```
(Windows) ORACLE_COMMON_HOME\jlib\cloningclient.jar
```

Do **not** copy the other scripts, such as pasteConfig. Those scripts are generated when you extract the files, as in step 6.

5. On Linux and UNIX, if the target system does not contain any installed Oracle products, you must create an oraInst.loc file, specifying a group whose members are given access to write to the Oracle inventory (oraInventory), and where you want to put Oracle inventory. For example, the oraInst.loc file could contain the following:

   ```
   inst_group=dba
   inventory_loc=/scratch/oracle1/oraInventory
   ```

   Then, if the location is not the default location (/etc/oraInst.loc.), use the -invPtrLoc option to the pasteBinary script to specify the location of the oraInst.loc file.

6. At the target, extract the files from the archive using the pasteBinary script, See Section 20.3.1.2 for the syntax of the pasteBinary script.

   Note: Before you run the pasteBinary script, the *parent* directory for the Middleware home must exist on the target. If it does not exist, you must create it and ensure that you have write permission to it. For example, if you want the target Middleware home to be /scratch/oracle/MW_Home_prod, create the following directory.:

   ```
   /scratch/oracle
   ```

   Note that the actual directory for the Middleware home (for example, MW_Home_prod) cannot exist.

   For example, to apply the archive to the directory /scratch/oracle/MW_Home_prod, use the following command:

   ```
   pasteBinary.sh -javaHome /scratch/Oracle/Middleware1/jrockit_160_20_D1.1.0-18
                  -archiveLoc  /tmp/mw_copy.jar
                  -targetMWHomeLoc /scratch/oracle/MW_Home_prod
   ```

   The Middleware home is extracted to /scratch/oracle/MW_Home_prod and the WebLogic Server home and all of the Oracle homes are extracted under it with the same names as that of the source Oracle home names.

7. At the target, delete the Node Manager directory and the files in it. The default location of the directory is:

   ```
   WL_hOME/common/nodemanager
   ```

   You will move the Node Manager configuration in Section 21.3.6, Step 9.

21.3.5 Moving Oracle Platform Security Services Data

If you use Oracle Platform Security Services and it is database-based, you must take the steps in this section before you execute the pasteConfig script.

Note: If the source is a file-based store and the target is an LDAP or database-based store, the policy and credential stores are moved as part of the copyConfig and pasteConfig operations.

Move the data from the source environment to the target environment:

If the data is in a database:

1. If you have not already done so, create the Oracle Platform Security Services schema using RCU as described in Section 21.3.4.
2. Set the environment variables and change to the Oracle home directory:

```
setenv ORACLE_HOME ORACLE_HOME
setenv ORACLE_SID ORACLE_SID
cd $ORACLE_HOME/bin
```

3. Export the data from the source schema:

```
expdp "sys/password@connect_id as sysdba"
           DIRECTORY=DATA_PUMP_DIR SCHEMAS=OPSS_schema_name
           DUMPFILE=export.dmp PARALLEL=2 LOGFILE=export.log
```

4. Copy the .dmp file to the DATA_PUMP_DIR directory in the target environment.
5. Import the data into the target schema:

```
impdp "sys/password@connect_id as sysdba"
          DIRECTORY=DATA_PUMP_DIR DUMPFILE=export.dmp
          PARALLEL=2 LOGFILE=import.log
          remap_schema=test_env_schema_name:prod_env_schema_name
          remap_tablespace=test_env_tablespace:prod_env_tablespace
          TABLE_EXISTS_ACTION=REPLACE
```

21.3.6 Moving the Configuration of Java Components

You can move a copy of the domain configuration for Java components, such as Oracle SOA Suite, using the copyConfig, extractMovePlan, and pasteConfig scripts. This step moves a copy of the configuration, including the domain, the Administration Server and Managed Servers. Then, it starts the Administration Server. You also move a copy of the Node Manager configuration.

Because in most cases the user-specific data is not the same in the target environment as in the source environment, this process does not move user-specific data.

Notes:

- When you move the configuration of a component, the scripts replicate the topology of the source. For example, if the source domain contains Managed Servers server_1 and server_2 on Host A and Managed Servers server_3 and server_4 on Host B, you must specify a similar relationship between Managed Servers and hosts at the target. (You specify the hosts for each Managed Server in the move plan.)
- The domain directory is local to each machine. The pasteConfig script is performed only on the Administration Server domain directory. Subsequently, if the Managed Servers are not in the same directory as the Administration Server, you must re-create the domain directory for those Managed Servers by using the Oracle WebLogic Server pack and unpack commands. For more information, see *Oracle Fusion Middleware Creating Templates and Domains Using the Pack and Unpack Commands*.

To move a copy of the domain configuration and Node Manager configuration:

1. At the source, make sure that the Administration Server and all Managed Servers are started.
2. At the source, copy the domain configuration by executing the copyConfig script.

   The copyConfig script is located in:

   ```
   (UNIX) ORACLE_COMMON_HOME/bin/copyConfig.sh
   (Windows) ORACLE_COMMON_HOME\bin\copyConfig.cmd
   ```

   See Section 20.3.1.3 for the syntax of the copyConfig script.

   For example, to copy the configuration of the Oracle SOA Suite domain named SOA_domain1 in the Middleware home /scratch/Oracle/Middleware1, use the following command:

   ```
   copyConfig.sh -javaHome /scratch/Oracle/Middleware1/jrockit_160_20_D1.1.0-18
                 -archiveLoc /tmp/soa.jar
                 -sourceDomainLoc /scratch/Oracle/Middleware1/user_
   projects/domains/SOA_domain1
                 -sourceMWHomeLoc /scratch/Oracle/Middleware1
                 -domainHostName example.com
                 -domainPortNum 8001
                 -domainAdminUserName domain_admin_username
                 -domainAdminPassword /scratch/admin/passwd.txt
                 -logDirLoc /tmp/logs
   ```

3. If you are copying the domain configuration to a different host, copy the archive file to that system.
4. At the source, extract the move plan from the archive, using the extractMovePlan script.

   The extractMovePlan script is located in:

   ```
   (UNIX) ORACLE_COMMON_HOME/bin/extractMovePlan.sh
   (Windows) ORACLE_COMMON_HOME\bin\extractMovePlan.cmd
   ```

   See Section 20.3.1.7 for the syntax of the extractMovePlan script.

   For example:

   ```
   extractMovePlan.sh -javaHome /scratch/Oracle/Middleware1/jrockit_160_20_
   D1.1.0-18
                      -archiveLoc /tmp/soa.jar
                      -planDirLoc /tmp/Oracle/t2p_plans/soa
   ```

5. Edit the move plan, modifying the properties to reflect the values for the target environment. See Table 20–12 to find the list of properties for the type of component you are moving.

   If you are moving only one domain in an environment that contains more than one domain, in the move plan, remove the entries for the domains that you are not moving.
6. Copy the edited move plan to the target. (During the pasteConfig operation, you specify the location using the -movePlanLoc option.)
7. At the target, run the following script to generate obfuscated password files required by the move plan. Run the script for each password file.

   ```
   (UNIX) ORACLE_COMMON_HOME/bin/obfuscatePassword.sh
   (Windows) ORACLE_COMMON_HOME\bin\obfuscatePassword.cmd
   ```

The script prompts you to enter the password and the location where the password file is to be written.

8. At the target, extract the files from the archive using the pasteConfig script.

   The pasteConfig script is located in:

   ```
   (UNIX) ORACLE_COMMON_HOME/bin/pasteConfig.sh
   (Windows) ORACLE_COMMON_HOME\bin\pasteConfig.cmd
   ```

   See Section 20.3.1.8 for the syntax of the script.

   For example, to apply the archive to the Middleware home /scratch/Oracle/Middleware1, use the following command:

   ```
   pasteConfig.sh -javaHome /scratch/Oracle/Middleware1/jrockit_160_20_D1.1.0-18
                  -archiveLoc /tmp/soa.jar
                  -movePlanLoc /tmp/Oracle/t2p_plans/soa/moveplan.xml
                  -targetDomainLoc /scratch/Oracle/Middleware1/user_
   projects/domains/SOA_domain1
                  -targetMWHomeLoc /scratch/Oracle/Middleware1/
                  -domainAdminPassword /scratch/pwd_dir/pass.txt
   ```

9. At the source, copy the Node Manager configuration, by executing the copyConfig script.

   The copyConfig script is located in:

   ```
   (UNIX) ORACLE_COMMON_HOME/bin/copyConfig.sh
   (Windows) ORACLE_COMMON_HOME\bin\copyConfig.cmd
   ```

   See Section 20.3.1.6 for the syntax of the script. For example, use the following command:

   ```
   copyConfig.sh -javaHome /scratch/Oracle/Middleware1/jrockit_160_20_D1.1.0-18
                 -archiveLoc /tmp/nm.jar
                 -sourceNMHomeLoc /scratch/Oracle/Middleware/wlserver_
   10.3/common/nodemanager
                 -logDirLoc /tmp/logs
   ```

10. If you are copying the Node Manager to a different host, copy the archive file to that system.

11. At the source, extract the move plan from the archive, using the extractMovePlan script.

    The extractMovePlan script is located in:

    ```
    (UNIX) ORACLE_COMMON_HOME/bin/extractMovePlan.sh
    (Windows) ORACLE_COMMON_HOME\bin\extractMovePlan.cmd
    ```

    See Section 20.3.1.7 for the syntax of the extractMovePlan script.

    For example:

    ```
    extractMovePlan.sh -javaHome /scratch/Oracle/Middleware1/jrockit_160_20_
    D1.1.0-18
                       -archiveLoc /tmp/nm.jar
                       -planDirLoc /tmp/Oracle/t2p_plans/nm
    ```

12. Edit the move plan, modifying the properties to reflect the values for the target environment. See Table 20–13 to find the list of properties for Node Manager.

13. Copy the edited move plan to the target. (During the pasteConfig operation, you specify the location using the -movePlanLoc option.)

14. At the target, run the following script to generate obfuscated password files required by the move plan. Run the script for each password file.

```
(UNIX) ORACLE_COMMON_HOME/bin/obfuscatePassword.sh
(Windows) ORACLE_COMMON_HOME\bin\obfuscatePassword.cmd
```

    The script prompts you to enter the password and the location where the password file is to be written.

15. At the target, extract the files from the archive using the pasteConfig script.

    The pasteConfig script is located in:

```
(UNIX) ORACLE_COMMON_HOME/bin/pasteConfig.sh
(Windows) ORACLE_COMMON_HOME\bin\pasteConfig.cmd
```

    See Section 20.3.1.11 for the syntax of the script.

    For example, use the following command:

```
pasteConfig -javaHome USER_HOME/jrockit_160_17_R28.0.0-679/
            -archiveLoc /tmp/nm.jar
            -targetNMHomeLoc /scratch/Oracle/Middleware1/wlserver_
10.3/common/nodemanager
            -targetMWHomeLoc /scratch/Oracle/Middleware1
            -movePlanLoc /tmp/Oracle/t2p_plans/nm/moveplan.xml
            -silent true
```

When you complete this task, you need to perform additional steps for each component, as described in Section 21.4.

21.3.7 Moving the Configuration of Oracle Instances and System Components

You can move and Oracle instance and system components, such as Oracle HTTP Server, Oracle Internet Directory, Oracle Virtual Directory, and Oracle BI EE, in one of the following ways:

- Moving an Oracle Instance and All of Its System Components
- Moving an Individual System Component

In both cases, you use the copyConfig, extractMovePlan, and pasteConfig scripts. The only difference is in the options that you pass to the scripts.

21.3.7.1 Moving an Oracle Instance and All of Its System Components

You can move the entire Oracle instance, including the configuration of all of the system components within that instance.

Take the following steps:

1. At the source, execute the copyConfig script.

    The copyConfig script is located in:

```
(UNIX) ORACLE_COMMON_HOME/bin/copyConfig.sh
(Windows) ORACLE_COMMON_HOME\bin\copyConfig.cmd
```

    See Section 20.3.1.4 for the syntax of the script.

    For example, to copy the Oracle instance located in /scratch/Oracle/Middleware1/webtier_1, use the following command:

```
copyConfig.sh -javaHome /scratch/Oracle/Middleware1/jrockit_160_20_D1.1.0-18
              -archiveLoc /tmp/ohs1.jar
              -sourceInstanceHomeLoc /scratch/Oracle/Middleware1/webtier_1
```

2. If you are copying the component to a different host, copy the archive file to that system.
3. At the source, extract the move plan from the archive, using the extractMovePlan script.

   The extractMovePlan script is located in:

   ```
   (UNIX) ORACLE_COMMON_HOME/bin/extractMovePlan.sh
   (Windows) ORACLE_COMMON_HOME\bin\extractMovePlan.cmd
   ```

   See Section 20.3.1.7 for the syntax of the extractMovePlan script.

   For example:

   ```
   extractMovePlan.sh -javaHome /scratch/Oracle/Middleware1/jrockit_160_20_
   D1.1.0-18
                   -archiveLoc /tmp/ohs1.jar
                   -planDirLoc /tmp/Oracle/t2p_plans/ohs
   ```

4. Edit the move plan, modifying the properties for the particular components to reflect the values for the target environment:
   - For Oracle HTTP Server, see Table 20–18.
   - For Oracle Internet Directory, see Table 20–19.
   - For Oracle Virtual Directory, see Table 20–20.
   - For Oracle BI EE, see Table 20–22.
5. Copy the edited move plan to the target. (During the pasteConfig operation, you specify the location using the -movePlanLoc option.)
6. At the target, run the following script to generate obfuscated password files required by the move plan. Run the script for each password file.

   ```
   (UNIX) ORACLE_COMMON_HOME/bin/obfuscatePassword.sh
   (Windows) ORACLE_COMMON_HOME\bin\obfuscatePassword.cmd
   ```

   The script prompts you to enter the password and the location where the password file is to be written.
7. At the target, extract the files from the archive using the pasteConfig script.

   The pasteConfig script is located in:

   ```
   (UNIX) ORACLE_COMMON_HOME/bin/pasteConfig.sh
   (Windows) ORACLE_COMMON_HOME\bin\pasteConfig.cmd
   ```

   See Section 20.3.1.9 for the syntax of the script.

   For example, to apply the archive to the Oracle instance webtier_2, use the following command:

   ```
   pasteConfig.sh -javaHome /scratch/Oracle/Middleware/jrockit_160_20_D1.1.0-18
              -archiveLoc /tmp/ohs1.jar
              -movePlanLoc /tmp/Oracle/t2p_plans/ohs/moveplan.xml
              -targetOracleHomeLoc /scratch/Oracle/Middleware/Oracle_WebTier
              -targetInstanceHomeLoc /scratch/Oracle/Middleware/webtier_2
              -targetInstanceName webtier_2
              -domainHostName myhost
   ```

```
-domainPortNum 7001
-domainAdminUserName domain_admin_username
-domainAdminPassword domain_admin_password_file
```

Note that the Oracle instance name must be unique in the domain. If you are applying the archive of the Oracle instance to the same domain, use the -targetInstanceName option to specify a different name for the instance.

When you complete this task, you need to perform additional steps for each component, as described in Section 21.4.

21.3.7.2 Moving an Individual System Component

You can move the configuration of an individual system component within an Oracle instance.

Take the following steps:

1. At the source, execute the copyConfig script. Copying an individual component, is similar to copying an Oracle instance, except that you add the -sourceComponentName option.

   The copyConfig script is located in:

   ```
   (UNIX) ORACLE_COMMON_HOME/bin/copyConfig.sh
   (Windows) ORACLE_COMMON_HOME\bin\copyConfig.cmd
   ```

   See Section 20.3.1.5 for the syntax of the script.

   For example, to copy the Oracle HTTP Server instance named ohs1 in the Oracle instance located in /scratch/Oracle/Middleware1/webtier_1, use the following command:

   ```
   copyConfig.sh -javaHome /scratch/Oracle/Middleware1/jrockit_160_20_D1.1.0-18
                 -archiveLoc /tmp/ohs1.jar
                 -sourceInstanceHomeLoc /scratch/Oracle/Middleware1/webtier_1
                 -sourceComponentName ohs1
   ```

2. If you are copying the component to a different host, copy the archive file to that system.

3. At the source, extract the move plan from the archive, using the extractMovePlan script.

   The extractMovePlan script is located in:

   ```
   (UNIX) ORACLE_COMMON_HOME/bin/extractMovePlan.sh
   (Windows) ORACLE_COMMON_HOME\bin\extractMovePlan.cmd
   ```

   See Section 20.3.1.7 for the syntax of the extractMovePlan script.

   For example:

   ```
   extractMovePlan.sh -javaHome /scratch/Oracle/Middleware1/jrockit_160_20_
   D1.1.0-18
                      -archiveLoc /tmp/ohs1.jar
                      -planDirLoc /tmp/Oracle/t2p_plans/ohs
   ```

4. Edit the move plan, modifying the properties for the particular components to reflect the values for the target environment:

   - For Oracle HTTP Server, see Table 20–18.
   - For Oracle Internet Directory, see Table 20–19.

- For Oracle Virtual Directory, see Table 20–20.
- For Oracle BI EE, see Table 20–22.

5. Copy the edited move plan to the target. (During the pasteConfig operation, you specify the location using the -movePlanLoc option.)
6. At the target, run the following script to generate obfuscated password files required by the move plan. Run the script for each password file.

   ```
   (UNIX) ORACLE_COMMON_HOME/bin/obfuscatePassword.sh
   (Windows) ORACLE_COMMON_HOME\bin\obfuscatePassword.cmd
   ```

   The script prompts you to enter the password and the location where the password file is to be written.
7. At the target, extract the files from the archive using the pasteConfig script. To apply the archive for an individual component, add the -targetComponentName option.

   The pasteConfig script is located in:

   ```
   (UNIX) ORACLE_COMMON_HOME/bin/pasteConfig.sh
   (Windows) ORACLE_COMMON_HOME\bin\pasteConfig.cmd
   ```

   See Section 20.3.1.10 for the syntax of the script.

   For example, to apply the archive to the Oracle instance webtier_2 and name the target Oracle HTTP Server instance ohs_cl, use the following command:

   ```
   pasteConfig.sh -javaHome /scratch/Oracle/Middleware/jrockit_160_20_D1.1.0-18
                  -archiveLoc /tmp/ohs1.jar
                  -movePlanLoc /tmp/Oracle/t2p_plans/ohs/moveplan.xml
                  -targetOracleHomeLoc /scratch/Oracle/Middleware/Oracle_WebTier
                  -targetInstanceHomeLoc /scratch/Oracle/Middleware/webtier_2
                  -targetInstanceName webtier_2
                  -targetComponentName ohs_cl
                  -domainHostName myhost
                  -domainPortNum 7001
                  -domainAdminUserName domain_admin_username
                  -domainAdminPassword domain_admin_password_file
   ```

   Note that the Oracle instance name must be unique in the domain and the component name must be unique in the Oracle instance. If you are applying the archive of the Oracle instance to the same domain, use the -targetInstanceName and -targetComponentName options to specify a different name for the instance and component.

When you complete this task, you need to perform additional steps for each component, as described in Section 21.4.

21.3.8 Configuring Users and Groups

You must configure security in the new target environment. The steps you take depends on the configuration of your environment and application.

The target environment LDAP identity store may not use the same users and groups as the source environment, or it may already be populated with users and groups. Take the following steps only if you need to move users, groups, and passwords from the source environment to the target environment:

1. Export the users and groups from LDAP identity store on the source environment, using the ldapsearch command. This produces an ldif file that you later import

into the LDAP identity store in the target environment. The ldapsearch command is located in the ORACLE_HOME/bin directory of the Identity Management components. For example:

```
ORACLE_HOME/bin/ldapsearch -h test_oid_host -p test_oid_port
  -D "cn=orcladmin" -w "test_orcladmin_passwd" -b "cn=Users,dc=us"
```

2. Import the ldif file that you exported from the source environment into the target environment, using the ldapaddmt command, as shown in the following example. (*ORACLE_HOME* is the Oracle home for Identity Management.)

```
ORACLE_HOME/bin/ldapaddmt -h production_oid_host
   -p production_oid_port -D "cn=orcladmin"
   -w "production_orcladmin_passwd" -r -f ldif_filename
```

21.4 Moving Oracle Fusion Middleware Components

The following sections describe the steps you must take to move Oracle Fusion Middleware components. In many cases, the steps use the common procedures described in Section 21.3. All components require additional steps as described in the following topics:

- Moving Identity Management Components to a Target Environment
- Moving Oracle SOA Suite to a Target Environment
- Moving Oracle WebCenter Portal to a Target Environment
- Moving Oracle WebCenter Content to a Target Environment
- Moving Oracle Hyperion Enterprise Performance Management System to a Target Environment
- Moving the Web Tier to a Target Environment
- Moving Oracle Business Intelligence to a Target Environment
- Moving Oracle Real-Time Decisions to a Target Environment
- Moving Oracle Portal, Oracle Forms Services, Oracle Reports, and Oracle BI Discoverer to a Target Environment
- Moving Oracle Data Integrator to a Target Environment

21.4.1 Moving Identity Management Components to a Target Environment

The following topics describe how to move Identity Management from the source environment to the target environment:

- Moving Identity Management to a New Target Environment
- Moving Identity Management to an Existing Target Environment

In both cases, you have performed the following in the source environment:

- Installed a database to be used for Identity Management components such as Oracle Internet Directory, Oracle Directory Integration Platform (which depends on Oracle Internet Directory,) and Identity Federation.
- Created the needed schemas in the source environment using RCU. See the *Oracle Fusion Middleware Repository Creation Utility User's Guide*.

- Installed and configured Identity Management, including some or all of the following components: Oracle Internet Directory, Oracle Virtual Directory, Oracle Web Services Manager, or Oracle Adaptive Access Manager.
- For Oracle Internet Directory, created the desired LDAP trees and entries, in particular, users and groups.
- For Oracle Virtual Directory, created adapters to various data sources, such as LDAP and databases, and you may have configured a Local Store Adapter (LSA) to create local LDAP data, which resides in the local file system.
- For Oracle Directory Integration Platform, created synchronization profiles to various targets. These profiles are in the form of LDAP entries residing in Oracle Internet Directory.
- For Identity Federation, configured various trusted identity providers and service providers.
- For Oracle Access Management Access Manager 11*g*, set up authentication with corresponding WebGates configured in the Web tier of the protected applications. The Access Manager configuration data resides in a file and the policy and configuration data resides in a database, as described in the *Oracle Fusion Middleware Administrator's Guide for Oracle Access Management*.
- For Oracle Platform Security, created security policies and stored credentials in the Credential Store Framework (CSF).
- For Oracle Web Services Manager, created Oracle Web Services Manager policies. These policies are also attached to Web services and clients.
- For SSL, configured self-signed certificates. (In the target environment, you use trusted CA-signed certificates.)

21.4.1.1 Moving Identity Management to a New Target Environment

In this procedure, you have installed Identity Management components, such as Oracle Internet Directory, Oracle Virtual Directory, and Oracle Directory Integration Platform, in the source environment and you want to move them to the target environment that does not exist.

Perform the following tasks, depending on which components you use. Note that Task 1 is required for all components.

- Task 1, "Move the Database, Middleware Homes, and Domain Configuration to the New Target Environment"
- Task 2, "Perform Prerequisite Task for Oracle Adaptive Access Manager"
- Task 4, "Move Oracle Internet Directory to the New Target Environment"
- Task 5, "Move Oracle Virtual Directory to a New Target Environment"
- Task 6, "Move Oracle Directory Integration Platform to a New Target Environment"
- Task 7, "Move Access Manager 11g to a New Target Environment"
- Task 8, "Move Oracle Access Manager 10g to a New Target Environment"
- Task 9, "Move Oracle Access Management Identity Federation to a New Target Environment"
- Task 10, "Move Oracle Adaptive Access Manager to a New Target Environment"
- Task 11, "Move Oracle Identity Navigator to a New Target Environment"

- Task 12, "Move Oracle Identity Manager to a New Target Environment"
- Task 13, "Move Oracle Privileged Account Manager to a New Target Environment:"
- Task 14, "Move Audit Policies to a New Target Environment"
- Task 15, "Move Oracle Web Services Manager to a New Target Environment"
- Task 16, "Move Oracle Unified Directory to a New Target Environment"

Task 1 Move the Database, Middleware Homes, and Domain Configuration to the New Target Environment

You move the database, a copy of all Identity Management Middleware homes, and the domain configuration to the target environment, using the following steps:

1. Move or create the database and the schemas, as described in Section 21.3.3.
2. Move a copy of the Middleware home containing the Identity Management components from the source environment to the target environment using the copyBinary and pasteBinary scripts, as described in Section 21.3.4.
3. For Oracle Adaptive Access Manager, perform the prerequisite task, as described in Task 2, "Perform Prerequisite Task for Oracle Adaptive Access Manager".
4. For Access Manager, if required, perform the prerequisite task, as described in Task 3, "Perform Prerequisite Task for Access Manager".
5. Move a copy of the configuration of each domain containing the Identity Management configuration, as described in Section 21.3.6. This step moves the configuration, including the domain, Administration Server, and Managed Servers. Moving the configuration also:
   - Reassociates the security store to an LDAP or database-based store, based on the values provided in move plan.
   - Configures Identity Federation. For optional tasks that you may take in certain situations, see Task 4, "Move Oracle Access Management Identity Federation to an Existing Target Environment".
   - Moves Oracle Platform Security. The policy and credential stores are moved as part of the copyConfig and pasteConfig operations only if the source is a file-based store and the target is an LDAP or database-based store.

     If you are using Oracle Web Services Manager, move audit policies as described in Task 14, "Move Audit Policies to a New Target Environment".
   - Moves Oracle Web Services Manager and any policies that are stored in the MDS Repository or deployment plans, and any custom policies that are stored in *DOMAIN_HOME*/lib. To move policies that are not stored in the MDS Repository, see Task 15, "Move Oracle Web Services Manager to a New Target Environment".
   - Moves Oracle Entitlements Server.
   - Moves Oracle Access Management Mobile and Social. The move plan properties for Access Manager (see Table 20–30) automatically update the Mobile and Social configuration.
   - Moves Oracle Access Management Secure Token Service. The move plan properties for Access Manager (see Table 20–30) automatically update the Secure Token Service configuration.
   - Configures data sources.

- Configures JMS resources.
- Starts the Administration Server.

Task 2 Perform Prerequisite Task for Oracle Adaptive Access Manager

Oracle Adaptive Access Manager uses system snapshots to easily migrate security data across environments. Before you run the copyConfig and pasteConfig scripts, you must create a snapshot, which loads the data from the database into a snapshot, then restore the system snapshots on the source system:

1. Log in to the OAAM Admin console as a system administration, using the following URL:

   ```
   http://oaam_admin_server_host:oaam_admin_server_port/oaam_admin
   ```

2. In the Navigation pane, under **Environment,** select **System Snapshots.**
3. Load and restore the system snapshots, as described in "Loading and Restoring a Snapshot" in the *Oracle Fusion Middleware Administrator's Guide for Oracle Adaptive Access Manager*.
4. After you restore the snapshot, update the configuration properties to the desired values for the source environment.

Task 3 Perform Prerequisite Task for Access Manager

If you are moving Access Manager, perform the following steps:

1. Unlock the WebLogic Server configuration by clicking **Release Configuration** on the WebLogic Server Administration Console,
2. If you have modified an Access Manager server port number using the Access Manager console, that port number is not modified in config.xml. To work around this issue, edit the following file to modify the port number:

   ```
   DOMAIN_HOME/config/config.xml
   ```

Task 4 Move Oracle Internet Directory to the New Target Environment

To move Oracle Internet Directory to a new target environment:

1. Move the Oracle Internet Directory configuration by moving the configuration of the Oracle instance, as described in Section 21.3.7.

   Note the following:

   - If an Oracle Internet Directory component is copied with the same database credentials as the source component, the name of the target OID component should be different than the source component to avoid conflicts in the OID schema.
   - If an Oracle Internet Directory component is copied with different database credentials from the source component, the name of the target Oracle Internet Directory component should be the same as the source component.

2. Under certain conditions, you may see the following errors when you run the copyConfig and pasteConfig scripts:

   ```
   OID Cloning: Error cleaning replication agreements
   OID Cloning: Error deleting replication dn
   OID Cloning: Error updating orclreplicaid
   ```

   If you do, take the following steps:

a. Run the following command:

```
ORACLE_HOME/ldap/bin/remtool -pcleanup
```

When prompted, enter the Oracle Internet Directory host, non-SSL port, and the ODS schema password.

b. Perform an ldapsearch on the root dn for the orclreplicaid value. Use the following command:

```
ORACLE_HOME/bin/ldapsearch -p port -h host
   -b "" -s base "(objectclass=*)" orclreplicaid
```

c. Using the value obtained in Step b, perform an ldapdelete, deleting the following dns from Oracle Internet Directory:

```
cn=replication dn, orclreplicaid=<replicaid>, cn=replication configuration
orclreplicaid=<replicaid>, cn=replication configuration
```

For example:

```
ldapdelete -p port -h host "cn=replication dn,
   orclreplicaid=replicaid, cn=replication configuration"
```

d. Set the orclreplicaid value in the root entry to 0. For example:

```
ORACLE_HOME/bin/ldapmodify -p port -h host -f file.ldif
```

The ldif file contains the following:

```
dn:
changetype: modify
replace: orclreplicaid
orclreplicaid: 0
```

e. Restart Oracle Internet Directory.

3. If you have configured Oracle Internet Directory replication in the source environment, you must reconfigure it again in the target environment after moving. The replication configuration is not moved from the source to the target environment. See "Setting Up Replication" in the *Oracle Fusion Middleware Administrator's Guide for Oracle Internet Directory.*

Task 5 Move Oracle Virtual Directory to a New Target Environment

To move Oracle Virtual Directory to a new target environment:

1. Move the Oracle Virtual Directory configuration by moving the configuration of the Oracle instance, as described in Section 21.3.7.

If you have already moved the Oracle instance that contains Oracle Virtual Directory, you do not need to take this step.

Note that, during the pasteConfig operation, if you have not provided a password file for the Oracle Virtual Directory adapter or you specify an incorrect location for the password file in the move plan, the adapter configuration is not changed and the script returns the following message:

```
Password file is either not provided or invalid for adapter adapter_name.
Nothing will be changed for this adapter configuration.
```

Task 6 Move Oracle Directory Integration Platform to a New Target Environment

To move Oracle Directory Integration Platform to a new target environment:

1. Move Oracle Internet Directory, as described in Task 4.

   Oracle Directory Integration Platform profiles reside in Oracle Internet Directory. If you have correctly moved Oracle Internet Directory to the target environment, the profiles are carried over to the target environment.

2. If you configured SSL on the source environment, that configuration is not moved to the target environment. You must configure SSL on the target environment. See Section 6.5.4.3.

Task 7 Move Access Manager 11*g* to a New Target Environment

When you move the configuration of the domain, update the move plan properties in Table 20–30 to use values for the target environment. However, note the following:

- Although you can specify the production host and port in the move plan, you must manually re-register the partners.
- For Webgate, the script generates the obAccessclient.xml file. If you are not also moving WebGate, you must manually copy this file to the agent at the following location:

  ```
  WebGate_instance_dir/webgate/config
  ```

- For mod_osso, you must copy the osso.conf file to the following location in the Middleware home in which Oracle HTTP Server has been installed:

  ```
  MW_HOME/instances/instance1/config/OHS/ohs1/osso/
  ```

- When Access Manager is integrated with Oracle Adaptive Access Manager, you should manually regenerate the partner key.
- When Access Manager is integrated with Identity Federation, you should manually regenerate the partner key.
- If Access Manager is integrated with Oracle Adaptive Access Manager, update the challengeURL for the Authentication Scheme in the Access Manager configuration. See "Viewing or Editing a Authentication Scheme" in the *Oracle Fusion Middleware Administrator's Guide for Oracle Access Management*.
- The passwords (in CSF) and the server key remain same for source and target. For Access Manager, if the target server key has to be different from the source, you must regenerated it after you move the domain configuration.

Moving the configuration of the domain moves the Access Manager configuration to the target environment.

Task 8 Move Oracle Access Manager 10*g* to a New Target Environment

To move Oracle Access Manager 10*g* to a new target environment:

1. Move the Directory Server from the source environment to the target environment. That is, migrate the o=oblix node. See "Preparing the New Directory Server Instance" in the *Oracle Access Manager Installation Guide.*
2. Remove the entries that are associated with the Identity Server, Policy Manager, and Access Servers. The entries are under the following:

   ```
   obcontainerId=DBAgents,<Configuration DN>
   ```

   Do not delete the container (obcontainerId=DBAgents).

3. Install and configure Oracle Access Manager, specifying the LDAP information for the target environment, as described in the *Oracle Access Manager Installation Guide*.

Oracle Access Manager stores policy and configuration data in the LDAP directory. If the LDAP directory is correctly configured (for example if you have correctly moved Oracle Internet Directory from the source environment to the target environment), Oracle Access Manager inherits the policy and configuration data from the LDAP directory.

4. On the target environment, install the Identity Server and WebPass using new identifiers. For more information, see:

   - "Installing the Identity Server" in the *Oracle Access Manager Installation Guide.*
   - "Installing WebPass" in the *Oracle Access Manager Installation Guide.*

   After installation, take the following steps:

   a. Start the server.

   b. Complete the identity system browser setup. See "Setting Up the Identity System" in the *Oracle Access Manager Installation Guide.*

5. Install the Policy Manager, as described in "Installing the Policy Manager" in the *Oracle Access Manager Installation Guide.* However, do not update the schema because you already updated it when you moved the Directory Server. Do not configure the authentication scheme because it already exists in the Directory Server.

   Note: After setting up the target Policy Manager, when you log in as the Oracle Access Manager Administrator, you may get the following error:

   ```
   There was a problem obtaining the user ID. One possible reason for
   this is a time difference between the Identity System and Access
   Systems (Policy Manager and Access System Console).
   ```

   To fix this, from the LDAP, delete the cookie encryption key (without changing the CPResponseEncryptionKey) under the o=oblix node, and restart the Identity Server. Note that you should make a backup of the cookie encryption entry into an ldif file before deletion.

6. Complete the browser setup from the Access System Console, adding the Access Server with a new identifier. See "Creating an Access Server Instance in the System Console" in the *Oracle Access Manager Installation Guide* for more information.

   Also see "About the Access Server and Installation" in the *Oracle Access Manager Installation Guide* for additional information.

7. This scenario reuses the existing WebGate identifier for the target WebGates. Take the following steps:

   a. Navigate to the Access System Console and select the Access System Configuration tab.

   b. Select **Host Identifiers.** On the List all host identifiers page, select the host identifier that is used by the source environment.

   c. Click **Modify.** Then, add the host name and port for the target Web server to the **Hostname variations** field.

Note: Resources may become unprotected if you have the same host and port in multiple host identifiers.

Ensure that only the host identifier used in the policy domain has the host:port in its definition. Remove host:port from other host identifiers.

---

d. Click **Save.**

e. From the Access System Configuration tab, select **Access Gate Configuration**. Then, select the relevant Access Gate.

f. In the Details for AccessGate page, click **Modify.**

g. Change the **Hostname** and **Port,** specifying the host name and port of the target Web server.

h. Change the **Preferred HTTP Host,** specifying the host name variation that you added in Step c.

i. Associate the WebGate to the newly added target Access Server, as described in "Associating AccessGates and WebGates with Access Servers" in the *Oracle Access Manager Access Administration Guide*.

j. Disable the WebGate temporarily. From the Access System Console, select the Access System Configuration tab, then select **AccessGate Configuration.** Click **Go** to search. From the results, select an AccessGate. Then, click **Modify.** Click **Disabled.** Then, click **Save.**

You enable it after you install the Access Server.

8. Install the Access Server using the new identifier that you used while creating the WebGates. See "Installing the Access Server" in the *Oracle Access Manager Installation Guide.*

9. Install the new WebGate. See "Installing the WebGate" in the *Oracle Access Manager Installation Guide.*

10. Verify entries and delete entries related to the source environment:

a. From the Identity System Console, select the System Configuration tab, then select **Directory Profiles.** Verify that the respective Directory Profiles are associated with the new Identity Server, Access Server, and Policy Manager.

b. From the Identity System Console, select the System Configuration tab, then select **Webpass** and delete the entry for the source WebPass.

c. From the Identity System Console, select the System Configuration tab, then select **Identity Server** and delete the entry for the source Identity Server.

d. From the Access System Console, select the Access System Configuration tab, then select **Access Server Configuration.** Delete the entry for the source environment Access Server.

11. From the Identity System Console, select the System Configuration tab, then select **Password Policy.** If the host and port are set for **Password Change Redirect URL,** change them to point to the new Identity Server.

12. From the Access System Console, select the Access System Configuration tab, then select **Authentication Management.** Select the authentication scheme for which Challenge redirect is set. Modify **Challenge Redirect** to specify the host and port of the new Web server, if the new authentication WebGate is installed.

13. From the Access System Console, select the Access System Configuration tab, then select **Authentication Management.** Select the authentication scheme for which a password policy is configured. Change the obWebPassURLprefix (if it exists) to accommodate the new host and port of the target Web server on which WebPass is installed, if WebPass and WebGate reside on different Web servers.

    For more information, see "Configuring Password Policies" in the *Oracle Access Manager Identity and Common Administration Guide*.

Task 9 Move Oracle Access Management Identity Federation to a New Target Environment

When you use the copyConfig and pasteConfig scripts, as described in step 5 in Task 1, and modified the move plan as described in Table 20–21, the following are configured with values for the target environment:

- The load balancer host and port and the SOAP port.
- The service provider ID URL
- The identity provider ID URL
- The data stores
- The Authentication Engines
- The Service Provider Integration Modules

To complete the movement of Identity Federation:

1. Start the Managed Servers.
2. If, when you use Fusion Middleware Control and you receive a message that Identity Federation is not running although it is actually running, you must update the monitoring user name to be able to make configuration changes using Fusion Middleware Control:
    a. Navigate to the Agent-Monitored Target configuration page, as described in Section I.3.1.3.
    b. Select the Identity Federation icon.
    c. On the Configuration page, update the WebLogic Monitoring Username and WebLogic Monitoring Password.
3. Delete old trusted partner:
    a. In Fusion Middleware Control, navigate to the Identity Federation instance.
    b. Select **Administration,** then **Federations.**
    c. Select the provider and click **Delete.**
4. Regenerate the metadata and reregister the provider, as described in "Generate Provider Metadata" and "Register the Providers" in the *Oracle Fusion Middleware Integration Guide for Oracle Access Manager*.
5. Optionally, regenerate the server key. See "Managing Oracle Access Management Identity Federation" the *Oracle Fusion Middleware Administrator's Guide for Oracle Access Management* for information about regenerating the keys.

In most cases, you do not need to take any additional steps. If you need to change the Security and Trust, or Authentication Mechanisms, take the following steps:

1. If you need to change or add partners, see "Add Trusted Providers" and "Delete Trusted Providers" in the *Oracle Fusion Middleware Administrator's Guide for Oracle Identity Federation*.
2. If you need to change the HTTP basic authentication, update the user name and password:
   a. In Fusion Middleware Control, from the target menu on the OIF page, choose **Administration,** then **Federations.**
   b. Select a Trusted Provider and click **Edit.** Update **HTTP Authentication Username** and **HTTP Authentication Password.** Then, confirm the password.
   c. Click **Apply.**
3. Start the Managed Servers.

Task 10 Move Oracle Adaptive Access Manager to a New Target Environment

When you move the configuration of the domain, update the move plan properties in Table 20–31 to use values for the target environment.

> **Note:** Before you run the copyConfig and pasteConfig scripts, you must perform the prerequisite task, as described in Task 2, "Perform Prerequisite Task for Oracle Adaptive Access Manager".

1. If Oracle Adaptive Access Manager is integrated with Access Manager, after you have moved the configuration of Oracle Adaptive Access Manager, update the redirect URL:
   a. Log in to the Access Manager console:

      ```
      https://hostname.com:7001/oamconsole
      ```

   b. In the Policy Configuration tab, select the Authentication Scheme.
   c. Change the **Challenge Redirect URL** to the value for the target system.
2. The passwords in the Credential Store Framework remain the same for the target environment as for the source environment. If you want the passwords to be different in the target environment, you must regenerate them on the target environment.

