

Oracle® Fusion Middleware

Administrator's Guide

11g Release 1 (11.1.1)

E10105-08

March 2011

Oracle Fusion Middleware Administrator's Guide, 11g Release 1 (11.1.1)

E10105-08

Copyright © 2009, 2011, Oracle and/or its affiliates. All rights reserved.

Primary Author: Helen Grembowicz

Contributing Author: Vinaye Misra

Contributors: Mike Blevins, Nick Fry, Greg Cook, Shalendra Goel, Harry Hsu, Srinu Indla, Pavana Jain, Gopal Kirsur, Kenneth Ma, Dan MacKinnon, Manoj Nayak, Mark Nelson, Praveen Sampath, Sachin Kapur, Sandeep Singh, 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 software or related documentation 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 USA, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

This software 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 which may create a risk of personal injury. If you use this software in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure the safe use of this software. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software in dangerous applications.

Oracle is a registered trademark of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

This software 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

Preface	xxx
Audience	xxx
Documentation Accessibility	xxx
Related Documents	xxxii
Conventions	xxxii
What's New in This Guide?	xxxiii
New and Changed Features for Oracle Fusion Middleware 11g Release 1 (11.1.1.4)	xxxiii
New Features for Oracle Fusion Middleware 11g Release 1 (11.1.1.3)	xxxiii
New Features for Oracle Fusion Middleware 11g Release 1 (11.1.1.2)	xxxiv
New Features for Oracle Fusion Middleware 11g Release 1 (11.1.1)	xxxiv
Part I Understanding Oracle Fusion Middleware	
1 Introduction to Oracle Fusion Middleware	
1.1 What Is Oracle Fusion Middleware?	1-1
1.2 Oracle Fusion Middleware Components	1-1
2 Understanding Oracle Fusion Middleware Concepts	
2.1 Understanding Key Oracle Fusion Middleware Concepts	2-1
2.2 What Is an Oracle WebLogic Server Domain?	2-3
2.2.1 What Is the Administration Server?	2-4
2.2.2 Understanding Managed Servers and Managed Server Clusters	2-4
2.2.3 What Is Node Manager?	2-5
2.3 What Is an Oracle Instance?	2-5
2.4 What Is a Middleware Home?	2-5
2.5 What Is a WebLogic Server Home?	2-6
2.6 What Is an Oracle Home and the Oracle Common Home?	2-6
2.7 What Is the Oracle Metadata Repository?	2-6
Part II Basic Administration	
3 Getting Started Managing Oracle Fusion Middleware	
3.1 Setting Up Environment Variables	3-1

3.2	Overview of Oracle Fusion Middleware Administration Tools	3-4
3.3	Getting Started Using Oracle Enterprise Manager Fusion Middleware Control	3-6
3.3.1	Displaying Fusion Middleware Control	3-6
3.3.2	Using Fusion Middleware Control Help.....	3-7
3.3.3	Navigating Within Fusion Middleware Control.....	3-7
3.3.4	Understanding Users and Roles for Fusion Middleware Control.....	3-10
3.3.5	Viewing and Managing the Farm.....	3-10
3.3.6	Viewing and Managing Components.....	3-11
3.3.7	Viewing the Status of Applications.....	3-13
3.4	Getting Started Using Oracle WebLogic Server Administration Console.....	3-14
3.4.1	Displaying the Oracle WebLogic Server Administration Console	3-14
3.4.2	Locking the WebLogic Server Configuration	3-15
3.5	Getting Started Using Command-Line Tools	3-15
3.5.1	Getting Started Using the Oracle WebLogic Scripting Tool (WLST).....	3-16
3.5.1.1	Using Custom WLST Commands	3-16
3.5.1.2	Using WLST Commands for System Components	3-17
3.5.2	Getting Started Using Oracle Process Manager and Notification Server	3-18
3.6	Getting Started Using the Fusion Middleware Control MBean Browsers	3-19
3.6.1	Using the System MBean Browser	3-19
3.6.2	Using the MBeans for a Selected Application	3-20
3.7	Managing Components.....	3-20
3.8	Changing the Administrative User Password	3-21
3.8.1	Changing the Administrative User Password Using the Command Line.....	3-21
3.8.2	Changing the Administrative User Password Using the Administration Console	3-21
3.9	Basic Tasks for Configuring and Managing Oracle Fusion Middleware	3-22

4 Starting and Stopping Oracle Fusion Middleware

4.1	Overview of Starting and Stopping Procedures.....	4-1
4.2	Starting and Stopping Oracle WebLogic Server Instances	4-1
4.2.1	Starting and Stopping Administration Servers	4-2
4.2.2	Starting and Stopping Managed Servers.....	4-2
4.2.2.1	Starting and Stopping Managed Servers Using Fusion Middleware Control.....	4-2
4.2.2.2	Starting and Stopping Managed Servers Using WLST	4-2
4.2.3	Enabling Servers to Start Without Supplying Credentials	4-3
4.2.4	Configuring Node Manager to Start Managed Servers	4-3
4.3	Starting and Stopping Components	4-4
4.3.1	Starting and Stopping Components Using Fusion Middleware Control.....	4-4
4.3.2	Starting and Stopping Components Using the Command Line.....	4-5
4.4	Starting and Stopping Fusion Middleware Control	4-5
4.5	Starting and Stopping Oracle Management Agent.....	4-5
4.6	Starting and Stopping Applications	4-6
4.6.1	Starting and Stopping Java EE Applications Using Fusion Middleware Control	4-6
4.6.2	Starting and Stopping Java EE Applications Using WLST.....	4-6
4.7	Starting and Stopping Your Oracle Fusion Middleware Environment	4-6
4.7.1	Starting an Oracle Fusion Middleware Environment	4-6
4.7.2	Stopping an Oracle Fusion Middleware Environment	4-7
4.8	Starting and Stopping: Special Topics	4-8

4.8.1	Starting and Stopping in High Availability Environments	4-8
4.8.2	Forcing a Shutdown of Oracle Database	4-9

5 Managing Ports

5.1	About Managing Ports	5-1
5.2	Viewing Port Numbers	5-1
5.2.1	Viewing Port Numbers Using the Command Line	5-1
5.2.2	Viewing Port Numbers Using Fusion Middleware Control	5-2
5.3	Changing the Port Numbers Used by Oracle Fusion Middleware	5-2
5.3.1	Changing the Oracle WebLogic Server Listen Ports	5-3
5.3.1.1	Changing the Oracle WebLogic Server Listen Ports Using the Administration Console	5-3
5.3.1.2	Changing the Oracle WebLogic Server Listen Ports Using WLST.....	5-4
5.3.2	Changing the Oracle HTTP Server Listen Ports.....	5-4
5.3.2.1	Enabling Oracle HTTP Server to Run as Root for Ports Set to Less Than 1024 (UNIX Only)	5-4
5.3.2.2	Changing the Oracle HTTP Server Non-SSL Listen Port	5-5
5.3.2.3	Changing the Oracle HTTP Server SSL Listen Port.....	5-5
5.3.3	Changing Oracle Web Cache Ports	5-7
5.3.4	Changing OPMN Ports (ONS Local, Request, and Remote).....	5-7
5.3.5	Changing Oracle Portal Ports	5-8
5.3.5.1	Changing the Oracle Portal Midtier Port	5-8
5.3.5.2	Changing the Oracle Web Cache Invalidation Port for Oracle Portal	5-8
5.3.5.3	Changing the Oracle Internet Directory Port for Oracle Portal	5-9
5.3.5.4	Changing the PPE Loopback Port	5-9
5.3.5.5	Changing Oracle Portal SQL*Net Listener Port	5-10
5.3.5.6	Restarting WLS_PORTAL Managed Server	5-10
5.3.6	Changing the Oracle Database Net Listener Port	5-10
5.3.6.1	Changing the KEY Value for an IPC Listener	5-14

Part III Secure Sockets Layer

6 Configuring SSL in Oracle Fusion Middleware

6.1	How SSL Works	6-2
6.1.1	What SSL Provides	6-2
6.1.2	About Private and Public Key Cryptography	6-2
6.1.3	Keystores and Wallets.....	6-3
6.1.4	How SSL Sessions Are Conducted	6-4
6.2	About SSL in Oracle Fusion Middleware.....	6-5
6.2.1	SSL in the Oracle Fusion Middleware Architecture	6-6
6.2.2	Keystores and Oracle Wallets	6-7
6.2.3	Authentication Modes.....	6-8
6.2.4	Tools for SSL Configuration.....	6-8
6.3	Configuring SSL for Configuration Tools	6-9
6.3.1	Oracle Enterprise Manager Fusion Middleware Control	6-9
6.3.2	Oracle WebLogic Server Administration Console.....	6-9

6.3.3	WLST Command-Line Tool	6-9
6.4	Configuring SSL for the Web Tier	6-9
6.4.1	Configuring Load Balancers.....	6-9
6.4.2	Enabling SSL for Oracle Web Cache Endpoints.....	6-10
6.4.2.1	Enable Inbound SSL for Oracle Web Cache Using Fusion Middleware Control	6-10
6.4.2.2	Enable Inbound SSL for Oracle Web Cache Using WLST	6-11
6.4.2.3	Enable Outbound SSL for Oracle Web Cache Using Fusion Middleware Control	6-12
6.4.2.4	Specify the Wallet for Outbound SSL from Oracle Web Cache Using WLST ..	6-14
6.4.3	Enabling SSL for Oracle HTTP Server Virtual Hosts	6-14
6.4.3.1	Enable SSL for Inbound Requests to Oracle HTTP Server Virtual Hosts Using Fusion Middleware Control.....	6-14
6.4.3.2	Enable SSL for Inbound Requests to Oracle HTTP Server Virtual Hosts Using WLST	6-16
6.4.3.3	Enable SSL for Outbound Requests from Oracle HTTP Server.....	6-16
6.5	Configuring SSL for the Middle Tier	6-17
6.5.1	Configuring SSL for Oracle WebLogic Server.....	6-18
6.5.1.1	Inbound SSL to Oracle WebLogic Server.....	6-18
6.5.1.2	Outbound SSL from Oracle WebLogic Server.....	6-18
6.5.1.2.1	Outbound SSL from Oracle Platform Security Services to LDAP.....	6-18
6.5.1.2.2	Outbound SSL from Oracle Platform Security Services to Oracle Database.....	6-19
6.5.1.2.3	Outbound SSL from LDAP Authenticator to LDAP	6-19
6.5.1.2.4	Outbound SSL to Database	6-19
6.5.2	Configuring SSL for Oracle SOA Suite	6-20
6.5.3	Configuring SSL for Oracle WebCenter	6-20
6.5.4	Configuring SSL for Oracle Identity and Access Management.....	6-20
6.5.4.1	Configuring SSL for Oracle Directory Integration Platform.....	6-21
6.5.4.2	Configuring SSL for Oracle Identity Federation.....	6-21
6.5.4.3	Configuring SSL for Oracle Directory Services Manager.....	6-21
6.5.5	SSL-Enable Oracle Reports, Forms, Discoverer, and Portal	6-21
6.5.5.1	SSL for Oracle Reports	6-22
6.5.5.2	SSL for Oracle Forms.....	6-22
6.5.5.3	SSL for Oracle Discoverer.....	6-23
6.5.5.4	SSL for Oracle Portal	6-23
6.5.6	Client-Side SSL for Applications	6-23
6.6	Configuring SSL for the Data Tier	6-23
6.6.1	Enabling SSL on Oracle Internet Directory Listeners.....	6-24
6.6.1.1	Enable Inbound SSL on an Oracle Internet Directory Listener Using Fusion Middleware Control.....	6-24
6.6.1.2	Enabling Inbound SSL on an Oracle Internet Directory Listener Using WLST	6-25
6.6.1.3	Enabling Outbound SSL from Oracle Internet Directory to Oracle Database..	6-25
6.6.2	Enabling SSL on Oracle Virtual Directory Listeners	6-26
6.6.2.1	Enable SSL for Oracle Virtual Directory Using Fusion Middleware Control ..	6-26
6.6.2.2	Enabling SSL on an Oracle Virtual Directory Listener Using WLST	6-28
6.6.3	Configuring SSL for the Database	6-29
6.6.3.1	SSL-Enable Oracle Database	6-29

6.6.3.2	SSL-Enable a Data Source	6-31
6.7	Advanced SSL Scenarios	6-32
6.7.1	Hardware Security Modules and Accelerators	6-32
6.7.2	CRL Integration with SSL	6-33
6.7.2.1	Configuring CRL Validation for a Component	6-34
6.7.2.2	Manage CRLs on the File System	6-34
6.7.2.3	Test a Component Configured for CRL Validation	6-35
6.7.3	Oracle Fusion Middleware FIPS 140-2 Settings	6-35
6.7.3.1	FIPS-Configurable Products	6-36
6.7.3.2	Setting the SSLFIPS_140 Parameter	6-36
6.7.3.3	Selecting Cipher Suites	6-36
6.7.3.4	Other Configuration Parameters	6-36
6.8	Best Practices for SSL	6-37
6.8.1	Best Practices for Administrators	6-37
6.8.2	Best Practices for Application Developers	6-37
6.9	WLST Reference for SSL	6-37
6.9.1	addCertificateRequest	6-39
6.9.1.1	Description	6-39
6.9.1.2	Syntax	6-39
6.9.1.3	Example	6-40
6.9.2	addSelfSignedCertificate	6-40
6.9.2.1	Description	6-40
6.9.2.2	Syntax	6-40
6.9.2.3	Example	6-40
6.9.3	changeKeyStorePassword	6-40
6.9.3.1	Description	6-40
6.9.3.2	Syntax	6-40
6.9.3.3	Example	6-41
6.9.4	changeWalletPassword	6-41
6.9.4.1	Description	6-41
6.9.4.2	Syntax	6-41
6.9.4.3	Example	6-41
6.9.5	configureSSL	6-41
6.9.5.1	Description	6-42
6.9.5.2	Syntax	6-42
6.9.5.3	Examples	6-42
6.9.6	createKeyStore	6-42
6.9.6.1	Description	6-42
6.9.6.2	Syntax	6-42
6.9.6.3	Example	6-43
6.9.7	createWallet	6-43
6.9.7.1	Description	6-43
6.9.7.2	Syntax	6-43
6.9.7.3	Examples	6-43
6.9.8	deleteKeyStore	6-43
6.9.8.1	Description	6-43
6.9.8.2	Syntax	6-44