Task 11 Move Oracle Identity Navigator to a New Target Environment

To move Oracle Identity Navigator to a new target environment:

1. On the target system, configure a proxy, as described in "Configuring a Proxy to Access News Feeds" in the *Oracle Fusion Middleware Administrator's Guide for Oracle Identity Navigator*.

Task 12 Move Oracle Identity Manager to a New Target Environment

You can use the Oracle Identity Manager Deployment Manager to move most metadata from the source environment to a target environment. See "Migrating Configurations and Customizations" in the *Oracle Fusion Middleware Developer's Guide for Oracle Identity Manager* for information about Deployment Manager.

The following table lists the entities that you can move using Deployment Manager:

Entities	Deployment Manager Category
Application Instances	Application instance
Catalog Definitions	Catalog definition
Plug-ins	Plug-in
JAR files	JAR file
Custom Resource Bundles	Custom resource bundles
Entity Publications	Entity publications
Roles	Role
Organizations	Organization
Access Policies	Access Policy
Attestation Processes	Attestation Process
Authorization Policies	Authorization Policy
User Metadata	User Metadata
Roles and Org Metadata	Roles and Org Metadata
Scheduled Tasks	Scheduled Task
Scheduled Jobs	Job
IT Resources	IT Resource
Resource Objects	Resource
Lookup Definitions	Lookup
Process Forms	Process Form
Provisioning Workflows and Process Task Adapters	Process
Data Object Definitions	Data Object Definition
Rules	Rule
Notification Templates	Notification Template
GTC Providers	Generic Technology Connector (GTC) Provider
Error Codes	Error Code
System Properties	System Property
EmailDef	Email Definition
EventHandler	Event Handlers
PasswordPolicy	Password Policy
GenericConnector	Generic Technology Connector
ITResourceDef	IT Resource Definition
Request Templates	Request Template
Request Datasets	Request Dataset
Approval Policies	Approval Policy
Prepopulation Adapters	Prepopulation adapters
Process Definitions	Process definitions

To move Oracle Identity Manager to a new target environment:

1. On the source environment, use the Deployment Manager to export the metadata for the entities listed in the preceding table. In the wizard, select the entities' children and dependencies. See "Exporting Deployments" in the *Oracle Fusion Middleware Developer's Guide for Oracle Identity Manager* for information about how to export metadata.

   The data is exported as .xml files.

2. On the target environment, use the Deployment Manager to import the metadata for the entities listed in the preceding table. See "Importing Deployments" in the *Oracle Fusion Middleware Developer's Guide for Oracle Identity Manager* for information about how to import metadata.

   The Deployment Manager does not manage Custom Reconciliation Profiles.

3. Move the Approval Workflows, which are SOA composite applications, using JDeveloper:

   a. Copy all of the files in the JDeveloper project from the source environment to the target environment, using any standard file transfer method.

   b. In the application, change any calls to external systems to point to the systems in the target environment. For example, if the workflow uses an LDAP server in the source environment, change references to point to an LDAP server in the target environment.

   c. Use JDeveloper to build the sca jar file from the SOA composite.

   d. Deploy the SOA composite application on the target environment, using the SOA Deployment wizard in Fusion Middleware Control (see Section 10.5.1) or JDeveloper.

4. Move any custom reconciliation profiles, as described in "Updating Reconciliation Profiles Manually" in the *Oracle Fusion Middleware Developer's Guide for Oracle Identity Manager*.

   a. Use the WLST command `exportMetadata` to export the custom reconciliation profiles from the source environment:

   ```
   connect('username','password','JNDI-URL')
   exportMetadata(application='OIM', server='server_name',
     toLocation='directory', docs='path_to_reconciliation_profiles')
   ```

   b. Copy the exported files to the target environment.

   c. If a reconciliation profile is imported in any MDS environment with the attribute configure="true," it automatically generates all the required configuration for that environment and updates this property to false. In this case, after you export this profile from source environment, edit the file and add the configure='true' property before importing to the target environment.

   d. Use the WLST command `importMetadata` to import the custom reconciliation profiles to the target environment:

   ```
   connect('username','password','JNDI-URL')
   importMetadata(application='OIM', server='server_name',
     fromLocation='directory', docs='/**')
   ```

5. For connectors, if there are any changes to the forms that need the older versions of these forms to be upgraded with the new definition on the target environment, move the connectors, then run the Form Version Control (FVC) utility. For more

information, see the section "Upgrading the Connector" of the Connector Patch Readme file. The Readme file is located in the top-level directory of the connector distribution media.

Task 13 Move Oracle Privileged Account Manager to a New Target Environment:

To move Oracle Privileged Account Manager to a new target environment:

1. Because the source environment uses a database-based policy store, the copyConfig and pasteConfig scripts do not move the policy store and credential store data. To move them, follow the steps in Section 21.3.5.

Task 14 Move Audit Policies to a New Target Environment

To move audit policies to a new target environment, see the following topics in the *Oracle Fusion Middleware Application Security Guide*:

- "Migrating Audit Policies"
- "Managing Audit Policies"

Task 15 Move Oracle Web Services Manager to a New Target Environment

To move Oracle Web Services Manager to a new target environment:

1. Migrate audit policies, as described in "Migrating Audit Policies" in the *Oracle Fusion Middleware Application Security Guide.*
2. Move policies that are not stored in the MDS Repository. For ADF BC and Oracle WebCenter Portal policy attachments, migrate them, as described in "Managing Application Migration Between Environments" in the *Oracle Fusion Middleware Security and Administrator's Guide for Web Services*.

   For other policy attachments, the attachments are moved with the application if you use the Oracle WebLogic Server cloning feature.

   See Also: "Managing Application Migration Between Environments" in the *Oracle Fusion Middleware Security and Administrator's Guide for Web Services*

Task 16 Move Oracle Unified Directory to a New Target Environment

Oracle Unified Directory does not use the procedures described in this chapter or the movement scripts described in Chapter 20. To move Oracle Unified Directory from a source to a target environment, see "Moving From a Test to a Production Environment" in the *Oracle Fusion Middleware Administrator's Guide for Oracle Unified Directory*.

21.4.1.2 Moving Identity Management to an Existing Target Environment

In this procedure, you have installed Identity Management components, such as Oracle Internet Directory, Oracle Directory Integration Platform, and Oracle Web Services Manager, in the source environment and you want to move them to the target environment that already exists.

On the existing target environment, you have installed and configured the components. You want to move an application from the source environment to the target environment while retaining its security-related configuration. This requires migrating application-specific data from the source Identity Management environment to the target Identity Management environment.

To move Identity Management to an existing target environment, perform the following tasks:

- Task 1, "Move Oracle Internet Directory to an Existing Target Environment"
- Task 2, "Move Access Manager 11g to an Existing Target Environment"
- Task 3, "Move Oracle Access Manager 10g to an Existing Target Environment"
- Task 4, "Move Oracle Access Management Identity Federation to an Existing Target Environment"
- Task 5, "Move Oracle Adaptive Access Manager to an Existing Target Environment"
- Task 6, "Move Oracle Identity Manager to an Existing Target Environment"
- Task 7, "Move Oracle Identity Navigator to an Existing Target Environment"
- Task 8, "Move Oracle Platform Security to an Existing Target Environment"
- Task 9, "Move Oracle Privileged Account Manager to an Existing Target Environment"
- Task 10, "Move Oracle Web Services Manager to an Existing Target Environment"

Task 1 Move Oracle Internet Directory to an Existing Target Environment

To move Oracle Internet Directory to an existing target environment:

1. You may have configured Oracle Platform Security to use the users and groups in the source environment. To move the users and groups from the source environment, take the following steps:

   a. Identify the Default Subscriber for the source Oracle Internet Directory instance by running the following command from the source Oracle home:

   ```
   ORACLE_HOME/bin/ldapsearch -h test_oid_host -p test_oid_port
     -D "cn=orcladmin" -w "test_orcladmin_passwd"
     -b "cn=Common,cn=Products,cn=OracleContext"
     -s base "objectclass=*"  orcldefaultsubscriber
   ```

   This query returns a value for the attribute orcldefaultSubscriber. The value is used in following steps as *default_subscriber*.

   b. Retrieve the users from the source Oracle Internet Directory instance by running the following command from the source Oracle home:

   ```
   ORACLE_HOME/bin/ldapsearch -h test_oid_host -p test_oid_port
     -D "cn=orcladmin" -w "test_orcladmin_passwd"
     -L -b "cn=users, default_subscriber"
     -s sub "objectclass=*" * orclguid > ldif_filename
   ```

   c. Move the users into the target Oracle Internet Directory instance by running the following command from the target Oracle home:

   ```
   ORACLE_HOME/bin/ldapaddmt -h production_oid_host
     -p production_oid_port -D "cn=orcladmin"
     -w "production_orcladmin_passwd" -r -f ldif_filename
   ```

   Specify the -r argument to move data and resolve conflicts. The *ldif_filename* is the file you obtained in the previous step.

2. If the source environment is set up as a staging environment to mimic the target environment, Oracle recommends that you set up one-way replication from the target Oracle Internet Directory to the source Oracle Internet Directory to ensure that any users or groups that exist in the target environment are available in the fan-out replica, which can be used to test applications. Fan-out replication also

provides the capability to keep the source Oracle Internet Directory synchronized with the target and to replicate any users or groups that are added into target on real-time basis.

For information about fan-out replication, see the following sections in the *Oracle Fusion Middleware Administrator's Guide for Oracle Internet Directory*:

- "Understanding Oracle Internet Directory Replication"
- "Setting Up an LDAP-Based Replication Agreement by Using the Replication Wizard"

3. If you use Oracle Forms Services or Oracle Reports, move Resource Access Descriptors (RAD). This procedure assumes that you have moved the Default Subscriber from the source environment to the target environment, as described in Step 1. It also assumes that the orclGUIDs of the users at the source Oracle Internet Directory are identical to those in the target Oracle Internet Directory.

   Take the following steps:

   a. Identify the Default Subscriber as described in Step 1a.

   b. Retrieve the RADs from the source Oracle Internet Directory instance using the following command:

   ```
   ORACLE_HOME/bin/ldapsearch -h test_oid_host -w test_orcladmin_passwd
      -p test_oid_port -D "cn=orcladmin"
      -L -b "cn=Extended Properties,cn=OracleContext, default_subscriber"
      -s sub "objectclass=*" * orclguid > ldif_filename
   ```

   c. Move the RADs into the target Oracle Internet Directory instance using the following command:

   ```
   ORACLE_HOME/bin/ldapaddmt -h production_oid_host
      -p production_oid_port -D "cn=orcladmin"
      -w "production_orcladmin_passwd" -r -f ldif_filename
   ```

   Specify the -r argument to move data and resolve conflicts. The *ldif_filename* is the file you obtained in the previous step.

   Note that this command generates the file add.log in the directory where you run it. Check the add.log file for errors encountered during RADs migration. If there are any errors, fix the errors and rerun the command.

Task 2 Move Access Manager 11*g* to an Existing Target Environment

In this procedure, you move incremental changes that you have made in the source environment to the target environment.

> **Note:** The Administration Servers in both the source environment and the target environment must be started.

To replicate the policy configuration information from the source environment into the target environment:

1. Set the environment variable JAVA_HOME and add JAVA_HOME to the PATH.
2. Export the policies from the source environment, using the following WLST command:

   ```
   exportPolicy(pathTempOAMPolicyFile='path_of_Temp_PolicyFile')
   ```

Note that you must run the WLST commands in this task from the following directory:

```
Identity_Mgmt_ORACLE_HOME/common/bin
```

For example, the following command generates the file oam_policy.xml:

```
exportPolicy(pathTempOAMPolicyFile='/tmp/oam_policy.xml')
```

3. Edit the oam_policy.xml policy file to change the host and port identifiers to the values for the target environment. These are specified in the <host-identifiers> section of the file.
4. Copy the policy file to the target environment.
5. Import the policies into the target environment, using the following WLST command:

```
importPolicy(pathTempOAMPolicyFile='path_of_Temp_PolicyFile')
```

For example:

```
importPolicy(pathTempOAMPolicyFile='/tmp/oam_policy.xml')
```

6. Export the partner information from the source environment, using the following WLST command:

```
exportPartners(pathTempOAMPartnerFile='path_of_Temp_PartnerFile')
```

For example, the following command generates the file oam_partner.xml:

```
exportPartners(pathTempOAMPartnerFile='/tmp/oam_partner.xml')
```

7. Copy the partner file to the target environment.
8. Import the partner information to the target environment, using the following WLST command:

```
importPartners(pathTempOAMPartnerFile='path_of_Temp_PartnerFile')
```

For example:

```
importPartners(pathTempOAMPartnerFile='/tmp/oam_partner.xml')
```

9. The following directory on the target system now contains the partner artifacts that were generated for the migrated partners.

```
DOMAIN_HOME/output
```

Copy the partner artifacts, ObAccessClient.xml and logout.html, to the partner Oracle HTTP Server and update with values for the target environment, if necessary.

10. Update the mod_wl_ohs.conf file with the URI, host, and port of WebLogic Server where the application is deployed. For example:

```
WebLogicHost=example.com|WebLogicPort=18357
```

11. Restart the partner Oracle HTTP Server.

Task 3 Move Oracle Access Manager 10*g* to an Existing Target Environment

To move Oracle Access Manager 10*g* to an existing target environment:

1. In the target environment, use the Oracle Access Manager OAMCfgTool to create the same policy domain for the application. Ensure that the following specify values for the target environment:

   ```
   web_domain (The Host identifier is derived from this entry)
   protected_uris="uri1,uri2,uri3"
   app_agent_password=password to be provisioned for the WebGate
   ldap_host=hostname_of_LDAP_server
   ldap_port=port_of_LDAP_server
   ldap_userdn=DN_of_LDAP_Admin_User
   ldap_userpassword=password_of_LDAP_Admin_User
   oam_aaa_host=host_of_OAM_server
   oam_aaa_port=port_of_OAM_server
   ```

   If you are using a uris_file to specify the protected and public URIs in a file, review the file to ensure that you are listed the corrected URIs.

2. If you made other changes to the Oracle Access Manager entities, such as the policy domain, in the source environment, make the same types of changes in the target environment.

Task 4 Move Oracle Access Management Identity Federation to an Existing Target Environment

To move Identity Federation to an existing target environment:

1. Set up the WLST environment on both the source and target environment.
2. On the source environment, extract the partner metadata and configuration properties by running the following script:

   ```
   java weblogic.WLST extractPartnerMetadataAndProperties.py providerID
            outputFilePrefix
   ```

   Two files are created: *outputFilePrefix*_metadata.xml and *outputFilePrefix*_ properties.txt.

3. Copy the files to the target system.
4. On the target environment, import the partner metadata and configuration properties by running the following script:

   ```
   java weblogic.WLST setPartnerMetadataAndProperties.py outputFilePrefix_
   metadata.xml
            outputFilePrefix_properties.txt description
   ```

5. If you have removed a partner from the source environment, remove it from the target environment, as described in "Delete Trusted Providers" in the *Oracle Fusion Middleware Administrator's Guide for Oracle Identity Federation.*
6. If you made any other configuration changes in the source environment that you want to replicate in the target environment, make those changes. See the *Oracle Fusion Middleware Administrator's Guide for Oracle Access Management* for more information.

Task 5 Move Oracle Adaptive Access Manager to an Existing Target Environment

To move Oracle Adaptive Access Manager to an existing target environment:

1. Export snapshots from the source environment. Use the Oracle Adaptive Access Manager Administration console to export the configuration to a zip file. See "System Snapshot Import/Export" in the *Oracle Fusion Middleware Administrator's Guide for Oracle Adaptive Access Manager* for more information.

You can export the following types of items:

- Policies
- Rule conditions
- Patterns
- Configurable actions
- Transaction definitions
- Entities
- KBA questions
- KBA validations
- All group types, including alert and action groups, and black list and white list groups used in rules

2. Import snapshots into the target environment. Use the Oracle Adaptive Access Manager Administration console to import the contents of the zip file saved in step 1. See "System Snapshot Import/Export" in the *Oracle Fusion Middleware Administrator's Guide for Oracle Adaptive Access Manager* for more information.
3. Manually update the target environment for the following items, when necessary:
   a. Because snapshot export and import only copies action and alert group types, you must export the group members from the source environment and import them into the target environment.

      To export the groups, see "Exporting a Group" in the *Oracle Fusion Middleware Administrator's Guide for Oracle Adaptive Access Manager.*

      To import the groups into the target environment, see "Importing a Group" in the *Oracle Fusion Middleware Administrator's Guide for Oracle Adaptive Access Manager.*

   b. Use the oaam_extensions shared library to package the configurable actions jar.

   c. Manually copy any items customized in the OAAM server, such as headers, footers, cascading style sheets (CSS), and JavaScript, from the source environment to target environment. These items are located in the oaam_ extensions shared library.

   d. Manually re-create the KBA logic, OTP logic, and policy set overrides using the Oracle Adaptive Access Manager Admin Console. See the *Oracle Fusion Middleware Administrator's Guide for Oracle Adaptive Access Manager.*

   e. Copy the following from the source environment to the target environment: properties files, resource bundles, and end user JSP screens. These items are located in the oaam_extensions shared library.

   f. Copy the VAD images, which are in a custom jar, from the source environment to the target environment.

   g. Copy the following from the source environment to the target environment: properties files, resource bundles, VAD images, and end user JSP screens.

Task 6 Move Oracle Identity Manager to an Existing Target Environment

To move Oracle Identity Manager to an existing target environment, follow the steps described in Section 21.4.1.1, Task 12, "Move Oracle Identity Manager to a New Target Environment".

Task 7 Move Oracle Identity Navigator to an Existing Target Environment

To move Oracle Identity Navigator to an existing target environment, perform the tasks described in "Managing Oracle Identity Navigator" in the *Oracle Fusion Middleware Administrator's Guide for Oracle Identity Navigator*. Note that you do not need to configure the identity store and policy store if they have already been configured.

Task 8 Move Oracle Platform Security to an Existing Target Environment

If you have made changes in the policy store, credential store, users, and groups, and you want to move them from the source environment to an existing target environment:

1. If the policy store on the source environment is not file-based, migrate the policy store, using the WLST command `migrateSecurityStore`, as described in "Migrating with the Script migrateSecurityStore" in the *Oracle Fusion Middleware Application Security Guide*.
2. If the credential store on the source environment is not file-based, migrate the credential store, using the script `migrateSecurityStore`, as described in "Migrating Credentials Manually" in the *Oracle Fusion Middleware Application Security Guide*.
3. Users and groups in the target LDAP may differ from that in the LDAP. There is a mapping between Oracle Platform Security application roles and LDAP roles. While the application roles may remain the same, the mapping to LDAP groups can be changed to map to the corresponding LDAP group in the target environment. See "Managing Application Roles" in the *Oracle Fusion Middleware Application Security Guide.*
4. If you are using Oracle Web Services Manager, migrate audit policies, as described in "Migrating Audit Policies" in the *Oracle Fusion Middleware Application Security Guide.*

Task 9 Move Oracle Privileged Account Manager to an Existing Target Environment

To move Oracle Privileged Account Manager to an existing target environment:

1. Because the source environment uses a database-based policy store, the copyConfig and pasteConfig scripts do not move the policy store and credential store data. To move them, follow the steps in Section 21.3.5.
2. If the target identity store is different from the source identity store, move users and groups, using the steps in Section 21.3.8.
3. If the *`ORACLE_HOME`* path is different in the source and the target environments, then you must manually update some references in `DOMAIN_HOME/config/fmwconfig/opam/-config.xml`. These references include lines with `bundleJar` locations that point to jar files in `ORACLE_HOME`.

   For example, in a default set-up, the lines with `bundleJar` references include:

```
<connector bundleJar="ORACLE_HOME/connectors/ldap/bundle/
org.identityconnectors.ldap-1.0.6380.jar" targetType="ldap">
<connector bundleJar="ORACLE_
HOME/connectors/genericunix/bundle/org.identityconnectors.genericunix-1.0.0.jar
"targetType="unix">
<connector bundleJar="ORACLE_HOME/connectors/dbum/bundle/
org.identityconnectors.dbum-1.0.1116.jar" targetType="database">
```

Ensure that *ORACLE_HOME* is the correct path for the target environment, because the value from a source environment would have been migrated.

Task 10 Move Oracle Web Services Manager to an Existing Target Environment

To move Oracle Web Services Manager to an existing target environment:

1. Move policies for SOA Composite applications, WebCenter Portal, or ADF applications, which are stored in the MDS Repository.

   To do so using Fusion Middleware Control:

   a. On the source environment, select the domain. Then, from the WebLogic Domain menu, choose **Web Services,** then **Policies.**

   b. Select a policy, then click **Export to File.**

      The policy is copied to a file on the source environment.

   c. Click **Save File,** then **OK.**

   d. Navigate to the location on your local directory to which you want to save the file and update the file name as desired. Click **Save.**

   e. Copy the file to the target environment.

   f. On the target environment, select the domain. Then, from the WebLogic Domain menu, choose **Web Services,** then **Policies.**

   g. Click **Import from File.** Browse to the file and click **OK.**

   h. On source environment, select the domain. Then, from the WebLogic Domain menu, choose **Web Services,** then **Policies.**

   i. Click **Web Services Assertion Templates** in the upper right corner of the page.

   j. Click **Export to File.**

   k. Click **Save File,** then **OK.**

   l. Navigate to the location on your local directory to which you want to save the file and update the filename as desired. Click **Save.**

   m. On the target environment, select the domain. Then, from the WebLogic Domain menu, choose **Web Services,** then **Policies.**

   n. Click **Import from File.** Browse to the file and click **OK.**

   o. Click **Web Services Assertion Templates** in the upper right corner of the page.

   p. Click **Import from File.** Browse to the file and click **OK.**

   To move policies using WLST:

   a. From the source environment, execute the following WLST commands:

```
exportMetadata(application='wsm-pm',server='server_name',
   docs='/assertiontemplates/assert_template_name',
   toLocation='/tmp/owsmexport/')
exportMetadata(application='wsm-pm',server='server_name',
   docs='/policies/policy_name',toLocation='/tmp/owsmexport/')
```

   b. Copy the /tmp/owsmexport directory from the source environment to the target environment.

   c. In the target environment, execute the following WLST commands:

```
importMetadata(application='wsm-pm',server='server_name',
  docs='/assertiontemplates/assert_template_name'',
```

```
  fromLocation='/tmp/owsmexport/')
importMetadata(application='wsm-pm',server='server_name',
  docs='/policies/policy_name',fromLocation='/tmp/owsmexport/')
```

d. If you have custom-built policies, move those by copying the jar files from the source to the target environment. The jar files are located in the following directory:

```
DOMAIN_HOME/lib
```

See Also: "Managing Application Migration Between Environments" in the *Oracle Fusion Middleware Security and Administrator's Guide for Web Services*

2. Oracle WebLogic Server JAX-WS applications use policies stored in wsm-seed-policies.jar instead of in MDS. Move these policies by copying the following file from the source environment to the target environment:

   ```
   ORACLE_HOME/modules/oracle.wsm.policies_11.1.1/wsm-seed-policies.jar
   ```

   You can also use the Oracle WebLogic Server Administration Console to move these policies.

3. Move any policy attachments for a SOA, ADF, or WebCenter Portal application if they have changed since the application was first deployed in the target environment. For example, policy A was initially configured in the source environment with the BASIC 128 algorithm and attached to the HelloWorld application. The application was deployed to the target environment. Then, on the source environment, you changed policy A to use the Basic 129 algorithm.
4. Move any policy attachments for JAX-WS applications if they have changed since the application was first deployed.

21.4.2 Moving Oracle SOA Suite to a Target Environment

The following topics describe how to move Oracle SOA Suite from the source environment to the target environment:

- Moving Oracle SOA Suite to a New Target Environment
- Moving Oracle SOA Suite to an Existing Target Environment

In both cases, you have performed the following in the source environment:

- Installed Oracle WebLogic Server and created the Middleware home.
- Created the needed schemas in the source environment using RCU. See the *Oracle Fusion Middleware Repository Creation Utility User's Guide*.
- Installed Oracle SOA Suite.
- Configured Oracle SOA Suite using the Configuration Wizard.
- If required for your environment, installed and configured Identity Management components, such as Oracle Internet Directory, Oracle Platform Security, and Oracle Web Services Manager.
- Configured security policies.
- Deployed one or more applications or SOA Composite applications. The applications have internal and external references.

- Changed some configuration settings. For example, you may have changed something in the config directory, in MDS, or another data source.
- Optionally, configured Oracle WebLogic Server dependent artifacts for Oracle Business Activity Monitoring, such as:
  - BAM Adapter
  - Data sources for the database or JMS
- Configured and populated the identity store for Oracle Business Activity Monitoring users.
- Set up UMS and all required subcomponents, configured UMS drivers and user preferences in the source environment.

Note: The Oracle User Messaging Service (UMS) is used in SOA and BAM procedures. The functionality and actions in both procedures are similar, but there are small differences. In particular, for BAM, only the email driver is supported, so the reconfiguration steps for UMS only apply to the email driver. Also, BAM does not make use of the UMS User Preferences in this release. Hence, the userprefs migration in UMS migration does not apply to BAM. See Task 7 for details on moving UMS from the source to the target environment.

21.4.2.1 Moving Oracle SOA Suite to a New Target Environment

See Also: *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle SOA Suite* for information about setting up an enterprise deployment for Oracle SOA Suite

To move Oracle SOA Suite to a new target environment, perform the following tasks:

- Task 1, "Move the Database, Middleware Home and Perform the Initial Configuration"
- Task 2, "Create Directory Structures"
- Task 3, "Export JKS Certificates"
- Task 4, "Move Human Workflow to the New Target Environment"
- Task 5, "Move Oracle Business Activity Monitoring Data to the New Target Environment"
- Task 6, "Move Oracle Business Process Management to the New Target Environment"
- Task 7, "Move Oracle User Messaging Service Details to the New Target Environment"
- Task 8, "Move Oracle Service Bus to a New Target Environment"

Task 1 Move the Database, Middleware Home and Perform the Initial Configuration

To move the database and Middleware home and perform the initial configuration:

1. Move or create the database and the schemas, as described in Section 21.3.3.
2. Move Identity Management components, as described in Section 21.4.1.

3. Move the Middleware home and binary files, as described in Section 21.3.4.
4. For Oracle Service Bus, if the SOA cluster migration is defined as a database data source, you must create the leasing schema, as described in "Setting Up the User and Tablespace for the Server Migration Leasing Table" in the *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle SOA Suite.*
5. Move the configuration, as described in Section 21.3.6.

   When you move the configuration, the pasteConfig script copies the configuration of the domain, including the Administration Server and Managed Servers. In addition, that step:

   - Moves SOA composite applications.
   - Moves Oracle Human Workflow attribute labels, flex field mappings, approval groups and standard views.
   - Moves Oracle B2B.
   - Reassociates the security store to an LDAP or database-based store, based on the values provided in move plan.
   - Moves Oracle Platform Security.
   - Moves Oracle Web Services Manager, any policies that are stored in the MDS Repository or deployment plans, and any custom policies that are stored in *DOMAIN_HOME*/lib.
   - Deploys applications in the target environment.
   - Configures adapters, such as the database adapters, AQ adapters, JMS adapters. Note, however, that you must edit the deployment plan of any adapters before you use the pasteConfig script.
   - Configures data sources.
   - Configures JMS resources.
   - Starts the Administration Server.
6. Configure users and groups, as described in Section 21.3.8.

Task 2 Create Directory Structures

> **See Also:** *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle SOA Suite* for information about setting up an enterprise deployment for Oracle SOA Suite

Create directory structures for any inbound or outbound files. For example, if you are using a file adapter that reads an inbound file from the /tmp/inbound_msg directory and writes outbound files to the /tmp/outbound_msg directory, create those directories on the target environment. Similarly, if Oracle B2B is using a listening channel that reads inbound messages from the /tmp/inbound directory and writes outbound messages to the /tmp/outbound directory, create those directories.

Task 3 Export JKS Certificates

Export any JKS certificates for B2B endpoints from the source environment to the target environment. Then, import them to the target environment. For information about exporting and importing JKS certificates, see Section 8.3.3.

Task 4 Move Human Workflow to the New Target Environment

When you moved a copy of the domain from the source environment to the target environment, the scripts moved the following Human Workflow entities:

- Attribute labels
- Flex field mappings
- Approval groups
- Standard views

Given that the movement scripts do not move user-specific artifacts, the following are not moved:

- User views
- User and group workflow rules

In most cases, the user-specific data is not the same in the target environment as in the source environment. However, if you want to move the user views and rules from the source environment to the target environment, see "Moving Human Workflow Data from a Test to a Production Environment" in the *Oracle Fusion Middleware Administrator's Guide for Oracle SOA Suite and Oracle Business Process Management Suite.*

Task 5 Move Oracle Business Activity Monitoring Data to the New Target Environment

To move Oracle Business Activity Monitoring to the new target environment:

1. At the source, export the ORACLEBAM database schema, using the following commands (*ORACLE_HOME* is the Oracle home for the Oracle Database):

   ```
   ORACLE_HOME/bin/sqlplus "sys/password as sysdba"
   create or replace directory directory as 'path';
   grant read,write on DIRECTORY directory to oraclebam;
   exit;

   ORACLE_HOME/bin/expdp userid=oraclebam/bam@connect_id
          directory=directory dumpfile=orabam.dmp
          schemas=oraclebam logfile=oraclebam_date.log
   ```

   See Also: "Overview of Oracle Data Pump" and other chapters on Oracle Data Pump in *Oracle Database Utilities*

   The Oracle BAM objects, such as reports, alerts, and data definitions from the source environment are exported.

2. At the target, import the ORACLEBAM database schema that you exported from the source environment, using the following commands (*ORACLE_HOME* is the Oracle home for the Oracle Database):

   ```
   ORACLE_HOME/bin/impdp userid=system/password dumpfile=ORACLEBAM.DMP
      remap_schema=oraclebam:oraclebam TABLE_EXISTS_ACTION=replace
   ORACLE_HOME/bin/sqlplus "sys/password as sysdba"
      alter user oraclebam account unlock;
      alter user oraclebam identified by bam;
   ```

   Note that impdp may report the following errors:

   - `ORA-00959: tablespace <source tablespace> does not exist.`

You can fix this error by creating the tablespace in the import database before the import or use REMAP_TABLESPACES to change the tablespace referenced in the table definition to a tablespace in the import database.

– You may see failure with restoring index statistics if you use an Oracle database version earlier than 11.2.0.2. You can work around this issue by rebuilding the index statistics after import.

3. Restart the Oracle Business Activity Monitoring Managed Server.

Task 6 Move Oracle Business Process Management to the New Target Environment

To move Oracle Business Process Management to the new target environment:

- To create organizational units, see "Managing Organizational Units in Process Workspace" in the *Oracle Fusion Middleware Getting Started With Installation for Oracle WebLogic Server*.
- To move dashboards, use the ant-t2p-workspace.xml migration tool. The migration tool is available as an ant target that can be executed in the command line. It calls a configuration file that you create specifying the input parameters for the migration of data, as described in this task.

  This script moves dashboards data with the BAM_WIDGET data type in the BPMUserApplicationData table to the target environment.

Note that the migration tool does not move any user-specific configuration because users in the source and target environments would not be same.

You use the following script:

```
ORACLE_HOME/bin/ant-t2p-workspace.xml
```

The command has the following format:

```
ant -f ant-t2p-workspace.xml
     -Dbea.home=BEA_HOME
     -Dbpm.home=BPM_HOME
     -Dbpm.t2p.migration.config=MIGRATION_CONFIG_FILE
```

Take the following steps:

1. Ensure that the PATH environment variable contains the required JAVA_HOME and ANT_HOME environment variables and that they point to the locations within the Oracle SOA Suite installation.
2. Export dashboards:
   a. Create a configuration file to export dashboards:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<testToProductionMigrationConfiguration
  xmlns="http://xmlns.oracle.com/bpm/t2p/migration/config"
  xmlns:ns2="http://xmlns.oracle.com/bpm/common"
  override="true" skip="true">
  <sourceEndPoint>
    <serverEndPoint>
      <serverURL>t3://host:port</serverURL>
      <adminUserLogin>admin_username</adminUserLogin>
      <adminUserPassword>admin_password</adminUserPassword>
      <realm>jazn.com</realm>
    </serverEndPoint>
  </sourceEndPoint>
```

```
  <targetEndPoint>
    <fileEndPoint>
       <migrationFile>/tmp/bpm_dashboard.xml</migrationFile>
    </fileEndPoint>
  </targetEndPoint>
  <operation>EXPORT</operation>
  <object>DASHBOARD</object>
  <objectDetails>
      <login>username</login>
      <password>password</password>
      <identityContext>jazn.com</identityContext>
      <userApplicationData>
         <ownerId>username/ownerId>
      </userApplicationData>
  </objectDetails>
</testToProductionMigrationConfiguration>
```

In the configuration file, you must specify the values for the source environment in the following elements:

- serverURL: The SOA server URL.
- adminUserLogin: The Administration user name.
- adminUserPassword: The password for the Administration user.
- migrationFile. The file that was generated by the export operation.
- objectDetails: The login and password elements.
- userApplicationData: The ownerID element.

b. Export dashboards, using the following command:

```
ant -f ant-t2p-workspace.xml
     -Dbea.home=BEA_HOME
     -Dbpm.home=BPM_HOME
     -Dbpm.t2p.migration.config=Dashboard_MIGRATION_CONFIG_FILE
```

3. Import dashboards:

a. Create a configuration file to import dashboards:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<testToProductionMigrationConfiguration
  xmlns="http://xmlns.oracle.com/bpm/t2p/migration/config"
  xmlns:ns2="http://xmlns.oracle.com/bpm/common"
  override="true" skip="true">
  <sourceEndPoint>
    <fileEndPoint>
       <migrationFile>/tmp/bpm_dashboard.xml</migrationFile>
    </fileEndPoint>
  </sourceEndPoint>
  <targetEndPoint>
    <serverEndPoint>
      <serverURL>t3://host:port</serverURL>
      <adminUserLogin>admin_username</adminUserLogin>
      <adminUserPassword>admin_password</adminUserPassword>
      <realm>jazn.com</realm>
    </serverEndPoint>
  </targetEndPoint>
  <operation>IMPORT</operation>
  <object>DASHBOARD</object>
  <objectDetails>
```

```
      <login>username</login>
      <password>password</password>
      <identityContext>jazn.com</identityContext>
      <userApplicationData>
        <ownerId>username/ownerId>
      </userApplicationData>
  </objectDetails>
</testToProductionMigrationConfiguration>
```

In the configuration file, you must update the following elements with the values for the target environment:

- serverURL: The SOA server URL.
- adminUserLogin: The Administration user name.
- adminUserPassword: The password for the Administration user.
- migrationFile. The file that was generated by the export operation.
- objectDetails: The login and password elements.
- userApplicationData: The ownerID element.

b. Import dashboards, using the following command:

```
ant -f ant-t2p-workspace.xml
     -Dbea.home=BEA_HOME
     -Dbpm.home=BPM_HOME
     -Dbpm.t2p.migration.config=Dashboard_MIGRATION_CONFIG_FILE
```

Task 7 Move Oracle User Messaging Service Details to the New Target Environment

To move UMS details to the new target environment:

1. Configure the required UMS drivers in the target environment.
   a. Use Fusion Middleware Control to configure the User Messaging Service drivers with target driver information.
   b. Use the WLST command `deployUserMessagingDriver` to deploy multiple drivers similar to the source environment.

   Note: To see different options for deploying additional drivers, execute `help('deployUserMessagingDriver')` at the `wls:/offline>` prompt.

   c. Re-create any custom-created *business terms* in the target environment. This step is essential in order to use the same set of *User Preferences* filter settings in the target environment, and to ensure that filters built with custom business terms are functional.
   d. Restart the target environment to apply the changes.
2. Move the User Messaging preferences from the source to the target environment:
   a. In the source environment, run the following WLST command to download the User Messaging preferences from the backend database to the specified .xml file:

```
wls:/offline> manageUserMessagingPrefs(operation='download',
    filename='/tmp/userprefs-dump.xml', url='t3://localhost:8001',
```

```
username='username', password='password')
```

In this example, 8001 is the Managed Server port on which UMS is running. Replace it with the appropriate value.

b. Copy the /tmp/userprefs-dump.xml file to the target environment.

c. In the target environment, run the following WLST command to upload the User Messaging preferences from file to the backend database:

```
wls:/offline> manageUserMessagingPrefs(operation='upload',
   filename='/tmp/userprefs-dump.xml', url='t3://localhost:8001',
   username='username', password='password')
```

In this example, 8001 is the Managed Server port on which UMS is running. Replace it with the appropriate value.

d. Observe the message displayed for successful upload. Exit the WLST command line tool.

Note: To see different options for performing download and upload operations, execute `help('manageUserMessagingPrefs')` at the `wls:/offline>` prompt. Please note that user devices provisioned in the LDAP store are dynamic. The assumption is that both the source and target environments point to the same LDAP store or you reconfigured it to use the same set of information.

---

e. Test the UMS drivers for send and receive capabilities for supported drivers.

f. Test the successful upload of user messaging preferences by invoking the `http://host:port/sdpmessaging/userprefs-ui` URL. Log in as the desired user and validate that the messaging channels and filters are identical to those in the test environment. Alternatively, send and receive messages that are expected to be delivered based on the User Messaging preferences.

Task 8 Move Oracle Service Bus to a New Target Environment

For information about moving Oracle Service Bus to a new target environment, see the chapter "Customization" in the *Oracle Fusion Middleware Administrator's Guide for Oracle Service Bus*. This chapter describes how to change environment values that differ between domains. Environment values are certain predefined fields in the configuration data whose values are very likely to change when you move your configuration from one domain to another (for example, from test to production).

21.4.2.2 Moving Oracle SOA Suite to an Existing Target Environment

In this procedure, you have a working target environment and want to test changes in your applications or configuration before rolling those changes into the target environment. In the source environment, you have the same environment as described in Section 21.4.2.

To move Oracle SOA Suite to an existing target environment:

- Task 1, "Move Oracle SOA Suite Changes to an Existing Target Environment"
- Task 2, "Move Oracle B2B Changes to an Existing Target Environment"
- Task 3, "Move Oracle Business Process Management Changes to an Existing Target Environment"

- Task 4, "Move Oracle Business Activity Monitoring Data to an Existing Target Environment"
- Task 5, "Move Oracle User Messaging Service Data to an Existing Target Environment"
- Task 6, "Move Oracle Service Bus to an Existing Target Environment"

Task 1 Move Oracle SOA Suite Changes to an Existing Target Environment

To move any changes that you have made to Oracle SOA Suite:

1. If you have added users and groups in the source environment, follow the steps in Section 21.3.8 to move them to the target environment.
2. If you have modified EJBs or Plain Old Java Objects (POJOs) in the source environment that support the composite references, move them to the target environment:
   a. To deploy EJB Modules, see "Deploy EJB Modules" in the Oracle WebLogic Server Administration Console Online Help.
   b. To deploy Enterprise Applications, see "Working with Enterprise Applications" in the Oracle WebLogic Server Administration Console Online Help.
   c. If you have made any changes to Human Workflow in the source environment, move them to the target environment. See "Moving Human Workflow Data from a Test to a Production Environment" in the *Oracle Fusion Middleware Administrator's Guide for Oracle SOA Suite and Oracle Business Process Management Suite*.
3. If you have modified any information in the configuration plans, copy those changes to the target environment. For more information about configuration plans, see "Customizing SOA Composite Applications for the Target Environment" in the *Oracle Fusion Middleware Developer's Guide for Oracle SOA Suite*.

Task 2 Move Oracle B2B Changes to an Existing Target Environment

If you have made any changes to Oracle B2B in the source environment, move those changes to the target environment.

Note that if you export selective agreements using the tpanames parameter, you must import each zip file individually.

To move Oracle B2B system changes:

1. Move Oracle B2B system configuration parameters by using the Oracle B2B interface to configure the properties. See "Configuring B2B System Parameters" in the *Oracle Fusion Middleware User's Guide for Oracle B2B* for details.
2. Move the B2B agreements and trading partners to the target environment:
   a. Export the data from the source environment. The following example exports multiple deployed and active agreements:

```
ant -f ant-b2b-util.xml b2bexport -Dtpanames="Acme_GC_Agreement1,
    GC_Acme_Agreement1" -Dactive=true -Dexportfile="/tmp/export.zip"
```

   b. Import the data to the target environment. The following example imports the elements in the file /tmp/export.zip:

```
ant -f ant-b2b-util.xml b2bimport -Dlocalfile=true
     -Dexportfile="/tmp/export.zip"
```

For more information about these commands, see "B2B Command Line Tools" in the *Oracle Fusion Middleware User's Guide for Oracle B2B*.