6.9.8.3	Example.....	6-44
6.9.9	deleteWallet	6-44
6.9.9.1	Description	6-44
6.9.9.2	Syntax	6-44
6.9.9.3	Example.....	6-44
6.9.10	exportKeyStore	6-44
6.9.10.1	Description	6-44
6.9.10.2	Syntax	6-45
6.9.10.3	Example.....	6-45
6.9.11	exportKeyStoreObject	6-45
6.9.11.1	Description	6-45
6.9.11.2	Syntax	6-45
6.9.11.3	Examples	6-46
6.9.12	exportWallet	6-46
6.9.12.1	Description	6-46
6.9.12.2	Syntax	6-46
6.9.12.3	Examples	6-46
6.9.13	exportWalletObject	6-47
6.9.13.1	Description	6-47
6.9.13.2	Syntax	6-47
6.9.13.3	Examples	6-47
6.9.14	generateKey	6-48
6.9.14.1	Description	6-48
6.9.14.2	Syntax	6-48
6.9.14.3	Examples	6-48
6.9.15	getKeyStoreObject	6-49
6.9.15.1	Description	6-49
6.9.15.2	Syntax	6-49
6.9.15.3	Examples	6-49
6.9.16	getSSL	6-49
6.9.16.1	Description	6-49
6.9.16.2	Syntax	6-49
6.9.16.3	Example.....	6-50
6.9.17	getWalletObject	6-50
6.9.17.1	Description	6-50
6.9.17.2	Syntax	6-50
6.9.17.3	Examples	6-50
6.9.18	importKeyStore	6-51
6.9.18.1	Description	6-51
6.9.18.2	Syntax	6-51
6.9.18.3	Example.....	6-51
6.9.19	importKeyStoreObject	6-51
6.9.19.1	Description	6-51
6.9.19.2	Syntax	6-52
6.9.19.3	Examples	6-52
6.9.20	importWallet.....	6-52
6.9.20.1	Description	6-52

6.9.20.2	Syntax	6-52
6.9.20.3	Examples	6-53
6.9.21	importWalletObject	6-53
6.9.21.1	Description	6-53
6.9.21.2	Syntax	6-53
6.9.21.3	Examples	6-54
6.9.22	listKeyStoreObjects	6-54
6.9.22.1	Description	6-54
6.9.22.2	Syntax	6-54
6.9.22.3	Examples	6-54
6.9.23	listKeyStores	6-55
6.9.23.1	Description	6-55
6.9.23.2	Syntax	6-55
6.9.23.3	Example.....	6-55
6.9.24	listWalletObjects	6-55
6.9.24.1	Description	6-55
6.9.24.2	Syntax	6-55
6.9.24.3	Examples	6-56
6.9.25	listWallets.....	6-56
6.9.25.1	Description	6-56
6.9.25.2	Syntax	6-56
6.9.25.3	Example.....	6-56
6.9.26	removeKeyStoreObject	6-56
6.9.26.1	Description	6-56
6.9.26.2	Syntax	6-57
6.9.26.3	Examples	6-57
6.9.27	removeWalletObject	6-57
6.9.27.1	Description	6-57
6.9.27.2	Syntax	6-57
6.9.27.3	Examples	6-58
6.9.28	Properties Files for SSL	6-58
6.9.28.1	Structure of Properties Files.....	6-58
6.9.28.2	Examples of Properties Files	6-60

7 Using the SSL Automation Tool

7.1	Introduction to the SSL Automation Tool	7-1
7.2	Prerequisites	7-2
7.2.1	Setting up Oracle Fusion Middleware Environment	7-2
7.2.2	Assembling Required Information.....	7-2
7.3	Generating the CA Certificate	7-3
7.3.1	Example: Generating a Certificate.....	7-4
7.4	Configuring a Component Server	7-5
7.4.1	Example: Configuring a WebLogic Server and Java EE Components.....	7-6
7.4.2	Example: Configuring an Oracle Internet Directory Server Component.....	7-6
7.4.3	Example: Configuring an Oracle Virtual Directory Server Component	7-7
7.4.4	Example: Configuring an Oracle Access Manager 10g Server Component.....	7-8
7.5	Configuring a Client	7-10

7.5.1	Example: Downloading the CA Certificate for SSL Clients	7-11
7.5.2	Example: Downloading the Certificate and Configuring a WebLogic Client	7-12
7.5.3	Example: Downloading the Certificate and Configuring a WebGate Client.....	7-12

8 Managing Keystores, Wallets, and Certificates

8.1	Key and Certificate Storage in Oracle Fusion Middleware	8-1
8.1.1	Types of Keystores.....	8-1
8.1.1.1	JKS Keystore and Truststore	8-1
8.1.1.2	Oracle Wallet.....	8-2
8.1.2	Keystore Management Tools.....	8-2
8.2	Command-Line Interface for Keystores and Wallets	8-4
8.3	JKS Keystore Management	8-4
8.3.1	About Keystores and Certificates.....	8-5
8.3.1.1	Sharing Keystores Across Instances	8-5
8.3.1.2	Keystore Naming Conventions	8-5
8.3.2	Managing the Keystore Life Cycle	8-5
8.3.3	Common Keystore Operations	8-6
8.3.3.1	Creating a Keystore Using Fusion Middleware Control	8-6
8.3.3.2	Creating a Keystore Using WLST	8-7
8.3.3.3	Exporting a Keystore Using Fusion Middleware Control	8-7
8.3.3.4	Exporting a Keystore Using WLST	8-8
8.3.3.5	Deleting a Keystore Using Fusion Middleware Control	8-8
8.3.3.6	Deleting a Keystore Using WLST.....	8-9
8.3.3.7	Importing a Keystore Using Fusion Middleware Control	8-9
8.3.3.8	Importing a Keystore Using WLST.....	8-9
8.3.3.9	Changing the Keystore Password Using Fusion Middleware Control	8-9
8.3.3.10	Changing the Keystore Password Using WLST	8-10
8.3.4	Managing the Certificate Life Cycle	8-10
8.3.5	Common Certificate Operations.....	8-10
8.3.5.1	Generating a New Key for the Keystore Using Fusion Middleware Control ..	8-11
8.3.5.2	Generating a New Key for the Keystore Using WLST.....	8-12
8.3.5.3	Generating a Certificate Signing Request Using Fusion Middleware Control	8-12
8.3.5.4	Generating a Certificate Signing Request Using WLST.....	8-13
8.3.5.5	Importing a Certificate or Trusted Certificate into a Keystore Using Fusion Middleware Control.....	8-13
8.3.5.6	Importing a Certificate or Trusted Certificate into a Keystore Using WLST....	8-14
8.3.5.7	Exporting a Certificate or Trusted Certificate from the Keystore Using Fusion Middleware Control.....	8-14
8.3.5.8	Exporting a Certificate or Trusted Certificate from the Keystore Using WLST	8-15
8.3.5.9	Deleting a Certificate or Trusted Certificate from the Keystore Using Fusion Middleware Control.....	8-15
8.3.5.10	Deleting a Certificate or Trusted Certificate from the Keystore Using WLST .	8-16
8.3.5.11	Converting a Self-Signed Certificate to a Third-Party Certificate Using Fusion Middleware Control.....	8-16
8.3.5.12	Converting a Self-Signed Certificate to a Third-Party Certificate Using WLST	8-17
8.3.6	Keystore and Certificate Maintenance.....	8-18
8.3.6.1	Location of Keystores.....	8-18