3. Configure B2B agreement external endpoints with target locations and credentials, as described in "Configuring Channels" in the *Oracle Fusion Middleware User's Guide for Oracle B2B*.
4. If your Oracle B2B environment has been configured with Java callouts, manually move the callout library. See "Managing Callouts" in the *Oracle Fusion Middleware User's Guide for Oracle B2B*.
5. Deploy the B2B agreements, as described in "Deploying an Agreement" in the *Oracle Fusion Middleware User's Guide for Oracle B2B*.

Task 3 Move Oracle Business Process Management Changes to an Existing Target Environment

If you have made any changes to Oracle Business Process Management in the source environment, re-create them or move them to the target environment:

- To create organizational units, see "Managing Organizational Units in Process Workspace" in the *Oracle Fusion Middleware Getting Started With Installation for Oracle WebLogic Server*.
- To move dashboards, as described in Task 6, "Move Oracle Business Process Management to the New Target Environment" in Section 21.4.2.1.

Task 4 Move Oracle Business Activity Monitoring Data to an Existing Target Environment

To move any Oracle BAM data that has changed:

1. Export Oracle BAM artifacts from the source environment using the icommand, which is located in the following directory:

   ```
   (UNIX) ORACLE_HOME\bam\bin\icommand.sh
   (Windows) ORACLE_HOME\bam\bin\icommand.bat
   ```

   For example:

   ```
   icommand -cmd export -type dataobject -all 1 -PERMISSIONS 1 -OWNER 1
     -file dataobject.xml
   icommand -cmd export -type folder -all 1 -PERMISSIONS 1 -OWNER 1
     -file folder.xml
   icommand -cmd export -type report -all 1 -file reports.xml
   icommand -cmd export -type rule -all 1 -file rules.xml
   icommand -cmd export -type ems -all 1 -file ems.xml
   icommand -cmd export -type eds -all 1 -file eds.xml
   ```

   In addition to exporting all artifacts of a particular type, you can export individual artifacts. For more information about using the icommand to export artifacts, see "Export" in the *Oracle Fusion Middleware Developer's Guide for Oracle SOA Suite.*

2. Export the BAM users from the LDAP identity store on the source environment, using the ldapsearch command. This produces an ldif file that you later import into the LDAP identity store in the target environment. The ldapsearch command is located in the *ORACLE_HOME*/bin directory of the Identity Management components. For example:

   ```
   ORACLE_HOME/bin/ldapsearch -h test_oid_host -p test_oid_port
   -D "cn=orcladmin"
   -w "test_orcladmin_passwd" -b "cn=Users,dc=us"
   ```

3. Import BAM data and artifacts into the target environment:

   a. Deactivate the rules that are set up by default, using Oracle BAM Architect. See "To change the activity status of an alert rule" in the *Oracle Fusion Middleware Developer's Guide for Oracle SOA Suite*.

   b. Import the BAM users from the ldif file that you exported from the source environment into the LDAP provider, such as Oracle Internet Directory, on the target environment. (*ORACLE_HOME* is the Oracle home for Identity Management.)

   ```
   ORACLE_HOME/bin/ldapadd -h production_oid_host -p production_oid_port
    -D "cn=orcladmin" -w production_orcladmin_passwd -vf ldif_filename
   ```

   c. Move the BAM application policy and roles to LDAP using Fusion Middleware Control:

      – From the navigation pane, right-click the domain that contains Oracle BAM and choose **Security,** then **Security Provider Configuration.**

      – Follow the steps in "Reassociating with Fusion Middleware Control" in the *Oracle Fusion Middleware Application Security Guide.*

   d. Import the Oracle BAM artifacts using the icommand, which is located in the following directory:

   ```
   (UNIX) ORACLE_HOME\bam\bin\icommand.sh
   (Windows) ORACLE_HOME\bam\bin\icommand.bat
   ```

   For example:

   ```
   icommand -cmd import -file dataobject.xml -UPDATELAYOUT 1
      -MODE UPDATE -CONTINUEONERROR
   icommand -cmd import -file folder.xml -MODE OVERWRITE -PRESERVEOWNER
   icommand -cmd import -file reports.xml -MODE OVERWRITE -PRESERVEOWNER
   icommand -cmd import -file ems.xml -MODE OVERWRITE
   icommand -cmd import -file eds.xml -MODE OVERWRITE
   ```

4. Start the BAM server.

Task 5 Move Oracle User Messaging Service Data to an Existing Target Environment

To move Oracle User Messaging Service data:

1. Configure the required UMS drivers in the target environment.

   Note: While moving Oracle User Messaging Service to an existing target environment configured against an LDAP Store, only use the *Userprefs-UI* option to change User Preferences. Using the WLST command `manageUserMessagingPrefs` is not recommended as it may not correctly migrate identity-store backed device preferences that have been removed from the source instance.

   a. Use Fusion Middleware Control to configure the User Messaging Service drivers with target driver information.

   b. Use the WLST command `deployUserMessagingDriver` to deploy multiple drivers similar to the source environment.

> **Note:** To see different options for deploying additional drivers, execute `help('deployUserMessagingDriver')` at the `wls:/offline>` prompt.

c. Re-create any custom-created business terms in the target environment. This step is essential in order to use the same set of User Preferences filter settings in the target environment, and to ensure that filters built with custom business terms are functional.

d. Restart the target environment to apply the changes.

2. Move the User Messaging preferences from the source environment to the target environment. Filters cannot be updated or appended to an existing filter set. You must do one of the following:
   - Delete the entire filter set and upload a new set if there are changes made to the filter set in the source environment.
   - Newly created or modified user devices and filters in the source environment must be created or modified using the following URL in the target environment:

     `http://host:port/sdpmessaging/userprefs-ui`

3. Test the UMS drivers for send and receive capabilities for supported drivers.
4. Test the successful upload of user messaging preferences by invoking the `http://host:port/sdpmessaging/userprefs-ui` URL. Log in as the desired user and validate that the messaging channels and filters are identical to those in the test environment. Alternatively, send and receive messages that are expected to be delivered based on the User Messaging preferences.

Task 6 Move Oracle Service Bus to an Existing Target Environment

For information about moving Oracle Service Bus to an existing target environment, see the chapter "Customization" in the *Oracle Fusion Middleware Administrator's Guide for Oracle Service Bus*. This chapter describes how to change environment values that differ between domains. Environment values are certain predefined fields in the configuration data whose values are very likely to change when you move your configuration from one domain to another (for example, from test to production).

21.4.3 Moving Oracle WebCenter Portal to a Target Environment

The following topics describe how to move Oracle WebCenter Portal from the source environment to the target environment:

- Moving Oracle WebCenter Portal to a New Target Environment
- Moving Oracle WebCenter Portal to an Existing Target Environment

In both cases, you have performed the following in the source environment:

- Installed Oracle WebLogic Server.
- Installed Oracle WebCenter Portal.
- Created the needed schemas in the source environment using RCU. See the *Oracle Fusion Middleware Repository Creation Utility User's Guide*.
- Installed and configured Oracle SOA Suite.

- Configured Oracle WebCenter Portal using the Configuration Wizard. You created a domain and Managed Servers for WebCenter Portal products.
- Installed and configured Oracle WebCenter Content.
- Installed Identity Management components, such as Oracle Internet Directory, Identity Federation, and Access Manager.
- Configured Oracle WebCenter Portal to use LDAP and created some users and groups in the embedded LDAP or an LDAP store.
- Created the required Oracle Platform Security Services policies in the policy store.
- Created the required user credentials in the credential store.
- Created your WebCenter Portal application by using WebCenter Spaces to build one or more spaces or by using JDeveloper to create and deploy your own portal application, or both.

21.4.3.1 Moving Oracle WebCenter Portal to a New Target Environment

To move Oracle WebCenter Portal to a new target environment, perform the following tasks:

- Task 1, "Move the Database, Middleware Home, and Perform the Initial Configuration"
- Task 2, "Move Discussion Server Data to the Target Environment (Optional)"
- Task 3, "Move Oracle WebCenter Portal: Spaces Data to the Target Environment (Optional)"
- Task 4, "Move Oracle WebCenter Content Documents and Folders Associated with Spaces (Optional)"

Task 1 Move the Database, Middleware Home, and Perform the Initial Configuration

Note that if the source environment includes portlet producers that are not being used, the portlet producer connection details are moved without the associated registrations and therefore, you must manually register those portlet producers in the target. When you register the portlet producers in the target, do not use the source connection names as they will conflict with the connections moved by this procedure.

To move the database and Middleware home, and perform the initial configuration:

1. Move or create the database, as described in Section 21.3.3.
2. Move Identity Management components, as described in Section 21.4.1.
3. Move the Middleware home and binary files, as described in Section 21.3.4.
4. Move Oracle HTTP Server (a requirement for Oracle WebCenter Content), as described in Section 21.4.6.1.1.
5. Move the configuration, as described in Section 21.3.6.

   Note that when you move the configuration, the pasteConfig script copies the configuration of the domain, including the Administration Server and Managed Servers. In addition, that step:

   - Creates authenticators for Identity Management in Oracle WebLogic Server.
   - Reassociates the policy and credential store.

- Moves WebCenter Portal application metadata from the source environment to the target environment.
- Starts the Administration Server.

Task 2 Move Discussion Server Data to the Target Environment (Optional)

If your Oracle WebCenter Portal application uses the Discussion service, move the discussion server data from the source environment to the target environment:

1. Export the discussion server data using the Oracle Database export utility from the *ORACLE_HOME*/bin (UNIX) and the *ORACLE_HOME*\bin (Windows) directories, where *ORACLE_HOME* is the Oracle home for the Oracle Database:

```
sqlplus "sys/password@connect_id as sysdba"
create or replace directory directory as 'path';
exit;

expdp "sys/password@connect_id as sysdba"
schemas=prefix_DISCUSSIONS directory=directory dumpfile=filename
```

2. Import the discussion server data using the following command, where *ORACLE_HOME* is the Oracle home for the Oracle Database:

```
ORACLE_HOME/bin/sqlplus "sys/password as sysdba"
create or replace directory directory as 'path';
exit;

ORACLE_HOME/bin/impdb "sys/password@connect_id as sysdba"
DIRECTORY=directory dumpfile=filename
TABLE_EXISTS_ACTION=REPLACE
```

Task 3 Move Oracle WebCenter Portal: Spaces Data to the Target Environment (Optional)

If you want to move Spaces application data such as spaces, space templates, and service-related data for lists, links, tags, people connections, to the target environment:

1. Export Spaces application data from the source database, using the following commands from the *ORACLE_HOME*/bin (UNIX) and the *ORACLE_HOME*\bin (Windows) directories, where *ORACLE_HOME* is the Oracle home for the Oracle Database:

```
sqlplus "sys/password as sysdba"
create or replace directory directory as 'path';
exit;

expdp "sys/password@connect_id as sysdba"
schemas=prefix_WEBCENTER directory=directory dumpfile=filename
```

2. Import Spaces application data to the target database, using the file you exported in Step 1. Execute the following commands, where *ORACLE_HOME* is the Oracle home for the Oracle Database:

```
ORACLE_HOME/bin/sqlplus "sys/password as sysdba"
create or replace directory directory as 'path';
exit;

ORACLE_HOME/bin/impdb "sys/password@connect_id as sysdba"
DIRECTORY=directory dumpfile=filename
TABLE_EXISTS_ACTION=REPLACE
```

Task 4 Move Oracle WebCenter Content Documents and Folders Associated with Spaces (Optional)

If you moved Spaces application data, as described in Task 3, or you want to move documents previously uploaded through Document service task flows to the target environment, move the Oracle WebCenter Content data, as described in Section 21.4.4.

21.4.3.2 Moving Oracle WebCenter Portal to an Existing Target Environment

In this procedure, you have a working target environment with Oracle WebCenter Portall installed and configured and you want to test changes in your applications or configuration before rolling those changes into the target environment.

Take the following steps:

1. To move changes to individual spaces or space templates to an existing target environment, refer to:
   - "Migrating individual spaces" in the *Oracle Fusion Middleware Administrator's Guide for Oracle WebCenter Portal*.
   - "Migrating spaces templates" in the *Oracle Fusion Middleware Administrator's Guide for Oracle WebCenter Portal*.
2. To propagate changes to WebCenter Portal applications built using WebCenter Portal: Framework to an existing target environment, refer to "Using the Propagation Tool to Propagate From Staging to Production" in the *Oracle Fusion Middleware Developer's Guide for Oracle WebCenter Portal*.

21.4.4 Moving Oracle WebCenter Content to a Target Environment

The following topics describe how to move Oracle WebCenter Content to the target environment:

- Moving Oracle WebCenter Content to a New Target Environment
- Moving Oracle WebCenter Content to an Existing Target Environment

In both cases, you have performed the following in the source environment:

- Installed a database to be used for required schemas.
- Created the needed schemas in the source environment using RCU. See the *Oracle Fusion Middleware Repository Creation Utility User's Guide*.
- Installed Oracle WebLogic Server and created the Middleware home.
- Installed and configured Oracle WebCenter Content.
- Configured Oracle WebCenter Content.
- Configured Oracle WebCenter Content: Imaging.

  If Imaging uses Oracle Universal Content Management 10*g* repository, ensure that the repository was manually configured for Imaging.
- If Oracle WebCenter Content: Imaging uses Workflow or Oracle Application Extension Framework (AXF), installed and configured Oracle SOA Suite.
- Configured Oracle WebCenter Content: Records.
- Defined several definitions, such as Connections, Applications, Searches, and Inputs for Imaging.
- Installed and configured Identity Management components, such as Oracle Internet Directory.

21.4.4.1 Moving Oracle WebCenter Content to a New Target Environment

To move Oracle WebCenter Content to a new target environment, perform the following tasks:

- Task 1, "Move the Database, Middleware Home, and Perform the Initial Configuration"
- Task 2, "Configure Full-Text for Oracle Universal Content Management 10g"
- Task 3, "Modify Oracle Information Rights Management Settings"
- Task 4, "Move Oracle WebCenter Content to a New Target Environment"
- Task 5, "Move Oracle WebCenter Content: Imaging to a New Target Environment"
- Task 6, "Move Oracle WebCenter Content: Records to a New Target Environment"
- Task 7, "Move Oracle WebCenter Content: Inbound Refinery to a New Target Environment"

Note that in a target environment, Oracle WebCenter Content applications need to use an external Lightweight Directory Application Protocol (LDAP) authentication provider rather than the Oracle WebLogic Server embedded LDAP server, which is part of the default configuration. You reassociate the identity store for your application with one of the following external LDAP authentication providers before you complete the configuration of a Managed Server, before you connect a Managed Server to a repository, and before the first user logs in to the application:

- Oracle Internet Directory
- Oracle Virtual Directory
- A third-party LDAP Server

Task 1 Move the Database, Middleware Home, and Perform the Initial Configuration

To move the database and the Middleware home, and perform the initial configuration:

1. Move or create the database, as described in Section 21.3.3.
2. Move Identity Management components, as described in Section 21.4.1.

   Note that because Oracle WebCenter Content: Imaging uses Oracle Internet Directory, you need to move users and groups into the LDAP identity store on the target system.
3. Move the Middleware home and binary files, as described in Section 21.3.4.
4. Move the configuration, as described in Section 21.3.6.

   Note the following:

   - For the Oracle WebCenter Content server or Oracle WebCenter Content: Records, you have two options for moving the component:
     - copy: This option copies the entire source system, including configuration and data, to the target system.
     - init: This option initializes a new Content Server or Records instance in the target system.

     You specify the copy or init option in the move plan, in the MoveType configProperty, as described in Table 20–27. Then, you modify the properties listed in that configGroup.

- If the Administration Server and the component servers (WebCenter Content server, Records server, and Inbound Refinery servers) are on separate hosts and the domain home is not on a shared disk, take the following steps:

  a. Before you run the copyConfig script, create soft links for each of the following:

  ```
  Admin_Server_domain_home/ucm/cs
  to
  Content_Server_domain_home/ucm/cs

  Admin_Server_domain_home/ucm/urm
  to
  URM_Server_domain_home/ucm/urm

  Admin_Server_domain_home/ucm/ibr
  to
  IBR_Server_domain_home/ucm/ibr
  ```

  Check the value of IntradocDir in the following files to make sure that the path is mounted and accessible from the Administration Server host:

  ```
  Admin_Server_domain_home/ucm/cs/bin/intradoc.cfg
  Admin_Server_domain_home/ucm/urm/bin/intradoc.cfg
  Admin_Server_domain_home/ucm/ibr/bin/intradoc.cfg
  ```

  Note that in a high-availability setting where you have multiple WebCenter Content hosts, you can create the soft link to any WebCenter Content host.

  b. If you are using the copy option and the IntradocDir is not a subdirectory of the *Admin_Server_domain_home*/ucm directory, take the following steps on the target system, before you run the pasteConfig script:

  Mount the IntradocDir on the Administration Server host. Modify the move plan, specifying the path of the IntradocDir for WebCenter Content server, Records, and Inbound Refinery on the Administration Server host.

  After you run the pasteConfig script, update the following file to specify the location of the IntradocDir on the WebCenter Content server host:

  ```
  Admin_Server_domain_home/ucm/cs/bin/intradoc.cfg
  Admin_Server_domain_home/ucm/urm/bin/intradoc.cfg
  Admin_Server_domain_home/ucm/ibr/bin/intradoc.cfg
  ```

  Note that in a high-availability setting where you have multiple WebCenter Content hosts, edit the file on each host.

- When you use the copy option, the pasteConfig script copies the configuration of the domain, including the Administration Server and Managed Servers. In addition, that step:
  - Copies the configuration, including the modified settings, of Oracle WebCenter Content and its components.
  - Copies the Imaging sample input files, if they are in the domain directory.
  - Copies the BPEL credentials.
  - Moves Oracle Web Services Manager policies.
  - Sets the Listen address for the Managed Server that contains Oracle Application Extension Framework (AXF).

  - Starts the Administration Server and Managed Servers.

- For the Oracle WebCenter Content server and Oracle WebCenter Content: Records, the init option copies the following initialization properties from the source system:

```
IDC_Name
IDC_Id
InstanceMenuLabel
InstanceDescription
IntradocServerPort
IdcCommandServerHost
SocketHostAddressSecurityFilter
HttpServerAddress
HttpRelativeWebRoot
UseSSL
MailServer
SysAdminAddress
IsAutoNumber
AutoNumberPrefix
AdditionalRegisteredComponents
AdditionalEnabledComponents
```

5. Configure users and groups, as described in Section 21.3.8.

Task 2 Configure Full-Text for Oracle Universal Content Management 10*g*

If you are using an Oracle Universal Content Management 10*g* repository, ensure that Full-Text is configured correctly on the target Oracle UCM system, if configured on the source Oracle Universal Content Management 10*g* system.

Note that if you are using an Oracle Universal Content Management 10*g* server, it is not configured when you move the Oracle WebCenter Content. you must install it, similar to the way you installed it in the source environment, using the procedures described in the Oracle Universal Content Management page at:

http://www.oracle.com/technetwork/middleware/content-management/overview/index.html

Task 3 Modify Oracle Information Rights Management Settings

To move Oracle IRM, you need to modify some Oracle IRM settings in the new target environment:

1. Set up SSL. For Oracle IRM, SSL should be enabled so that Oracle IRM Desktop does not show prompts to accept certificates when it contacts the Managed Server. The certificate used must be trusted by Microsoft Internet Explorer on computers running Oracle IRM Desktop. Follow the standard SSL setup instructions for Oracle WebLogic Server, as described in "Configuring SSL" in *Oracle Fusion Middleware Securing Oracle WebLogic Server.*

2. Each Oracle IRM installation requires access to a keystore with installation specific keys. The unpacked domain may have a keystore. If it does and if the Content Tracker component is enabled and being used in their source environment., delete this keystore, clear the passwords details, and create a new keystore:

   a. Delete the keystore file. By default, the keystore is located in the following directory:

      DOMAIN_HOME/config/fmwconfig

By default, the file name is irm.jks. It may be named differently or use a different type, depending on the template used.

b. Keystore passwords are stored in the credential store. If passwords have been set in the template domain, clear the passwords using the following WLST commands:

```
connect('username', 'password', 'localhost:7001')
deleteCred('IRM', 'keystore:keystore_filename')
deleteCred('IRM', 'key:irm.jks:oracle.irm.wrap')
```

For the key, you use the keystore file name stored in the template.

c. Create a new keystore, as described in "Configuring the Keystore for Oracle IRM" in the *Oracle WebCenter Content Installation Guide*.

3. If the target environment is not using the same LDAP store as the source environment, migrate the users from the source environment to the target environment. See "Reassociating the Identity Store with an External LDAP Authentication Provider" in the *Oracle WebCenter Content Installation Guide*.

Task 4 Move Oracle WebCenter Content to a New Target Environment

If you selected the init option in the move plan, the only step you need to take is Step 3, but you need to do that only if your environment has a full text search solution using external databases.

If you selected the copy option in the move plan, take the following steps:

1. Export the OCS database schema from the source environment, using the following command (*ORACLE_HOME* is the Oracle home for the Oracle Database):

```
ORACLE_HOME/bin/expdp \"sys/password as sysdba\"
       schemas=test_env_schema_name
       directory=directory dumpfile=ucm.dmp
```

Make sure that the dumpfile is in a location that can be accessed by the target database.

2. Import the OCS database schema that you exported from the source environment, using the following commands (*ORACLE_HOME* is the Oracle home for the Oracle Database):

```
ORACLE_HOME/bin/impdp \"sys/password as sysdba\"
      remap_schema=test_env_schema_name:prod_env_schema_name
      directory=directory dumpfile=ucm.dmp
      TABLE_EXISTS_ACTION=REPLACE
```

3. For a system that has a full text search solution using external databases, set up Oracle Secure Enterprise Search and configure it for WebCenter Content on the target system:

a. Install Oracle Secure Enterprise Search as described in the *Oracle Secure Enterprise Search Installation and Upgrade Guide*.

b. If you selected init in the move plan, on the Oracle WebCenter Content Post Configuration page, choose **External Full Text Search,** and enter the name of the data source. See "Configuring Oracle SES and Oracle UCM" in the *Oracle WebCenter Content System Administrator's Guide for Content Server*.

4. If you configured the IntradocDir, WeblayoutDir, VaultDir, and UserProfilesDir directories to be outside of the domain structure, copy the directories to the target environment.
5. Restart the Administration Server and the Managed Server.

Task 5 Move Oracle WebCenter Content: Imaging to a New Target Environment

Note that when you use the copy mode with Oracle WebCenter Content, Imaging data is moved. However, if you have created instances within SOA, those instances are not moved, because the Oracle SOA Suite procedures do not move this data.

Before you begin this procedure, if you use Workflow Integration or Oracle Application Extension Framework (AXF), make sure that you have:

- Installed and configured Oracle SOA Suite and moved its source environment to the target environment, as described in Section 21.4.2.1.
- Configured Imaging, extending the SOA domain, as described in "Extending an Existing Domain" in the *Oracle WebCenter Content Installation Guide*.

To complete the movement of Imaging to the new target environment:

1. Start the Administration Server and the Imaging Managed Server.
2. Move the Oracle Application Extension Framework (AXF) configuration database:
   a. If you have installed the EBS adapter as part of AXF, export the following tables from the source EBS database schema and insert them into the target database schema:
      - AXF_COMMAND_PARAMETERS
      - AXF_COMMANDS
      - AXF_CONFIGS
      - AXF_FND_MAP
      - AXF_PROPERTIES
   b. Modify the AXF_CONFIGS table in the EBS schema to point to the solution endpoint in the target AXF system, using the following command:

```
UPDATE AXF_CONFIGS SET SOLUTIONENDPOINT = 'AXFConnectionURL'
```

Task 6 Move Oracle WebCenter Content: Records to a New Target Environment

If you selected the init option in the move plan, you do not need to take any additional steps.

If you selected the copy option in the move plan, take the following steps:

1. Export the Records database schema (*prefix*_urmserver) from the source environment, using the following command (*ORACLE_HOME* is the Oracle home for the Oracle Database):

```
ORACLE_HOME/bin/expdp \"sys/password as sysdba\"
      schemas=test_env_schema_name
      directory=directory dumpfile=urm.dmp
```

   Make sure that the dumpfile is in a location that can be accessed by the target database.

2. Import the Records database schema that you exported from the source environment, using the following commands (*ORACLE_HOME* is the Oracle home for the Oracle Database):

```
ORACLE_HOME/bin/impdp \"sys/password as sysdba\"
      remap_schema=test_env_schema_name:prod_env_schema_name
      directory=directory dumpfile=urm.dmp
      TABLE_EXISTS_ACTION=REPLACE
```

3. If you configured the IntradocDir, WeblayoutDir, VaultDir, and UserProfilesDir directories to be outside of the domain structure, copy the directories to the target environment.
4. Restart the Administration Server and the Managed Server.

Task 7 Move Oracle WebCenter Content: Inbound Refinery to a New Target Environment

To complete the movement of Oracle WebCenter Content: Inbound Refinery to a new target environment:

1. If you configured the IntradocDir, WeblayoutDir, VaultDir, and UserProfilesDir directories to be outside of the domain structure, copy the directories to the target environment.
2. If you have set up third-party software to convert certain types of content, additional steps may be required. The third-party software could include FlipFactory to convert video, Microsoft Office to perform native conversion for Office documents, or Adobe Distiller to convert PDF files. If you have installed this software, note the following:
   - You must install the software on the target system.
   - Oracle recommends that you install the third-party software and fonts at the exact same absolute path on the target system as they are on the source system. This ensures that Inbound Refinery is properly configured at the start up of the target system.
   - If you do not install the third-party software and fonts at the exact same absolute path, you must perform the same manual steps to configure the software on the target system as you did on the source system.
   - Do not submit any jobs that require the software before you complete the configuration. Otherwise, the conversion will fail.
3. Start the Managed Server.

21.4.4.2 Moving Oracle WebCenter Content to an Existing Target Environment

In this procedure, you have installed Oracle WebCenter Content components in the source environment and you want to move them to the target environment that already exists.

To move Oracle WebCenter Content to an existing target environment, perform the following tasks:

- Task 1, "Move Oracle Information Rights Management to an Existing Target Environment"
- Task 2, "Move Oracle WebCenter Content to an Existing Target Environment"
- Task 3, "Move Oracle WebCenter Content: Imaging to an Existing Target Environment"

- Task 4, "Move Oracle WebCenter Content: Records to an Existing Target Environment."

Task 1 Move Oracle Information Rights Management to an Existing Target Environment

Organizations that run a proof of concept or pilot (source) deployment can copy the operational service into the target environment and continue to use all existing source content, contexts, and rights.

The IRM server URL (for example *protocol_schema*:*hostname*:*port*\irm_desktop) is sealed into source content. Therefore, this value must not change on moving from source to target. For this reason, make sure you consider the following points when installing the source deployment:

- Configure SSL in the source deployment because switching from the HTTP protocol in the source environment to the HTTPS protocol in the target environment would prevent source-sealed content from working in the target environment.
- Use a generic host name such as irm.example.com for the source deployment rather than a machine-specific host name such as mytestdeploymachine.example.com.

After the source to target installation has been completed, the DNS entries for domain name can be switched from the source server to the target environment. If needed, you can use port redirection to ensure that the source deployment IRM Server URL points to the target environment deployment.

To move the source deployment into the target environment:

1. If the target database is different from the source database, you should back up the Oracle IRM schema. Restore the backup into the target database.
2. Copy the Oracle IRM keystore set up during the source installation to the target environment. This is typically called `irm.jks`. This file is usually located in the following directory:

   ```
   DOMAIN_HOME/config/fmwconfig
   ```

3. The Oracle IRM Java EE application needs a password for the keystore copied in the previous step and each key stored in that keystore. If the passwords are not specified, the Oracle IRM Java EE application will not be able to retrieve the keys.

   To switch to using more secure passwords than those used in the source environment, use the keytool command line to change the passwords before proceeding. See the keytool Help for syntax.

4. With secure passwords in place, use WLST commands to specify these passwords to the Oracle IRM Java EE application. The following example connects to an Administration Server and sets the keystore credentials:

   ```
   connect("username", "password", "t3://adminServerHost:adminServerPort")
   createCred("IRM", "keystore:irm.jks", "dummy", "secureproductionpassword")
   createCred("IRM", "key:irm.jks:oracle.irm.wrap", "dummy",
   "secureproductionpassword")
   ```

   For more information, see "Adding Key Store Passwords to the Credential Store" in the *Oracle WebCenter Content Installation Guide*.

5. Copy the Oracle IRM configuration file, irm-config.xml, which is usually located in the following directory, from the source environment to the target environment:

DOMAIN_HOME/config/fmwconfig

6. Because the source environment configuration may contain source-specific settings, you should review the contents of the file. You can use Fusion Middleware Control, WLST, or you can edit the configuration file, irm-config.xml. To use Fusion Middleware Control, expand the navigation tree and click **IRM.** From the IRM menu, choose **Administration,** then **General Settings.** The following settings may need to be changed:
   - **Privacy URL:** A URL to a page hosting the Oracle IRM usage privacy policy for the installation. There is no default value, so typically you do not need to alter this setting after unpacking a domain. The default behavior is to show the built-in privacy page.
   - **Status Page Redirection:** An optional URL to a page hosting alternative Oracle IRM Desktop status pages. There is no default value, so typically you do not need to alter this setting after a domain is unpacked. The default behavior is to use the built-in status pages.
   - **Keystore location:** The path should reflect the location of the restored source environment keystore. The following is the suggested location of the file:

     DOMAIN_HOME/config/fmwconfig

7. If the target environment is not using the same user store as the source environment, migrate the users from the source environment to the target environment. See "Reassociating the Identity Store with an External LDAP Authentication Provider" in the *Oracle WebCenter Content Installation Guide*.

Task 2 Move Oracle WebCenter Content to an Existing Target Environment

To move Oracle WebCenter Content to an existing target environment:

1. Select the Configuration Templates option from the Migration Options or from the top menu on any Migration screen.
2. From **Actions,** select **Create New Template.**
3. For **Server Config,** select **SearchIndexEngineName.**
4. For **Content Metadata,** select the text fields that you want to export.
5. For **Content Profile Rules,** select the rules that you want to export.
6. For **Personalization Data,** select the profiles that you want to export.
7. From **Action,** select **Save.**
8. From **Action,** select **Export.**
9. Click **Configuration Bundles.**
10. On the Configuration Bundles page, select the bundle you created when you exported the data. Then, from **Action,** select **Download.**
11. If you are using Records Manager for UCM and you want to perform incremental migrations from the source environment to the target environment, export archives from the source environment and then import them into the target environment, as described in "Managing Imports and Exports" in the *Oracle WebCenter Content Administrator's Guide for Records.*

Task 3 Move Oracle WebCenter Content: Imaging to an Existing Target Environment

To move Oracle WebCenter Content: Imaging from the source environment to an existing target environment, you use the same steps as described in Task 5, "Move Oracle WebCenter Content: Imaging to a New Target Environment". However, note the following about updating definitions on the target environment:

- When you import a definition from the source environment to the existing target environment and the definition has the same name as an existing definition, the original definition is overwritten. The following rules apply to importing existing definitions:
  - If an application deletes a field, it is not imported if the any of the existing search or input definitions refer to the deleted field.
  - If a search or input definition references a field that is not in the currently defined in the application, the definition is not imported.
- You cannot delete definitions through the export and import process. If you delete a search in the source environment, you must manually delete it in the target environment using the Manage Search functions.
- You cannot import an input definition if there is an existing definition with the same name and that input definition is online. To import the definition, you must first place the definition offline:
  1. On the target environment, open the Managed Inputs folder and select the input that you want to import.
  2. Select **Toggle On-Line.**

Task 4 Move Oracle WebCenter Content: Records to an Existing Target Environment.

To move Records to an existing target environment:

1. On the source environment, export any configuration settings that have changed, as described in "Exporting an Archive" in *Oracle WebCenter Content Administrator's Guide for Records.*
2. Copy the archive to the target environment.
3. Import the archive to the target environment, as described in "Importing an Archive" in *Oracle WebCenter Content Administrator's Guide for Records.*

21.4.5 Moving Oracle Hyperion Enterprise Performance Management System to a Target Environment

The section describes how to move Oracle Hyperion Enterprise Performance Management System components to the target environment.

In this procedure, you have performed the following in the source environment:

- Installed a database to be used for required schemas.
- Created the needed schemas in the source environment using RCU. See the *Oracle Fusion Middleware Repository Creation Utility User's Guide*.
- Installed Oracle WebLogic Server and created the Middleware home.
- Installed and configured Oracle Hyperion Enterprise Performance Management components.

To move Oracle Hyperion Enterprise Performance Management to the target environment, perform the following tasks:

- Task 1, "Move the Database, Middleware Home, and Perform the Initial Configuration"
- Task 2, "Move Oracle Essbase to a Target Environment"
- Task 3, "Move Oracle Hyperion Calculation Manager to a Target Environment"
- Task 4, "Move Oracle Hyperion Financial Reporting to a Target Environment"
- Task 5, "Move Oracle Hyperion Provider Services to a Target Environment"
- Task 6, "Move Oracle Hyperion Smart View to a Target Environment"
- Task 7, "Move Oracle EPM Workspace to a Target Environment"

Task 1 Move the Database, Middleware Home, and Perform the Initial Configuration

To move the database and Middleware home, and perform the initial configuration:

1. Move or create the database, as described in Section 21.3.3.
2. Move Identity Management components, as described in Section 21.4.1.
3. Move the Middleware home and binary files, as described in Section 21.3.4.
4. Move the configuration, as described in Section 21.3.6.
5. Configure users and groups, as described in Section 21.3.8.

Task 2 Move Oracle Essbase to a Target Environment

For Oracle Essbase, only configuration settings need to be moved from the source environment to the target environment. For example, copy `essbase.cfg` from the source to the target environment:

```
ORACLE_INSTANCE/Essbase/essbaseserver1/bin/essbase.cfg
```

Task 3 Move Oracle Hyperion Calculation Manager to a Target Environment

To move Oracle Hyperion Calculation Manager from the source environment to the target environment, do one of the following:

- Back up the repository

  Since the Calculation Manager schema does not change, you can use the same repository with the new Calculation Manager environment. All pre-existing Calculation Manager objects and other related information are available in the new environment.

- Export/import allocation rules and rule sets

  Export Calculation Manager rules and rule sets from the source environment (in Calculation Manager, select **File,** and then **Export**) and then import them back into the target environment (in Calculation Manager, select **File,** and then **Import**).

Notes:

- Use only one (not both) of the above options to move from the source environment to the target environment. Use the option that makes the most sense for your environment. The export/import option is simpler; however, backing up the repository saves more information in the database.
- Rules that have already been deployed to Fusion GL and have since been modified in Calculation Manager will not be handled by Calculation Manager. (Calculation Manager does not keep versions of rules).

Task 4 Move Oracle Hyperion Financial Reporting to a Target Environment

You can export Oracle Hyperion Financial Reporting content from the source environment and import it into the target environment:

1. Export Financial Reporting report content from the source environment:
   a. Log in to the source environment.
   b. Select **File,** and then **Export.**
   c. Navigate to and select the content or directory that you want to move to the target environment.
   d. Export the selected content or directory to the local file system.
2. Import Financial Reporting report content into the target environment:
   a. Log in to the target environment and select **File,** then **Import,** and then **Financial Reporting.**
   b. Browse to the target location where you want to import the Financial Reporting report content, and select the local file to which you saved the exported content.

Note: Oracle Hyperion Financial Reporting annotations and scheduler output cannot be migrated from the source environment to the target environment.

Task 5 Move Oracle Hyperion Provider Services to a Target Environment

Oracle Hyperion Provider Services artifacts need to be copied from source environment to the target environment.

- To move Provider services when it is used by Smart View:
  1. Use the Smart View client to manually add any Oracle Essbase servers created in the source server to the target server.
  2. Copy the following file to the target environment:

     ORACLE_INSTANCE/products/Essbase/aps/bin/essbase.properties

  3. From Smart View, re-create any Cube views created under the source environment in the target environment.
- To move Provider Services when it is used by the Oracle Essbase Java API:

If the default Java API preferences are changed in the `essbase.properties` file on the Java API client program configuration, then copy `essbase.properties` to the target environment.

Task 6 Move Oracle Hyperion Smart View to a Target Environment

Because Oracle Hyperion Smart View for Office is a client side application, spreadsheets and other Microsoft Office documents created with source servers must be associated with target server connections. You can point an existing report from source to target if the metadata remains the same.

To associate shared connections:

1. Open the existing report.

   The location of existing reports depends on where you stored the reports when you first created them.

2. In Excel with Smart View installed, select **Smart View,** then **Options,** and then **Advanced.**

3. Change the Shared Connections URL to the new connection URL; for example:

   ```
   https://host.example.com/workspace/SmartViewProviders
   ```

4. Select **Smart View,** then **Open,** then **Smart View Panel,** then **Shared Connections,** and do one of the following:

   - If Essbase Server is not listed, click **Create New Connection**.
   - If Essbase Server is listed, enter the user name and password, and click **Connect**.

5. From the drop-down list, select **Essbase Server.**

   a. From the drop-down list, select **Locate worksheet connection.**

      The connection is created under cube.

   b. Select the connection and right-click to connect.

6. Click **Refresh.**

7. For ad-hoc analysis, you need to connect to a new server and choose to maintain POV. To do this, select **Reuse sheet contents and POV** when prompted.

To associate private connections:

1. Select **Smart View,** then **Open,** and then **Smart View Panel.**

2. Select **Private Connections.**

3. Enter the Provider Services URL: for example:

   ```
   https://host.example.com/aps/SmartView
   ```

4. Log in to Oracle Hyperion Provider Services.

5. Select the Oracle Essbase application and log in.

6. Select the Oracle Essbase application, right-click, and select **Add to Private Connections.**

7. Enter the connection name or use the default name, and click **OK.**

8. Associate the connection (**SVC,** then **Open,** and then **Active Connection**) and select the connection. Click **OK** to confirm the message.

9. Click **Refresh.**
10. For ad-hoc analysis, you need to connect to a new server and choose to maintain POV. Select **Reuse sheet contents and POV** when prompted.

Task 7 Move Oracle EPM Workspace to a Target Environment

When you move Oracle Enterprise Performance Management Workspace information from the source environment to the target environment, the system settings and user preferences must be manually migrated. Any changes to server settings and user preferences must be made in the target system. See "Administering EPM Workspace" in the *Oracle Enterprise Performance Management Workspace, Fusion Edition Administrator's Guide*.

21.4.6 Moving the Web Tier to a Target Environment

In this procedure, you have installed Oracle HTTP Server and Oracle Web Cache in the source environment and you want to move them to the target environment.

The following topics describe how to move the Web tier from the source environment to the target environment:

- Moving the Web Tier to a New Target Environment
- Moving the Web Tier to an Existing Target Environment

21.4.6.1 Moving the Web Tier to a New Target Environment

The following topics describe how to move the Web tier to a new target environment:

- Moving Oracle HTTP Server to a New Target Environment
- Moving Oracle Web Cache to a New Target Environment

21.4.6.1.1 Moving Oracle HTTP Server to a New Target Environment In this procedure, you have installed Oracle HTTP Server in the source environment and you want to move it to the target environment, which does not yet exist. In the source environment, you have:

- Installed Oracle HTTP Server.
- Created an Oracle instance and one or more Oracle HTTP Server component instances.
- Registered the Oracle instance and the Oracle HTTP Server component instances, with an existing JRF-enabled Oracle WebLogic Server Administration Server if you want to manage the components with Fusion Middleware Control.
- Configured mod_wl_ohs to route requests to one or more virtual hosts.
- Configured SSL for one or more virtual hosts.
- Configured Oracle Single Sign-On.
- Configured mod_plsql.
- Configured mod_oradav.
- In addition, you may be using Access Manager. In this procedure, the Access Manager Access Servers are not in the source environment. They reside on a separate target environment. However, WebGate is running in the source environment.

To move this environment to a new target environment, perform the following tasks:

- Task 1, "Move Access Manager If It Uses Oracle HTTP Server"
- Task 2, "Move the Middleware Home and Perform the Initial Configuration"
- Task 3, "Start the Processes"

Task 1 Move Access Manager If It Uses Oracle HTTP Server

If Oracle HTTP Server is used by WebGate, you must first move Access Manager to the target environment, as described in Section 21.4.1, Task 7 or Task 8, depending on your version of Access Manager.

Note the following:

- The WebGateInstalldir property and references to this path are updated in the webgate.conf file.
- The WebGate directory must be in the following directory:

  Oracle_Instance/config/OHS/*ohs_component_name*

Task 2 Move the Middleware Home and Perform the Initial Configuration

To move the Middleware home and perform the initial configuration:

1. Move the Middleware home and binary files, as described in Section 21.3.4.
2. Move the configuration. You can choose to:
   - Move the Oracle instance and all of the components in that particular Oracle Instance, as described in Section 21.3.7.1.
   - Move the Oracle instance and only one component in that particular Oracle instance, as described in Section 21.3.7.2.

   This step moves the configuration. In addition, it:
   - Updates the Listen address and the name of the virtual host.
   - Configures SSL, if it was configured in the source environment.
   - Updates the httpd.conf file with new values for the environment and topology directives, such as host name and IP address.
   - Updates the WebLogicHost, WebLogicPort, or WebLogicCluster directives in the mod_wl_ohs.conf file with the host name, IP address, and port number for the target environment.
   - Configures SSL for mod_wl_ohs, if SSL is configured for mod_wl_ohs.
   - Configures mod_osso, if it was configured in the source environment.
   - Configures PL/SQL, if it was configured in the source environment.
   - Configures mod_osso, if it was configured in the source environment.
   - Updates audit.config.xml, if any changes were made to it in the source environment.
   - Updates *component*-log.xml, if any changes were made to it in the source environment.
   - Configures WebGate if you are using Access Manager.

Task 3 Start the Processes

Start the processes in the Oracle instance:

ORACLE_INSTANCE/bin/opmnctl stopall

```
ORACLE_INSTANCE/bin/opmnctl startall
```

21.4.6.1.2 Moving Oracle Web Cache to a New Target Environment In this procedure, you have installed Oracle Web Cache in the source environment and you want to move it to the target environment, which does not yet exist. In the source environment, you have:

- Installed Oracle Web Cache.
- Configured two or more Oracle instances, each containing an Oracle Web Cache instance.
- Registered the Oracle instances and the Oracle Web Cache instances with an existing JRF-enabled Oracle WebLogic Server Administration Server, if you want to manage the components with Fusion Middleware Control.
- Configured the Oracle Web Cache instances as a Oracle Web Cache cluster.
- Created a site and configured site-to-server mapping.
- Configured Oracle Web Cache to have an SSL-enabled listening address.
- Configured caching rules, and defined filters for request filtering.

> **Note:** If you have already moved the entire Oracle instance, as described in Section 21.3.7.1, any Oracle Web Cache instance is already moved. For example, if you moved Oracle HTTP Server, and chose to move the entire Oracle instance and that Oracle instance contains Oracle Web Cache, Oracle Web Cache is moved along with Oracle HTTP Server.
>
> In that case, you can omit Task 1.

To move this environment to a new target environment, perform the following tasks:

- Task 1, "Create the Oracle Instances and the Oracle Web Cache Instances"
- Task 2, "Update Oracle Web Cache"

Task 1 Create the Oracle Instances and the Oracle Web Cache Instances

In the target environment, move the binary files and create the Oracle instances and Oracle Web Cache instances:

1. Move the Middleware home and binary files, as described in Section 21.3.4.
2. Create the Oracle instances and the Oracle Web Cache instances:
   a. From the command line, go the following directory:

      ```
      (UNIX) ORACLE_HOME/opmn/bin
      (Windows) ORACLE_HOME\opmn\bin
      ```

   b. Create the Oracle instances, using the opmnctl `createinstance` command. For example:

      ```
      opmnctl createinstance -oracleInstance /scratch/Oracle/Middleware/inst1
         -adminHost hostname -adminPort 7001
      ```

      This command creates the Oracle instance and, by default, registers the instance with the Oracle WebLogic Server Administration Server.

c. Create the Oracle Web Cache instances, using the opmnctl `createcomponent` command. For example:

```
opmnctl createcomponent -componentType WebCache
    -oracleInstance /scratch/Oracle/Middleware/inst1
    -componentName webcache1
```

3. Register the Oracle instances, along with all of its components, with the Administration Server, using the opmnctl `registerinstance` command. For example:

```
opmnctl registerinstance -adminHost admin_server_host
     -adminPort admin_server_port -adminUsername username
     -adminPassword password
     -oracleInstance ORACLE_INSTANCE_dir -oracleHome ORACLE_HOME_dir
     -instanceName Instance_name -wlserverHome Middleware_Home
```

Task 2 Update Oracle Web Cache

For each Oracle Web Cache instance, take the following steps:

1. Copy the webcache.xml file, which is located in the following directory, from the source environment to a temporary location:

```
(UNIX) ORACLE_INSTANCE/config/WebCache/webcache_name
(Windows) ORACLE_INSTANCE\config\WebCache\webcache_name
```

2. Make the following changes to webcache.xml in the temporary location:
   - If the Web Cache Administration password at the target environment is different from the password at the source environment:
     - Copy the value of the PASSWORDHASH attribute of the <USER TYPE="INVALIDATION"> element from the webcache.xml file for the target environment Web Cache instance and replace the current value of the corresponding PASSWORDHASH attribute in this temporary webcache.xml.
     - Copy the value of the PASSWORDHASH attributes of the <USER TYPE="MONITORING"> element from the webcache.xml file for the target environment Web Cache instance and replace the current value of the corresponding PASSWORDHASH attribute in this temporary webcache.xml.
   - Update the NAME and PORT attributes of each <HOST> and <VIRTUALHOSTMAP> elements with the new host name or IP address and port number of the origin servers at the target environment.
   - For each <CACHE> element in webcache.xml, change the following, substituting the values that correspond to the host where the target environment Oracle Web Cache instance is located:
     - Update the NAME, ORACLE_HOME and HOSTNAME attributes.
     - Search for and replace the Oracle instance path.

       Note: Update this information on one Oracle Web Cache instance at a time. Do not do a global search and replace, because other Oracle Web Cache instances might be configured in a different Oracle instance running at a different path.

- For each <LISTEN> element, update IPADDR (if it is configured other than ANY) and PORT (if Oracle Web Cache uses different ports at the target environment).
- Update the wallet location (if different) for a SSL-enabled listen address. The wallet location is specified within the <WALLET> element for each SSL listen port.
- Update the USERID and GROUPID attributes of the <IDENTITY> element.
- In the <OSWALLET> element, update the wallet location (if different on the target environment) for the original servers. This is the wallet used by Oracle Web Cache to talk to an SSL-enabled origin server).

3. Copy the edited webcache.xml to the following location on the target environment:

```
(UNIX) ORACLE_INSTANCE/config/WebCache/webcache_name
(Windows) ORACLE_INSTANCE\config\WebCache\webcache_name
```

4. If any changes have been made to auditconfig.xml, copy the following file from the source environment to the corresponding target environment.

```
(UNIX) ORACLE_INSTANCE/config/WebCache/webcache_name/auditconfig.xml
(Windows) ORACLE_INSTANCE\config\WebCache\webcache_name\auditconfig.xml
```

5. If any changes have been made to *component*-log.xml, first, edit the file to update the log path, and then copy the file from the source environment to the corresponding target environment.
6. If any changes have been made to the Oracle Web Cache error pages, which are located in the following directory, copy the error pages from the source environment to the target environment location:

```
(UNIX) ORACLE_INSTANCE/config/WebCache/webcache_name/files
(Windows) ORACLE_INSTANCE\config\WebCache\webcache_name\files
```

7. If a non-default wallet was used at the source environment for either an SSL-enabled listen address or an OSwallet, or both, export the wallets from the source environment and import them at the target environment. For information about exporting and importing wallets, see Section 8.4.4.

21.4.6.2 Moving the Web Tier to an Existing Target Environment

In this procedure, you have a working target environment and want to test changes in your applications or configuration before rolling those changes into the target environment.

- For Oracle HTTP Server, see Section 21.4.6.2.1.
- For Oracle Web Cache, perform Task 2 in Section 21.4.6.1.2.

21.4.6.2.1 Moving Oracle HTTP Server to an Existing Target Environment

To move Oracle HTTP Server to an existing target environment, you update the configuration:

1. Copy any custom contents, such as contents that have been changed or added to the htdocs directory, to the target environment Oracle HTTP Server.
2. If any changes have been made to auditconfig.xml, which is located in the following directory, make a backup copy of the file in the target environment.

Then, copy auditconfig.xml from the source environment to the corresponding target environment:

ORACLE_INSTANCE/config/OHS/*ohs_component_name*/auditconfig.xml

3. If any changes have been made to *component*-log.xml, make a backup copy of the file in the target environment. Then, copy the file, which is located in the following directory, from the source environment to the target environment:

ORACLE_INSTANCE/diagnostics/logs/OHS/*ohs_component_name*

21.4.7 Moving Oracle Business Intelligence to a Target Environment

This section describes the steps for moving Oracle Business Intelligence from the source environment to the target environment.

> **See Also:** "Managing the Repository Lifecycle in a Multiuser Development Environment" in *Oracle Fusion Middleware Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition* for detailed information about the life cycle for the Oracle Business Intelligence repository, including source to target considerations for the repository.

The following procedures assume that you have already installed and configured Oracle Business Intelligence components in the source environment and that you want to move them to either a new or an existing target environment:

- Moving Oracle Business Intelligence to a New Target Environment
- Moving Oracle Business Intelligence to an Existing Target Environment When There Are Few Patches to Apply
- Moving Oracle Business Intelligence Components to an Existing Target Environment When There are Many Patches to Apply

If you are applying patches to an existing target environment, the steps you take depend on how many patches you need to apply. If there are few patches, you use the steps in Section 21.4.7.2, which apply the patches to the master host and all cluster hosts in the environment. If there are many patches to apply, consider using the steps in Section 21.4.7.3, which apply the patches to one host and use different means to propagate that to the other hosts, depending on whether or not new hardware is available.

21.4.7.1 Moving Oracle Business Intelligence to a New Target Environment

This section describes the steps for moving Oracle Business Intelligence from the source environment to a new target environment.

This procedure assumes that you have already installed and configured Oracle Business Intelligence components in the source environment and that you have patched the source environment, if necessary, and tested the environment. You want to move them to a new target environment.

To move Oracle Business Intelligence components to a new target environment, perform the following tasks:

- Task 1, "Move the Database, Middleware Home, and Perform the Initial Configuration"
- Task 2, "Patch-Merge the Repository File"

- Task 3, "Configure Security in the New Target Environment"
- Task 4, "Move the Configuration of the Oracle BI Enterprise Edition Components"
- Task 5, "Copy and Scale Out to New Cluster Hosts in the Target Environment"
- Task 6, "Enable New Agents and Oracle BI Publisher Scheduled Jobs"
- Task 7, "Update Links to External Systems"
- Task 8, "(Optional) Move Oracle Business Intelligence Related Applications"

Task 1 Move the Database, Middleware Home, and Perform the Initial Configuration

To move the database and Middleware home, and perform the initial configuration:

1. Move or create the database, as described in Section 21.3.3.
2. Move Identity Management components, as described in Section 21.4.1.
3. Move the Middleware home and binary files, as described in Section 21.3.4.
4. Move the configuration of the domain and Node Manager, as described in Section 21.3.6.

   Note that when you move the configuration of the domain, the pasteConfig script copies the configuration of the domain, including the Administration Server and Managed Servers.
5. Configure users and groups, as described in Section 21.3.8.

Task 2 Patch-Merge the Repository File

In the source environment, use the Administration Tool and the Oracle BI Server XML API to perform a patch merge of the source repository file (.rpd) with the target file.

See "Performing Patch Merges" in *Oracle Fusion Middleware Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition* for more information.

Task 3 Configure Security in the New Target Environment

Configure security if you use something other than the default Oracle WebLogic Server LDAP. For information, see *Oracle Fusion Middleware Security Guide for Oracle Business Intelligence Enterprise Edition*.

For information about migrating security data (for example, users, groups, and roles), see the appropriate documentation for your authentication provider. The following list provides sources for various components:

- Oracle Internet Directory: See Task 4, "Move Oracle Internet Directory to the New Target Environment" in Section 21.4.1.1.
- Oracle WebLogic Server: See "Migrating Security Data" in *Oracle Fusion Middleware Securing Oracle WebLogic Server*.
- Oracle Platform Security Services: See *Oracle Fusion Middleware Application Security Guide*.

Task 4 Move the Configuration of the Oracle BI Enterprise Edition Components

You move the configuration of the following Oracle BI EE components using the copyConfig, extractMovePlan, and pasteConfig scripts:

- Oracle BI server
- Oracle BI Presentation Services

- Oracle BI Cluster Controller
- Oracle BI Scheduler
- JavaHost
- Oracle Essbase server, if it is installed in your environment

To move the configuration of the components:

1. At the source Middleware home, execute the copyConfig script for the Oracle BI EE components.

   The copyConfig script is located in:

   ```
   (UNIX) ORACLE_COMMON_HOME/bin/copyConfig.sh
   (Windows) ORACLE_COMMON_HOME\bin\copyConfig.cmd
   ```

   See Section 20.3.1.5 for the syntax of the script.

   The following shows examples of the commands for each.

   ```
   Oracle BI Server
   copyConfig.cmd -javahome c:\Utilities\Java\jrockit-jdk1.6.0_24-R28.1.3-4.0.1
                  -archiveLoc  c:\toTarget\biconfig_bis.jar
                  -sourceInstanceHomeLoc c:\Oracle\Middleware\instance\instance1
                  -sourceComponentName coreapplication_obis1

   Oracle BI Presentation Services
   copyConfig.cmd -javahome c:\Utilities\Java\jrockit-jdk1.6.0_24-R28.1.3-4.0.1
                   -archiveLoc   c:\toTarget\biconfig_bips.jar
                   -sourceInstanceHomeLoc c:\Oracle\Middleware\instance\instance1
                   -sourceComponentName  coreapplication_obips1

   Cluster Controller
   copyConfig.cmd -javahome c:\Utilities\Java\jrockit-jdk1.6.0_24-R28.1.3-4.0.1
                   -archiveLoc c:\toTarget\biconfig_biccs.jar
                   -sourceInstanceHomeLoc c:\Oracle\Middleware\instance\instance1
                   -sourceComponentName coreapplication_obiccs1

   Scheduler
   copyConfig.cmd -javahome c:\Utilities\Java\jrockit-jdk1.6.0_24-R28.1.3-4.0.1
                   -archiveLoc c:\toTarget\biconfig_bisch.jar
                   -sourceInstanceHomeLoc c:\Oracle\Middleware\instance\instance1
                   -sourceComponentName coreapplication_obisch1

   JavaHost
   copyConfig.cmd -javahome c:\Utilities\Java\jrockit-jdk1.6.0_24-R28.1.3-4.0.1
                   -archiveLoc c:\toTarget\biconfig_bijh.jar
                   -sourceInstanceHomeLoc c:\Oracle\Middleware\instance\instance1
                   -sourceComponentName coreapplication_obijh1

   Oracle Essbase server
   copyConfig.cmd -javahome c:\Utilities\Java\jrockit-jdk1.6.0_24-R28.1.3-4.0.1
                   -archiveLoc c:\toTarget\biconfig_biess.jar
                   -sourceInstanceHomeLoc c:\Oracle\Middleware\instance\instance1
                   -sourceComponentName essbaseserver1
                   -domainAdminUserName domain_admin_username
                   -domainAdminPassword  c:\toTarget\password.txt
   ```

2. If you are copying the component to a different host, copy the archive file to that system.

3. Extract the move plan from the archive for *each* component, using the extractMovePlan script.

   The extractMovePlan script is located in:

   ```
   (UNIX) ORACLE_COMMON_HOME/bin/extractMovePlan.sh
   (Windows) ORACLE_COMMON_HOME\bin\extractMovePlan.cmd
   ```

   See Section 20.3.1.7 for the syntax of the extractMovePlan script.

   For example, to extract the move plan for the Oracle BI Server:

   ```
   extractMovePlan.cmd -javahome c:\Utilities\Java\jrockit-jdk1.6.0_
   24-R28.1.3-4.0.1
                       -archiveLoc c:\toTarget\biconfig_bis.jar
                       -planDirLoc c:\totarget\plans\bis
   ```

4. Edit the move plan, modifying the properties for the particular component to reflect the values for the target environment. See Table 20–22 for the properties to change.

   Note that for Oracle Essbase, you must specify valid file locations, disk volume customization locations, and the Oracle Essbase administration user name and password. Otherwise, you will receive an error.

5. Copy the edited move plan to the target. (During the pasteConfig operation, you specify the location using the -movePlanLoc option.)

6. At the target, extract the files from each archive using the pasteConfig script.

   The pasteConfig script is located in:

   ```
   (UNIX) ORACLE_COMMON_HOME/bin/pasteConfig.sh
   (Windows) ORACLE_COMMON_HOME\bin\pasteConfig.cmd
   ```

   See Section 20.3.1.10 for the syntax of the script.

   For example, to apply the archive of the BI Server to the Oracle instance instance1:

   ```
   pasteConfig.cmd -javahome c:\Utilities\Java\jrockit-jdk1.6.0_24-R28.1.3-4.0.1
               -archiveLoc  c:\fromSource\biconfig_bis.jar
               -targetOracleHomeLoc c:\NewOra\Oracle_BI1
               -targetComponentName coreapplication_obis1
               -targetInstanceHomeLoc c:\NewOra\instance\instance1
               -targetInstanceName  instance1
               -movePlanLoc  c:\fromSource\plans\bis\moveplan.xml
               -domainHostName example.com
               -domainPortNum 7001
               -domainAdminUserName domain_admin_username
               -domainAdminPassword c:\fromSource\password.txt
   ```

   Note that the Oracle instance name and the component name in the target environment must be the same as the name in the source environment.

Task 5 Copy and Scale Out to New Cluster Hosts in the Target Environment

1. Copy the archive file (created in Task 1, "Move the Database, Middleware Home, and Perform the Initial Configuration") to the new cluster host.

2. Copy the following files to the new cluster host:

   - UNIX:

     ```
     ORACLE_COMMON_HOME/bin/pasteBinary.sh
     ORACLE_HOME/jlib/cloningclient.jar
     ```

- Windows:

```
ORACLE_COMMON_HOME\bin\pasteBinary.cmd
ORACLE_HOME\jlib\cloningclient.jar
```

3. Use the pasteBinary script to copy the Middleware home to the new cluster host, as described in Section 20.2.1.

   Note: You must use exactly the same Middleware home name on the new cluster host that is used on the master host.

4. Use Fusion Middleware Control to scale out to the new cluster host.

   For information, see "Using Fusion Middleware Control to Scale System Components" in *Oracle Fusion Middleware System Administrator's Guide for Oracle Business Intelligence Enterprise Edition*.

5. Repeat the previous steps for each new cluster host.

Task 6 Enable New Agents and Oracle BI Publisher Scheduled Jobs

If new agents were created in the source environment, click each agent in the Oracle BI Presentation Services Catalog Manager (in the target environment) to enable it.

Because Oracle BI Publisher reports are stored in the Oracle BI Presentation Catalog, existing reports and new reports that are created in the source environment should be available.

In a target environment, Oracle WebLogic Server administrators should create JNDI connections (to be used by Oracle BI Publisher reports), using the same names as in the source environment. The connections should point to the target databases instead of the source databases. In this way, all reports automatically point to the target environment databases, instead of source environment databases, without any modification.

Task 7 Update Links to External Systems

To ensure that you move the static content that relates to external systems to the target environment, edit the Action Framework configuration file and ensure that the endpoints refer to relevant resources in the target system.

For information on configuring for different types of actions, see "Configuring the Action Framework" in *Oracle Fusion Middleware Integrator's Guide for Oracle Business Intelligence Enterprise Edition*.

Task 8 (Optional) Move Oracle Business Intelligence Related Applications

Move Oracle Business Intelligence related applications (such as Calculation Manager, Financial Reporting, and Oracle BI for Microsoft Office) to the new target environment. For information, see Section 21.4.5.

21.4.7.2 Moving Oracle Business Intelligence to an Existing Target Environment When There Are Few Patches to Apply

This section describes the steps for moving Oracle Business Intelligence from the source environment to an existing target environment when there are few patches to apply. (See Section 21.4.7.3 if you have many patches to apply).

The following steps assume that you have already installed and configured Oracle Business Intelligence components in the source environment and that you want to move them to an existing target environment.

To move Oracle Business Intelligence components to an existing target environment when there are few patches to apply, perform the following tasks:

- Task 1, "Patch the Source and Existing Target Environments"
- Task 2, "Deploy the Source Repository File to the Existing Target Environment"
- Task 3, "Deploy the Source Oracle BI Presentation Catalog to the Existing Target Environment"
- Task 4, "(Optional) Refresh Global Unique Identifiers (GUIDs)"
- Task 5, "Enable New Agents and Oracle BI Publisher Scheduled Jobs"
- Task 6, "Update Links to External Systems"
- Task 7, "(Optional) Move Oracle Business Intelligence Related Applications"

Task 1 Patch the Source and Existing Target Environments

A patch applies a collection of bug fixes to an existing environment and includes new binary files and metadata updates.

1. Patch the source environment as required, and test.
2. Patch the existing target environment to the same level as the source environment on the master host and on all cluster hosts.

   Note: Patching also includes non-Oracle Business Intelligence patches and one-off patches.

   For information, see "Patching Oracle Business Intelligence Systems" in *Oracle Fusion Middleware System Administrator's Guide for Oracle Business Intelligence Enterprise Edition*.

Task 2 Deploy the Source Repository File to the Existing Target Environment

1. In the source environment, use the Administration Tool and the Oracle BI Server XML API to perform a patch merge of the source repository file (.rpd) with the target file.

   You must complete this task only if you are moving to an existing target environment and have made changes to the RPD file in the source environment.

   See "Performing Patch Merges" in the *Oracle Fusion Middleware Metadata Repository Builder's Guide for Oracle Business Intelligence Enterprise Edition* for more information.
2. Use Fusion Middleware Control in the target environment to upload the RPD file.

   For information, see "Using Fusion Middleware Control to Upload a Repository and Set the Oracle BI Presentation Catalog Location" in the *Oracle Fusion Middleware System Administrator's Guide for Oracle Business Intelligence Enterprise Edition*.
3. If necessary, use the Administration Tool or the Oracle BI Server XML API to update connection pool and database settings in the repository. The RPD file might contain data source connection information from the source environment that must be changed to the target environment connection settings.

   See "Moving from Test to Production Environments" in the *Oracle Fusion Middleware XML Schema Reference for Oracle Business Intelligence Enterprise Edition* for information about performing this step using the Oracle BI Server XML API.

4. (Optional) Make the target repository file read-only by selecting **Disallow Online RPD Updates** in the Performance tab of the Capacity Management page in Fusion Middleware Control.

Task 3 Deploy the Source Oracle BI Presentation Catalog to the Existing Target Environment

1. Drag and drop new or updated folders from the source catalog into the target catalog as follows:

   a. Open two Catalog Manager windows: one with the source catalog and another with the target catalog.

   b. Selectively copy and paste the folders that you want from the source catalog into the target catalog.

      Note: If you copy and paste folders where the same content has been changed in the source or target environments, then the source content overwrites the target content.

      For information, see the *Oracle Fusion Middleware System Administrator's Guide for Oracle Business Intelligence Enterprise Edition*.

2. Use Fusion Middleware Control in the existing target environment to specify the location of the new catalog.

   For information, see "Using Fusion Middleware Control to Upload a Repository and Set the Oracle BI Presentation Catalog Location" in the *Oracle Fusion Middleware System Administrator's Guide for Oracle Business Intelligence Enterprise Edition*.

Task 4 (Optional) Refresh Global Unique Identifiers (GUIDs)

You do not normally refresh GUIDs in the LDAP directory (identity store users) between source and target environments, because the LDAP directories that contain the GUIDs should be fan-out replicas in both the source and the target environments. Possible scenarios for refreshing are described in the following list:

- Oracle Business Intelligence source servers and target servers are both configured against the corporate LDAP directory.

  There is no need to refresh LDAP GUIDs.

- Oracle Business Intelligence source servers are configured against a source LDAP and the target servers against the corporate LDAP, but the source LDAP is a fan-out replica of the corporate LDAP directory.

  There is no need to refresh LDAP GUIDs.

- Oracle Business Intelligence source servers are configured against a source LDAP and the target servers against the corporate LDAP, but the source LDAP is not a fan-out copy of the corporate LDAP directory.

  A refresh of the LDAP GUIDs is needed. Follow the procedures in this section.

After changing the directory server that is used as the data source for the authentication provider, it is best practice to update the user GUIDs. If the same user name exists in both directory servers (original and new), then the original user GUID might conflict with the user GUID that is contained in new directory server. A refresh forces the system to reference the user GUID that is contained in the new directory server. Authentication errors might result if the GUIDs are not refreshed and the system detects a mismatch for the user GUID.

The GUIDs that are stored in the Oracle BI Presentation Catalog or in the RPD file can be resynchronized and refreshed as described in the following procedure. Before you begin this procedure, ensure that you are familiar with the information in "Manually Updating Oracle Business Intelligence Configuration Settings Not Normally Managed by Fusion Middleware Control" in the *Oracle Fusion Middleware System Administrator's Guide for Oracle Business Intelligence Enterprise Edition*.

This procedure requires that you manually edit the configuration files to instruct Oracle BI Server and Oracle BI Presentation Services to refresh the GUIDs on restart. Once completed, you edit these files to remove the modification. For information about where to locate Oracle Business Intelligence configuration files, see the section that describes where configuration files are located, in *Oracle Fusion Middleware System Administrator's Guide for Oracle Business Intelligence Enterprise Edition*.

To refresh the user GUIDs:

1. Open the NQSConfig.INI file for editing. For information, see "Where are Configuration Files Located?" in *Oracle Fusion Middleware System Administrator's Guide for Oracle Business Intelligence Enterprise Edition*.
2. Locate the setting `FMW_UPDATE_ROLE_AND_USER_REF_GUIDS = NO` and change its value to `YES`.
3. Modify the instanceconfig.xml file to instruct Presentation Services to refresh GUIDs on restart. Edit the file and find the following section:

   ```
   <Catalog>
      <UpgradeAndExit>false</UpgradeAndExit>
   </Catalog>
   ```

   Comment out the <UpgradeAndExit> line and add an extra line in this section as in the following example:

   ```
   <Catalog>
      <!--UpgradeAndExit>false</UpgradeAndExit-->
      <UpdateAccountGUIDs>UpdateAndExit</UpdateAccountGUIDs>
   </Catalog>
   ```

4. Stop and restart the managed processes using the `opmnctl` command with the parameters `stopall` and `startall`. You can use the parameter `status` to verify process status throughout.

   The following components are involved: Presentation Services, Oracle BI Server, Oracle BI Scheduler, Oracle BI Cluster Controller, and Oracle BI JavaHost.

   For information about using `opmnctl` commands, see "Using the OPMN command line to Start and Stop Oracle Business Intelligence System Components" in *Oracle Fusion Middleware System Administrator's Guide for Oracle Business Intelligence Enterprise Edition*.
5. Edit the NQSConfig.INI file to reset the `FMW_UPDATE_ROLE_AND_USER_REF_GUIDS = YES` to `NO` and restart the Oracle BI Servers.
6. Comment out the line added in Step 3 and remove the commenting from the original line so that it reads as shown in the following example:

   ```
   <Catalog>

     <UpgradeAndExit>false</UpgradeAndExit>

     <!--UpdateAccountGUIDs>UpdateAndExit</UpdateAccountGUIDs-->
   </Catalog>
   ```

7. Restart Presentation Services for the instanceconfig.xml file that was updated.
8. Ensure that Oracle WebLogic Server and the system components are also running. If they are not running, then restart them.

   For information, see "Starting and Stopping the Oracle Business Intelligence Components" in *Oracle Fusion Middleware System Administrator's Guide for Oracle Business Intelligence Enterprise Edition*.

Task 5 Enable New Agents and Oracle BI Publisher Scheduled Jobs

If new agents were created in the source environment, then click each agent in the Oracle BI Presentation Services Catalog Manager (in the target environment) to enable it.

Because Oracle BI Publisher reports are stored in the Oracle BI Presentation Catalog, existing reports and new reports created in the source environment should be available.

In the target environment, Oracle WebLogic Server administrators should create JNDI connections (to be used by Oracle BI Publisher reports), using the same names as in the source environment. The connections should point to the target databases instead of the source databases. In this way, all reports automatically point to the target environment databases, instead of source environment databases, without any modification.

Task 6 Update Links to External Systems

Ensure that you move the static content that relates to external systems to the target environment, as described in the following steps:

1. Copy the Action Framework configuration file from its location on the source system to the same location on the target system. In addition, the ActionFramework configuration file might contain policy elements that refer to files in the same directory as the configuration file. Copy these files to the same location on the target system.
2. Edit the Action Framework configuration file and ensure that the endpoints refer to relevant resources in the target system.

For information on configuring for different types of actions, see "Configuring the Action Framework" in *Oracle Fusion Middleware Integrator's Guide for Oracle Business Intelligence Enterprise Edition*.

Task 7 (Optional) Move Oracle Business Intelligence Related Applications

Move Oracle Business Intelligence related applications (such as Calculation Manager, Financial Reporting, and Oracle BI for Microsoft Office) to the existing target environment. For information, see Section 21.4.5.

21.4.7.3 Moving Oracle Business Intelligence Components to an Existing Target Environment When There are Many Patches to Apply

This section describes the steps for moving Oracle Business Intelligence from the source environment to an existing target environment when there are many patches to apply.

The following procedures assume that you have already installed and configured Oracle Business Intelligence components in the source environment and that you want to move them to an existing target environment.

Use one of the following strategies to move Oracle Business Intelligence components to an existing target environment when there are many patches to apply:

- Moving Oracle BI EE to an Existing Target Environment When New Hardware Is Available
- Moving Oracle BI EE to an Existing Target Environment When New Hardware Is Not Available

21.4.7.3.1 Moving Oracle BI EE to an Existing Target Environment When New Hardware Is Available

Perform the following tasks to move Oracle Business Intelligence components to an existing target environment when there are many patches to apply and new hardware is available:

- Task 1, "Follow the Steps for Moving to a New Target Environment"
- Task 2, "Switch Users from the Existing Target Environment to the New One"
- Task 3, "Remove the Existing Target Environment and Prepare It for the Next Patch"

Task 1 Follow the Steps for Moving to a New Target Environment

Complete the steps in Section 21.4.7.1 for moving to a new target environment.

These steps include merging the new source RPD file and catalog with those in the existing target environment. Ideally you merge once and resolve the issues while users continue using the existing environment. When the files are correct, you lock the target environment and repeat the merge to access the latest changes.

Task 2 Switch Users from the Existing Target Environment to the New One

Use a load balancer such as Oracle Web Cache to redirect users from a standard URL to the new target environment.

Task 3 Remove the Existing Target Environment and Prepare It for the Next Patch

Shut down the existing environment and deinstall all software. When needed, you can apply the next patchset to this host, and the sequence can start all over again.

21.4.7.3.2 Moving Oracle BI EE to an Existing Target Environment When New Hardware Is Not Available

Perform the following tasks to move Oracle Business Intelligence components to an existing target environment when there are many patches to apply and new hardware is not available:

- Task 1, "Scale the Target Environment Back to One Host"
- Task 2, "Patch the Host in the Target Environment"
- Task 3, "Remove the Existing Software on the Cluster Hosts"
- Task 4, "Move the Target Environment and Then Copy to the Cluster Hosts"

Task 1 Scale the Target Environment Back to One Host

Use the Capacity Management tab of the Scalability page in Fusion Middleware Control in the target environment to scale back system components to apply only to the first host in the list. This scaling makes it much easier to patch the existing target environment.

For more information, see the Fusion Middleware Control Help system.

Task 2 Patch the Host in the Target Environment

Patch the host in the target environment. Doing so imposes less downtime on users than having to patch multiple cluster hosts.

For information, see the *Oracle Fusion Middleware Patching Guide.*

Task 3 Remove the Existing Software on the Cluster Hosts

Deinstall all the Oracle Business Intelligence software on the cluster hosts. For information, see the *Oracle Fusion Middleware Installation Guide for Oracle Business Intelligence*.

Task 4 Move the Target Environment and Then Copy to the Cluster Hosts

Complete the tasks beginning with Task 5, "Copy and Scale Out to New Cluster Hosts in the Target Environment" in Section 21.4.7.1.

21.4.8 Moving Oracle Real-Time Decisions to a Target Environment

The following topics describe how to move Oracle Real-Time Decisions (Oracle RTD) from the source environment to a new target environment:

- Moving Oracle Real-Time Decisions to a New Target Environment
- Moving Oracle Real-Time Decisions to an Existing Target Environment

21.4.8.1 Moving Oracle Real-Time Decisions to a New Target Environment

To move the environment to the target environment, perform the following tasks:

- Task 1, "Move the Database, Middleware Home, and Perform the Initial Configuration"
- Task 2, "Install Oracle RTD Clients (If Used) on the Target Environment"
- Task 3, "Move Oracle RTD Inline Services"
- Task 4, "Edit Additional Oracle RTD Components for the Target"

Task 1 Move the Database, Middleware Home, and Perform the Initial Configuration

To move the database, Middleware home, and Oracle RTD software and perform the initial configuration:

1. Move or create the database and the required schemas, as described in Section 21.3.1.
2. Move the a copy of the Middleware home and binary files, as described in Section 21.4.1.

   If your environment includes Oracle BI EE and you have already moved Oracle BI EE to a new target environment, as described in Section 21.4.7.1, you do not need to take this step because the binary files for Oracle RTD were moved as well those for Oracle BI EE.
3. Move the configuration, as described in Section 21.3.6.

   Note that when you move the configuration, the pasteConfig script copies the configuration of the domain, including the Administration Server and Managed Servers.

Task 2 Install Oracle RTD Clients (If Used) on the Target Environment

If used for the integration of Oracle RTD to a customer's front-end applications, Oracle RTD clients must be installed in the target environment, according to the setup steps outlined in the *Oracle Fusion Middleware Administrator's Guide for Oracle Real-Time Decisions*.

Configuration of client parameters should reflect values specific to the target architecture.

Task 3 Move Oracle RTD Inline Services

Move Oracle RTD Inline Services that exist on the source environment to the target environment:

1. Moving Inline Services to the target environment can be performed in two ways:
   - Command-line deployment: For more information, see the chapter "Command Line Deployment of Inline Services" in *Oracle Fusion Middleware Administrator's Guide for Oracle Real-Time Decisions*.
   - Decision Studio deployment: For information about Oracle RTD deployment in Decision Studio, see the chapter "Deploying, Testing, and Debugging Inline Services" in *Oracle Fusion Middleware Platform Developer's Guide for Oracle Real-Time Decisions*.

   Note: Prior to moving an Inline Service, if changes have been made to the Inline Service used by the Oracle RTD server, for example, through the Decision Center, you should first download the latest Inline Service version to Decision Studio before redeploying to the production environment.

2. When moving Inline Services from one environment to another, note the following areas that may also need editing within the Inline Service:
   - Calls to third party APIs and third party JAR files.

     Any addition of new jar files must be put into the corresponding location in the new environment.
   - Calls to third party web services.

     Location paths, web service parameters, and so on, if different in the new environment, need to be modified.
   - References to custom tables, such as location, user names, and passwords, within the Inline Service, if different in the production environment, must be edited before redeploying.
   - References to the data sources, if different in the target environment, should be edited before deploying. This includes modifying the data sources for dynamic choices, if used.
   - References to any debugging code (logInfo statements, logTrace statements, and so on) that may not be desired in the new environment should be commented out or removed in the Inline Service before redeploying.
3. For Inline Services that include external objects, such as dynamic choices or external rules, the following considerations apply:
   - For dynamic choices:

If dynamic choices are part of the Inline Service configuration, you must re-create both the data and the tables that store the dynamic choices, if the source and target environment do not share the same source.

Data source elements in the Inline Service also need to be modified as appropriate.

- For external rules:

  If external rules are part of the Inline Service configuration, you must re-create both the data and the tables that store the rule data if the source and target environment do not share the same source.

  Data source elements in the Inline Service need to be modified as appropriate.

  In addition, the external rule editor used in the target environment should be configured to point to the target database.

Task 4 Edit Additional Oracle RTD Components for the Target

Additional tasks that you may need to perform with Oracle RTD include the following:

1. Creating and configuring the model snapshot tables:

   a. You can create the Oracle RTD model snapshot tables in the target environment in two ways: using RCU or the tool sdexec/SDDBTool, which is provided with the installation.

      RCU creates the necessary snapshot tables in the same schema as the Oracle RTD platform tables, while sdexec/SDDBTool allows you to create the tables in another location.

   b. After the model snapshot tables are created, use the Enterprise Manager console to configure the settings needed to populate the tables. For details, see the chapter "Setting Up and Using Model Snapshots" in the *Oracle Fusion Middleware Administrator's Guide for Oracle Real-Time Decisions*.

2. Modifying the loadgen files.

   If you have created loadgen files that will also be used in the target environment, you must modify the following parameters according to the new environment (each must be modified within the specific loadgen configuration file):

   - ClientHttpEndpoints.properties files
   - Inline Service name (if changed)
   - Path references to data files if used as inputs to a loadgen script
   - Path to the loadgen log file

3. Modifying batch processing files.

   If using the RTD Batch module, you should also pay attention to any data sources referenced in the batch files that are environment specific and modify the files accordingly.

21.4.8.2 Moving Oracle Real-Time Decisions to an Existing Target Environment

After the target environment has been created, typical Oracle RTD incremental changes include the following:

- Task 1, "Oracle RTD Patch Updates"
- Task 2, "Update Inline Services"

- Task 3, "Update Data Sources"

Task 1 Oracle RTD Patch Updates

Because each specific patch addresses unique functional enhancements and known bugs, you should always refer to the release notes that come with each patch for specific instructions on how to apply it.

Task 2 Update Inline Services

For incremental Inline Service changes, moving the Inline Service to the target follows the same steps as outlined for a moving the full product source environment to the target environment.

Task 3 Update Data Sources

If additional data sources are to be added incrementally to an Inline Service, refer to the "Configuring Data Access" chapter in the *Oracle Fusion Middleware Administrator's Guide for Oracle Real-Time Decisions*.

21.4.9 Moving Oracle Portal, Oracle Forms Services, Oracle Reports, and Oracle BI Discoverer to a Target Environment

In these procedures, you have installed Oracle Portal, Oracle Forms Services, Oracle Reports, and Oracle Business Intelligence Discoverer in the source environment and you want to move them to the target environment.

The following topics describe how to move these components from the source environment to the target environment:

- Moving Oracle Portal, Oracle Forms Services, Oracle Reports, and Oracle Business Intelligence Discoverer to a New Target Environment
- Moving Oracle Portal, Oracle Forms Services, Oracle Reports, and Oracle Business Intelligence Discoverer to an Existing Target Environment

In both cases, you have performed the following in the source environment:

- Installed a database to be used for these components.
- Created the needed schemas in the source environment using RCU. See the *Oracle Fusion Middleware Repository Creation Utility User's Guide*.
- For Oracle BI Discoverer, installed an additional database to be used for the End User Layer (EUL), Discoverer catalog, and OLAP catalog.
- Installed Oracle WebLogic Server and created a Middleware home.
- Installed and configured Identity Management, including Oracle Internet Directory and Oracle Single Sign-On, and a database for Identity Management data.
- Installed and configured Oracle Portal, Oracle Forms Services, Oracle Reports, and Oracle BI Discoverer.
- For Oracle Portal:
  - Created users and groups and assigned page access permissions to the groups.
  - Created new page groups, new templates, and new pages, and added contents, such as items and portlets, to the pages.
  - Customized pages, layouts, items, and portlets.