8.3.6.2	Replacing Expiring Certificates	8-18
8.3.6.3	Effect of Host Name Change on Keystores	8-19
8.4	Wallet Management.....	8-20
8.4.1	About Wallets and Certificates	8-20
8.4.1.1	Password-Protected and Autologin Wallets	8-20
8.4.1.2	Self-Signed and Third-Party Wallets	8-21
8.4.1.3	Sharing Wallets Across Instances.....	8-21
8.4.1.4	Wallet Naming Conventions	8-21
8.4.2	Accessing the Wallet Management Page in Fusion Middleware Control.....	8-22
8.4.3	Managing the Wallet Life Cycle	8-22
8.4.4	Common Wallet Operations	8-23
8.4.4.1	Creating a Wallet Using Fusion Middleware Control	8-23
8.4.4.2	Creating a Wallet Using WLST.....	8-24
8.4.4.3	Creating a Self-Signed Wallet Using Fusion Middleware Control	8-24
8.4.4.4	Creating a Self-Signed Wallet Using WLST.....	8-25
8.4.4.5	Changing a Self-Signed Wallet to a Third-Party Wallet Using Fusion Middleware Control.....	8-26
8.4.4.6	Changing a Self-Signed Wallet to a Third-Party Wallet Using WLST.....	8-26
8.4.4.7	Exporting a Wallet Using Fusion Middleware Control.....	8-26
8.4.4.8	Exporting a Wallet Using WLST	8-26
8.4.4.9	Importing a Wallet Using Fusion Middleware Control.....	8-27
8.4.4.10	Importing a Wallet Using WLST.....	8-27
8.4.4.11	Deleting a Wallet Using Fusion Middleware Control.....	8-27
8.4.4.12	Deleting a Wallet Using WLST	8-28
8.4.5	Managing the Certificate Life Cycle.....	8-28
8.4.6	Accessing the Certificate Management Page for Wallets in Fusion Middleware Control.....	8-28
8.4.7	Common Certificate Operations.....	8-29
8.4.7.1	Adding a Certificate Request Using Fusion Middleware Control	8-29
8.4.7.2	Adding a Certificate Request Using WLST	8-30
8.4.7.3	Exporting a Certificate, Certificate Request, or a Trusted Certificate Using Fusion Middleware Control.....	8-30
8.4.7.4	Exporting a Certificate, Certificate Request, or a Trusted Certificate Using WLST	8-31
8.4.7.5	Importing a Certificate or a Trusted Certificate Using Fusion Middleware Control	8-31
8.4.7.6	Importing a Certificate or a Trusted Certificate Using WLST	8-32
8.4.7.7	Deleting a Certificate Request, a Certificate, or a Trusted Certificate Using Fusion Middleware Control.....	8-32
8.4.7.8	Deleting a Certificate Request, a Certificate, or a Trusted Certificate Using WLST	8-32
8.4.7.9	Converting a Self-Signed Certificate into a Third-Party Certificate Using Fusion Middleware Control.....	8-33
8.4.7.10	Converting a Self-Signed Certificate into a Third-Party Certificate Using WLST	8-34
8.4.8	Wallet and Certificate Maintenance.....	8-35
8.4.8.1	Location of Wallets.....	8-35
8.4.8.2	Effect of Host Name Change on a Wallet	8-36

8.4.8.3	Changing a Self-Signed Wallet to a Third-Party Wallet	8-37
8.4.8.4	Replacing an Expiring Certificate in a Wallet	8-37

Part IV Deploying Applications

9 Understanding the Deployment Process

9.1	What Is a Deployer?	9-1
9.2	General Procedures for Moving from Application Design to Production Deployment ..	9-1
9.2.1	Designing and Developing an Application	9-1
9.2.2	Deploying an Application to Managed Servers	9-2
9.2.3	Automating the Migration of an Application to Other Environments.....	9-5
9.3	Diagnosing Typical Problems	9-5

10 Deploying Applications

10.1	Overview of Deploying Applications	10-1
10.1.1	What Types of Applications Can You Deploy?.....	10-1
10.1.2	Understanding Deployment, Redeployment, and Undeployment.....	10-3
10.2	Understanding and Managing Data Sources.....	10-3
10.2.1	Creating and Managing JDBC Data Sources	10-4
10.2.1.1	Creating a JDBC Data Source Using Fusion Middleware Control.....	10-5
10.2.1.2	Editing a JDBC Data Source Using Fusion Middleware Control	10-6
10.2.1.3	Monitoring a JDBC Data Source Using Fusion Middleware Control	10-6
10.2.1.4	Controlling a JDBC Data Source Using Fusion Middleware Control.....	10-6
10.3	Deploying, Undeploying, and Redeploying Java EE Applications.....	10-7
10.3.1	Deploying Java EE Applications	10-7
10.3.1.1	Deploying Java EE Applications Using Fusion Middleware Control	10-7
10.3.1.2	Deploying Java EE Applications Using WLST.....	10-10
10.3.2	Undeploying Java EE Applications.....	10-10
10.3.2.1	Undeploying Java EE Applications Using Fusion Middleware Control.....	10-10
10.3.2.2	Undeploying Java EE Applications Using WLST	10-11
10.3.3	Redeploying Java EE Applications	10-11
10.3.3.1	Redeploying Java EE Applications Using Fusion Middleware Control	10-11
10.3.3.2	Redeploying Java EE Applications Using WLST.....	10-12
10.4	Deploying, Undeploying, and Redeploying Oracle ADF Applications	10-13
10.4.1	Deploying Oracle ADF Applications	10-13
10.4.1.1	Deploying ADF Applications Using Fusion Middleware Control	10-13
10.4.1.2	Deploying ADF Applications Using WLST or the Administration Console..	10-16
10.4.2	Undeploying Oracle ADF Applications	10-17
10.4.3	Redeploying Oracle ADF Applications.....	10-17
10.5	Deploying, Undeploying, and Redeploying SOA Composite Applications.....	10-19
10.5.1	Deploying SOA Composite Applications	10-19
10.5.2	Undeploying SOA Composite Applications.....	10-21
10.5.3	Redeploying SOA Composite Applications	10-21
10.6	Deploying, Undeploying, and Redeploying WebCenter Applications	10-22
10.6.1	Deploying WebCenter Applications.....	10-22
10.6.2	Undeploying WebCenter Applications	10-24

10.6.3	Redeploying WebCenter Applications	10-24
10.7	Managing Deployment Plans.....	10-26
10.8	About the Common Deployment Tasks in Fusion Middleware Control	10-26
10.9	Changing MDS Configuration Attributes for Deployed Applications.....	10-28
10.9.1	Changing the MDS Configuration Attributes Using Fusion Middleware Control.....	10-29
10.9.2	Changing the MDS Configuration Using WLST.....	10-32
10.9.3	Restoring the Original MDS Configuration for an Application	10-32

Part V Monitoring Oracle Fusion Middleware

11 Monitoring Oracle Fusion Middleware

11.1	Monitoring the Status of Oracle Fusion Middleware	11-1
11.1.1	Viewing General Information	11-2
11.1.2	Monitoring an Oracle WebLogic Server Domain.....	11-3
11.1.3	Monitoring an Oracle WebLogic Server Administration or Managed Server	11-4
11.1.4	Monitoring a Cluster	11-5
11.1.5	Monitoring a Java Component	11-6
11.1.6	Monitoring a System Component	11-7
11.1.7	Monitoring Java EE Applications.....	11-8
11.1.8	Monitoring ADF Applications	11-9
11.1.9	Monitoring SOA Composite Applications.....	11-9
11.1.10	Monitoring Oracle WebCenter Applications.....	11-10
11.1.11	Monitoring Applications Deployed to a Cluster.....	11-11
11.2	Viewing the Performance of Oracle Fusion Middleware	11-12
11.3	Viewing the Routing Topology.....	11-13

12 Managing Log Files and Diagnostic Data

12.1	Overview of Oracle Fusion Middleware Logging	12-1
12.2	Understanding ODL Messages and ODL Log Files.....	12-2
12.3	Viewing and Searching Log Files	12-5
12.3.1	Viewing Log Files and Their Messages	12-5
12.3.1.1	Viewing Log Files and Their Messages Using Fusion Middleware Control....	12-6
12.3.1.2	Viewing Log Files and Their Messages Using WLST	12-7
12.3.2	Searching Log Files.....	12-9
12.3.2.1	Searching Log Files Using Fusion Middleware Control.....	12-9
12.3.2.1.1	Searching Log Files: Basic Searches	12-9
12.3.2.1.2	Searching Log Files: Advanced Searches.....	12-10
12.3.2.2	Searching Log Files Using WLST	12-11
12.3.3	Downloading Log Files.....	12-12
12.3.3.1	Downloading Log Files Using Fusion Middleware Control.....	12-12
12.3.3.2	Downloading Log Files Using WLST	12-13
12.4	Configuring Settings for Log Files.....	12-13
12.4.1	Changing Log File Locations	12-14
12.4.1.1	Changing Log File Locations Using Fusion Middleware Control	12-15
12.4.1.2	Changing Log File Locations Using WLST.....	12-15

12.4.2	Configuring Log File Rotation	12-15
12.4.2.1	Specifying Log File Rotation Using Fusion Middleware Control	12-16
12.4.2.2	Specifying Log File Rotation Using WLST	12-17
12.4.3	Setting the Level of Information Written to Log Files.....	12-17
12.4.3.1	Configuring Message Levels Using Fusion Middleware Control.....	12-19
12.4.3.2	Configuring Message Levels Using WLST	12-20
12.4.4	Specifying the Log File Format.....	12-20
12.4.4.1	Specifying the Log File Format Using Fusion Middleware Control.....	12-21
12.4.4.2	Specifying the Log File Format Using WLST	12-21
12.4.5	Specifying the Log File Locale	12-21
12.4.5.1	Specifying the Log File Encoding Using WLST	12-21
12.4.5.2	Specifying the Log File Encoding in logging.xml.....	12-22
12.5	Correlating Messages Across Log Files and Components.....	12-22

13 Diagnosing Problems

13.1	Understanding the Diagnostic Framework.....	13-1
13.1.1	About Incidents and Problems	13-3
13.1.1.1	Incident Flood Control.....	13-3
13.1.2	Diagnostic Framework Components	13-3
13.1.2.1	Automatic Diagnostic Repository	13-3
13.1.2.2	Diagnostic Dumps	13-5
13.1.2.3	Management MBeans	13-5
13.1.2.4	WLST Commands for Diagnostic Framework.....	13-5
13.1.2.5	ADCRI Command-Line Utility.....	13-5
13.2	How the Diagnostic Framework Works.....	13-6
13.3	Configuring the Diagnostic Framework.....	13-9
13.3.1	Configuring Diagnostic Framework Settings.....	13-9
13.3.2	Configuring WLDF Watch and Notification for the Diagnostic Framework	13-11
13.4	Investigating, Reporting, and Solving a Problem	13-13
13.4.1	Roadmap—Investigating, Reporting, and Resolving a Problem.....	13-13
13.4.2	Viewing Problems and Incidents	13-15
13.4.2.1	Viewing Problems	13-15
13.4.2.2	Viewing Incidents.....	13-16
13.4.3	Working with Diagnostic Dumps	13-17
13.4.3.1	Listing Diagnostic Dumps.....	13-17
13.4.3.2	Viewing a Description of a Diagnostic Dump.....	13-18
13.4.3.3	Executing Dumps	13-18
13.4.4	Managing Incidents	13-18
13.4.4.1	Creating an Incident Manually.....	13-19
13.4.4.2	Packaging an Incident.....	13-20
13.4.4.3	Generating an RDA Report.....	13-22
13.4.4.4	Purging Incidents	13-22

Part VI Advanced Administration

14 Managing the Metadata Repository

14.1	Understanding a Metadata Repository.....	14-1
14.2	Creating a Database-Based Metadata Repository.....	14-2
14.3	Managing the MDS Repository.....	14-2
14.3.1	Understanding the MDS Repository.....	14-3
14.3.1.1	Databases Supported by MDS.....	14-4
14.3.1.2	Understanding MDS Operations.....	14-6
14.3.2	Registering and Deregistering a Database-Based MDS Repository.....	14-8
14.3.2.1	Registering a Database-Based MDS Repository.....	14-8
14.3.2.1.1	Registering a Database-Based MDS Repository Using Fusion Middleware Control.....	14-8
14.3.2.1.2	Registering a Database-Based MDS Repository Using WLST.....	14-10
14.3.2.2	Adding or Removing Servers Targeted to the MDS Repository.....	14-10
14.3.2.3	Deregistering a Database-Based MDS Repository.....	14-11
14.3.2.3.1	Deregistering a Database-Based MDS Repository Using Fusion Middleware Control.....	14-11
14.3.2.3.2	Deregistering a Database-Based MDS Repository Using WLST.....	14-11
14.3.3	Registering and Deregistering a File-Based MDS Repository.....	14-11
14.3.3.1	Creating and Registering a File-Based MDS Repository.....	14-12
14.3.3.2	Deregistering a File-Based MDS Repository.....	14-12
14.3.4	Changing the System Data Source.....	14-13
14.3.5	Using System MBeans to Manage an MDS Repository.....	14-13
14.3.6	Viewing Information About an MDS Repository.....	14-14
14.3.6.1	Viewing Information About an MDS Repository Using Fusion Middleware Control.....	14-14
14.3.6.2	Viewing Information About an MDS Repository Using System MBeans.....	14-15
14.3.7	Configuring an Application to Use a Different MDS Repository or Partition.....	14-15
14.3.7.1	Cloning a Partition.....	14-16
14.3.7.2	Creating a New Partition and Reassociating the Application to It.....	14-18
14.3.8	Moving Metadata from a Test System to a Production System.....	14-18
14.3.8.1	Transferring Metadata Using Fusion Middleware Control.....	14-19
14.3.8.2	Transferring Metadata using WLST.....	14-20
14.3.9	Moving from a File-Based Repository to a Database-Based Repository.....	14-21
14.3.10	Deleting a Metadata Partition from a Repository.....	14-21
14.3.10.1	Deleting a Metadata Partition Using Fusion Middleware Control.....	14-22
14.3.10.2	Deleting a Metadata Partition Using WLST.....	14-22
14.3.11	Purging Metadata Version History.....	14-22
14.3.11.1	Purging Metadata Version History Using Fusion Middleware Control.....	14-23
14.3.11.2	Purging Metadata Version History Using WLST.....	14-23
14.3.11.3	Enabling Auto-Purge.....	14-23
14.3.12	Managing Metadata Labels in the MDS Repository.....	14-23
14.3.12.1	Creating Metadata Labels.....	14-24
14.3.12.2	Listing Metadata Labels.....	14-24
14.3.12.3	Promoting Metadata Labels.....	14-24
14.3.12.4	Purging Metadata Labels.....	14-25
14.3.12.4.1	Purging Metadata Labels Using Fusion Middleware Control.....	14-25
14.3.12.4.2	Purging Metadata Labels Using WLST.....	14-26