  - Registered producers (database, Web, and WSRP) and customized the portlet from the producers.
  - Registered external applications.
- Set up Forms applications.
- Configured Oracle Reports instances and created connections to the database.
- For Oracle BI Discoverer:
  - For Discoverer Plus, created a new workbook with parameters, calculations, conditions, and totals. Saved the workbook.
  - For Discoverer Viewer, opened the workbook created in Discoverer Plus and performed some formatting, sorting, exporting, and drilling.
  - For Discoverer Plus OLAP, created a new workbook in Discoverer Plus OLAP with custom members, custom expressions, and saved selections. Saved the workbook.
  - For Viewer OLAP, opened the workbook created in Discoverer Plus OLAP and performed some operations such as exporting, linking and unlinking layouts.

21.4.9.1 Moving Oracle Portal, Oracle Forms Services, Oracle Reports, and Oracle Business Intelligence Discoverer to a New Target Environment

In this procedure, you have installed Oracle Portal, Oracle Forms Services, Oracle Reports, and Oracle Business Intelligence Discoverer in the source environment and you want to move the components to the target environment that does not exist.

Although this section describes how to move all of the components to the target environment, you can choose to move only some of them.

To move this environment to a new target environment, perform the following tasks:

- Task 1, "Move the Database, Middleware Home and Perform the Initial Configuration"
- Task 2, "Move Oracle Portal to the New Target Environment"
- Task 3, "Move Oracle Forms Services to the New Target Environment"
- Task 4, "Move Oracle Reports to the New Target Environment"
- Task 5, "Move Oracle Business Intelligence Discoverer to the New Target Environment"

Task 1 Move the Database, Middleware Home and Perform the Initial Configuration

To move the database and Middleware home, and perform the initial configuration:

1. Move or create the database and the schemas, as described in Section 21.3.3.
2. Move the Middleware home and binary files, as described in Section 21.3.4.
3. Configure the components, as described in the *Oracle Fusion Middleware Installation Guide for Oracle Portal, Forms, Reports and Discoverer*. For Oracle Portal, this includes installing Oracle Internet Directory and Oracle Single Sign-On Release 10.1.3.4.

   For Oracle Portal, specify the credentials to connect to Oracle Internet Directory at the Configure Components screen.

Task 2 Move Oracle Portal to the New Target Environment

To move the Oracle Portal configuration to a new target environment:

1. Create a transport set on the source instance that contains the list of page groups to be moved. For information about creating a transport set, see "Creating Transport Sets" in the *Oracle Fusion Middleware Administrator's Guide for Oracle Portal.*
2. Export the data from the source environment, as described in "Exporting Data" in the *Oracle Fusion Middleware Administrator's Guide for Oracle Portal.*
3. On the target environment, create a database link to the source environment, as described in "Creating a Database Link" in the *Oracle Fusion Middleware Administrator's Guide for Oracle Portal.*
4. Before moving data from a source portal, you must register the portal. Once registered, the source portal can be selected and used to specify the data source in the Transport Sets. See "Register a Source Portal" in the *Oracle Fusion Middleware Administrator's Guide for Oracle Portal.*
5. Before importing your objects, you must move the contents of the transport set to the transport set tables on the target system. You do this by acquiring the transport set from the source environment, using the registered database link described in Step 1. For information about acquiring the transport set, see "Moving Data to the Target System" in the *Oracle Fusion Middleware Administrator's Guide for Oracle Portal.*
6. Import the data, as described in "Import in Oracle Portal" in the *Oracle Fusion Middleware Administrator's Guide for Oracle Portal.*
7. Move users and groups from the LDAP directory in the source environment to the LDAP directory in the target environment, as described in "Migrating Users and Groups" in the *Oracle Fusion Middleware Administrator's Guide for Oracle Portal.*
8. Import the external applications list using the SSOMig utility:
   a. Run ssomig in export mode on the source environment. The command creates a dump file. For example:

      ```
      ssomig -export -s orasso -p orasso_schema_password
       -c tns_alias_for_sso_schema
       -log_d directory_where_dump_needs_to_be_created
       -log_f ssomig.log -d ssomig.dmp
      ```

   b. Run ssomig in import mode on the target environment, specifying the dump file created in the previous step. For example:

      ```
      ssomig -import -overwrite -s orasso -p orasso_schema_password
       -c tns_alias_for_sso_schema -d ssomig.dmp
       -log_d directory_where_dump_is_located -discoforce
      ```

9. For the following files, copy any customizations that you want to maintain from the source environment file to the target environment file:

   ```
   DOMAIN_HOME/config/fmwconfig/servers/WLS_
   PORTAL/applications/portal/configuration/portal_plsql.conf
   DOMAIN_HOME/config/fmwconfig/servers/WLS_
   PORTAL/applications/portal/configuration/portal_dads.conf
   DOMAIN_HOME/config/fmwconfig/servers/WLS_
   PORTAL/applications/portal/configuration/appConfig.xml
   ```

10. If you modified any configuration files, restart the Managed Server WLS_ PORTAL.

Note that when Oracle Portal is moved from source to target using export and import, portlet customizations are included in transport set. You do not need to take any additional steps.

Task 3 Move Oracle Forms Services to the New Target Environment

To move Oracle Forms Services to a new target environment:

1. Stop the Oracle instances processes and the Oracle Forms Services Managed Servers in the target environment, using the following commands:

   ```
   ORACLE_INSTANCE/bin/opmnctl stopall
   DOMAIN_NAME/bin/stopManagedWebLogic.sh
               managed_server_name admin_url username password
   ```

2. Copy the Oracle Forms Services application files (fmx, mmx, obx and plx) from the source environment to the target environment. The location of the files may be specified in the Forms environment configuration file, default.env.

   Note that if the files are in a shared network location, you do not need to copy them to the target environment. Make sure the network path exists and is accessible in the target environment.

3. Move the application-related data from the source environment to a database in the target environment using database migration tools.

4. Create the relevant target database entries in the SQL*Net configuration file, tnsnames.ora.

5. Forms applications have single sign-on user names and passwords mapped to the database connect strings. This information is stored in Oracle Internet Directory. Move the Forms RAD data from Oracle Internet Directory in the source environment to Oracle Internet Directory in the target environment. See Step 3 in Task 1, "Move Oracle Internet Directory to an Existing Target Environment" in Section 21.4.1.

6. Copy any customizations in the following files that you want to maintain from the source environment file to the target environment file:

Type of File	Location
Forms application configuration	*DOMAIN_HOME*/config/fmwconfig/servers/WLS_FORMS/applications/formsapp_*version*/config/formsweb.cfg
Forms server configuration	*DOMAIN_HOME*/config/fmwconfig/servers/WLS_FORMS/applications/formsapp_*version*/config/default.env
Forms HTML template	*ORACLE_INSTANCE*/config/FormsComponent/forms/server/base.htm *ORACLE_INSTANCE*/config/FormsComponent/forms/server/basejpi.htm
WebUtil configuration	*ORACLE_INSTANCE*/config/FormsComponent/forms/server/webutil.cfg
WebUtil HTML template	*ORACLE_INSTANCE*/config/FormsComponent/forms/server/webutiljpi.htm *ORACLE_INSTANCE*/config/FormsComponent/forms/server/webutilbase.htm
Forms OHS directives configuration	*ORACLE_INSTANCE*/config/OHS/*OHS_name*/moduleconf/forms.conf

7. If you modified the Oracle HTTP Server forms.conf file, restart Oracle HTTP Server:

```
ORACLE_INSTANCE/bin/opmnctl restartproc ias-component=ohs_name
```

8. Copy the following files from the source environment to the target environment:

Type of File	Location
Forms application configuration client-side downloadable pluggable contents	These files are user customizations such as images and are in a location accessible to a Web browser.
Forms trace configuration	*ORACLE_INSTANCE*/config/FormsComponent/forms/server/ftrace.cfg
Any customized Forms Java EE applications .ear file	*ORACLE_HOME*/forms/j2ee
JVM Controllers configuration	*ORACLE_INSTANCE*/config/FRComponent/frcommon/tools/jvm/jvmcontrollers.cfg
FMA configuration	*ORACLE_INSTANCE*/config/FormsComponent/forms/search_replace.properties *ORACLE_INSTANCE*/config/FormsComponent/forms/converter.properties
Forms utilities-specific configuration wrapper shell scripts	UNIX: *ORACLE_INSTANCE*/bin/frmbld.sh *ORACLE_INSTANCE*/bin/frmcmp.sh *ORACLE_INSTANCE*/bin/frmplsqlconv.sh *ORACLE_INSTANCE*/bin/frmxmlsg.sh *ORACLE_INSTANCE*/bin/frmcmp_batch.sh *ORACLE_INSTANCE*/bin/frmf2xml.sh *ORACLE_INSTANCE*/bin/frmxml2f.sh *ORACLE_INSTANCE*/bin/frmxmlv.sh Windows: *ORACLE_INSTANCE*\bin\frmplsqlconv.bat *ORACLE_INSTANCE*\bin\frmxmlsg.bat *ORACLE_INSTANCE*\bin\frmf2xml.bat *ORACLE_INSTANCE*\bin\frmxml2f.bat *ORACLE_INSTANCE*\bin\frmxmlv.bat

For the Forms utilities-specific configuration wrapper shell scripts, replace any occurrences of the Oracle home and Oracle instance with the details for the target environment.

9. Start the components in the instance and start the Managed Server, using the following commands:

```
ORACLE_INSTANCE/bin/opmnctl startall
DOMAIN_NAME/bin/startManagedWebLogic.sh
    managed_server_name admin_url
```

10. If you had overridden the default context root or Forms servlet alias in the source environment, copy the customized Forms EE application ear file to the target environment and redeploy it. Refer to "Using Oracle Forms Services with the HTTP Listener and Oracle WebLogic Server" of the *Oracle Fusion Middleware Forms Services Deployment Guide* for details on custom deployment of the Forms Java EE application.

Task 4 Move Oracle Reports to the New Target Environment

To move Oracle Reports to the target environment:

1. For the following Oracle Reports Server configuration files, merge changes made from the source environment to the target environment files. Note that you cannot just copy the files from the source environment to the target environment, because they may have environment-specific information such as Oracle home and Oracle instance names or locations and port numbers.

Type of File	Location
Reports standalone server configuration	*ORACLE_INSTANCE*/config/ReportsServerComponent/*server_name*/rwserver.conf *ORACLE_INSTANCE*/config/ReportsServerComponent/*server_name*/jdbcpds.conf *ORACLE_INSTANCE*/config/ReportsServerComponent/*server_name*/xmlpds.conf *ORACLE_INSTANCE*/config/ReportsServerComponent/*server_name*/textpds.conf *ORACLE_INSTANCE*/config/ReportsServerComponent/*server_name*/rwnetwork.conf *ORACLE_INSTANCE*/config/ReportsServerComponent/*server_name*/component-logs.xml *ORACLE_INSTANCE*/config/ReportsServerComponent/*server_name*/logging.xml
Reports in-process server and servlet configuration	*DOMAIN_HOME*/config/fmwconfig/servers/*server_name*/applications/reports_*version*/configuration/cgicmd.dat *DOMAIN_HOME*/config/fmwconfig/servers/*server_name*/applications/reports_*version*/configuration/rwservlet.properties *DOMAIN_HOME*/config/fmwconfig/servers/*server_name*/applications/reports_*version*/configuration/rwserver.conf *DOMAIN_HOME*/config/fmwconfig/servers/*server_name*/applications/reports_*version*/configuration/jdbcpds.conf *DOMAIN_HOME*/config/fmwconfig/servers/*server_name*/applications/reports_*version*/configuration/xmlpds.conf *DOMAIN_HOME*/config/fmwconfig/servers/*server_name*/applications/reports_*version*/configuration/textpds.conf *DOMAIN_HOME*/config/fmwconfig/servers/*server_name*/applications/reports_*version*/configuration/rwnetwork.conf *DOMAIN_HOME*/config/fmwconfig/servers/*server_name*/applications/reports_*version*/configuration/logging.xml *DOMAIN_HOME*/config/fmwconfig/servers/*server_name*/applications/reports_*version*/configuration/logmetadata.xml *DOMAIN_HOME*/config/fmwconfig/servers/*server_name*/applications/reports_*version*/configuration/jazn-data.xml
Reports Tools configuration	*ORACLE_INSTANCE*/config/ReportsToolsComponent/ReportsTools/rwbuilder.conf *ORACLE_INSTANCE*/config/ReportsToolsComponent/ReportsTools/rwnetwork.conf *ORACLE_INSTANCE*/config/ReportsToolsComponent/ReportsTools/jdbcpds.conf *ORACLE_INSTANCE*/config/ReportsToolsComponent/ReportsTools/xmlpds.conf *ORACLE_INSTANCE*/config/ReportsToolsComponent/ReportsTools/textpds.conf *ORACLE_INSTANCE*/config/ReportsToolsComponent/ReportsTools/component-logs.xml *ORACLE_INSTANCE*/config/ReportsToolsComponent/ReportsTools/logging.xml
Reports Bridge configuration	*ORACLE_INSTANCE*/config/ReportsBridgeComponent/*bridge_name/rwbridge.conf* *ORACLE_INSTANCE*/config/ReportsBridgeComponent/*bridge_name*/rwnetwork.conf *ORACLE_INSTANCE*/config/ReportsBridgeComponent/*bridge_name*/component-logs.xml *ORACLE_INSTANCE*/config/ReportsBridgeComponent/*bridge_name*/logging.xml

Type of File	Location
Reports shell scripts	(UNIX) *ORACLE_INSTANCE*/config/reports/bin/rw*.sh (Windows) *ORACLE_INSTANCE*\config\reports\bin\rw*.bat (UNIX) *ORACLE_INSTANCE*/config/reports/bin/reports.sh (Windows) *ORACLE_INSTANCE*\config\reports\bin\reports.bat (UNIX) *ORACLE_INSTANCE*/config/reports/bin/namingservice.sh (Windows) *ORACLE_INSTANCE*\config\reports\bin\namingservice.bat

2. For the following Oracle Fusion Middleware configuration files, which are related to Oracle Reports Server configuration files, merge changes made from the source environment to the target environment files. Note that you cannot just copy the files from the source environment to the target environment, because they may have environment-specific information such as Oracle home and Oracle instance names or locations and port numbers.

Type of File	Location
JPS configuration	*DOMAIN_HOME*/config/fmwconfig/jps-config.xml *DOMAIN_HOME*/config/fmwconfig/jps-config-jse.xml *DOMAIN_HOME*/config/fmwconfig/system-jazn-data.xml
Forms and Reports common files	Font setup, aliasing, subsetting, embedding: *ORACLE_INSTANCE*/config/FRComponent/frcommon/guicommon/tk/admin/uifont.ali Printer configuration (UNIX only): *ORACLE_INSTANCE*/config/FRComponent/frcommon/guicommon/tk/admin/uiprint.txt Toolkit configuration, encoding (UNIX only): *ORACLE_INSTANCE*/config/FRComponent/frcommon/guicommon/tk/admin/Tk2Motif.rgb PPD files (UNIX only): *ORACLE_INSTANCE*/config/FRComponent/frcommon/guicommon//tk/admin/PPD/* AFM files (UNIX only): *ORACLE_INSTANCE*/config/FRComponent/frcommon/guicommon/tk/admin/AFM/*

3. If you created additional Oracle Reports Server component instances in the source environment, create these in the target environment using opmnctl.
4. For resources related to Oracle Reports Server, take the following actions:
   - Copy any fonts used in the source environment from the directory specified by environment variable REPORTS_FONT_DIRECTORY to target environment. By default, they are in *ORACLE_INSTANCE*/reports/fonts.
   - Move the Common UNIX Printing System (CUPS) printing configuration to the target environment, if applicable.

     For more information about using CUPS with Oracle Reports, see "Enhanced Printing on Linux Using CUPS" in the *Oracle Fusion Middleware Publishing Reports to the Web with Oracle Reports Services.*
5. For Reports definition files and data tables, take the following actions:
   - Copy the reports files, such as RDF, JSP, REP, and XML files, used in the source environment to the target environment.
   - Deploy JSP Web reports to the target environment in the following location:

     DOMAIN_HOME/servers/WLS_REPORTS/tmp/_WL_user/reports_*version*/uxabaw/web.war
   - Move Reports-specific data tables that are referred to in the RDF files to the database in the target environment using database migration tools, such as the Oracle Database export and import utilities.

6. For Reports job-related configuration files, take the following actions:
   - Copy Reports Server cache files to the following location in the target environment:

     ORACLE_INSTANCE/reports/cache

   - For Reports scheduled job information, copy the server data (*server_name*.dat) file to the following location in target environment:

     ORACLE_INSTANCE/reports/server

     Note that because the server name is generated automatically when it is created and the .dat file is named with the server name, the name of the .dat file differs between the source environment and the target environment. Depending on whether it is a standalone server or an in-process server, the name takes one of the following forms:

     ReportsServer_*hostname_instanceName*
     rep_wls_reports_*hostname_instanceName*

     Change the name of the file to reflect the host name and Oracle instance name in the target environment.
7. If the job repository or job status repository is configured in the database, you must create the same schemas in the target environment database and move the data:
   a. Use the following script:

      ORACLE_HOME/reports/admin/sql/rw_job_repos.sql

   b. Move any data from the source database for the schemas RW_JOBS, RW_SERVER_JOB_QUEUE, and RW_SERVER_QUEUE to target database using database migration tools, such as the Oracle Database export and import utilities.
8. Move any user and reports server security policy information. See "Securing Oracle Reports" in the *Oracle Fusion Middleware Publishing Reports to the Web with Oracle Reports Services*.
9. If you use Oracle Internet Directory as the identity and policy store, move the Forms RAD data from Oracle Internet Directory in the source environment to Oracle Internet Directory in the target environment. See Step 3 in Task 1, "Move Oracle Internet Directory to an Existing Target Environment" in Section 21.4.1.
10. If you used JAZN-XML-based identity and policy store in the source environment, move them to the LDAP in the target environment. You can use the WLST command `migrateSecurityStore`, as described in "Migrating Policies with the Command migrateSecurityStore" in the *Oracle Fusion Middleware Application Security Guide*.
11. Migrate the credential store, using the script `migrateSecurityStore`, as described in "Migrating Credentials Manually" in the *Oracle Fusion Middleware Application Security Guide*.
12. Move any database proxy users to the target database using database cloning tools.
13. If any Reports plug-ins are registered, copy the corresponding .jar files to the target environment and add the path to the files to the environment variable REPORTS_CLASSPATH.

Task 5 Move Oracle Business Intelligence Discoverer to the New Target Environment

To move Oracle BI Discoverer to the new target environment:

1. If you have modified the default user preferences, copy the following files from the source environment to the target environment:

   ORACLE_INSTANCE/config/PreferenceServer/*disco-comp-name*/.reg_key.dc
   ORACLE_INSTANCE/config/PreferenceServer/*disco-comp-name*/pref.txt
   ORACLE_INSTANCE/config/PreferenceServer/*disco-comp-name*/defaults.txt

2. If you have changed the Oracle BI Discoverer settings, copy following files from the source environment to the target environment:

   DOMAIN_HOME/config/fmwconfig/servers/WLS_DISCO/applications/discoverer_*version*/configuration/configuration.xml
   *DOMAIN_HOME*config/fmwconfig/servers/WLS_DISCO/applications/discoverer_*version*/configuration/configuration-preview.xml

   In the configuration.xml file, change the values of the following elements to reflect the target environment:

   - applicationURL
   - oracleInstance
   - discovererComponentName

3. If you have changed the server configuration files, copy the following file from the source environment to the target environment:

   ORACLE_INSTANCE/config/OPMN/opmn/opmn.xml

4. Copy the following file from the source environment to the target environment:

   ORACLE_INSTANCE/config/OHS/*ohs_name*/moduleconf/module_disco.conf

   In the file, change the values of the following elements to reflect the target environment:

   - WebLogicCluster. Valid only if a cluster exists.
   - WebLogicHost
   - WebLogicPort

5. Copy the following files from the source environment to the target environment:

   DOMAIN_HOME/servers/WLS_DISCO/stage/discoverer/*version*/discoverer/configuration/base-descktop.xss
   DOMAIN_HOME/servers/WLS_DISCO/stage/discoverer/*version*/discoverer/configuration/blstyles.xss
   DOMAIN_HOME/servers/WLS_DISCO/stage/discoverer/*version*/discoverer/configuration/dc-blaf-review.xss
   DOMAIN_HOME/servers/WLS_DISCO/stage/discoverer/*version*/discoverer/configuration/dc-blaf.xsd
   DOMAIN_HOME/servers/WLS_DISCO/stage/discoverer/*version*/discoverer/configuration/dc-blaf.xss
   DOMAIN_HOME/servers/WLS_DISCO/stage/discoverer/*version*/discoverer/configuration/minimal-desktop.xss
   DOMAIN_HOME/servers/WLS_DISCO/stage/discoverer/*version*/discoverer/configuration/minimal-pda.xss
   DOMAIN_HOME/servers/WLS_DISCO/stage/discoverer/*version*/discoverer/configuration/oracle-desktop.xss
   DOMAIN_HOME/servers/WLS_

```
DISCO/stage/discoverer/version/discoverer/configuration/oracle-pda.xss
DOMAIN_HOME/servers/WLS_
DISCO/stage/discoverer/version/discoverer/configuration/pocketPC.xss
DOMAIN_HOME/servers/WLS_
DISCO/stage/discoverer/version/discoverer/configuration/simple-desktop.xss
DOMAIN_HOME/servers/WLS_
DISCO/stage/discoverer/version/discoverer/configuration/swan-desktop.xss
```

6. Copy some or all of the files in the following directory, depending on which files you use:

   ```
   DOMAIN_HOME/servers/WLS_
   DISCO/stage/discoverer/version/discoverer/discoverer.war/custom_logos/
   ```

   The files that are used are listed in the configuration.xml file.

7. To use the same database service entries, copy the following file from the source environment to the target environment:

   ```
   ORACLE_HOME/network/admin/tnsnames.ora
   ```

8. Move the DISCOVERER schema from the source environment to the target environment. You can use the Oracle Database export and import utilities to move the schema.

   Note that if you choose to use the same database for source and target, you do not need to move the data.

9. Move the EUL data from the source environment to the target environment:

   a. Create the EUL user and an empty EUL on the target database. See "How to Create an End User Layer in a New Database User" in the *Oracle Fusion Middleware Administrator's Guide for Oracle Business Intelligence Discoverer.*

   b. Move the EUL schema from the source database by using the Discoverer Administrator to export the schema from the source database and import it into the database in the target environment. For more information, see "About Using the Discoverer Export Wizard and Import Wizard" in the *Oracle Fusion Middleware Administrator's Guide for Oracle Business Intelligence Discoverer.*

   c. Run the eul5_id.sql script to give the new EUL a unique reference number. Then, grant the entire Discoverer end user community access to the EUL. The script is located in:

      ```
      ORACLE_ HOME/discoverer/util/eul5_id.sql
      ```

      For more information, see "Creating and Maintaining End User Layers" in the *Oracle Fusion Middleware Administrator's Guide for Oracle Business Intelligence Discoverer*.

10. Move the catalog data from the source environment to the target environment:

    a. Install the catalog in the target OLAP database, using the following command:

       ```
       java -classpath d4o.jar oracle.dss.d4o.administration.D4OCommand install
         -h hostname -po port -sid sid -su "sys as sysdba"
         -sp password -p d4osys-password -t users
       ```

    b. Authorize users in the target OLAP database, using the following command:

       ```
       java -classpath d4o.jar oracle.dss.d4o.administration.D4OCommand
        authorize -h hostname -po port -sid sid -p d4osys-password -u user
       ```

c. Export the Discoverer catalog from the source database and import it into the database in the target environment by using the OLAP command utility. For more information see "Using the Discoverer Plus OLAP Command Line Utility to Manage the Discoverer Catalog" in the *Oracle Fusion Middleware Configuration Guide for Oracle Business Intelligence Discoverer*.

11. Move Portlet data from the source Discoverer metadata repository to the target Discoverer metadata repository:

    a. Use the Oracle Database export and import utilities.

       Note that you may need to perform the import multiple times to ensure that parent tables are populated before child tables. Use the following order to avoid SQL errors: PTM5_PARTITION, PTM5_PORTLET, PTM5_VERSION, PTM5_INSTANCE, PTM5_SCHEDULE, PTM5_CACHE,PTM5_CUSTOMINFO.

    b. Modify the Portlet Provider URL in the Portal to point to the new target setup.

12. Move PStore data:

    a. Delete the default encryption key from the table WWSSO_PS_CONFIGURATION_INFO_T.

    b. Move the PStore data for the Discoverer metadata repository using Oracle Database export and import utilities.

       Note that the user names and schema names must be the same in the target environment as in the source environment.

21.4.9.2 Moving Oracle Portal, Oracle Forms Services, Oracle Reports, and Oracle Business Intelligence Discoverer to an Existing Target Environment

In this procedure, you have installed Oracle Portal, Oracle Forms Services, Oracle Reports, and Oracle Business Intelligence Discoverer in the source environment and you want to move the components to the target environment that already exists.

To move to an existing target environment, perform the following tasks:

- Task 1, "Move Oracle Portal to an Existing Target Environment"
- Task 2, "Move Oracle Forms Services to an Existing Target Environment"
- Task 3, "Move Oracle Reports to an Existing Target Environment"
- Task 4, "Move Oracle Business Intelligence Discoverer to an Existing Target Environment"

Task 1 Move Oracle Portal to an Existing Target Environment

This procedure assumes that you have made changes to Oracle Portal in the source environment, such as adding pages, adding content to pages, creating new users and groups, and assigning page access permissions for newly created pages to new users and groups.

To move Oracle Portal to an existing target environment, take the steps described in Task 2, "Move Oracle Portal to the New Target Environment" in Section 21.4.9.1.

Task 2 Move Oracle Forms Services to an Existing Target Environment

To move Oracle Forms Services to the existing target environment:

1. Copy the Oracle Forms Services application files (FMX, MMX, and PLX) from the source environment to the target environment. The location of the files may be specified in the Forms environment configuration file, default.env.

   Note that if the files are in a shared network location, you do not need to copy them to the target environment. Instead, add the location to the default.env file.

2. Make any necessary configuration changes as described in "Deploying Your Application" in the *Oracle Fusion Middleware Forms Services Deployment Guide.*

3. Restart the components:

   ```
   ORACLE_INSTANCE/bin/opmnctl stopall
   ORACLE_INSTANCE/bin/opmnctl startall
   ```

Task 3 Move Oracle Reports to an Existing Target Environment

To move Oracle Reports to an existing target environment, take the same steps as described in Task 4, "Move Oracle Reports to the New Target Environment" in Section 21.4.9.1.

Task 4 Move Oracle Business Intelligence Discoverer to an Existing Target Environment

In this procedure, you primarily use the source environment to create EULs for developing a business area without compromising the performance of target environments.

To move Oracle BI Discoverer to an existing target environment:

1. Move the configuration files that are listed in Steps 1 and 5 in Task 5, "Move Oracle Business Intelligence Discoverer to the New Target Environment" in Section 21.4.9.1.

2. Move the DISCOVERER schema from the source environment to the target environment. You can use the Oracle Database export and import utilities to move the schema.

   Note that if you choose to use the same database for source and target, you do not need to move the data.

3. Move the EUL schema from the source environment to the target environment by using the Oracle database export and import utilities to export the schema from the source database and import it into the database in the target environment.

   Note that the user names and schema names must be the same in the target environment as in the source environment.

21.4.10 Moving Oracle Data Integrator to a Target Environment

The following topics describe how to move Oracle Data Integrator from the source environment to the target environment:

- Moving Oracle Data Integrator to a New Target Environment
- Moving Oracle Data Integrator to an Existing Target Environment

In both cases, you have performed the following in the source environment:

- Installed Oracle WebLogic Server and created the Middleware home for the Java components.
- Created the needed schemas in the source environment using RCU. See the *Oracle Fusion Middleware Repository Creation Utility User's Guide*.

- Installed Oracle Data Integrator.
- Configured and deployed Oracle Data Integrator Java components using the Configuration Wizard. The Java components can connect and use the source repositories.
- The source environment should be fully functional in terms of Oracle Data Integrator agents in Oracle WebLogic Server and have a working repository.

21.4.10.1 Moving Oracle Data Integrator to a New Target Environment

In this procedure, you have installed Oracle Data Integrator in the source environment and you want to move it to a target environment which does not yet exist.

To move Oracle Data Integrator to a new target environment, perform the following tasks:

- Task 1, "Move the Database, Middleware Home, and Perform the Initial Configuration"
- Task 2, "Review the Settings for the Target Environment"
- Task 3, "Restart the Java EE Agents in the Target Environment"

Task 1 Move the Database, Middleware Home, and Perform the Initial Configuration

To move the database, Middleware home, and perform the initial configuration on the target environment:

1. Create the required master and work repositories schemas in the target database using RCU. See the *Oracle Fusion Middleware Repository Creation Utility User's Guide*.

   Make sure that both the work and master repositories in the target environment are created with unique IDs across your entire organization, including your development and source repositories. Also make sure that the target work repository is created with the same type as the source repository (For example, if the source work repository is created as a development repository, the target work repository must also be created as a development repository).

2. Move the configuration of Oracle Data Integrator and its repository from the source environment to the target environment using the copyConfig and pasteConfig scripts, as described in Section 21.3.6.

   When you run the copyConfig script, note the following:

   - You must pass a configuration file to the copyConfig script. You pass this using the additionalParams option. For example:

     ```
     ./copyConfig.sh -javaHome /private/Middleware/jrockit_160_26_D1.2.0-5
        -archiveLocation /tmp/ar.jar
        -sourceMWHomeLoc /private/Middleware
        -sourceDomainLoc /private/Middleware/user_projects/domains/base_domain
        -domainHostName host1.example.com
        -domainPortNo 7001
        -domainAdminUserName weblogic
        -domainAdminPassword /tmp/wls_pswd.txt
        -additionalParams odiCustomArg=/private/t2p/odiCustomArg.xml
     ```

     The file odiCustomArg.xml is the configuration file.

- The configuration file that you pass to the script contains the connection information for all Oracle Data Integrator master repositories. The following shows a sample configuration file:

```
<?xml version="1.0" encoding="UTF-8" ?>
<config>
 <jps-config-path>/private/t2p/jps-config.xml</jps-config-path>
   <masterRepositories>
       <masterRepository>
           <driver>oracle.jdbc.OracleDriver</driver>
           <url>jdbc:oracle:thin:@localhost:1521/example.com</url>
           <schema>odi_master_11g</schema>
           <schema_password_file>/tmp/all_pswd.txt</schema_password_file>
            <supervisor>SUPERVISOR</supervisor>
            <supervisor_password_file>/tmp/sup_pswd.txt</supervisor_
password_file>
       </masterRepository>
       <masterRepository>
                        .....content for 2nd master repository
       </masterRepository>
    </masterRepositories>
</config>
```

  Note that when you move the configuration, the pasteConfig script copies the configuration of the domain, including the Administration Server and Managed Servers.

Task 2 Review the Settings for the Target Environment

The movement scripts update the physical architecture in the target environment according to the information you specified in the move plan. Review the following items in the physical architecture in the target environment before proceeding:

- Physical Agents: Change the host, port, and Web application context (for Java EE Agent) to match the configuration of the target environment.
- Data Servers: Change the data server connection information (JDBC, JNDI, data source name) to match the configuration of the target environment.
- Physical Schemas: The schemas (including file folder location) defined for the data servers must match the configuration of the target environment.

Task 3 Restart the Java EE Agents in the Target Environment

Restart the Java EE agents in the target environment. These agents start processing the scheduled scenarios.

21.4.10.2 Moving Oracle Data Integrator to an Existing Target Environment

In this procedure, you have a number of new or regenerated scenarios in the source environment and you want to move them to the target environment that already exists.

The movement scripts support repeated runs to the target environment. To overwrite the target environment with the latest source environment follow the process in Section 21.4.10.1, but take one of the following actions:

- If the repository is in internal authentication mode, supply the supervisor password in move plan before running the pasteConfig script.

- If the repository is in external authentication mode, change it to internal authentication mode and supply the supervisor password in move plan before running the pasteConfig script.

21.5 Considerations in Moving to and from an Oracle RAC Environment

If you are moving your environment to or from an Oracle Real Application Cluster (Oracle RAC) environment, note the following:

- If you are moving from a source environment that is not an Oracle RAC environment to a target environment that uses Oracle RAC, the move plan will have one entry for a generic data source (for example mds-soa.) You update the move plan to point to one of the Oracle RAC instances and complete the move from the source environment to the target environment.

  Then, you configure your target environment for Oracle RAC, as described in the *Oracle Fusion Middleware High Availability Guide*, especially "Considerations for High Availability Oracle Database Access."

- Multi data sources are moved to the target environment, even though they are not listed in the move plan.

- If you are moving from a source environment that uses Oracle RAC to a target environment that does not use Oracle RAC, the move plan will have multiple entries for generic data sources. For example, if you have four Oracle RAC instances, you will have four generic data sources that are named mds-soa-rac1 through mds-soa-rac4. You update the move plan to point all generic data sources to the single non-RAC instance in the target environment.

- If you are moving from a source environment that uses Oracle RAC to a target environment that uses Oracle RAC, but you have more Oracle RAC instances in the target environment, the move plan will have multiple entries for generic data sources. For example, if you have three Oracle RAC instances on the source environment, you will have three generic data sources that are named mds-soa-rac1 through mds-soa-rac3. You have four Oracle RAC instances in the target environment. You update the move plan to point the generic data sources to the first three generic data sources in the target environment.

- If you are moving from a source environment that uses Oracle RAC to a target environment that uses Oracle RAC, but you have fewer Oracle RAC instances in the target environment, the move plan will have multiple entries for generic data sources. For example, if you have four Oracle RAC instances on the source environment, you will have four generic data sources that are named mds-soa-rac1 through mds-soa-rac4. You have three Oracle RAC instances in the target environment. You update the move plan to point the first three generic data sources to the three generic data sources in the target environment. You point the last generic data source to the third generic data source. (The third Oracle RAC instance will contain both mds-soa-rac3 and mds-soa-rac4).

  Then, you can add an additional data source, as described in Section 10.2.2.1.

21.6 Limitations in Moving from Source to Target

Note the following limitations and restrictions:

- The target environment must be on the same operating system as the source environment. Also, the operating system architecture must be the same in both environments. For example, both environments must be running 32-bit operating systems or 64-bit operating systems.

- The target environment must have the same superuser or administrative user as the user at the source environment. The user's password can be different if you are using an external LDAP; you specify it on the command line when you use the pasteConfig script. After you complete the movement of the installation, you can modify the user on the target environment.

  Note, however, that you cannot change the password if you are using an embedded LDAP.

- When you move the configuration of a component, the scripts replicate the topology of the source. For example, if the source domain contains Managed Servers server_1 and server_2 on Host A and Managed Servers server_3 and server_4 on Host B, you must specify a similar relationship between Managed Servers and hosts at the target. (You specify the hosts for each Managed Server in the move plan.)

- All Oracle homes in the Middleware home must be registered in the same Oracle inventory. If you have installed multiple components under the same Middleware home, but used different Oracle inventory locations, the scripts are not able to detect all of the Oracle homes.

- If the source Oracle WebLogic Server machine is specified as unix-machineType in the config.xml file, the copyConfig script fails with the error CLONE-20408. To work around this problem, change the machine type in the following file on the source environment:

  ```
  DOMAIN_HOME/config/config.xml
  ```

  In the following line in the file, remove `xsi:type="unix-machineType"`:

  ```
  <machine xsi:type="unix-machineType">
     <name>machine_name</name>
  ```

  For example:

  ```
  <machine>
     <name>machine_name</name>
  ```

- If a custom application uses an internal data source (for example, the application was created and deployed with an internal data source using JDeveloper), the internal data source is not migrated during the pasteConfig operation.

  To work around this, create an external data source in the domain, modify the application to use that data source, and deploy the application again.

- If you are applying the clone of a Middleware home on a host that does not yet have Oracle Fusion Middleware installed, the host must have JDK 1.6.04 or higher installed. In addition, the PATH, CLASSPATH, and JAVA_HOME environment variables must point to the JDK.

- If the source Middleware home uses a JDK that is external to the Middleware Home, the pasteBinary operation must also use an external JDK.

- If there is not enough space in the temporary directory when you are moving an entity, an error is returned, noting the space needed. To work around this problem, specify a different location for the temporary directory by using the T2P_JAVA_OPTIONS environment variable as described in Section 20.3.

- When you use the pasteBinary script to move a Web Tier installation, you may receive the following error:

  ```
  SEVERE : Jun 14, 2011 12:35:27 AM - SEVERE - CLONE-20901  Unable to restore
  permission of a few files of the Oracle home
  ```

You can ignore this message, because the files are not needed.

- When you are moving Oracle Access Management Access Manager to a target environment and an additional server instance was created in the source environment, the pasteConfig script will fail. To work around this:
  - Do not create an additional Oracle Access Protocol host and port on the source system before you execute the movement scripts.
  - After you perform the movement scripts, create an additional Oracle Access Protocol host and port on the source environment, test environment, or both.

21.7 Recovering from Test to Production Errors

When you execute the pasteBinary or pasteConfig scripts and enter incorrect information in the move plan, the scripts return an error. In some cases, the scripts may have partially completed the paste operation. To recover, take the following actions, depending on the script that returned the error:

- On Windows if you are using the Sun JDK, the copyBinary, pasteBinary, copyConfig, or pasteConfig operations may fail with the following error:

  ```
  java.nio.channels.OverlappingFileLockException
  ```

  In this case, use the T2P_JAVA_OPTIONS to set the system property sun.nio.ch.disableSystemWideOverlappingFileLockCheck as shown in the following example:

  ```
  set T2P_JAVA_OPTIONS=
  -Dsun.nio.ch.disableSystemWideOverlappingFileLockCheck=true
  ```

  Then, retry the operation.

- If the pasteBinary script returns an error while moving the Middleware home directory at the target:
  1. Delete the target Middleware home.
  2. Remove the Oracle home entry from the Oracle inventory, if it is present.
  3. For Windows, remove the shortcut for the Middleware home and Oracle home.
- The copyConfig script requires that all servers be running, but that they are idle, so that no directories are being modified. If a server is not idle, the copyConfig script reports that the cloning operation completed successfully and the copyConfig error log file will remain at 0 bytes. However, the copyConfig standard log file will contain an error regarding writing to the packed_domain.jar. That error will cause the pasteConfig process to fail.

  To work around this issue, wait for a short period of time, then retry the copyConfig operation again.
- If the pasteConfig script returns an error while moving Java components:
  1. Stop all processes related to the domain.
  2. Delete the following directories:

     ```
     MW_HOME/user_projects/domains/domain_home
     MW_HOME/user_projects/applications/domain_name
     ```

  3. Drop the schemas and re-create them using RCU.

In addition, if the Oracle Platform Security reassociation failed:

- For an LDAP store, delete the domain node or specify a different value in the move plan.
- For a database-based store, drop the schema and re-create it using RCU.

- If the pasteConfig script returns an error while moving system components:

  1. Deinstall the instance.
  2. If you cannot deinstall the instance, stop all processes related to that instance and delete the Oracle instance.

- If the machine is specified as unix-machineType in the config.xml file, the pasteConfig script fails with the error CLONE-20408. To work around this problem, change the machine type in the following file:

  ```
  DOMAIN_HOME/config/config.xml
  ```

  In the following line in the file, remove xsi:type="unix-machineType":

  ```
  <machine xsi:type="unix-machineType">
     <name>machine_name</name>
  ```

  For example:

  ```
  <machine>
     <name>machine_name</name>
  ```

- If you encounter an out-of-memory error when you are using the pasteConfig script, you can work around this in one of the following ways:

  - Increase the JVM heap size: Use the option -Xmx for maximum heap size, and -Xms for initial heap size. For example:

    ```
    CONFIG_JVM_ARGS="-Xms512m -Xmx1024m"
    ```

  - Often, the Oracle WebLogic Server domain directory structure contains some large, unnecessary files, such as large older log files. You can delete these files, then run the copyConfig and pasteConfig scripts again.

- If you encounter the following error when you are using the copyConfig script for a Oracle SOA Suite installation, use the T2P_JAVA_OPTIONS environment variable to increase the message size:

  ```
  weblogic.socket.MaxMessageSizeExceededException: Incoming message of size:
  '10000080' bytes exceeds the configured maximum of: '10000000' bytes for
  protocol: 't3'.
  ```

  You use the T2P_JAVA_OPTIONS environment variable, as described in Section 20.3, to pass the -Dweblogic.MaxMessageSize=20000000 property to both the copyConfig and pasteConfig scripts.