14.3.12.5	Deleting Metadata Labels.....	14-27
14.4	Managing Metadata Repository Schemas	14-27
14.4.1	Changing Metadata Repository Schema Passwords	14-27
14.4.2	Changing the Character Set of the Metadata Repository.....	14-27
14.5	Purging Data.....	14-28
14.5.1	Purging Oracle Infrastructure Web Services Data.....	14-30
14.5.2	Purging Oracle WebCenter Data.....	14-30
14.5.2.1	Purging Oracle WebCenter Activity Stream Data.....	14-30
14.5.2.2	Purging Oracle WebCenter Analytics Data	14-30
14.5.2.2.1	Loading the Oracle WebCenter Purge Package.....	14-30
14.5.2.2.2	Running the Oracle WebCenter Purge Script	14-31
14.5.2.3	Partitioning Oracle WebCenter Analytics Data	14-32

15 Changing Network Configurations

15.1	Changing the Network Configuration of Oracle Fusion Middleware.....	15-1
15.1.1	Changing the Network Configuration of a Managed Server.....	15-1
15.1.2	Changing the Network Configuration of Web Tier Components.....	15-2
15.2	Changing the Network Configuration of a Database	15-3
15.3	Moving Between On-Network and Off-Network.....	15-6
15.3.1	Moving from Off-Network to On-Network (Static IP Address).....	15-7
15.3.2	Moving from Off-Network to On-Network (DHCP)	15-7
15.3.3	Moving from On-Network to Off-Network (Static IP Address).....	15-7
15.4	Changing Between a Static IP Address and DHCP	15-7
15.4.1	Changing from a Static IP Address to DHCP.....	15-8
15.4.2	Changing from DHCP to a Static IP Address.....	15-8
15.5	Using IPv6	15-8
15.5.1	Supported Topologies for IPv6 Network Protocols	15-10
15.5.2	Configuring Oracle HTTP Server for IPv6.....	15-11
15.5.3	Disabling IPv6 Support for Oracle Web Cache	15-12
15.5.4	Configuring Oracle Single Sign-On to Use Oracle HTTP Server with IPv6	15-12
15.5.5	Configuring Oracle Access Manager Support for IPv6.....	15-14
15.5.5.1	Simple Authentication with IPv6	15-14
15.5.5.2	Configuring IPv6 with an Authenticating WebGate and Challenge Redirect	15-15
15.5.5.3	Considerations	15-16
15.5.5.4	Prerequisites	15-16
15.5.5.5	Configuring IPv6 with Simple Authentication	15-17
15.5.5.6	Configuring IPv6 with an Authenticating WebGate and Challenge Redirect	15-18
15.5.5.7	Configuring IPv6: Separate Proxy for Authentication and Resource WebGates.....	15-20

Part VII Advanced Administration: Backup and Recovery

16 Introducing Backup and Recovery

16.1	Understanding Oracle Fusion Middleware Backup and Recovery.....	16-1
16.1.1	Impact of Administration Server Failure	16-2
16.1.2	Managed Server Independence (MSI) Mode	16-2
16.1.3	Configuration Changes in Managed Servers.....	16-2

16.2	Oracle Fusion Middleware Directory Structure	16-3
16.3	Overview of the Backup Strategies	16-3
16.3.1	Types of Backups	16-4
16.3.2	Recommended Backup Strategy.....	16-5
16.4	Overview of Recovery Strategies.....	16-6
16.4.1	Types of Recovery.....	16-7
16.4.2	Recommended Recovery Strategies.....	16-7
16.5	Backup and Recovery Recommendations for Oracle Fusion Middleware Components.....	16-7
16.5.1	Backup and Recovery Recommendations for Oracle WebLogic Server.....	16-8
16.5.1.1	Backup and Recovery Recommendations for Oracle WebLogic Server.....	16-8
16.5.1.2	Backup and Recovery Recommendations for Oracle WebLogic Server JMS ..	16-9
16.5.2	Backup and Recovery Recommendations for Oracle Identity Management.....	16-11
16.5.2.1	Backup and Recovery Recommendations for Oracle Internet Directory	16-11
16.5.2.2	Backup and Recovery Recommendations for Oracle Virtual Directory	16-12
16.5.2.3	Backup and Recovery Recommendations for Oracle Directory Integration Platform	16-12
16.5.2.4	Backup and Recovery Recommendations for Oracle Directory Services Manager	16-13
16.5.2.5	Backup and Recovery Recommendations for Oracle Identity Federation.....	16-13
16.5.2.6	Backup and Recovery Recommendations for Oracle Access Manager	16-13
16.5.2.7	Backup and Recovery Recommendations for Oracle Adaptive Access Manager	16-14
16.5.2.8	Backup and Recovery Recommendations for Oracle Identity Manager	16-14
16.5.2.9	Backup and Recovery Recommendations for Oracle Identity Navigator.....	16-15
16.5.3	Backup and Recovery Recommendations for Oracle SOA Suite.....	16-15
16.5.3.1	Backup and Recovery Recommendations for Oracle BPEL Process Manager	16-16
16.5.3.2	Backup and Recovery Recommendations for Oracle Business Activity Monitoring.....	16-16
16.5.3.3	Backup and Recovery Recommendations for Oracle B2B.....	16-17
16.5.3.4	Backup and Recovery Recommendations for Oracle Service Bus.....	16-17
16.5.3.5	Backup and Recovery Recommendations for Oracle Mediator	16-18
16.5.3.6	Backup and Recovery Recommendations for Oracle Business Rules.....	16-18
16.5.3.7	Backup and Recovery Recommendations for Oracle Business Process Management.....	16-19
16.5.4	Backup and Recovery Recommendations for Oracle WebCenter	16-19
16.5.4.1	Backup and Recovery Recommendations for Oracle WebCenter	16-20
16.5.4.2	Backup and Recovery Recommendations for Oracle WebCenter Portlet.....	16-20
16.5.4.3	Backup and Recovery Recommendations for Oracle WebCenter Discussions Server.....	16-20
16.5.4.4	Backup and Recovery Recommendations for Oracle WebCenter Wiki and Blog Server.....	16-21
16.5.4.5	Backup and Recovery Recommendations for Oracle WebCenter Activity Graph.....	16-21
16.5.4.6	Backup and Recovery Recommendations for Oracle WebCenter Analytics ..	16-22
16.5.4.7	Backup and Recovery Recommendations for Oracle Content Server	16-22
16.5.5	Backup and Recovery Recommendations for Oracle JRF Installations	16-22
16.5.5.1	Backup and Recovery Recommendations for Oracle Web Services Manager	16-23

16.5.5.2	Backup and Recovery Recommendations for Oracle Platform Security Services.....	16-23
16.5.6	Backup and Recovery Recommendations for Web Tier Installations.....	16-23
16.5.6.1	Backup and Recovery Recommendations for Oracle HTTP Server.....	16-24
16.5.6.2	Backup and Recovery Recommendations for Oracle Web Cache.....	16-24
16.5.7	Backup and Recovery Recommendations for Oracle Portal, Oracle Forms Services, Oracle Reports, and Oracle BI Discoverer Installations.....	16-24
16.5.7.1	Backup and Recovery Recommendations for Oracle Portal.....	16-25
16.5.7.2	Backup and Recovery Recommendations for Oracle Forms Services.....	16-25
16.5.7.3	Backup and Recovery Recommendations for Oracle Reports.....	16-26
16.5.7.4	Backup and Recovery Recommendations for Oracle Business Intelligence Discoverer.....	16-27
16.5.8	Backup and Recovery Recommendations for Oracle Business Intelligence.....	16-28
16.5.8.1	Backup and Recovery Recommendations for Oracle BI Enterprise Edition...	16-28
16.5.8.2	Backup and Recovery Recommendations for Oracle Business Intelligence Publisher.....	16-29
16.5.8.3	Backup and Recovery Recommendations for Oracle Real-Time Decisions....	16-30
16.5.9	Backup and Recovery Recommendations for Oracle Data Integrator.....	16-30
16.5.10	Backup and Recovery Recommendations for Oracle Enterprise Content Management Suite.....	16-31
16.5.10.1	Backup and Recovery Recommendations for Oracle Information Rights Management.....	16-31
16.5.10.2	Backup and Recovery Recommendations for Oracle Imaging and Process Management.....	16-32
16.5.10.3	Backup and Recovery Recommendations for Oracle Universal Content Management.....	16-32
16.5.10.4	Backup and Recovery Recommendations for Oracle Universal Records Management.....	16-33
16.6	Assumptions and Restrictions.....	16-33