- When you use the pasteConfig operation when Oracle B2B inbound/outbound dispatcher is configured, you may receive the following error:

  ```
  oracle.mds.exception.MDSRuntimeException: java.sql.SQLException: Data Source
  mds-soa does not exist.
  Data Source mds-soa does not exist.
  ```

  In this situation, after the failure, kill the Managed Server process and manually restart the Managed Server.

- If you receive an error when you attempt to start the Oracle SOA Suite Managed Server, you must modify system parameters using the Administration Console after you run the pasteConfig script. (Note that the pasteConfig script sets these system parameters with temporary values.)

  a. Log into the Oracle WebLogic Server Administration Console.

  b. In the Domain Structure window, expand the **Environment.**

  c. Click **Servers.** The Summary of Servers page appears.

  d. Select the server.

  e. Select the Server Start tab.

  f. In the **Arguments** field, enter the following parameters:

  ```
  -Dtangosol.coherence.wkan=hostname
  -Dtangosol.coherence.localhost=hostname
  -Dtangosol.coherence.localport=localport_number
  -Dtangosol.coherence.wka1.port=port_number_for_Coherence
  ```

  g. Click **Save** and **Activate Changes.**

  h. Start the server.

21.8 A Case Study: Moving Oracle SOA Suite and the Fusion Order Demo to a New Target Environment

In this case study, you move Oracle SOA Suite, along with the deployed SOA composite application, the Fusion Order Demo, to a new target environment.

In this procedure, you have performed the following in the source environment:

- Installed Oracle WebLogic Server and created the Middleware home.
- Created the required schemas in the source database using RCU.
- Installed Oracle SOA Suite.
- Configured Oracle SOA Suite using the Configuration Wizard.
- If required for your environment, installed and configured Identity Management components, such as Oracle Internet Directory, Oracle Platform Security, and Oracle Web Services Manager.
- Configured security policies.
- Deployed the Fusion Order Demo sample application.

  See "Introduction to the SOA Sample Application" in the *Oracle Fusion Middleware Developer's Guide for Oracle SOA Suite* for information on how to download the zip file for the application and how to deploy it.

To move the Oracle SOA Suite to a new target environment, perform the following tasks:

- Task 1, "Move the Database and the Middleware Home"
- Task 2, "Move the Domain Configuration"
- Task 3, "Deploy Oracle B2B Agreements and Enable Listening Channel"
- Task 4, "Validate the Fusion Order Demo"

Task 1 Move the Database and the Middleware Home

To move the Middleware home and binary files:

1. Move or create the database and the schemas, as described in Section 21.3.3.
2. Move Identity Management components, as described in Section 21.4.1.
3. Move the Middleware home and binary files:
   a. On Windows, at the source Middleware home, stop the Administration Server and any Managed Servers running in the Middleware home. (On UNIX, you do not need to stop the servers.)
   b. At the source Middleware home, execute the copyBinary script, which copies the WebLogic Server home and the Oracle homes contained within the Middleware home. If there are no Oracle homes in the source Middleware home, no Oracle homes are present in the archive.

      For example, to copy a Middleware home that is located at /scratch/Oracle /Middleware1, use the following command:

      ```
      copyBinary.sh -javaHome /scratch/Oracle/Middleware1/jrockit_160_20_
      D1.1.0-18
                    -archiveLoc /tmp/mw_copy.jar
                    -sourceMWHomeLoc /scratch/Oracle/Middleware1
                    -invPtrLoc /scratch/oracle/oraInst.loc
      ```

   c. If you are copying the Middleware home to a different host, copy the archive file to that system.
   d. Copy the pasteBinary script and the cloningclient.jar file to the target system and ensure that they have execute permission. See Section 20.3 for the locations of the files.

      Do **not** copy the other scripts, such as pasteConfig. Those scripts are generated when you extract the files, as in step e.
   e. At the target, extract the files from the archive using the pasteBinary script.

      For example, to apply the archive to the directory /scratch/oracle/MW_ Home_prod, use the following command:

      ```
      pasteBinary.sh -javaHome /scratch/Oracle/Middleware1/jrockit_160_20_
      D1.1.0-18
                     -archiveLoc  /tmp/mw_copy.jar
                     -targetMWHomeLoc  /scratch/oracle/MW_Home_prod
      ```

      The Middleware home is extracted to /scratch/oracle/MW_Home_prod and the WebLogic Server home and all of the Oracle homes are extracted under it with the same names as that of the source Oracle home names.

Task 2 Move the Domain Configuration

To move a copy of the domain configuration and Node Manager configuration:

1. At the source Middleware home, make sure that the Administration Server and all Managed Servers are started.
2. At the source, execute the copyConfig script to copy the domain configuration.

   See Section 20.3.1.3 for the syntax of the copyConfig script.

   For example, to copy the configuration of the Oracle SOA Suite domain named SOA_domain1 in the Middleware home /scratch/Oracle/Middleware1, use the following command:

```
copyConfig.sh -javaHome /scratch/Oracle/Middleware1/jrockit_160_20_D1.1.0-18
              -archiveLoc /tmp/soa.jar
              -sourceDomainLoc /scratch/Oracle/Middleware1/user_
projects/domains/SOA_domain1
              -sourceMWHomeLoc /scratch/Oracle/Middleware1
              -domainHostName example.com
              -domainPortNum 8001
              -domainAdminUserName domain_admin_username
              -domainAdminPassword /scratch/admin/passwd.txt
              -logDirLoc /tmp/logs
```

3. If you are copying the component to a different host, copy the archive file to that system.
4. Extract the move plan from the archive, using the extractMovePlan script.

   See Section 20.3.1.7 for the syntax of the extractMovePlan script.

   For example:

   ```
   extractMovePlan.sh -javaHome /scratch/Oracle/Middleware1/jrockit_160_20_
   D1.1.0-18
                   -archiveLoc /tmp/soa.jar
                   -planDirLoc /tmp/Oracle/t2p_plans/soa
   ```

5. Edit the move plan, modifying the properties to reflect the values for the target environment. See Table 20–14, Table 20–15, Table 20–16 and Table 20–17 for the list of properties for Oracle SOA Suite.

   For example, edit the host and port numbers for all properties, updating the properties with the correct values for the target environment.
6. Edit the adapters deployment plan files, which are located in the following directory:

   move_plan_dir/adapters

   (The location is specified in the move plan, under the configGroup Adapter.)

   If you only have the Fusion Order Demo configured, the files are:

   - FileAdapter_plan.xml
   - JmsAdapter_plan.xml
   - OracleBamAdapter_plan.xml

   If you updated other adapters, those files will be located in the adapter directory.

   For each file, edit the <config-root> element to specify the location of the adapters configuration plan on the target. For example, modify the following line to specify the location of the adapters configuration plan files on the target.

   ```
   <config-root>/scratch/Oracle/Middleware/Oracle_
   SOA1/soa/connectors/plan</config-root>
   ```

7. Edit the Composites configuration plan files, which are located in the following directory:

   move_plan_dir/composites

   The location is specified in the move plan, under the configProperty, Config Plan Location. For more information about Composites configuration plan files, see "Introduction to a Configuration Plan File" in the *Oracle Fusion Middleware Developer's Guide for Oracle SOA Suite*.

The files are:

- B2BX12OrderGateway_1.0_soaFusionOrderDemo.xml

  In this file, update the host and Managed Server port in the URLs in the location attribute to specify the values for the target. For example:

```
<attribute name="location">
    <replace>
http://example.com:18937/soa-infra/services/soaFusionOrderDemo/OrderSDOComp
osite!1.0/StoreFrontService?wsdl</replace>
</attribute>
```

- PartnerSupplierComposite_1.0_soaFusionOrderDemo.xml

  In this file, for each <wsdlAndSchema> element, change the location, if necessary, to specify a different location on the target. (The location is specified in the <replace> element.)

- OrderBookingComposite_1.0_soaFusionOrderDemo.xml

  In this file, update the host and Managed Server port in the URLs in the location attribute. For example:

```
<attribute name="location">
   <replace>http://example.com:18937/WebServices_WebLogicFusionOrderDemo_
CreditCardAuthorization/CreditAuthorizationPort?wsdl</replace>
</attribute>
```

- OrderSDOComposite_1.0_soaFusionOrderDemo.xml

  This file does not require any changes.

8. Copy the edited move plan, adapters deployment plan files, and the Composites configuration plan files to the target. (During the pasteConfig operation, you specify the location of the move plan using the -movePlanLoc option.)

9. At the target, extract the files from the archive using the pasteConfig script, which is described in Section 20.3.1.8. The pasteConfig scripts creates the domain and deploys all the configurations and composites.

   For example, to apply the archive to the Middleware home /scratch/Oracle/Middleware1, use the following command:

```
pasteConfig.sh -javaHome /scratch/Oracle/Middleware1/jrockit_160_20_D1.1.0-18
            -archiveLoc /tmp/soa.jar
            -movePlanLoc /tmp/Oracle/t2p_plans/soa/moveplan.xml
            -targetDomainLoc /scratch/Oracle/Middleware1/user_
projects/domains/SOA_domain1
            -targetMWHomeLoc /scratch/Oracle/Middleware1/
            -domainAdminPassword /scratch/pwd_dir/pass.txt
```

   After the script completes, the Administration Server is running, but the Managed Servers are not.

10. At the source, execute the copyConfig script to copy the Node Manager configuration.

    See Section 20.3.1.6 for the syntax of the copyConfig script. For example, use the following command:

```
copyConfig.sh -javaHome /scratch/Oracle/Middleware1/jrockit_160_20_D1.1.0-18
              -archiveLoc /tmp/nm.jar
              -sourceNMHomeLoc /scratch/Oracle/Middleware/wlserver_
10.3/common/nodemanager
```

```
                -logDirLoc /tmp/logs
```

11. If you are copying the Node Manager to a different host, copy the archive file to that system.
12. Extract the move plan from the archive, using the extractMovePlan script.

    See Section 20.3.1.7 for the syntax of the extractMovePlan script.

    For example:

```
extractMovePlan.sh -javaHome /scratch/Oracle/Middleware1/jrockit_160_20_
D1.1.0-18
                -archiveLoc /tmp/nm.jar
                -planDirLoc /tmp/Oracle/t2p_plans/nm
```

13. Edit the move plan, modifying the properties to reflect the values for the target environment. See Table 20–13 to find the list of properties for Node Manager.
14. Copy the edited move plan to the target. (During the pasteConfig operation, you specify the location using the -movePlanLoc option.)
15. At the target, extract the files from the archive using the pasteConfig script, See

    Section 20.3.1.11 for the syntax of the script.

    For example, use the following command:

```
pasteConfig -javaHome USER_HOME/jrockit_160_17_R28.0.0-679/
            -archiveLoc /tmp/nm.jar
            -targetNMHomeLoc /scratch/Oracle/Middleware1/wlserver_
10.3/common/nodemanager
            -targetMWHomeLoc /scratch/Oracle/Middleware1
            -movePlanLoc /tmp/Oracle/t2p_plans/nm/moveplan.xml
            -silent true
```

    After the script completes, the Administration Server is running, but the Managed Servers are not.
16. Start all Managed Servers.

Task 3 Deploy Oracle B2B Agreements and Enable Listening Channel

You must explicitly deploy theOracle B2B agreements. (See "Deploying an Agreement" in the *Oracle Fusion Middleware User's Guide for Oracle B2B* for more information.)

To deploy the Oracle B2B agreements and enable the listening channel:

1. Log in to the Oracle B2B console, by entering the following URL, and providing the user name and password:

```
http://host:8001/b2bconsole
```

2. Deploy the Oracle B2B agreements
   a. Select the Administration tab, then the Deploy tab.
   b. Use the search parameters to find the agreement you want to deploy and click **Search.**
   c. Highlight one or more agreements and click **Deploy.**
3. Enable the listening channel:
   a. Select the Administration tab, then the Listening Channel tab, and then the Channel Attributes tab.

b. Select the channel and click **Enable.**

Task 4 Validate the Fusion Order Demo

To validate the Fusion Order Demo:

For more information about running Fusion Order Demo, see "Running Fusion Order Demo" in the *Oracle Fusion Middleware Developer's Guide for Oracle SOA Suite*.

1. Access the Store Front from the following URL:

   `http://hostname:port/StoreFrontModule/faces/home.jspx`

   In the example, *hostname* is the DNS name or IP address of the Oracle WebLogic Server for Oracle SOA Suite and *port* is the address of the port, for example, port 8001, on which the Store Front module is deployed.

   You begin the order process by browsing the product catalog. When you click **Add** next to a product, the site updates the shopping cart region to display the item.

2. Begin the order process by browsing the product catalog. Click **Add** under the Tre 650 Phone/PDA for $299.99.

3. Click **Checkout** and log in using `ngreenbe` and `welcome1` in the **Username** and **Password** fields.

4. Provide the shipping and invoice details.

5. Click **Logout.**

Part IX

Appendixes

This part contains the following appendixes:

A

Oracle Fusion Middleware Command-Line Tools

This appendix summarizes the command-line tools that are available in Oracle Fusion Middleware.

Table A–1 Oracle Fusion Middleware Command-Line Tools

Command	Path	Description
adrci	UNIX: *MW_HOME*/wlserver_*n*/server/adr Windows: *MW_HOME*\wlserver_*n*\server\adr	Package incident and problem information into a zip file for transmission to Oracle Support.
bulkdelete	UNIX: *ORACLE_HOME*/ldap/bin/bulkdelete.sh Windows: *ORACLE_HOME*\ldap\bin\bulkdelete.bat	Delete a subtree efficiently in Oracle Internet Directory. **See:** *Oracle Fusion Middleware Reference for Oracle Identity Management*
bulkload	UNIX: *ORACLE_HOME*/ldap/bin/bulkload.sh Windows: *ORACLE_HOME*\ldap\bin\bulkload.bat	Create Oracle Internet Directory entries from data residing in or created by other applications. **See:** *Oracle Fusion Middleware Reference for Oracle Identity Management*
bulkmodify	UNIX: *ORACLE_HOME*/ldap/ bin/bulkmodify Windows: *ORACLE_HOME*\ldap\bin\bulkmodify	Modify a large number of existing Oracle Internet Directory entries in an efficient way. **See:** *Oracle Fusion Middleware Reference for Oracle Identity Management*
catalog	UNIX: *ORACLE_HOME*/ldap/bin/catalog.sh Windows: *ORACLE_HOME*\ldap\bin\catalog.bat	Add and delete catalog entries in Oracle Internet Directory. **See:** *Oracle Fusion Middleware Reference for Oracle Identity Management*
chgiphost	UNIX: *ORACLE_HOME*/chgip/scripts/chpiphost.sh Windows: *ORACLE_HOME*\chgip\scripts\chpiphost.bat	Changes the network configuration of Oracle HTTP Server and Oracle Web Cache. **See:** Section 15.1.2
config	UNIX: *ORACLE_HOME*/common/bin/config.sh Windows: *ORACLE_HOME*\common\bin\config.cmd	Invoke the Configuration Wizard to created and configure a domain or extend a domain. **See:** The Installation Guide for the component.
eulbuilder.jar	UNIX: *ORACLE_HOME*/bin/eulbuilder.jar Windows: *ORACLE_HOME*\bin\eulbuilder.jar	Discoverer EUL Java command-line interface. Create and manipulate Discoverer EULs without installing Oracle Discoverer Administrator. **See:** *Oracle Business Intelligence Discoverer EUL Command Line for Java User's Guide*
iasua	UNIX: *ORACLE_HOME*/upgrade/iasua.sh Windows: *ORACLE_HOME*\upgrade\iasua.bat	Oracle Fusion Middleware Upgrade Assistant. **See:** *Oracle Fusion Middleware Upgrade Planning Guide*
frmcmp	UNIX: *ORACLE_HOME*/bin/frmcmp.sh Windows: *ORACLE_HOME*\bin\frmcmp.exe	Start Form Compiler to generate a form. **See:** Oracle Forms Services Online Help

Table A–1 (Cont.) Oracle Fusion Middleware Command-Line Tools

Command	Path	Description
ldapadd	UNIX: *ORACLE_HOME*/bin/ldapadd Windows: *ORACLE_HOME*\bin\ldapadd	Add entries, their object classes, attributes, and values to Oracle Internet Directory. **See:** *Oracle Fusion Middleware Reference for Oracle Identity Management*
ldapaddmt	UNIX: *ORACLE_HOME*/bin/ldapaddmt Windows: *ORACLE_HOME*\bin\ldapaddmt	Add entries, their object classes, attributes, and values to Oracle Internet Directory. Like `ldapadd`, except supports multiple threads for adding entries concurrently. **See:** *Oracle Fusion Middleware Reference for Oracle Identity Management*
ldapbind	UNIX: *ORACLE_HOME*/bin/ldapbind Windows: *ORACLE_HOME*\bin\ldapbind	Determine if you can authenticate a client to a server. **See:** *Oracle Fusion Middleware Reference for Oracle Identity Management*
ldapcompare	UNIX: *ORACLE_HOME*/bin/ldapcompare Windows: *ORACLE_HOME*\bin\ldapcompare	Match attribute values you specify in the command line with the attribute values in the Oracle Internet Directory entry. **See:** *Oracle Fusion Middleware Reference for Oracle Identity Management*
ldapdelete	UNIX: *ORACLE_HOME*/bin/ldapdelete Windows: *ORACLE_HOME*\bin\ldapdelete	Remove entire entries from Oracle Internet Directory. **See:** *Oracle Fusion Middleware Reference for Oracle Identity Management*
ldapmoddn	UNIX: *ORACLE_HOME*/bin/ldapmoddn Windows: *ORACLE_HOME*\bin\ldapmoddn	Modify the DN or RDN of an Oracle Internet Directory entry. **See:** *Oracle Fusion Middleware Reference for Oracle Identity Management*
ldapmodify	UNIX: *ORACLE_HOME*/bin/ldapmodify Windows: *ORACLE_HOME*\ bin\ldapmodify	Perform actions on attributes in Oracle Internet Directory. **See:** *Oracle Fusion Middleware Reference for Oracle Identity Management*
ldapmodifymt	UNIX: *ORACLE_HOME*/bin/ldapmodifymt Windows: *ORACLE_HOME*\bin\ldapmodifymt	Modify several Oracle Internet Directory entries concurrently. **See:** *Oracle Fusion Middleware Reference for Oracle Identity Management*
ldapsearch	UNIX: *ORACLE_HOME*/bin/ldapsearch Windows: *ORACLE_HOME*\bin\ldapsearch	Search and retrieve specific entries in Oracle Internet Directory. **See:** *Oracle Fusion Middleware Reference for Oracle Identity Management*
ldifmigrator	UNIX: *ORACLE_HOME*/ bin/ldifmigrator Windows: *ORACLE_HOME*\bin\ldifmigrator.bat	Migrate data from application-specific repositories into Oracle Internet Directory. **See:** *Oracle Fusion Middleware Reference for Oracle Identity Management*
ldifwrite	UNIX: *ORACLE_HOME*/ldap/ bin/ldifwrite Windows: *ORACLE_HOME*\ldap\bin\ldifwrite.bat	Convert to LDIF all or part of the information residing in an Oracle Internet Directory. **See:** *Oracle Fusion Middleware Reference for Oracle Identity Management*
oidcmprec	UNIX: *ORACLE_HOME*/ldap/bin/oidcmprec Windows: *ORACLE_HOME*\ldap\bin\oidcmprec	Compare one Oracle Internet Directory with another, detect conflicts or discrepancies, and optionally resolve them. **See:** *Oracle Fusion Middleware Reference for Oracle Identity Management*
oidcred	UNIX: *ORACLE_HOME*/ldap/bin/oidcred Windows: *ORACLE_HOME*\ldap\ bin\oidcred	Add, update, or delete a credential that has been created in the Credential Store Framework. **See:** *Oracle Fusion Middleware Reference for Oracle Identity Management*

Table A–1 (Cont.) Oracle Fusion Middleware Command-Line Tools

Command	Path	Description
oidctl	UNIX: *ORACLE_HOME*/bin/oidctl Windows: *ORACLE_HOME*\ bin\oidctl	Start and stop Oracle Internet Directory. **See:** *Oracle Fusion Middleware Reference for Oracle Identity Management*
oiddiag	UNIX: *ORACLE_HOME*/ldap/bin/oiddiag Windows: *ORACLE_HOME*\ldap\ bin\oiddiag	Collects diagnostic information for Oracle Internet Directory. **See:** *Oracle Fusion Middleware Reference for Oracle Identity Management*
oidmon	UNIX: *ORACLE_HOME*/ bin/oidmon Windows: *ORACLE_HOME*\bin\oidmon	Monitor Oracle Internet Directory processes. **See:** *Oracle Fusion Middleware Reference for Oracle Identity Management*
oidpasswd	UNIX: *ORACLE_HOME*/ldap/ bin/oidpasswd Windows: *ORACLE_HOME*\ldap\bin\oidpasswd	Change the Oracle Internet Directory password and otherwise restricts access for Oracle Internet Directory **See:** *Oracle Fusion Middleware Reference for Oracle Identity Management*
oidprovtool	UNIX: *ORACLE_HOME*/bin/oidprovtool Windows: *ORACLE_HOME*\bin\oidprovtool.bat	Administer provisioning profile entries in Oracle Internet Directory. **See:** *Oracle Fusion Middleware Reference for Oracle Identity Management*
oidrealm	UNIX: *ORACLE_HOME*/ldap/bin/oidrealm Windows: *ORACLE_HOME*\ldap\bin\oidrealm.bat	Create multiple realms in Oracle Internet Directory. **See:** *Oracle Fusion Middleware Reference for Oracle Identity Management*
oidstats	UNIX: SQL command, oidstats.sql Windows: SQL command, oidstats.sql	Analyze the various database ods schema objects to estimate statistics. **See:** *Oracle Fusion Middleware Reference for Oracle Identity Management*
opmnctl	UNIX: *ORACLE_INSTANCE*/bin/opmnctl.exe Windows: *ORACLE_INSTANCE*\bin\opmnctl.exe	Start, stop, and get status on OPMN-managed processes. **See:** *Oracle Fusion Middleware Oracle Process Manager and Notification Server Administrator's Guide*
orapki	UNIX: *ORACLE_HOME*/bin/orapki Windows: *ORACLE_HOME*\bin\orapki.bat	Manages wallets and certificates. See Appendix H.
remtool	UNIX: *ORACLE_HOME*/ldap/bin/remtool Windows: *ORACLE_HOME*\ldap\bin\remtool	Search for problems and seek to rectify them in the event of an Oracle Internet Directory replication failure. **See:** *Oracle Fusion Middleware Reference for Oracle Identity Management*
rwbuilder	UNIX: *ORACLE_HOME*/bin/rwbuilder Windows: *ORACLE_HOME*\bin\rwbuilder	Invoke the Reports Builder. **See:** *Oracle Fusion Middleware Publishing Reports to the Web with Oracle Reports Services*
rwclient	UNIX: *ORACLE_HOME*/ bin/rwclient Windows: *ORACLE_HOME*\bin\rwclient	Parse and transfer a command line to the specified (or default) Reports Server. **See:** *Oracle Fusion Middleware Publishing Reports to the Web with Oracle Reports Services*
rwconverter	UNIX: *ORACLE_HOME*/bin/rwconverter Windows: *ORACLE_HOME*\bin\rwconverter	Convert one or more report definitions or PL/SQL libraries from one storage format to another. **See:** *Oracle Fusion Middleware Publishing Reports to the Web with Oracle Reports Services*
rwrun	UNIX: *ORACLE_HOME*/bin/rwrun Windows: *ORACLE_HOME*\bin\rwrun	Run a report using the Oracle Reports Services in-process server. **See:** *Oracle Fusion Middleware Publishing Reports to the Web with Oracle Reports Services*

Table A–1 (Cont.) Oracle Fusion Middleware Command-Line Tools

Command	Path	Description
rwserver	UNIX: *ORACLE_HOME*/bin/rwserver Windows: *ORACLE_HOME*\bin\rwserver.bat	Invoke the Reports Server. **See:** *Oracle Fusion Middleware Publishing Reports to the Web with Oracle Reports Services*
ssocfg	UNIX: sso/bin/ssocfg.sh Windows: sso\bin\ssocfg.bat	Update host, port, and protocol of Oracle Single Sign-On URL. **See:** *Oracle Fusion Middleware Administrator's Guide for Oracle Single Sign-On*, Release 10.1.3.4
ssooconf.sql	UNIX: *ORACLE_HOME*/portal/admin/plsql/sso/ssooconf.sql Windows: *ORACLE_HOME*\portal\admin\plsql\sso\ssooconf.sql	Script to point Oracle Single Sign-On server to a different Oracle Internet Directory. **See:** *Oracle Fusion Middleware Administrator's Guide for Oracle Single Sign-On* Release 10.1.3.4
wlst	UNIX: *WLS_HOME*/common/bin/wlst.sh Windows: *WLS_HOME*\common\bin\wlst.cmd UNIX: *ORACLE_HOME_for_component*/common/bin/wlst.sh Windows: *ORACLE_HOME_for_component*\common\bin\wlst.cmd	(WebLogic Scripting tool) Manages Oracle WebLogic Server and the components in a Oracle WebLogic Server domain. **See:** Section 3.5.1 and *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*

B

URLs for Components

This appendix provides the URLs needed to access Oracle Fusion Middleware components.

Table B–1 shows the URLs, and the default user to access components after installation.

The URLs in the table are shown with the default ports. The components in your environment might use different ports. To determine the port numbers, from the WebLogic Domain menu in Fusion Middleware Control, select **Port Usage**.

Unless otherwise noted, the password for each user is the password supplied during installation or the password you assigned to the user when you either created the user or changed the user's password.

Table B–1 URLs for Components

Component	URL (with Default Port Number)	Default User and Password
Oracle B2B	http://*host*:8001/b2b	`weblogic`
Oracle Business Activity Monitoring	http://*host*:9001/oracleBAM	`weblogic`
Oracle Business Intelligence Discoverer Plus	http://*host*:7777/discoverer/plus	n/a
Oracle Business Intelligence Discoverer Portlet Provider	http://*host*:7777/discoverer/portletprovider	n/a
Oracle Business Intelligence Discoverer Viewer	http://*host*:7777/discoverer/viewer	n/a
Oracle Directory Services Manager	https://*host*:7001/odsm	The superuser, such as `cn=orcladmin`
Oracle Enterprise Manager Fusion Middleware Control	http://*host*:7001/em	`weblogic`
Oracle Forms Services	http://*host*:*http_listen_port*/forms/frmservlet	Not Applicable
Oracle HTTP Server	http://*host*:7777	Not Applicable
Oracle Portal	http://*host*:*http_listen_port*/pls/portal	`orcladmin` Use the password that you supplied during installation.

Table B–1 (Cont.) URLs for Components

Component	URL (with Default Port Number)	Default User and Password
Oracle Reports Services	http://*host*:*http_listen_port*/reports/rwservlet	`orcladmin` The default password is the same as the `weblogic` password *of the Infrastructure* instance used by Oracle Reports.
Oracle WebCenter Content: Imaging	http://*host*:16000/imaging	First user to log in to Imaging
Oracle WebCenter Portal: Spaces	http://*host*:8888/webcenter	`weblogic`
Oracle WebCenter Portal's Activity Graph	http://*host*:8891/activitygraph-engines	`weblogic`
Oracle WebCenter Portal's Personalization	http://*host:port*/wcps/api/property/resourceIndex	`weblogic`
Oracle WebCenter Portal's Portlet Producer	http://*host*:8889/wsrp-tools/info http://*host*:8889/portalTools http://*host*:8889/pagelets	`weblogic`
Oracle WebCenter Portal's Discussion Server	http://*host*:8890/owc_discussions	`weblogic`
Oracle WebLogic Server Administration Console	http://*host*:7001/console	`weblogic`

C

Port Numbers

This appendix provides information about Oracle Fusion Middleware port numbers.

It contains the following topics:

C.1 Port Numbers by Component

This section provides the following information for each Oracle Fusion Middleware component or service that uses a port:

- **Component or Service:** The name of the component and service.
- **Default Port Number:** The first port number Oracle Fusion Middleware attempts to assign to a component. It is usually the lowest number in the allotted port range. If the port is in use, the next available port number, within the allotted range, is assigned.
- **Allotted Port Range:** The set of port numbers Oracle Fusion Middleware attempts to use when assigning a port.

Port numbers for Oracle WebLogic Server servers are assigned sequentially for each server created. For example, the first Administration Server is assigned the port 7001, the second 7002. Managed Servers created during installation and configuration for particular components may have specific default port numbers.

Table C–1 shows the default port number and the port number range for components, sorted alphabetically by component.

Table C–1 Port Numbers Sorted by Component

Component or Service	Default Port Number	Allotted Port Range
Oracle Access Management Identity Federation Server Managed Server	7499	7499-9000
Oracle Business Activity Monitoring	9001	9000-9080
Oracle Data Integrator	15000	15500
Oracle Directory Integration Platform	7005	7005-9000
Oracle Directory Services Manager	7005	7005-9000
Oracle Forms Services Managed Server	9001	9001-9100
Oracle HTTP Server non-SSL Listen Port	7777 or 8888	7777-7877, 8888

Table C–1 (Cont.) Port Numbers Sorted by Component

Component or Service	Default Port Number	Allotted Port Range
Oracle HTTP Server SSL Listen Port	4443	4443-4543
Oracle Information Rights Management	16100	16100-16199
Oracle Internet Directory (non-SSL)	3060	3061 to 3070, 13060 to 13070
Oracle Internet Directory (SSL)	3131	3132 to 3141, 13131 to 13141
Oracle Management Agent (used by Fusion Middleware Control)	5162	5162-6162
Oracle Notification Server Local Port	6100	6100 - 6199
Oracle Notification Server Remote Port	6200	6200 - 6299
Oracle Notification Server Request Port	6003	6003 - 6099
Oracle Portal Managed Server	9001	9001-9100
Oracle Reports Managed Server	9001	9001-9100
Oracle Virtual Directory (non-SSL)	6501	6501-6510
Oracle Virtual Directory (SSL)	7501	7501-7510
Oracle Web Cache Administration Port	7786	7781-7790
Oracle Web Cache Invalidation Port	7788	7781-7790
Oracle Web Cache Listen Port	7785	7781-7790
Oracle Web Cache SSL Listen Port	7789	7781-7790
Oracle Web Cache Statistics Port	7787	7781-7790
Oracle WebCenter Content Content Server	16200	16200-16299
Oracle WebCenter Content: Imaging	16000	16000-16099
Oracle WebCenter Content: Records	16300	16300-16399
Oracle WebCenter Personalization	8891	8891
Oracle WebCenter Portal Custom Portal managed server	8793	8793
Oracle WebCenter Portal Custom Portal managed server	8892	8892
Oracle WebCenter Portal: Spaces	8888	8881-8890
Oracle WebCenter Portal's Activity Graph Engines	8891	8891
Oracle WebCenter Portal's Pagelet Producer	8889	8881-8890
Oracle WebCenter Portal's Portlet Producer	8889	8881-8890
Oracle WebCenter Portal's Discussion Server	8890	8881-8890
Oracle WebLogic Server Listen Port for Administration Server	7001	7001-9000
Oracle WebLogic Server Listen Port for Managed Server	8001	8000 - 8080
Oracle WebLogic Server Node Manager Port	5556	5556

Table C–1 (Cont.) Port Numbers Sorted by Component

Component or Service	Default Port Number	Allotted Port Range
Oracle WebLogic Server SSL Listen Port for Administration Server	7002	7002-9000

C.2 Port Numbers (Sorted by Number)

Table C–2 lists Oracle Fusion Middleware ports numbers and components, sorted in ascending order by port number.

Table C–2 Port Numbers Sorted by Number

Default Port Number	Component or Service
3060	Oracle Internet Directory (non-SSL)
3131	Oracle Internet Directory (SSL)
4443	Oracle HTTP Server (SSL)
5162	Oracle Management Agent
5556	Oracle WebLogic Server Node Manager Port
6003	Oracle Notification Server Request Port
6100	Oracle Notification Server Local Port
6200	Oracle Notification Server Remote Port
6501	Oracle Virtual Directory (non-SSL)
7001	Oracle WebLogic Server Listen Port for Administration Server
7002	Oracle WebLogic Server SSL Listen Port for Administration Server
7005	Oracle Directory Integration Platform
7005	Oracle Directory Services Manager
7499	Oracle Access Management Identity Federation Server Managed Server
7501	Oracle Virtual Directory (SSL)
7777	Oracle HTTP Server (non-SSL)
7785	Oracle Web Cache (non-SSL)
7786	Oracle Web Cache Administration Port
7787	Oracle Web Cache Statistics Port
7788	Oracle Web Cache Invalidation Port
7789	Oracle Web Cache (SSL)
8001	Oracle WebLogic Server Listen Port for Managed Server
8793	Oracle WebCenter Portal Custom Services Producer managed server
8888	Oracle HTTP Server, Oracle WebCenter Portal: Spacess
8889	Oracle WebCenter Portal's Pagelet Producer
8889	Oracle WebCenter Portal's Portlet Producer
8890	Oracle WebCenter Portal's Discussion Server

Table C–2 (Cont.) Port Numbers Sorted by Number

Default Port Number	Component or Service
8891	Oracle WebCenter Analytics Collector
8891	Oracle WebCenter Personalization
8891	Oracle WebCenter Portal Activity Graph Engines
8892	Oracle WebCenter Portal Custom Portal managed server
9001	Oracle Business Activity Monitoring Managed Server
9001	Oracle Forms Services Managed Server
9001	Oracle Portal Managed Server
9001	Oracle Reports Managed Server
15000	Oracle Data Integrator
16000	Oracle WebCenter Content: Imaging
16100	Oracle Information Rights Management
16200	Oracle WebCenter Content
16300	Oracle WebCenter Content: Records

D

Metadata Repository Schemas

Oracle Fusion Middleware components store metadata in a repository. Many components require a database repository to store schemas to support the component. This appendix provides information about those schemas.

This appendix contains the following topics:

- Metadata Repository Schema Descriptions
- Metadata Repository Schemas, Tablespaces, and Data Files

D.1 Metadata Repository Schema Descriptions

Table D–1 lists the schemas used by Oracle Fusion Middleware components, sorted alphabetically by component. Note that the schema names are prefixed by the prefix you supplied when you ran the Repository Creation Utility.

Table D–1 Metadata Schemas Created by Repository Creation Utility

Component	Schema	Description
Oracle Access Management Access Manager	OAM	Contains information for Oracle Access Management Access Manager.
Oracle Access Management Identity Federation	OIF	Contains metadata for Identity Federation.
Oracle Adaptive Access Manager	OAAM OAAM_PARTN	Contains information for Oracle Adaptive Access Manager.
Oracle B2B	SOAINFRA	Contains the design and run-time repository. The design repository has modeling metadata and profile data for an integration. These describe the behavior of the integration and sequence of steps required to execute the business process. The modeling and profile metadata is the design of the integration prior to deployment and execution. Once the integration is deployed, the run-time repository contains the metadata required to execute the integration as well as the business process instance, event instances, role instances, and other data created during execution.
Oracle BPEL Process Manager	MDS SOAINFRA	MDS contains process definitions and configuration information. SOAINFRA contains instance and metadata database objects for Oracle Business Activity Monitoring and Oracle BPEL Process Manager.
Oracle Business Activity Monitoring	ORABAM MDS	Contains instance and metadata database objects for Oracle Business Activity Monitoring.

Table D–1 (Cont.) Metadata Schemas Created by Repository Creation Utility

Component	Schema	Description
Oracle Business Intelligence	BIPLATFORM	Contains metadata for Business Intelligence Server.
Oracle Business Intelligence Discoverer	DISCOVERER DISCOVERER_PS	Contains metadata for Discoverer Portlet Provider, portlet definitions for user portlets, and cached data obtained by running scheduled Discoverer queries. Has `RESOURCE` and `CONNECT` privileges.
Oracle Business Process Management	SOAINFRA	Contains metadata related to Oracle Business Process Management, as well as other SOA components.
Oracle Business Rules	MDS	Contains configuration information for Oracle Business Rules.
Oracle Content Server	OCS	Contains metadata information for Oracle Content Server 11*g*.
Oracle Data Integrator	ODI_REPO	Contains information for Oracle Data Integrator
Oracle Deployment Server	ODSSERVER	Contains metadata information for Oracle Deployment Server.
Oracle Directory Integration Platform	ODSSM	Contains configuration data for Oracle Directory Integration Platform.
Oracle Event Processing	MDS	MDS stores .MAR files, which store Oracle Event Processing .cqlsx files
Oracle Hyperion Enterprise Performance Management	EPM	Contains metadata for Oracle Hyperion Enterprise Performance Management.
Oracle Identity Manager	OIM	Contains metadata for applications that use Oracle Identity Manager.
Oracle Information Rights Management	ORAIRM	Contains metadata for Oracle Information Rights Management.
Oracle Internet Directory	ODS ODSSM	For internal use.
Oracle Mediator	MDS SOAINFRA	Contains metadata for Oracle Mediator.
Oracle Metadata Services	MDS	Contains metadata for applications that use MDS.
Oracle Portal	PORTAL	Contains Oracle Portal database objects and code.
Oracle Real-Time Decisions	RTD	For internal use.
Oracle Single Sign-On	ORASSO	For internal use.
Oracle SOA Suite Infrastructure	SOAINFRA	Contains metadata related to Oracle B2B, Oracle BPEL Process Manager, Oracle Business Process Management, Workflow, Sensor, Mediator, and CEP.
Oracle User Messaging	ORASDPM	Contains metadata related to User Messaging.
Oracle Web Services Manager	MDS	Contains configuration information.
Oracle WebCenter Content	OCS OCSSEARCH	Contains metadata for Oracle WebCenter Content.
Oracle WebCenter Content: Imaging	IPM	Contains metadata for Oracle WebCenter Content: Imaging.
Oracle WebCenter Content: Records	URMSERVER	Contains metadata for Oracle WebCenter Content: Records

Table D–1 (Cont.) Metadata Schemas Created by Repository Creation Utility

Component	Schema	Description
Oracle WebCenter Portal	WEBCENTER MDS	Contains information for Oracle WebCenter Portal.
Oracle WebCenter Portal: Spaces	WEBCENTER MDS	Contains information for WebCenter Portal services such as Links, lists, Tags, and Events.
Oracle WebCenter Portal's Activity Graph and Analytics	ACTIVITIES	Contains information for Oracle WebCenter Portal's Activity Graph and Analytics services.
Oracle WebCenter Portal's Portlet Producer	PORTLET	Contains information for WebCenter Portal's Portlet Producers.
Oracle WebCenter Portal's Discussion Server	DISCUSSIONS DISCUSSIONS_CRAWLER	Contains information for Oracle WebCenter Portal discussion server.

D.2 Metadata Repository Schemas, Tablespaces, and Data Files

Table D–2 lists the tablespace and default data file for each Metadata Repository schema. It is sorted alphabetically by schema name. Note that the default data files are prefixed by the prefix you assigned the schemas in RCU.

In addition to the tablespaces listed, the tablespace IAS_TEMP is always created when you create a schema with RCU. Its data file is iastemp.dbf.