17 Backing Up Your Environment

17.1	Overview of Backing Up Your Environment.....	17-1
17.2	Limitations and Restrictions for Backing Up Data.....	17-2
17.3	Performing a Backup.....	17-3
17.3.1	Performing a Full Offline Backup.....	17-4
17.3.2	Performing an Online Backup of Run-Time Artifacts.....	17-5
17.3.3	Exporting Oracle BI EE Registry Entries On Windows.....	17-6
17.4	Creating a Record of Your Oracle Fusion Middleware Configuration.....	17-6

18 Recovering Your Environment

18.1	Overview of Recovering Your Environment.....	18-1
18.2	Recovering After Data Loss, Corruption, Media Failure, or Application Malfunction.	18-2
18.2.1	Recovering a Middleware Home.....	18-2
18.2.2	Recovering an Oracle WebLogic Server Domain.....	18-2
18.2.3	Recovering an Oracle Home.....	18-3
18.2.4	Recovering an Oracle Instance Home.....	18-3
18.2.4.1	Recovering After Oracle Instance Home Deleted from File System.....	18-3
18.2.4.2	Recovering After Oracle Instance Home Deregistered.....	18-4

18.2.5	Recovering the Administration Server Configuration.....	18-4
18.2.6	Recovering a Managed Server	18-5
18.2.6.1	Recovering a Managed Server When It Cannot Be Started.....	18-5
18.2.6.2	Recovering a Managed Server When It Does Not Function Correctly.....	18-6
18.2.6.3	Recovering an Oracle SOA Suite Managed Server That Has a Separate Directory	18-7
18.2.7	Recovering Components.....	18-7
18.2.7.1	Recovering a Component That Is Not Functioning Properly	18-8
18.2.7.2	Recovering a Component After Cluster Configuration Change	18-8
18.2.7.3	Recovering Oracle Identity Manager.....	18-9
18.2.7.4	Recovering Oracle Identity Navigator	18-10
18.2.7.5	Recovering Oracle Access Manager.....	18-10
18.2.7.6	Recovering Oracle Adaptive Access Manager	18-10
18.2.7.7	Recovering Oracle Business Process Management.....	18-10
18.2.7.8	Recovering Oracle WebCenter Activities Graph	18-10
18.2.7.9	Recovering Oracle WebCenter Analytics.....	18-11
18.2.7.10	Recovering Oracle BI Enterprise Edition	18-11
18.2.7.10.1	Recovering Oracle BI Enterprise Edition in a Non-Clustered Environment	18-11
18.2.7.10.2	Recovering Oracle BI Enterprise Edition in a Clustered Environment....	18-11
18.2.7.10.3	Reconciling the LDAP Database with the Web Catalog and RPD.....	18-12
18.2.7.11	Recovering Oracle Business Intelligence Publisher.....	18-12
18.2.7.12	Recovering Oracle Real-Time Decisions	18-12
18.2.7.13	Recovering Oracle Data Integrator	18-13
18.2.7.14	Recovering Oracle Information Rights Management	18-13
18.2.7.15	Recovering Oracle Imaging and Process Management	18-13
18.2.7.16	Recovering Oracle Universal Content Management	18-13
18.2.7.17	Recovering Oracle Universal Records Management.....	18-14
18.2.8	Recovering a Cluster	18-14
18.2.8.1	Recovering a Cluster After Deletion or Cluster-Level Configuration Changes.....	18-14
18.2.8.2	Recovering a Cluster After Membership Is Mistakenly Modified	18-15
18.2.9	Recovering Applications.....	18-15
18.2.9.1	Recovering Application Artifacts.....	18-16
18.2.9.2	Recovering a Redeployed Application That Is No Longer Functional.....	18-16
18.2.9.3	Recovering an Undeployed Application.....	18-16
18.2.9.4	Recovering a Composite Application.....	18-17
18.2.10	Recovering a Database	18-17
18.3	Recovering After Loss of Host	18-17
18.3.1	Recovering an Oracle WebLogic Server Domain.....	18-17
18.3.2	Recovering After Loss of Administration Server Host	18-18
18.3.2.1	Recovering the Administration Server to the Same Host.....	18-18
18.3.2.2	Recovering the Administration Server to a Different Host.....	18-19
18.3.3	Recovering After Loss of Managed Server Host.....	18-20
18.3.3.1	Recovering a Managed Server to the Same Host.....	18-21
18.3.3.2	Recovering a Managed Server to a Different Host	18-22
18.3.3.3	Recovering an Oracle SOA Suite Managed Server That Has a Separate Directory	18-24

18.3.4	Recovering After Loss of Component Host	18-24
18.3.4.1	Recovering a Java Component to the Same Host	18-25
18.3.4.2	Recovering a Java Component to a Different Host	18-26
18.3.4.3	Recovering a System Component to the Same Host	18-26
18.3.4.4	Recovering a System Component to a Different Host	18-26
18.3.4.5	Recovering Identity Management Components to a Different Host	18-27
18.3.4.5.1	Recovering Oracle Internet Directory to a Different Host	18-27
18.3.4.5.2	Recovering Oracle Virtual Directory to a Different Host	18-27
18.3.4.5.3	Recovering Oracle Directory Integration Platform to a Different Host ...	18-28
18.3.4.5.4	Recovering Oracle Identity Federation to a Different Host	18-28
18.3.4.5.5	Recovering Oracle Identity Manager to a Different Host	18-29
18.3.4.5.6	Recovering Oracle Identity Navigator to a Different Host	18-29
18.3.4.5.7	Recovering Oracle Access Manager to a Different Host	18-30
18.3.4.5.8	Recovering Oracle Adaptive Access Manager to a Different Host	18-30
18.3.4.6	Recovering Oracle SOA Suite After Loss of Host	18-30
18.3.4.7	Recovering Web Tier Components to a Different Host	18-31
18.3.4.7.1	Recovering Oracle HTTP Server to a Different Host	18-31
18.3.4.7.2	Recovering Oracle Web Cache to a Different Host	18-32
18.3.4.8	Recovering Oracle Portal, Oracle Reports, Oracle Forms Services, and Oracle Business Intelligence Discoverer to a Different Host	18-32
18.3.4.8.1	Recovering Oracle Portal to a Different Host	18-32
18.3.4.8.2	Recovering Oracle Forms Services to a Different Host	18-34
18.3.4.8.3	Recovering Oracle Reports to a Different Host	18-35
18.3.4.8.4	Recovering Oracle Business Intelligence Discoverer to a Different Host	18-37
18.3.4.9	Recovering Oracle BI Enterprise Edition to a Different Host	18-38
18.3.4.9.1	Recovering Oracle BI Enterprise Edition to a Different Host in a Non-Clustered Environment	18-38
18.3.4.9.2	Recovering Oracle BI Enterprise Edition to a Different Host in a Clustered Environment	18-39
18.3.4.9.3	Additional Steps for Recovering Oracle BI EE	18-40
18.3.4.9.4	Importing Oracle BI Enterprise Edition Registry Entries	18-41
18.3.4.10	Recovering Oracle Business Intelligence Publisher to a Different Host	18-41
18.3.4.11	Recovering Oracle Real-Time Decisions to a Different Host	18-41
18.3.4.12	Recovering Oracle Data Integrator to a Different Host	18-41
18.3.4.13	Recovering Oracle Enterprise Content Management Suite to a Different Host	18-42
18.3.4.13.1	Recovering Oracle Universal Content Management to a Different Host.	18-42
18.3.4.13.2	Recovering Oracle Universal Records Management After Loss of Host .	18-43
18.3.5	Additional Actions for Recovering Entities After Loss of Host	18-43
18.3.5.1	Recovering Fusion Middleware Control to a Different Host	18-43
18.3.5.2	Changing the Host Name in the targets.xml File for Fusion Middleware Control	18-44
18.3.5.3	Recovering Oracle Management Agent When Components Are Recovered to a Different Host	18-44
18.3.5.4	Modify the mod_wl_ohs.conf File	18-45
18.3.5.5	Creating a New Machine for Certain Components	18-45
18.3.5.6	Reassociate Users to Groups for Certain Identity Management Components	18-46