Table D–2 Metadata Repository Tablespaces and Data Files

Schema	Tablespace	Default Data File
ACTIVITIES	IAS_ACTIVITIES	activities.dbf
BIPLATFORM	BIPLATFORM	biplatform.dbf
DISCOVERER	DISCO_PTM5_META	discoptm5meta.dbf
	DISCO_PTMS_CACHE	discoptm5cache.dbf
	DISCO_PSTORE	discopstore.dbf
DISCUSSIONS	IAS_DISCUSS	iasjive.dbf
DISCUSSIONS_CRAWLER	IAS_DSCRAWL	iasjivecrawl.dbf
EPM	EPM	epm.dbf
IPM	IPM	ipm.dbf
MDS	MDS	iasmds.dbf
OAAM	BRSADATA	brsdata*n*.dbf
OAAM_PARTN	OAAM_DATA	partn_brsadata*n*.dbf
OAM	OAM	oam.dbf
OAM	OAM	oam.dbf
OCS	OCS	ocs.dbf
OCSSEARCH	OCSSEARCH	ocssearch.dbf
ODI_REPO	ODI_USER	*prefix*_odi_user_*n*.dbf
ODS	OLTS_DEFAULT	default1_oid.dbf
ODSERVER	ODSERVER	odserver.dbf

Table D–2 (Cont.) Metadata Repository Tablespaces and Data Files

Schema	Tablespace	Default Data File
ODSSM	ODSSM	odssm.dbf
OIF	IAS_OIF	iasoif.dbf
OIM	OIM	oim.dbf
ORABAM	ORABAM	orabam.dbf
ORAIRM	ORAIRM	orairm.dbf
ORASDPLS	IAS_ORASDPLS	orasdpls.dbf
ORASDPM	IAS_ORASDPM	iassdpm.dbf
	IAS_ORASDPM_AQ	iassdpmaq.dbf
ORASDPSDS	IAS_ORASDPSDS	orasdpsds.dbf
ORASDPSXDMS	IAS_ORASDPSXDMS	orasdpsxdms.dbf
ORASSO	IAS_ORASSO	iasorasso.dbf
PORTAL	PORTAL	portal.dbf
	PORTAL_IDX	portalidx.dbf
	PORTAL_LOG	portallog.dbf
	PORTAL_DOC	portaldoc.dbf
PORTLET	IAS_PORTLET	webcenter_portlet.dbf
RTD	IAS_RTD	iasrtd.dbf
SOAINFRA	SOAINFRA	soainfra.dbf
URMSERVER	URMSERVER	urmserver.dbf
WEBCENTER	IAS_WEBCENTER	iaswebcenter.dbf

E

Using Oracle Fusion Middleware Accessibility Options

This appendix includes information about using Oracle Fusion Middleware accessibility options. It includes:

- Install and Configure Java Access Bridge (Windows Only)
- Enabling Fusion Middleware Control Accessibility Mode
- Fusion Middleware Control Keyboard Navigation

E.1 Install and Configure Java Access Bridge (Windows Only)

If you are installing on a Windows computer, you can install and configure Java Access Bridge for Section 508 Accessibility:

1. Download Java Access Bridge from the following URL:

   `http://www.oracle.com/technetwork/java/javase/tech/index-jsp-136191.html`

2. Install Java Access Bridge.
3. Copy the `access-bridge.jar` and `jaccess-1_4.jar` files from your installation location to the `jre/lib/ext` directory.
4. Copy the `WindowsAccessBridge.dll`, `JavaAccessBridge.dll`, and `JAWTAccessBridge.dll` files from your installation location to the `jre/bin` directory.
5. Copy the `accessibility.properties` file to the `jre/lib` directory.

E.2 Enabling Fusion Middleware Control Accessibility Mode

The following sections provide information on the benefits of running Fusion Middleware Control in accessibility mode, as well as instructions for enabling accessibility mode:

- Making HTML Pages More Accessible
- Viewing Text Descriptions of Fusion Middleware Control Charts

E.2.1 Making HTML Pages More Accessible

In Fusion Middleware Control, you can enable screen reader support. Screen reader support improves behavior with a screen reader. This is accomplished by adding

accessibility-specific constructs to the HTML, and by altering some navigation elements on the pages.

To enable screen reader mode in Fusion Middleware Control:

1. Choose **Setup,** then **My Preferences,** then **Accessibility.**

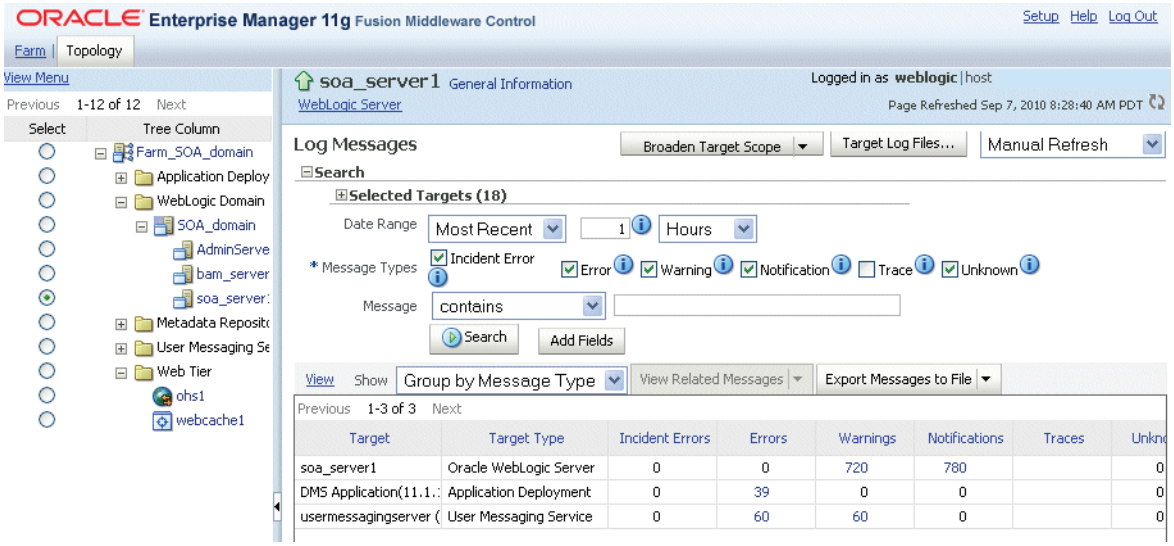
   The Accessibility Preference page is displayed.

2. Select one or both of the following options:

   - **I use a screen reader:** Accessibility-specific constructs are added to improve behavior with a screen reader.
   - **Show me the Accessibility Preference dialog when I log in:** When you log in, the Accessibility Preference dialog is displayed, with the following options:
     - I use a screen reader
     - Do not show me these options again

When you select screen reader support, Fusion Middleware Control renders the Web pages so that they can be read by a screen reader. For example, each node in the navigation tree includes a Select button.

The following figure shows the navigation pane and the Administration Server Performance Summary after enabling screen reader support:



E.2.2 Viewing Text Descriptions of Fusion Middleware Control Charts

Throughout Fusion Middleware Control, charts are used to display performance data. For most users, these charts provide a valuable graphical view of the data that can reveal trends and help identify minimum and maximum values for performance metrics.

However, charts do not convey information in a manner that can be read by a screen reader. To remedy this problem, you can configure Fusion Middleware Control to provide a complete textual representation of each performance chart. When you enable screen reader mode, Fusion Middleware Control displays the information in tables, instead of charts.

To view a representation of the data in a table, instead of a chart, without enabling screen reader mode, click **Table View** below a chart.

E.3 Fusion Middleware Control Keyboard Navigation

This section describes the keyboard navigation in Fusion Middleware Control.

Much of the keyboard navigation is the same whether or not you use screen reader mode.

Generally, you use the following keys to navigate:

- Tab key: Move to the next control, such as a dynamic target menu, navigation tree, content pane, or tab in a page. Tab traverses the page left to right, top to bottom. Use Shift +Tab to move to the previous control.
- Up and Down Arrow keys: Move to the previous or next item in the navigation tree, menu, or table. Down Arrow also opens a menu.
- Left and Right Arrow keys: Collapse and expand an item in the navigation tree or a submenu.
- Esc: Close a menu.
- Spacebar: Activate a control. For example, in a check box, spacebar toggles the state, checking or unchecking the box. On a link, spacebar navigates to the target of the link.
- Enter: Activate a button.

Table E–1 shows some common tasks and the keyboard navigation used.

Table E–1 Keyboard Navigation for Common Tasks

Task	Navigation
Move to next control, such as navigation tree or menu	Tab
Move to previous control, such as navigation tree or menu	Shift+Tab
Move to navigation pane	Tab until navigation tree has input focus
Move down the navigation tree	Down Arrow
Move up the navigation tree	Up Arrow
Expand a folder	Right Arrow
Collapse a folder	Left Arrow
Open a menu	Down Arrow
Move to the next item in a menu	Down Arrow
Move to the previous item in a menu	Up Arrow
Select a menu item	Enter
Open a submenu	Right Arrow
Close a submenu	Left Arrow
Move out of a menu	Esc
Activate a button	Enter
Open a tab in a content pane	Tab to the content pane, Tab to the tab to get input focus, then Enter to select the tab
Select an item, such as Message type in Log Messages screen	Spacebar

Table E–1 (Cont.) Keyboard Navigation for Common Tasks

Task	Navigation
Select a row in a table	Tab to the header of the table, then Down Arrow to move to a row
Select a cell in a table	Tab to the header of the table, then Tab until you reach the cell you want to select, then Enter

Table E–2 shows the keyboard navigation for the Topology Viewer. The navigation from one node to another is based on the geometry of the topology.

Table E–2 Keyboard Navigation for Topology Viewer

Task	Navigation
Navigate into the topology	Tab, until you have reached a node.
Navigate nodes based on geometry	Arrow keys
In a top-to-bottom orientation, navigate to a destination link	Ctrl+Shift+Down Arrow
In a top-to-bottom orientation, navigate to a source link	Ctrl+Shift+Up Arrow
In a left-to-right orientation, navigate to a destination link	Ctrl+Shift+Right Arrow
In a left-to-right orientation, navigate to a source link	Ctrl+Shift+Left Arrow
In a top-down orientation, when on a link, navigate to other links. The focus moves to another link based on the geometry.	Right Arrow or Left Arrow
In a left-to-right orientation, when on a link, navigate to other links. The focus moves to another link based on the geometry.	Up Arrow or Down Arrow
Move into or out of a group node	Shift+Arrow
Simulate a mouse click on the node. This can bring up a popup or it can navigate to another page.	Enter
Simulate a mouse over. Typically, this brings ups a popup.	Shift+Enter
Simulate a right-mouse click. Typically, this brings up a context menu.	m
Expand or contract a node subtree	e
Expand or contract a group. Note that if you use Shift+Arrow to move into a group, the group automatically expands.	g
Pan up or down, left or right	Ctrl+Arrow keys
Zoom in	Ctrl+Alt+Plus Key(+)
Zoom out	Ctrl+Alt+Minus Key(-)
Move out of a menu.	Esc

F

Examples of Administrative Changes

This appendix provides examples of administrative changes that can be performed on an Oracle Fusion Middleware environment. It is a companion to Part VII, "Advanced Administration: Backup and Recovery" in this book, and to the Disaster Recovery section in *Oracle Fusion Middleware High Availability Guide*.

It contains the following topics:

F.1 How to Use This Appendix

Some administrative operations cause configuration changes to your Oracle Fusion Middleware environment. These are called **administrative changes**, and include deploying and undeploying applications, adding or deleting Managed Servers or components, changing ports, creating and deleting users, and changing passwords. As an administrator, you should be aware when administrative changes occur because you may need to back up your environment or perform some synchronization procedures.

This appendix provides examples of administrative changes, listed by component. You can use this as a guide for performing the following procedures:

- Backup and Recovery

  Oracle recommends you perform a backup after each administrative change to your environment. You can use this appendix to determine the types of administrative changes that require you to back up your environment.

  See Also: Part VII, "Advanced Administration: Backup and Recovery"

- Disaster Recovery Synchronization Between the Primary and Standby Sites

  When you implement Disaster Recovery, you must update standby sites when you make an administrative change to your environment. You can use this appendix to determine the types of administrative changes that require you to update your standby sites.

  See Also: *Oracle Fusion Middleware High Availability Guide*

F.2 Examples of Administrative Changes (by Component)

Table F–1 provides examples of administrative changes, by component. Consult your component documentation to learn more about these operations.

Table F–1 Examples of Administrative Changes

Component	Examples of Administrative Changes
Directory Integration and Provisioning	Directory Integration and Provisioning administrative and configuration operations, such as running the ldapsearch utility
Dynamic Monitoring Service (DMS)	Manual edits to DMS configuration files, such as `dms.conf`
Fusion Middleware Control	Domain-wide or component-specific administrative and configuration operations performed using Fusion Middleware Control, changing port numbers, deploying and undeploying applications, and operations that result in configuration file changes
Oracle HTTP Server	Oracle HTTP Server administrative and configuration operations performed using Fusion Middleware Control, such as configuring modules, such as mod_wl_ohs, and creating virtual hosts Manual edits to Oracle HTTP Server configuration files Oracle HTTP Server administrative and configuration operations, such as registering a component with a domain, using the `opmnctl` utility
Oracle Internet Directory	Oracle Internet Directory administrative and configuration operations, such as running the `oidpasswd` utility (password management), and installing and removing components
Oracle Forms Services	Oracle Forms Services administrative and configuration operations performed using Fusion Middleware Control
Oracle Portal	Oracle Portal administrative and configuration operations performed using Fusion Middleware Control Oracle Portal administrative and configuration operations using the Administration screen in the Portal User Interface Manual edits to Oracle Portal configuration files Running the `ptlconfig` script Running any Portal-specific scripts that modify the database-side configuration for Portal, for example, disabling Oracle Web Cache or changing some background job frequencies in Portal
Oracle BPEL Process Analytics	Oracle BPEL Process Analytics administrative and configuration operations performed using Fusion Middleware Control
Oracle Reports Services	Oracle Reports Services administrative and configuration operations performed using Fusion Middleware Control, such as operations on the "Reports/Configuration" page Manual edits to Oracle Reports Services configuration files When the Reports server receives a job insert or update, such as when adding a new job or moving a job from one queue to another. *Note: Oracle recommends that you perform backup and file synchronization more frequently when running Oracle Reports Services.*
Oracle Web Cache	Oracle Web Cache configuration properties performed using Fusion Middleware Control. (**Web Cache** menu, then **Administration**)
Oracle WebLogic Server Administration Console	Domain-wide or component-specific administrative and configuration operations performed using the Administration Console, such as changing passwords, deploying and undeploying applications, and operations that result in configuration file changes

G

Viewing Release Numbers

This appendix describes how to view Oracle Fusion Middleware release numbers.

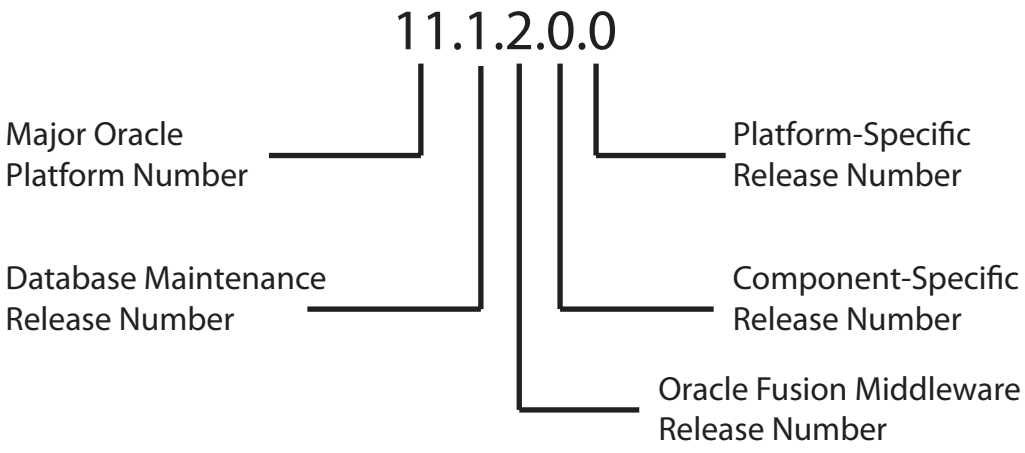
This appendix contains the following topics:

- Release Number Format
- Viewing the Software Inventory and Release Numbers

G.1 Release Number Format

To understand the release level nomenclature used by Oracle, examine the example of an Oracle Fusion Middleware release number shown in Figure G–1.

Figure G–1 Example of an Oracle Fusion Middleware Release Number



In Figure G–1, each digit is labeled:

- Major Oracle platform number

  This is the most general identifier. It represents a major new edition (or version) of an application, such as Oracle database server or Oracle Fusion Middleware, and indicates that the release contains significant new functionality.

- Database maintenance release number

  This digit represents a maintenance release level. Some new features may also be included.

- Oracle Fusion Middleware release number

  This digit reflects the release level of Oracle Fusion Middleware.

- Component-specific release number

  This digit identifies a release level specific to a component. Different components can have different numbers in this position depending upon, for example, component patch sets or interim releases.

- Platform-specific release number

  This digit identifies a platform-specific release.

G.2 Viewing the Software Inventory and Release Numbers

The following sections describe how to obtain the release numbers of Oracle Fusion Middleware:

- Viewing Oracle Fusion Middleware Installation Release Numbers
- Viewing Component Release Numbers
- Viewing Oracle Internet Directory Release Numbers
- Viewing Metadata Repository Release Numbers

G.2.1 Viewing Oracle Fusion Middleware Installation Release Numbers

All Oracle Fusion Middleware installations have a release number. This number is updated when you apply a patch set release or upgrade the installation.

You can view the release number of an Oracle Fusion Middleware installation using Opatch. Run the following command:

```
(UNIX) ORACLE_HOME/Opatch/opatch lsinventory
(Windows) ORACLE_HOME\Opatch\opatch lsinventory
```

For example, on UNIX:

```
 ./opatch lsinventory
Invoking OPatch 11.1.0.8.3

Oracle Interim Patch Installer version 11.1.0.8.3
Copyright (c) 2010, Oracle Corporation.  All rights reserved.


Oracle Home       : /scratch/oracle1/Oracle/Middleware/Oracle_SOA1
Central Inventory : /scratch/oracle1/oraInventory
   from           : /etc/oraInst.loc
OPatch version    : 11.1.0.8.3
OUI version       : 11.1.0.9.0
OUI location      : /scratch/oracle1/Oracle/Middleware/Oracle_SOA1/oui
Log file location : /scratch/oracle1/Oracle/Middleware/Oracle_
SOA1/cfgtoollogs/opatch/opatch2011-07-18_13-55-10PM.log

Patch history file: /scratch/oracle1/Oracle/Middleware/Oracle_
SOA1/cfgtoollogs/opatch/opatch_history.txt


OPatch detects the Middleware Home as "/scratch/oracle1/Oracle/Middleware"

Lsinventory Output file location : /scratch/oracle1/Oracle/Middleware/Oracle_
SOA1/cfgtoollogs/opatch/lsinv/lsinventory2011-07-18_13-55-10PM.txt

--------------------------------------------------------------------------------
```

```
Installed Top-level Products (1):

Oracle SOA Suite 11g                                                 11.1.1.5.0
There are 1 products installed in this Oracle Home.


There are no Interim patches installed in this Oracle Home.


--------------------------------------------------------------------------------

OPatch succeeded.
```

G.2.2 Viewing Oracle WebLogic Server Release Numbers

You can use the following command to view the release number of Oracle WebLogic Server:

```
(UNIX) cat $MW_HOME/wlserver_10.3/.product.properties | grep WLS_PRODUCT_VERSION
(Windows) type %MW_HOME%\wlserver_10.3\.product.properties | findstr WLS_PRODUCT_
VERSION
```

For example, on UNIX:

```
cat $MW_HOME/wlserver_10.3/.product.properties | grep WLS_PRODUCT_VERSION
WLS_PRODUCT_VERSION=10.3.6.0
```

G.2.3 Viewing Component Release Numbers

All Oracle Fusion Middleware components have a release number and many contain services that have release numbers. These numbers *may* be updated when you apply a patch set release or upgrade the installation.

You can view the release number of components and their services by using the following commands:

- On UNIX:

  ```
  cd ORACLE_HOME/inventory
  ls -d Components*/*/*
  ```

- On Windows:

  ```
  cd ORACLE_HOME/inventory/Componentsn
  dir /S /A:D
  ```

G.2.4 Viewing Oracle Internet Directory Release Numbers

Oracle Internet Directory has a server release number, which is the version of the binaries. It also has schema and context versions. All of these numbers correspond to the Oracle Fusion Middleware installation release number through the third digit. These numbers *may* be updated when you apply a patch set release or upgrade the installation.

Viewing the Oracle Internet Directory Server Release Number

The Oracle Internet Directory server release number is the version of the binaries. You can view the Oracle Internet Directory server release number as follows:

1. Ensure that the ORACLE_HOME environment variable is set.
2. Run the following command:

```
(UNIX) ORACLE_HOME/bin/oidldapd -version
(Windows) ORACLE_HOME\bin\oidldapd -version
```

Viewing the Oracle Internet Directory Schema and Context Versions

You can view the Oracle Internet Directory schema and context versions in this file:

```
(UNIX) ORACLE_HOME/ldap/schema/versions.txt
(Windows) ORACLE_HOME\ldap\schema\versions.txt
```

The contents of this file are kept up-to-date, however, you can also query the schema and context release from Oracle Internet Directory, just to be sure.

To view the schema version:

1. Ensure that the ORACLE_HOME environment variable is set.
2. Run the following command:

   ```
   ldapsearch -h oid_host -p oid_port -D "cn=orcladmin"
    -q -b "cn=base,cn=oracleschemaversion"
    -s base "objectclass=*" orclproductversion
   ```

   Because you use the -q option, the command prompts you for your password.

   The output is in this form:

   ```
   cn=BASE,cn=OracleSchemaVersion
   orclproductversion=90500
   ```

To view the context version:

1. Ensure that the ORACLE_HOME environment variable is set.
2. Run the following command:

   ```
   ldapsearch -h oid_host -p oid_port -D "cn=orcladmin"
    -q -b "cn=oraclecontext" -s base "objectclass=*" orclversion
   ```

   Because you use the -q option, the command prompts you for your password.

   The output is in this form:

   ```
   cn=oraclecontext
   orclversion=101200
   ```

G.2.5 Viewing Metadata Repository Release Numbers

If you are using an Oracle Database instance for your metadata repository, you can view the release number of the database using SQL*Plus as follows (you can be connected to the database as any user to issue these commands):

```
SQL> COL PRODUCT FORMAT A40
SQL> COL VERSION FORMAT A15
SQL> COL STATUS FORMAT A15
SQL> SELECT * FROM PRODUCT_COMPONENT_VERSION;

PRODUCT                                  VERSION         STATUS
---------------------------------        --------------  ----------------
NLSRTL                                   11.2.0.1.0      Production
Oracle Database 11g Enterprise Edition   11.2.0.1.0      Production
PL/SQL                                   11.2.0.1.0      Production
TNS for Linux:                           11.2.0.1.0      Production
```

G.2.6 Viewing Schema Release Numbers

If you are using an Oracle Database instance for your metadata repository, you can view the release number of the schema using SQL*Plus, as follows:

```
SQL> COL COMP_ID FORMAT A20
SQL> COL COMP_NAME A30
SQL> COL VERSION FORMAT A20
SQL> SELECT COMP_ID, COMP_NAME, VERSION FROM SCHEMA_VERSION_REGISTRY;

COMP_ID              COMP_NAME                      VERSION
-------------------- ------------------------------ --------------------
APM                  Authorization Policy Manager  11.1.1.3.0

BAM                  BAM Services                   11.1.1.5.0

BIPLATFORM           OracleBI and EPM               11.1.1.4.0
 .
 .
 .
```

H

Oracle Wallet Manager and orapki

Oracle Application Server 10*g* provided two utilities for managing wallets and certificates:

- Oracle Wallet Manager, a graphical user interface tool to manage PKI certificates
- The `orapki` utility, a command-line tool to manage certificate revocation lists (CRLs), create and manage Oracle wallets, and create signed certificates for testing purposes

Additionally, Oracle Application Server 10*g* provided the SSL Configuration Tool.

Oracle Fusion Middleware 11*g* Release 2 (11.1.2) provides:

- Additional `orapki` features
- The ability to manage JKS-based keystores, wallets, and certificates using Fusion Middleware Control
- Both command-line and graphical user interfaces to configure SSL. See Chapter 6 for details.

Use this appendix to learn about `orapki` updates, and to help transition to the new certificate, wallet management, and SSL configuration tools provided in 11*g* Release 2 (11.1.2). The appendix contains these topics:

See Also: *Oracle Advanced Security Administrator's Guide* for details of Oracle Wallet Manager and `orapki` usage:

`http://docs.oracle.com/cd/E11882_01/network.112/e10746/toc.htm`

Note: The `orapki` utility is located in the binary directory of Oracle Common home, that is, `$MIDDLEWARE_HOME/oracle_common/bin`.

H.1 New orapki Features

The `orapki` command-line utility contains these new features in Oracle Fusion Middleware 11*g* Release 2 (11.1.2):

- orapki Usage Examples
- New CRL Management Features
- New Version 3 Certificate Support
- Trust Chain Export
- Wallet Password Change
- Converting Between Oracle Wallet and JKS Keystore

H.1.1 orapki Usage Examples

See Also: Doc ID 1226654.1, "How To Create a Wallet via ORAPKI in FMW 11g" on the OTN Knowledge Base.

Here are a few examples of using orapki:

```
# Create root wallet (for example, CA wallet)
orapki wallet create -wallet ./root -pwd mypasswd

# Add a self-signed certificate (CA certificate) to the root wallet
orapki wallet add -wallet ./root -dn 'CN=root_test,C=US' -keysize 1024 -self_
signed -validity 3650 -pwd mypasswd

# Export self-signed certificate from the wallet
orapki wallet export -wallet ./root -dn 'CN=root_test,C=US' -cert
./root/b64certificate.txt -pwd mypasswd

# Create a user wallet (for example, a customer wallet)
orapki wallet create -wallet ./user -pwd mypasswd

# Add a certificate request
orapki wallet add -wallet ./user -dn 'CN=user_test,C=US' -keysize 1024 -pwd
mypasswd

# Export the certificate request
orapki wallet export -wallet ./user -dn 'CN=user_test,C=US' -request
./user/creq.txt -pwd mypasswd

# Create a certificate (issued by CA)
orapki cert create -wallet ./root -request ./user/creq.txt -cert ./user/cert.txt
-validity 3650 -pwd mypasswd

# Add a trusted certificate (CA certificate) to the wallet
orapki wallet add -wallet ./user -trusted_cert -cert ./root/b64certificate.txt
-pwd mypasswd

# Add a user certificate
orapki wallet add -wallet ./user -user_cert -cert ./user/cert.txt -pwd mypasswd

# Display contents of wallet
orapki wallet display -wallet ./root -pwd mypasswd
```

H.1.2 New CRL Management Features

`orapki` supports several new command options to work with CRLs:

Creating a CRL

You use orapki crl create to create a CRL.

See Section H.2.6.3, "orapki crl create."

Revoking a Certificate

You use orapki crl revoke to revoke a certificate.

See Section H.2.6.8, "orapki crl revoke."

Verifying a CRL Signature

You use orapki crl verify to verify a CRL signature.

See Section H.2.6.11, "orapki crl verify."

Checking If a Certificate Is Revoked in a CRL

You use orapki crl status to check if a certificate is revoked.

See Section H.2.6.9, "orapki crl status."

H.1.3 New Version 3 Certificate Support

orapki provides:

- The ability to add a subject key identifier extension to a certificate request
- The ability to add a version3 self-signed certificate to a wallet

See Section H.2.6.12, "orapki wallet add" for information about these features.

H.1.4 Trust Chain Export

You use orapki wallet export_trust_chain to export a chain of trust (certificate chain) for a user.

See Section H.2.6.17, "orapki wallet export_trust_chain."

H.1.5 Wallet Password Change

You use orapki wallet change_pwd to change a wallet password.

See Section H.2.6.13, "orapki wallet change_pwd."

H.1.6 Converting Between Oracle Wallet and JKS Keystore

You can convert a JKS keystore to an Oracle wallet, and convert an Oracle wallet to JKS.

Converting JKS to Oracle Wallet

Use this command to migrate entries from JKS store to p12 wallet:

```
jks_to_pkcs12 -wallet wallet -pwd pwd -keystore keystore
-jkspwd jkspwd [-aliases [alias:alias..]]
```

where the parameters are as follows:

- `wallet` is the wallet location; entries from the JKS keystore will be migrated to this wallet.
- `pwd` is the wallet password.
- `keystore` is the keystore location; this JKS will be migrated to the p12 wallet.
- `jkspwd` is the JKS password.
- `aliases` are optional. If specified, only entries corresponding to the specified alias are migrated. If not specified, all the entries are migrated.

To illustrate this command, start by creating a self-signed JKS keystore:

```
keytool -genkey -alias myalias -keyalg RSA -keysize 1024 -dname CN=root,C=US
-validity 3650 -keystore ./ewallet.jks -storetype jks -storepass password
-keypass password
```

Next, create an Oracle wallet:

```
orapki wallet create -wallet ./ -pwd password
```

Migrate the JKS keystore entries to the wallet:

```
orapki wallet jks_to_pkcs12 -wallet ./ -pwd password -keystore ./ewallet.jks
-jkspwd password
```

> **Note:** In this example the wallet was newly created and is empty. However, in practice the wallet need not be empty when you use this command; pre-existing entries are preserved.

Converting Oracle Wallet to JKS

Use this command to migrate entries from a p12 wallet to a JKS keystore:

```
pkcs12_to_jks -wallet p12wrl -pwd p12pwd
[-jksKeyStoreLoc jksKSloc -jksKeyStorepwd jksKS_pwd]
[-jksTrustStoreLoc loc -jksTrustStorepwd pwd]
```

where the parameters are as follows:

- `wallet` is the p12 wallet location
- `pwd` is the wallet password
- `jksKeyStoreLoc` is the JKS keystore location
- `jksKeyStorepwd` is the JKS keystore password
- `jksTrustStoreLoc` is the JKS truststore location
- `jksTrustStorepwd` is the JKS truststore password

> **Note:** Passwords must have a minimum length of eight characters and contain alphabetic characters combined with numbers or special characters.

This example migrates all wallet entries to the same JKS keystore:

```
orapki wallet pkcs12_to_jks -wallet ./ -pwd mypasswd -jksKeyStoreLoc ./ewallet.jks
-jksKeyStorepwd mypasswd2
```

This example migrates keys and trusted certificate entries into separate JKS keystores:

```
orapki wallet pkcs12_to_jks -wallet ./ -pwd mypasswd
-jksKeyStoreLoc ./ewalletK.jks -jksKeyStorepwd mypasswd2
-jksTrustStoreLoc ./ewalletT.jks -jksTrustStorepwd mypasswd2
```

H.2 Using the orapki Utility for Certificate Validation and CRL Management

This section contains these topics:

- orapki Overview
- Displaying orapki Help
- Creating Signed Certificates for Testing Purposes
- Managing Oracle Wallets with the orapki Utility
- Managing Certificate Revocation Lists (CRLs) with orapki Utility
- orapki Utility Commands Summary

H.2.1 orapki Overview

The `orapki` utility is provided to manage public key infrastructure (PKI) elements, such as wallets and certificate revocation lists, on the command line so the tasks it performs can be incorporated into scripts. This enables you to automate many of the routine tasks of maintaining a PKI.

This command-line utility can be used to perform the following tasks:

- Creating signed certificates for testing purposes
- Managing Oracle wallets:
  - Creating and displaying Oracle wallets
  - Adding and removing certificate requests
  - Adding and removing certificates
  - Adding and removing trusted certificates
- Managing certificate revocation lists (CRLs):
  - Renaming CRLs with a hash value for certificate validation
  - Uploading, listing, viewing, and deleting CRLs in Oracle Internet Directory

`orapki` allows you to import certificates in both DER and PEM formats.

H.2.1.1 orapki Syntax

The basic syntax of the `orapki` command-line utility is as follows:

```
orapki module command -parameter value
```

In the preceding command, *`module`* can be `wallet` (Oracle wallet), `crl` (certificate revocation list), or `cert` (PKI digital certificate). The available commands depend on the `module` you are using. For example, if you are working with a `wallet`, then you can add a certificate or a key to the wallet with the `add` command. The following example adds the user certificate located at `/private/lhale/cert.txt` to the wallet located at `ORACLE_HOME/wallet/ewallet.p12`:

```
orapki wallet add -wallet ORACLE_HOME/wallet/ewallet.p12
-user_cert -cert /private/lhale/cert.txt
```

H.2.1.2 Environment Setup for orapki

When running orapki, ensure that one of these following environment settings is in place:

- If running in the context of Identity Management or Web Tier or Classic installations, set `ORACLE_HOME` to point to the product installation location.
- If running in the context of Oracle SOA Suite or Oracle WebCenter Portal installations, set `JAVA_HOME` to point to a valid JDK location that contains Java 1.5 or higher.

H.2.2 Displaying orapki Help

You can display all the `orapki` commands that are available for a specific mode by entering the following at the command line:

```
orapki mode help
```

For example, to display all available commands for managing certificate revocation lists (CRLs), enter the following at the command line:

```
orapki crl help
```

Note: Using the `-summary`, `-complete`, or `-wallet` command options is always optional. A command will still run if these command options are not specified.

H.2.3 Creating Signed Certificates for Testing Purposes

This command-line utility provides a convenient, lightweight way to create signed certificates for testing purposes. The following syntax can be used to create signed certificates and to view certificates:

To create a signed certificate for testing purposes:

```
orapki cert create [-wallet wallet_location] -request
 certificate_request_location
-cert certificate_location -validity number_of_days [-summary]
```

This command creates a signed certificate from the certificate request. The `-wallet` parameter specifies the wallet containing the user certificate and private key that will be used to sign the certificate request. The `-validity` parameter specifies the number of days, starting from the current date, that this certificate will be valid. Specifying a certificate and certificate request is mandatory for this command.

To view a certificate:

```
orapki cert display -cert certificate_location [-summary | -complete]
```

This command enables you to view a test certificate that you have created with `orapki`. You can choose either `-summary` or `-complete`, which determines how much detail the command will display. If you choose `-summary`, the command will display the certificate and its expiration date. If you choose `-complete`, it will display additional certificate information, including the serial number and public key.

H.2.4 Managing Oracle Wallets with the orapki Utility

The following sections describe the syntax used to create and manage Oracle wallets with the `orapki` command-line utility. You can use these `orapki` utility `wallet` module commands in scripts to automate the wallet creation process.

- Creating and Viewing Oracle Wallets with orapki
- Adding Certificates and Certificate Requests to Oracle Wallets with orapki
- Exporting Certificates and Certificate Requests from Oracle Wallets with orapki

---

Note: The `-wallet` parameter is mandatory for all `wallet` module commands.

See Also: For examples of how to create either a password-protected wallet or an auto-login wallet, see Doc ID 1226654.1, "How To Create a Wallet via ORAPKI in FMW 11g" on the OTN Knowledge Base.

H.2.4.1 Creating and Viewing Oracle Wallets with orapki

To create an Oracle wallet:

```
orapki wallet create -wallet wallet_location
```

This command will prompt you to enter and re-enter a wallet password. It creates a wallet in the location specified for `-wallet`.

To create an Oracle wallet with auto-login enabled:

```
orapki wallet create -wallet wallet_location -auto_login
```

This command creates a wallet with auto-login enabled, or it can also be used to enable auto-login on an existing wallet. If the `wallet_location` already contains a wallet, then auto-login will be enabled for it. To disable the auto-login feature, delete `cwallet.sso`.

Note: For wallets with the auto-login feature enabled, you are prompted for a password only for operations that modify the wallet, such as `add`.

To view an Oracle wallet:

```
orapki wallet display -wallet wallet_location
```

This command displays the certificate requests, user certificates, and trusted certificates contained in the wallet.

H.2.4.2 Adding Certificates and Certificate Requests to Oracle Wallets with orapki

To add a certificate request to an Oracle wallet:

```
orapki wallet add -wallet wallet_location -dn user_dn -keysize 512|1024|2048|4096
```

This command adds a certificate request to a wallet for the user with the specified distinguished name (`user_dn`). The request also specifies the requested certificate's key size (512, 1024, or 2048 bits). To sign the request, export it with the export option.

See Section H.2.4.3, "Exporting Certificates and Certificate Requests from Oracle Wallets with orapki."

To add a trusted certificate to an Oracle wallet:

```
orapki wallet add -wallet wallet_location -trusted_cert -cert
certificate_location
```

This command adds a trusted certificate, at the specified location (`-cert certificate_location`), to a wallet. You must add all trusted certificates in the certificate chain of a user certificate before adding a user certificate, or the command to add the user certificate will fail.

To add a root certificate to an Oracle wallet:

```
orapki wallet add -wallet wallet_location -dn
certificate_dn -keysize 512|1024|2048 -self_signed -validity number_of_days
```

This command creates a new self-signed (root) certificate and adds it to the wallet. The `-validity` parameter (mandatory) specifies the number of days, starting from the current date, that this certificate will be valid. You can specify a key size for this root certificate (`-keysize`) of 512, 1024, 2048, or 4096 bits.

To add a user certificate to an Oracle wallet:

```
orapki wallet add -wallet wallet_location -user_cert -cert certificate_location
```

This command adds the user certificate at the location specified with the `-cert` parameter to the Oracle wallet at the *`wallet_location`*. Before you add a user certificate to a wallet, you must add all the trusted certificates that make up the certificate chain. If all trusted certificates are not installed in the wallet before you add the user certificate, then adding the user certificate will fail.

H.2.4.3 Exporting Certificates and Certificate Requests from Oracle Wallets with orapki

To export a certificate from an Oracle wallet:

```
orapki wallet export -wallet wallet_location -dn
certificate_dn -cert certificate_filename
```

This command exports a certificate with the subject's distinguished name (`-dn`) from a wallet to a file that is specified by `-cert`.

To export a certificate request from an Oracle wallet:

```
orapki wallet export -wallet wallet_location -dn
certificate_request_dn -request certificate_request_filename
```

This command exports a certificate request with the subject's distinguished name (`-dn`) from a wallet to a file that is specified by `-request`.

H.2.5 Managing Certificate Revocation Lists (CRLs) with orapki Utility

CRLs must be managed with `orapki`. This utility creates a hashed value of the CRL issuer's name to identify the CRLs location in your system. If you do not use `orapki`, your Oracle server cannot locate CRLs to validate PKI digital certificates. The following sections describe CRLs, how you use them, and how to use orapki to manage them:

- Section H.2.5.1, "About Certificate Validation with Certificate Revocation Lists"
- Section H.2.5.2, "Certificate Revocation List Management"

> **See Also:** "Certificate Revocation List Management" in the *Oracle Advanced Security Administrator's Guide* for details about managing CRLs with `orapki`:
>
> `http://docs.oracle.com/cd/E11882_01/network.112/e10746/asossl.htm`

H.2.5.1 About Certificate Validation with Certificate Revocation Lists

The process of determining whether a given certificate can be used in a given context is referred to as certificate validation. Certificate validation includes determining that:

- A trusted certificate authority (CA) has digitally signed the certificate.
- The certificate's digital signature corresponds to the independently-calculated hash value of the certificate itself and the certificate signer's (CA's) public key.
- The certificate has not expired.
- The certificate has not been revoked.

The SSL network layer automatically performs the first three validation checks, but you must configure certificate revocation list (CRL) checking to ensure that certificates have not been revoked. CRLs are signed data structures that contain a list of revoked certificates. They are usually issued and signed by the same entity who issued the original certificate.

H.2.5.1.1 What CRLs Should You Use? You should have CRLs for all of the trust points that you honor. The trust points are the trusted certificates from a third-party identity that is qualified with a level of trust. Typically, the certificate authorities you trust are called trust points.

H.2.5.1.2 How CRL Checking Works Certificate revocation status is checked against CRLs which are located in file system directories, Oracle Internet Directory, or downloaded from the location specified in the CRL Distribution Point (CRL DP) extension on the certificate. If you store your CRLs on the local file system or in the directory, then you must update them regularly. If you use CRL DPs then CRLs are downloaded when the corresponding certificates are first used.

The server searches for CRLs in the following locations in the order listed. When the system finds a CRL that matches the certificate CA's DN, it stops searching.

1. Local file system

   The system checks the `sqlnet.ora` file for the `SSL_CRL_FILE` parameter first, followed by the `SSL_CRL_PATH` parameter. If these two parameters are not specified, then the system checks the wallet location for any CRLs.

   Note: if you store CRLs on your local file system, then you must use the `orapki` utility to periodically update them. See Section H.2.5.2.1, "Renaming CRLs with a Hash Value for Certificate Validation."

2. Oracle Internet Directory

   If the server cannot locate the CRL on the local file system and directory connection information has been configured in the `ORACLE_HOME/ldap/admin/ldap.ora` file, then the server searches in the directory. It

searches the CRL subtree by using the CA's distinguished name (DN) and the DN of the CRL subtree.

The server must have a properly configured `ldap.ora` file to search for CRLs in the directory. It cannot use the Domain Name System (DNS) discovery feature of Oracle Internet Directory. Also note that if you store CRLs in the directory, then you must use the `orapki` utility to periodically update them. See Section H.2.5.2.2, "Uploading CRLs to Oracle Internet Directory."

3. CRL DP

   If the CA specifies a location in the CRL DP X.509, version 3, certificate extension when the certificate is issued, then the appropriate CRL that contains revocation information for that certificate is downloaded. Currently, Oracle Advanced Security supports downloading CRLs over HTTP and LDAP.

Notes:

- For performance reasons, only user certificates are checked.
- Oracle recommends that you store CRLs in the directory rather than the local file system.

H.2.5.2 Certificate Revocation List Management

Before you can enable certificate revocation status checking, you must ensure that the CRLs you receive from the CAs you use are in a form (renamed with a hash value) or in a location (uploaded to the directory) in which your system can use them. Oracle Advanced Security provides a command-line utility, `orapki`, that you can use to perform the following tasks:

- Renaming CRLs with a Hash Value for Certificate Validation
- Uploading CRLs to Oracle Internet Directory
- Listing CRLs Stored in Oracle Internet Directory
- Viewing CRLs in Oracle Internet Directory
- Deleting CRLs from Oracle Internet Directory

Note: CRLs must be updated at regular intervals (before they expire) for successful validation. You can automate this task by using `orapki` commands in a script.

You can also use LDAP command-line tools to manage CRLs in Oracle Internet Directory.

> **See Also:** Command-Line Tools Overview in the *Oracle Fusion Middleware Reference for Oracle Identity Management* for information about LDAP command-line tools and their syntax.

H.2.5.2.1 Renaming CRLs with a Hash Value for Certificate Validation

When the system validates a certificate, it must locate the CRL issued by the CA who created the certificate. The system locates the appropriate CRL by matching the issuer name in the certificate with the issuer name in the CRL.

When you specify a CRL storage location for the **Certificate Revocation Lists Path** field in Oracle Net Manager (sets the `SSL_CRL_PATH` parameter in the `sqlnet.ora`

file), use the `orapki` utility to rename CRLs with a hash value that represents the issuer's name. Creating the hash value enables the server to load the CRLs.

On UNIX systems, `orapki` creates a symbolic link to the CRL. On Windows systems, it creates a copy of the CRL file. In either case, the symbolic link or the copy created by `orapki` are named with a hash value of the issuer's name. Then when the system validates a certificate, the same hash function is used to calculate the link (or copy) name so the appropriate CRL can be loaded.

Depending on your operating system, enter one of the following commands to rename CRLs stored in the file system.

To rename CRLs stored in UNIX file systems:

```
orapki crl hash -crl crl_filename [-wallet wallet_location]
-symlink crl_directory [-summary]
```

To rename CRLs stored in Windows file systems:

```
orapki crl hash -crl crl_filename
[-wallet wallet_location] -copy crl_directory [-summary]
```

In the preceding commands, `crl_filename` is the name of the CRL file, *`wallet_location`* is the location of a wallet that contains the certificate of the CA that issued the CRL, and *`crl_directory`* is the directory in which the CRL is located.

Using `-wallet` and `-summary` are optional. Specifying `-wallet` causes the tool to verify the validity of the CRL against the CA's certificate prior to renaming the CRL. Specifying the `-summary` option causes the tool to display the CRL issuer's name.

H.2.5.2.2 Uploading CRLs to Oracle Internet Directory Publishing CRLs in the directory enables CRL validation throughout your enterprise, eliminating the need for individual applications to configure their own CRLs. All applications can use the CRLs stored in the directory in which they can be centrally managed, greatly reducing the administrative overhead of CRL management and use.

The user who uploads CRLs to the directory by using `orapki` must be a member of the directory group `CRLAdmins` (`cn=CRLAdmins,cn=groups,%s_OracleContextDN%`). This is a privileged operation because these CRLs are accessible to the entire enterprise. Contact your directory administrator to be added to this administrative directory group.

To upload CRLs to the directory, enter the following at the command line:

```
orapki crl upload -crl crl_location
-ldap hostname:ssl_port -user username [-wallet wallet_location] [-summary]
```

In the preceding command, *`crl_location`* is the file name or URL in which the CRL is located, *`hostname`* and *`ssl_port`* (SSL port with no authentication) are for the system on which your directory is installed, *`username`* is the directory user who has permission to add CRLs to the CRL subtree, and *`wallet_location`* is the location of a wallet that contains the certificate of the CA that issued the CRL.

Using `-wallet` and `-summary` are optional. Specifying `-wallet` causes the tool to verify the validity of the CRL against the CA's certificate prior to uploading it to the directory. Specifying the `-summary` option causes the tool to print the CRL issuer's name and the LDAP entry in which the CRL is stored in the directory.

Note:

- The `orapki` utility will prompt you for the directory password when you perform this operation.
- Ensure that you specify the directory SSL port on which the Diffie-Hellman-based SSL server is running. This is the SSL port that does not perform authentication. Neither the server authentication nor the mutual authentication SSL ports are supported by the `orapki` utility.

H.2.5.2.3 Listing CRLs Stored in Oracle Internet Directory

You can display a list of all CRLs stored in the directory with `orapki`, which is useful for browsing to locate a particular CRL to view or download to your local system. This command displays the CA who issued the CRL (Issuer) and its location (DN) in the CRL subtree of your directory.

To list CRLs in Oracle Internet Directory, enter the following at the command line:

```
orapki crl list -ldap hostname:ssl_port
```

In the preceding command, the *`hostname`* and *`ssl_port`* are for the system on which your directory is installed. Note that this is the directory SSL port with no authentication as described in the preceding section.

H.2.5.2.4 Viewing CRLs in Oracle Internet Directory

You can view specific CRLs that are stored in Oracle Internet Directory in a summarized format or you can request a complete listing of revoked certificates for the specified CRL. A summary listing provides the CRL issuer's name and its validity period. A complete listing provides a list of all revoked certificates contained in the CRL.

To view a summary listing of a CRL in Oracle Internet Directory, enter the following at the command line:

```
orapki crl display -crl crl_location [-wallet wallet_location] -summary
```

In the preceding command, *`crl_location`* is the location of the CRL in the directory. It is convenient to paste the CRL location from the list that displays when you use the `orapki crl list` command. See "Section H.2.5.2.3, "Listing CRLs Stored in Oracle Internet Directory".

To view a list of all revoked certificates contained in a specified CRL, which is stored in Oracle Internet Directory, enter the following at the command line:

```
orapki crl display -crl crl_location [-wallet wallet_location] -complete
```

For example, the following `orapki` command:

```
orapki crl display -crl $T_WORK/pki/wlt_crl/nzcrl.txt -wallet $T_WORK/pki/wlt_crl
-complete
```

produces the following output, which lists the CRL issuer's DN, its publication date, date of its next update, and the revoked certificates it contains:

```
issuer = CN=root,C=us, thisUpdate = Sun Nov 16 10:56:58 PST 2003,
nextUpdate = Mon Sep 30 11:56:58 PDT 2013, revokedCertificates =
{(serialNo = 153328337133459399575438325845117876415,
revocationDate - Sun Nov 16 10:56:58 PST 2003)}
CRL is valid
```

Using the `-wallet` option causes the `orapki crl display` command to validate the CRL against the CA's certificate.

Depending on the size of your CRL, choosing the `-complete` option may take a long time to display.

You can also use Oracle Directory Manager, a graphical user interface tool that is provided with Oracle Internet Directory, to view CRLs in the directory. CRLs are stored in the following directory location:

```
cn=CRLValidation,cn=Validation,cn=PKI,cn=Products,cn=OracleContext
```

H.2.5.2.5 Deleting CRLs from Oracle Internet Directory

The user who deletes CRLs from the directory by using `orapki` must be a member of the directory group `CRLAdmins`. See Section H.2.5.2.2, "Uploading CRLs to Oracle Internet Directory" for information about this directory administrative group.

To delete CRLs from the directory, enter the following at the command line:

```
orapki crl delete -issuer issuer_name -ldap hostname:ssl_port
-user username [-summary]
```

In the preceding command, *`issuer_name`* is the name of the CA who issued the CRL, the *`hostname`* and *`ssl_port`* are for the system on which your directory is installed, and *`username`* is the directory user who has permission to delete CRLs from the CRL subtree. Note that this must be a directory SSL port with no authentication. See Section H.2.5.2.2, "Uploading CRLs to Oracle Internet Directory" for more information about this port.

Using the `-summary` option causes the tool to print the CRL LDAP entry that was deleted.

For example, the following `orapki` command:

```
orapki crl delete -issuer "CN=root,C=us"
-ldap machine1:3500 -user cn=orcladmin -summary
```

produces the following output, which lists the location of the deleted CRL in the directory:

```
Deleted CRL at cn=root
cd45860c.rN,cn=CRLValidation,cn=Validation,cn=PKI,cn=Products,cn=OracleContext
```

H.2.6 orapki Utility Commands Summary

This section lists and describes the following `orapki` commands:

- orapki cert create
- orapki cert display
- orapki crl create
- orapki crl delete
- orapki crl display
- orapki crl hash
- orapki crl list
- orapki crl revoke
- orapki crl status

- orapki crl upload
- orapki crl verify
- orapki wallet add
- orapki wallet change_pwd
- orapki wallet create
- orapki wallet display
- orapki wallet export
- orapki wallet export_trust_chain

H.2.6.1 orapki cert create

The following sections describe this command.

H.2.6.1.1 Purpose Use this command to create a signed certificate for testing purposes.

H.2.6.1.2 Syntax
```
orapki cert create [-wallet wallet_location]
-request certificate_request_location
-cert certificate_location -validity number_of_days [-summary]
```

- The `-wallet` parameter specifies the wallet containing the user certificate and private key that will be used to sign the certificate request.
- The `-request` parameter (mandatory) specifies the location of the certificate request for the certificate you are creating.
- The `-cert` parameter (mandatory) specifies the directory location in which the tool places the new signed certificate.
- The `-validity` parameter (mandatory) specifies the number of days, starting from the current date, that this certificate will be valid.

H.2.6.2 orapki cert display

The following sections describe this command.

H.2.6.2.1 Purpose Use this command to display details of a specific certificate.

H.2.6.2.2 Syntax
```
orapki cert display -cert certificate_location
[-summary|-complete]
```

- The `-cert` parameter specifies the location of the certificate you want to display.
- You can use either the `-summary` or the `-complete` parameter to display the following information:
  - `-summary` displays the certificate and its expiration date
  - `-complete` displays additional certificate information, including the serial number and public key

H.2.6.3 orapki crl create

The following sections describe this command.

H.2.6.3.1 Purpose Use this command to create a CRL.

H.2.6.3.2 Syntax
```
orapki crl create [-crl [url|filename]]
```

```
[-wallet [cawallet]]
[-nextupdate [days]]
[-pwd pwd]
```

- -crl is the location where the CRL will be created (for example `./nzcrl.txt`)
- -wallet is the cawallet, which contains self-signed certificate and corresponding private key
- -nextupdate is the number of days until the next update
- -pwd is the password of cawallet

H.2.6.4 orapki crl delete

The following sections describe this command.

H.2.6.4.1 Purpose Use this command to delete CRLs from Oracle Internet Directory. Note that the user who deletes CRLs from the directory by using `orapki` must be a member of the `CRLAdmins` (`cn=CRLAdmins,cn=groups,%s_OracleContextDN%`) directory group.

H.2.6.4.2 Syntax `orapki crl delete -issuer` *`issuer_name`*
`-ldap` *`hostname:ssl_port`* `-user` *`username`* `[-summary]`

- The `-issuer` parameter specifies the name of the certificate authority (CA) who issued the CRL.
- The `-ldap` parameter specifies the hostname and SSL port for the directory in which the CRLs are to be deleted. Note that this must be a directory SSL port with no authentication. See Section H.2.5.2.2, "Uploading CRLs to Oracle Internet Directory" for more information about this port.
- The `-user` parameter specifies the username of the directory user who has permission to delete CRLs from the CRL subtree in the directory.
- The `-summary` parameter is optional. Using it causes the tool to print the CRL LDAP entry that was deleted.

H.2.6.5 orapki crl display

The following sections describe this command.

H.2.6.5.1 Purpose Use this command to display specific CRLs that are stored in Oracle Internet Directory.

H.2.6.5.2 Syntax `orapki crl display -crl` *`crl_location`*
`[-wallet` *`wallet_location`*`] [-summary|-complete]`

- The `-crl` parameter specifies the location of the CRL in the directory. It is convenient to paste the CRL location from the list that displays when you use the `orapki crl list` command. See Section H.2.6.7, "orapki crl list".
- The `-wallet` parameter (optional) specifies the location of the wallet that contains the certificate of the certificate authority (CA) who issued the CRL. Using it causes the tool to verify the validity of the CRL against the CA's certificate prior to displaying it.
- Choosing either the -summary or the -complete parameters displays the following information:

- `-summary` provides a listing that contains the CRL issuer's name and the CRL's validity period
- `-complete` provides a list of all revoked certificates that the CRL contains. Note that this option may take a long time to display, depending on the size of the CRL.

H.2.6.6 orapki crl hash

The following sections describe this command.

H.2.6.6.1 Purpose Use this command to generate a hash value of the certificate revocation list (CRL) issuer to identify the location of the CRL in your file system for certificate validation.

H.2.6.6.2 Syntax
```
orapki crl hash -crl crl_filename|URL
 [-wallet wallet_location] [-symlink|-copy] crl_directory [-summary]
```

- The `-crl` parameter specifies the filename that contains the CRL or the URL in which it can be found.
- The `-wallet` parameter (optional) specifies the location of the wallet that contains the certificate of the certificate authority (CA) who issued the CRL. Using it causes the tool to verify the validity of the CRL against the CA's certificate prior to uploading it to the directory.
- Depending on your operating system, use either the `-symlink` or the `-copy` parameter:
  - On UNIX: Use `-symlink` to create a symbolic link to the CRL at the *`crl_directory`* location
  - On Windows: Use `-copy` to create a copy of the CRL at the *`crl_directory`* location
- The `-summary` parameter (optional) causes the tool to display the CRL issuer's name.

H.2.6.7 orapki crl list

The following sections describe this command.

H.2.6.7.1 Purpose Use this command to display a list of CRLs stored in Oracle Internet Directory. This is useful for browsing to locate a particular CRL to view or download to your local file system.

H.2.6.7.2 Syntax
```
orapki crl list -ldap hostname:ssl_port
```

The `-ldap` parameter specifies the hostname and SSL port for the directory server from which you want to list CRLs. Note that this must be a directory SSL port with no authentication. See Section H.2.5.2.2, "Uploading CRLs to Oracle Internet Directory" for more information about this port.

H.2.6.8 orapki crl revoke

The following sections describe this command.

H.2.6.8.1 Purpose Use this command to revoke a certificate.

H.2.6.8.2 Syntax
```
orapki crl revoke [-crl [url|filename]]
```

```
[-wallet [cawallet]]
[-cert [revokecert]]
[-pwd pwd]
```

where:

- -crl specifies the CRL as either a URL or a filename
- -wallet is the cawallet, which contains self-signed certificate and corresponding private key
- -cert: certificate to be revoked
- -pwd is the password of cawallet.

H.2.6.9 orapki crl status

The following sections describe this command.

H.2.6.9.1 Purpose Use this command to check if a certificate is revoked in a CRL.

H.2.6.9.2 Syntax `orapki crl status [-crl [url|filename]] [-cert [cert]]`

- -crl specifies the CRL as either a URL or a filename
- -cert is the CA's certificate

H.2.6.10 orapki crl upload

The following sections describe this command.

H.2.6.10.1 Purpose Use this command to upload certificate revocation lists (CRLs) to the CRL subtree in Oracle Internet Directory. Note that you must be a member of the directory administrative group `CRLAdmins` (`cn=CRLAdmins,cn=groups,%s_OracleContextDN%`) to upload CRLs to the directory.

H.2.6.10.2 Syntax `orapki crl upload -crl crl_location -ldap hostname:ssl_port -user username [-wallet wallet_location] [-summary]`

- The `-crl` parameter specifies the directory location or the URL of the CRL that you are uploading to the directory.
- The `-ldap` parameter specifies the hostname and SSL port for the directory to which you are uploading the CRLs. Note that this must be a directory SSL port with no authentication. See Section H.2.5.2.2, "Uploading CRLs to Oracle Internet Directory" for more information about this port.
- The `-user` parameter specifies the username of the directory user who has permission to add CRLs to the CRL subtree in the directory.
- The `-wallet` parameter specifies the location of the wallet that contains the certificate of the certificate authority (CA) who issued the CRL. This is an optional parameter. Using it causes the tool to verify the validity of the CRL against the CA's certificate prior to uploading it to the directory.
- The `-summary` parameter is also optional. Using it causes the tool to display the CRL issuer's name and the LDAP entry in which the CRL is stored in the directory.

H.2.6.11 orapki crl verify

The following sections describe this command.

H.2.6.11.1 Purpose Use this command to verify a CRL signature.

H.2.6.11.2 Syntax

```
orapki crl verify [-crl [url|filename]]
[-cert [cacert]]
```

where:

- -crl specifies the CRL as either a URL or a filename
- -cert specifies the certificate to be checked

H.2.6.12 orapki wallet add

The following sections describe this command.

H.2.6.12.1 Purpose Use this command to add certificate requests and certificates to an Oracle wallet.

H.2.6.12.2 Syntax To add certificate requests:

```
orapki wallet add -wallet wallet_location -dn user_dn -keysize 512|1024|2048
```

- The `-wallet` parameter specifies the location of the wallet to which you want to add a certificate request.
- The `-dn` parameter specifies the distinguished name of the certificate owner.
- The `-keysize` parameter specifies the key size for the certificate.
- To sign the request, export it with the export option. See Section H.2.6.16, "orapki wallet export".

To add trusted certificates:

```
orapki wallet add -wallet wallet_location -trusted_cert -cert certificate_location
```

- The `-trusted_cert` parameter causes the tool to add the trusted certificate, at the location specified with `-cert`, to the wallet.

To add root certificates:

```
orapki wallet add -wallet wallet_location -dn
certificate_dn -keysize 512|1024|2048 -self_signed
-valid_from [mm/dd/yyyy] -valid_until [mm/dd/yyyy]
-validity number_of_days
```

- The `-self_signed` parameter causes the tool to create a root certificate.
- The `-validity` parameter can be used to specify the number of days, starting from the current date, that this root certificate will be valid.
- The `-valid_from` and `valid_until` parameters can be used to specify an exact date range for which this root certificate will be valid. You may specify validity in this way instead of `-validity` *number_of_days*.

To add user certificates:

```
orapki wallet add -wallet wallet_location -user_cert -cert certificate_location
```

- The `-user_cert` parameter causes the tool to add the user certificate at the location specified with the `-cert` parameter to the wallet. Before you add a user certificate to a wallet, you must add all the trusted certificates that make up the certificate chain. If all trusted certificates are not installed in the wallet before you add the user certificate, then adding the user certificate will fail.

To add a subject key identifier extension to a certificate request:

```
orapki wallet add -wallet wallet_location -dn user_dn -keysize 512|1024|2048
-addext_ski
```

To add a Version 3 self-signed certificate to a wallet:

```
orapki wallet add -wallet wallet_location -dn certificate_dn -keysize
512|1024|2048 -self_signed -validity number_of_days -addext_ski
```

H.2.6.13 orapki wallet change_pwd

The following sections describe this command.

H.2.6.13.1 Purpose Use this command to change the password for an Oracle wallet.

H.2.6.13.2 Syntax `orapki wallet change_pwd [-wallet [wallet_location]] [-oldpwd oldpassword] [-newpwd newpassword]`

- The `-wallet` parameter specifies the location of the wallet whose password you want to change.
- The `-oldpwd` parameter specifies the existing wallet password.
- The `-newpwd` parameter specifies the new wallet password.

H.2.6.14 orapki wallet create

The following sections describe this command.

H.2.6.14.1 Purpose Use this command to create an Oracle wallet or to set auto-login on for an Oracle wallet.

H.2.6.14.2 Syntax `orapki wallet create -wallet wallet_location [-auto_login]`

- The `-wallet` parameter specifies a location for the new wallet or the location of the wallet for which you want to turn on auto-login.
- The `-auto_login` parameter creates an auto-login wallet, or it turns on automatic login for the wallet specified with the `-wallet` option.

H.2.6.15 orapki wallet display

The following sections describe this command.

H.2.6.15.1 Purpose Use this command to view the certificate requests, user certificates, and trusted certificates in an Oracle wallet.

H.2.6.15.2 Syntax `orapki wallet display -wallet wallet_location`

- The `-wallet` parameter specifies a location for the wallet you want to open if it is not located in the current working directory.

H.2.6.16 orapki wallet export

The following sections describe this command.

H.2.6.16.1 Purpose Use this command to export certificate requests and certificates from an Oracle wallet.

H.2.6.16.2 Syntax `orapki wallet export -wallet` *`wallet_location`* `-dn` *`certificate_dn`* `-cert` *`certificate_filename`*

- The `-wallet` parameter specifies the directory where the wallet, from which you want to export the certificate, is located.
- The `-dn` parameter specifies the distinguished name of the certificate.
- The `-cert` parameter specifies the path and filename of the file that contains the exported certificate.

To export a certificate request from an Oracle wallet:

```
orapki wallet export -wallet wallet_location -dn
certificate_request_dn -request certificate_request_filename
```

- The `-request` parameter specifies the path and filename of the file that contains the exported certificate request.

H.2.6.17 orapki wallet export_trust_chain

The following sections describe this command.

H.2.6.17.1 Purpose Use this command to export a chain of trust (certificate chain) for a user.

H.2.6.17.2 Syntax

```
orapki wallet export_trust_chain [-wallet [wallet]]
[-certchain [filename]]
[-dn [user_cert_dn] ]
[-pwd pwd]
```

- The `-wallet` parameter specifies the location of the wallet from which you want to export the certificate chain.
- The `-certchain` parameter specifies the name of the file to contain the exported certificate chain.
- The `-dn` parameter specifies the distinguished name of the entry to be exported.
- The -pwd specifies the wallet password.

H.3 Equivalent Features for Oracle Wallet Manager

Table H–1 shows the wallet management features provided by Oracle Wallet Manager, and the commands or options that provide equivalent functionality in 11*g* Release 2 (11.1.2).

Table H–1 Mapping for Oracle Wallet Manager Features for Wallets

Oracle Wallet Manager Feature	How Implemented in 11gR1 Fusion Middleware Control	Notes
Creating a standard PKCS #12 wallet	**Security**, then **Wallets**	
Creating a PKCS#11 wallet	Not supported	Use Oracle Wallet Manager or the orapki command line tool
Opening a wallet	**Security**, then **Wallets**	Click on the wallet and enter a password, unless it is an auto-login wallet
Closing a wallet		Navigating to the wallets page, or opening another wallet, automatically closes the existing wallet.
Uploading a wallet to an LDAP directory	Not supported	Use the orapki command line tool
Downloading a wallet from an LDAP directory	Not supported	Use the orapki command line tool
Saving changes to an open wallet	See Notes.	Any changes made on the Manage Certificate page are automatically saved when the operation is completed.
Saving the open wallet to a new location	**Security**, then **Wallets**, then **Export**	
Saving in System Default	**Security**, then **Wallets**, then **Export**	
Deleting the wallet	**Security**, then **Wallets**, then **Delete**	
Changing the password	Not supported	Use WLST or orapki command line tools.
Enabling auto-login	See Notes.	An Auto-login wallet is automatically created with every password protected wallet.
Disabling auto-login	Not supported	You cannot disable generation of an auto-login wallet since it is always required for runtime.

Table H–2 shows the certificate management features provided by Oracle Wallet Manager, and the equivalent commands or options in 11*g* Release 2 (11.1.2).

Table H–2 Mapping for Oracle Wallet Manager Features for Certificates

Oracle Wallet Manager Feature	How Implemented in 11gR1 Fusion Middleware Control	Notes
Adding a certificate request	**Security**, then **Wallets**. Select a wallet, then **Add Certificate Request**	
Importing a user certificate	**Security**, then **Wallets**, select a wallet, then **Import**	Select User Certificate in the drop down box
Importing a trusted certificate	**Security**, then **Wallets**, select a wallet, then **Import**	Select Trusted Certificate in the drop down box
Remove certificate request	**Security**, then **Wallets**, select a wallet, select a certificate request, then **Delete**	
Remove user certificate	**Security**, then **Wallets**, select a wallet, select a user certificate, then **Delete**	
Remove trusted certificate	**Security**, then **Wallets**, select a wallet, select a trusted certificate, then **Delete**	
Export user certificate	**Security**, then **Wallets**, select a wallet, select a user certificate, then **Export**	
Export certificate request	**Security**, then **Wallets**, select a wallet, select a certificate request, then **Export**	
Export trusted certificate	**Security**, then **Wallets**, select a wallet, select a trusted certificate, then **Export**	

Table H–2 (Cont.) Mapping for Oracle Wallet Manager Features for Certificates

Oracle Wallet Manager Feature	How Implemented in 11gR1 Fusion Middleware Control	Notes
Export all trusted certificates	Not supported	Use WLST or `orapki` command-line tools
Importing a PKCS#7 certificate chain into the wallet	Not supported	Use WLST or `orapki` command-line tools
Exporting a PKCS#7 certificate chain from the wallet	Not supported	Use WLST or `orapki` command-line tools

Location of Default Wallet

The default location of the wallet depends on the ORACLE_HOME setting:

- When `ORACLE_HOME` is set, the default wallet location is `$ORACLE_HOME/owm/wallets/`*`username`*.
- When `ORACLE_HOME` is not set, the default wallet location is *`CurrentDir`*`/owm/wallets/`*`username`*.

H.4 Equivalent Features for orapki

Table H–3 shows the features provided by the `orapki` utility for Oracle wallets and CRLs, and the equivalent commands and options in 11*g* Release 2 (11.1.2).

Table H–3 Mapping for orapki Features for Wallets and CRLs

orapki Feature	How Implemented in 11gR1	Notes
Creating a standard PKCS#12 wallet	WLST createWallet()	To manage a password-protected and auto-login wallet, provide a non-empty password value. To manage just an auto-login wallet, provide an empty password value (that is, '')
Creating a PKCS#11 wallet	Not supported	Use orapki command-line tool
Uploading a wallet to an LDAP Directory	Not supported	Use orapki command-line tool
Downloading a wallet from an LDAP directory	Not supported	Use orapki command-line tool
Deleting a wallet	WLST deleteWallet()	
Changing the wallet password	WLST changeWalletPassword()	For obvious reasons, password can only be changed for a password-protected wallet
Enabling auto-login		Auto-login wallet is automatically created with every password-protected wallet.
Enabling auto-login wallet that works only on local machine	Not supported	Use orapki command line tool
Create, revoke, hash, verify, upload, list, display, delete CRLs	Not supported	Use orapki command line tool

Table H–4 shows the features provided by the `orapki` utility for certificates, and the equivalent commands or options in 11*g* Release 2 (11.1.2).

Table H–4 Mapping for orapki Features for Certificates

orapki Feature	How Implemented in WLST in 11gR1	Notes
Adding a certificate request	addCertificateRequest()	
Adding a self-signed certificate	addSelfSignedCertificate()	
Listing all entries in a wallet	listWalletObjects()	Provide a valid value of type ("CertificateRequest", "Certificate" or "TrustedCertificate")
Importing a user certificate	importWalletObject()	Enter type as "Certificate"
Importing a trusted certificate	importWalletObject()	Enter type as "TrustedCertificate"
Removing a certificate request	removeWalletObject()	Enter type as "CertificateRequest"
Removing a user certificate	removeWalletObject()	Enter type as "Certificate"
Removing a trusted certificate	removeWalletObject()	Enter type as "TrustedCertificate"
Removing all trusted certificates	removeWalletObject()	Enter type as "TrustedAll"
Exporting a user certificate	exportKeyStoreObject()	Enter type as "Certificate"
Exporting a certificate request	exportWalletObject()	Enter type as "CertificateRequest"
Exporting a trusted certificate	exportWalletObject()	Enter type as "TrustedCertificate"
Exporting a certificate chain	exportWalletObject()	Enter type as "CertificateChain"
Importing a PKCS#7 certificate chain into the wallet	importWalletObject()	Enter type as "TrustedChain"

H.5 Equivalent Features for the SSL Configuration Tool

Table H–5 shows the features provided by the pre-11*g* Release 2 (11.1.2) SSL Configuration Tool, and the equivalent commands or options in 11*g* Release 2 (11.1.2).

Table H–5 Equivalent Features for the SSL Configuration Tool

SSL Configuration Tool	SSL Configuration in 11*g* Release 2 (11.1.2)
No support for wallet management	Supports management of Oracle Wallets and Java Keystores, in addition to SSL configuration
Oracle Web Cache was the only standalone type supported for SSL	Oracle HTTP Server, Oracle Web Cache, Oracle Internet Directory, and Oracle Virtual Directory are supported for standalone SSL configuration
Provided only command line interface	Provides both command line interface (WLST) and graphical interface (Fusion Middleware Control)
Configuration file was required to run this tool. If the file was not provided, the tool prompted for values.	Configuration file is optional in the WLST command. If not provided, default values are used for SSL attributes.
Supported SSL configuration for Web tier only.	Supports SSL configuration for both Web tier and data tier.
Tool had to be run on the same physical host where component was installed.	Allows remote management of components.

Troubleshooting Oracle Fusion Middleware

This appendix provides information on how to troubleshoot problems that you might encounter when using Oracle Fusion Middleware. It contains the following topics:

- Diagnosing Oracle Fusion Middleware Problems
- Common Problems and Solutions
- Troubleshooting Fusion Middleware Control
- Troubleshooting SSL
- Need More Help?

I.1 Diagnosing Oracle Fusion Middleware Problems

Oracle Fusion Middleware components generate log files containing messages that record all types of events, including startup and shutdown information, errors, warning messages, access information on HTTP requests, and additional information. The log files can be used to identify and diagnose problems. See Chapter 12, "Managing Log Files and Diagnostic Data" for more information about using and reading log files.

Oracle Fusion Middleware includes a Diagnostic Framework which aids in detecting, diagnosing, and resolving problems. The problems that are targeted in particular are critical errors such as those caused by code bugs, metadata corruption, and customer data corruption, deadlocked threads, and inconsistent state.

When a critical error occurs, it is assigned an incident number, and diagnostic data for the error (such as log files) are immediately captured and tagged with this number. The data is then stored in the Automatic Diagnostic Repository (ADR), where it can later be retrieved by incident number and analyzed. See Chapter 13, "Diagnosing Problems" for more information about the Diagnostic Framework.

I.2 Common Problems and Solutions

This section describes common problems and solutions. It contains the following topics:

- Running out of Data Source Connections
- Using a Different Version of Spring
- ClassNotFound Errors When Starting Managed Servers

I.2.1 Running out of Data Source Connections

If the database performance has slowed or you receive the following message in the Oracle WebLogic Server log files, you may have leaks in the data source connections:

```
No resources currently available in pool datasource name

Any product functionality that depend on the datasource will not function as it
can't connect database to get required data.
```

If you receive this message, monitor the connection usage from the Administration Console data source monitoring page:

1. From Domain Structure, expand **Services,** then **Data Sources.**
2. Click the data source that you want to monitor.
3. Select the Monitoring tab, then the Statistics tab.
4. If the table does not display **Active Connection Current Count,** click **Customize this table.**
5. In Column Display, select **Active Connection Current Count** and move it from the Available to the Chosen box. Click **Apply.**
6. In the table, note the number in the **Active Connection Current Count** column.

If the active current count for a data source keeps increasing and does not go down, this data source is leaking connections. Contact Oracle Support.

I.2.2 Using a Different Version of Spring

When you configure a Managed Server with JRF, Spring 2.0.6 is installed and is placed in the Oracle WebLogic Server system classpath. If a custom application running in a JRF environment requires a different version of Spring, you must use the Filtering ClassLoader mechanism to specify the version of Spring.

Oracle WebLogic Server provides the FilteringClassLoader mechanism so that you can configure deployment descriptors to explicitly specify that certain packages should always be loaded from the application, rather than being loaded by the system classloader. This allows you to use alternate versions of applications such as Spring or Ant.

For more information about using the FilteringClassLoader mechanism, see "Using a Filtering ClassLoader" in the *Oracle Fusion Middleware Developing Applications for Oracle WebLogic Server*.

I.2.3 ClassNotFound Errors When Starting Managed Servers

If a Managed Server is started by Node Manager (as is the case when the servers are started by the Oracle WebLogic Server Administration Console or Fusion Middleware Control), you may receive a ClassNotFound error if Node Manager has not been configured to use the start scripts when starting Managed Servers. See Section 4.2.1 for information about resolving this problem.

I.3 Troubleshooting Fusion Middleware Control

The following sections describe problems and issues when using Fusion Middleware Control:

I.3.1 Troubleshooting the Display of Performance Metrics and Charts in Fusion Middleware Control

If you are using Fusion Middleware Control to manage system components, then you might encounter situations where the performance metrics and charts do not display properly for certain managed targets.

The following sections provide information about managed targets and describe some common troubleshooting tasks to perform if Fusion Middleware Control displays errors when attempting to display performance metrics, such as response time and load metrics:

I.3.1.1 What Are Agent-Monitored Targets?

To discover and view the following components with Fusion Middleware Control, an Oracle Management Agent must be available and running:

- Oracle Internet Directory
- Oracle Virtual Directory
- Oracle Directory Integration Platform
- Oracle Access Management Identity Federation
- Oracle Reports Application, Oracle Reports Server

These components can be referred to as agent-monitored targets.

When you install and configure an Oracle Fusion Middleware environment that includes these components, a management agent, Oracle Management Agent, is also installed and running in the Oracle instance.

In contrast, Java components and some system components can be managed by Fusion Middleware Control without a management agent.

For more information about the Oracle Management Agent, refer to the Oracle Enterprise Manager documentation on the Oracle Technology Network (OTN):

`http://www.oracle.com/technetwork/oem/grid-control/documentation/index.html`

I.3.1.2 Setting Monitoring Credentials for All Agent-Monitored Targets in a Farm

To make it easier to manage the monitoring credentials for all of your agent-monitored targets, you can use the Monitoring Credentials page to set the monitoring credentials for all of the agent-monitored targets in the farm:

1. From the **Farm** menu, select **Monitoring Credentials**.

2. Enter the user name and password of an Oracle WebLogic Server user account that has at least the `monitoring` level of privileges.

When you set the monitoring credentials on this page, you override all the monitoring credentials for the agent-monitored targets in the farm. However, after you set the monitoring credentials for all the targets, you can override the credentials for a specific target by using the Agent-Monitored Targets page, as described in Section I.3.1.3.

I.3.1.3 Changing the Monitoring Credentials for a Specific Agent-Monitored Target

To manage a target (an Oracle Fusion Middleware component), the Oracle Management Agent uses an Oracle WebLogic Server administration account to connect to the target. After it connects to the target, the Oracle Management Agent can gather performance metrics and send them back to the Fusion Middleware Control where they appear on monitoring pages and in performance charts.

This administration account and its password are called the monitoring credentials for an agent-monitored target.

If the monitoring credentials for a particular target are changed in Oracle WebLogic Server, then the Oracle Management Agent can no longer obtain the performance metrics. As a result, no metrics for the target appear on the Fusion Middleware Control pages and the performance charts are not rendered.

To fix this problem, you can modify the monitoring credentials of the Agent-Monitored target in Fusion Middleware Control:

1. From the **Farm** menu, select **Monitoring Credentials**.

   The Monitoring Credentials page is displayed.

2. Click **Agent-Monitored Targets.**

   The Agent-Monitored Targets page is displayed.

3. Click the Configure icon for the target that you need to modify.

4. On the Configuration page, locate the monitoring credentials fields and change the credentials to match those of an Oracle WebLogic Server user account that has at least the `monitoring` level of privileges.

I.3.1.4 Verifying or Changing the Oracle Management Agent URL

If the performance metrics for all of the agent-monitored targets in the farm are unavailable, and you have verified that the monitoring credentials for the agent-monitored targets are correct, then you might have to modify the URL used by the Oracle Management Agent to communicate with Fusion Middleware Control.

This situation can occur if you have backed up your environment and restored it to another host, or if you have moved your test environment to a production environment. In either case, the host name required in the Oracle Management Agent URL must be changed before the Oracle Management Agent can once again communicate with Fusion Middleware Control.

To modify the Oracle Management Agent URL:

1. From the **Farm** menu, select **Monitoring Credentials**.

   The Monitoring Credentials page is displayed.

2. Click **Agent-Monitored Targets.**

   The Agent-Monitored Targets page is displayed.

3. Click the Configure icon for one of the agent-monitored targets listed on the page.

4. Change the Oracle Management Agent URL.

I.3.2 Securing the Connection from Fusion Middleware Control to Oracle WebLogic Server Administration Console

By default, if you access Oracle WebLogic Server Administration Console from Fusion Middleware Control, the connection is a non-SSL connection. To access the Oracle WebLogic Server Administration Console using an SSL connection, you need to access it manually using the SSL port. Alternatively, you can enable a secure Administration port.

See "Understanding Network Channels" in the *Oracle Fusion Middleware Configuring Server Environments for Oracle WebLogic Server* for information about the admin channel and how to establish a channel.

To enable a secure mode of communication with the Administration Server domain and to disable all other non-secure modes, you may need to perform the following explicit steps to enable Oracle Management Agent to monitor agent-monitored targets in Fusion Middleware Control. (See Section I.3.1.1 for information about agent-monitored targets.) These steps are needed only if you are using the default self-signed certificates on the Administration Server instance or other signed certificates whose Certification Authorities (CAs) are not available in the default trust store of the JVM used by Oracle Management Agent.

In this case, take the following steps:

1. Stop the Oracle Management Agent using the following command:

   ```
   ORACLE_HOME/bin/emctl stop agent
   ```

2. Export the certificate from Oracle WebLogic Server:

   ```
   JAVA_HOME/jdk/bin/keytool -export -alias demoidentity -file /tmp/wlcert
   -keystore MW_HOME/wlserver_10.3/server/lib/DemoIdentity.jks
   ```

   When prompted, enter the password.

3. Update the JDKs default trust store (*JAVA_HOME*/jre/lib/security/cacerts) with the certificate. (This is the JDK being used by Oracle Management Agent.)

   ```
   keytool -import -alias demoidentity -trustcacerts -file /temp/wlcert -keystore
   JAVA_HOME/jre/lib/security/cacerts -storepass password
   ```

   When asked if you trust this certificate, enter `yes`.

4. Start the Oracle Management Agent using the following command:

   ```
   ORACLE_HOME/bin/emctl start agent
   ```

I.4 Troubleshooting SSL

This section describes common problems and solutions when working with SSL configuration. It contains the following topic:

- Components May Enable All Supported Ciphers

I.4.1 Components May Enable All Supported Ciphers

You should be aware that when no cipher is explicitly configured, some 11*g* Release 2 (11.1.2) components enable all supported SSL ciphers including `DH_Anon` (Diffie-Hellman Anonymous) ciphers.

At this time, Oracle HTTP Server is the only component known to set ciphers like this.

Configure the components with the desired cipher(s) if `DH_Anon` is not wanted.

I.5 Need More Help?

You can find more solutions on My Oracle Support, `http://support.oracle.com`. If you do not find a solution for your problem, log a service request.

You can also use the Remote Diagnostic Agent, as described in Section I.5.1.

> **See Also:** *Oracle Fusion Middleware Release Notes*, available on the Oracle Technology Network:
>
> `http://docs.oracle.com/index.html#middleware`

I.5.1 Using Remote Diagnostic Agent

Remote Diagnostic Agent (RDA) is a command-line diagnostic tool that provides a comprehensive picture of your environment. Additionally, RDA can provide recommendations on various topics, for example configuration and security. This aids you and Oracle Support in resolving issues.

RDA is designed to be as unobtrusive as possible; it does not modify systems in any way. A security filter is provided if required.

For more information about RDA, see the readme file, which is located at:

```
(UNIX) ORACLE_HOME/rda/README_Unix.txt
(Windows) ORACLE_HOME\rda\README_Windows.txt
```

Index

A

B

C

D

E

F

G

H

I

J

K

L

M

N

O

P

Q

R

S

T

U

V

W