18.3.5.7	Updating Oracle Inventory	18-46
18.3.5.8	Recovering the Windows Registry.....	18-46
18.3.6	Recovering After Loss of Database Host.....	18-47

Part VIII Advanced Administration: Expanding Your Environment

19 Scaling Your Environment

19.1	Overview of Scaling Your Environment	19-1
19.2	Extending a Domain to Support Additional Components.....	19-2
19.3	Adding Additional Managed Servers to a Domain.....	19-4
19.3.1	Applying Oracle JRF Template to a Managed Server or Cluster.....	19-5
19.4	Creating Clusters.....	19-6
19.5	Cloning a Middleware Home or Component	19-7

20 Cloning Oracle Fusion Middleware

20.1	Introduction to Cloning	20-1
20.2	What You Can Clone.....	20-1
20.3	Understanding the Cloning Process.....	20-2
20.3.1	Understanding Cloning a Middleware Home	20-2
20.3.2	Understanding Cloning Components	20-2
20.4	Cloning Syntax	20-3
20.4.1	Cloning Scripts.....	20-5
20.4.1.1	copyBinary Script	20-6
20.4.1.2	pasteBinary Script	20-7
20.4.1.3	copyConfig Script for Java Components.....	20-9
20.4.1.4	copyConfig Script for System Components	20-11
20.4.1.5	extractMovePlan Script	20-12
20.4.1.6	pasteConfig Script for Java Components.....	20-13
20.4.1.7	pasteConfig Script for System Components.....	20-15
20.5	Cloning Oracle Fusion Middleware Entities.....	20-17
20.5.1	Cloning a Middleware Home	20-17
20.5.2	Cloning Java Components.....	20-19
20.5.2.1	Considerations for Cloning Oracle SOA Suite	20-20
20.5.3	Cloning System Components.....	20-21
20.5.3.1	Cloning Oracle HTTP Server	20-22
20.5.3.2	Cloning Oracle Internet Directory	20-22
20.5.3.3	Cloning Oracle Virtual Directory	20-23
20.5.4	Customizing Move Plans When Cloning Components	20-24
20.5.4.1	Locating ConfigGroup Elements.....	20-24
20.5.4.2	Move Plans for Components.....	20-25
20.6	Recovering from Cloning Errors.....	20-35
20.7	Considerations and Limitations for Cloning	20-35

21 Moving from a Test to a Production Environment

21.1	Overview of Procedures for Moving from a Test to a Production Environment.....	21-1
21.2	Moving Identity Management Components to a Production Environment.....	21-3

21.2.1	Moving Identity Management to a New Production Environment.....	21-4
21.2.2	Moving Identity Management to an Existing Production Environment.....	21-18
21.3	Moving Oracle SOA Suite to a Production Environment	21-24
21.3.1	Moving Oracle SOA Suite to a New Production Environment	21-24
21.3.2	Moving Oracle SOA Suite to an Existing Production Environment	21-37
21.4	Moving Oracle WebCenter to a Production Environment	21-44
21.4.1	Moving Oracle WebCenter to a New Production Environment.....	21-45
21.4.2	Moving Oracle WebCenter to an Existing Production Environment	21-51
21.5	Moving the Web Tier to a Production Environment	21-53
21.5.1	Moving the Web Tier to a New Production Environment	21-53
21.5.1.1	Moving Oracle HTTP Server to a New Production Environment	21-53
21.5.1.2	Moving Oracle Web Cache to a New Production Environment	21-55
21.5.2	Moving the Web Tier to an Existing Production Environment	21-57
21.5.2.1	Moving Oracle HTTP Server to an Existing Production Environment	21-57
21.6	Moving Oracle Portal, Oracle Forms Services, Oracle Reports, and Oracle BI Discoverer Components to a Production Environment	21-58
21.6.1	Moving Oracle Portal, Oracle Forms Services, Oracle Reports, and Oracle Business Intelligence Discoverer to a New Production Environment	21-59
21.6.2	Moving Oracle Portal, Oracle Forms Services, Oracle Reports, and Oracle Business Intelligence Discoverer to an Existing Production Environment.....	21-68
21.7	Moving Oracle Business Intelligence Components to a Production System	21-69
21.7.1	Moving Oracle Business Intelligence Components to a New Production Environment.....	21-70
21.7.2	Moving Oracle Business Intelligence Components to an Existing Production Environment When There are Few Patches to Apply	21-74
21.7.3	Moving Oracle Business Intelligence Components to an Existing Production Environment When There are Many Patches to Apply	21-76
21.7.3.1	Moving to an Existing Production Environment When New Hardware Is Available	21-77
21.7.3.2	Moving to an Existing Production Environment When New Hardware Is Not Available	21-77
21.7.4	Refreshing the User GUIDs	21-78
21.8	Moving Oracle Real-Time Decisions to a Production System.....	21-79
21.8.1	Moving Oracle Real-Time Decisions to a New Production Environment	21-79
21.8.2	Moving Oracle Real-Time Decisions to an Existing Production Environment	21-82
21.9	Moving Oracle Enterprise Content Management to a Production System	21-83
21.9.1	Moving Oracle Enterprise Content Management Suite to a New Production Environment	21-83
21.9.2	Moving Oracle Enterprise Content Management Suite to an Existing Production Environment.....	21-90
21.10	Moving Oracle Data Integrator to a Production Environment	21-94
21.10.1	Moving Oracle Data Integrator to a New Production Environment	21-94
21.10.2	Moving Oracle Data Integrator Scenarios to an Existing Production Environment.....	21-96
21.11	Considerations in Moving to and from an Oracle RAC Environment	21-97

Part IX Appendixes

A Oracle Fusion Middleware Command-Line Tools

B URLs for Components

C Port Numbers

C.1	Port Numbers by Component.....	C-1
C.2	Port Numbers (Sorted by Number).....	C-2

D Metadata Repository Schemas

D.1	Metadata Repository Schema Descriptions	D-1
D.2	Metadata Repository Schemas, Tablespaces, and Data Files	D-3

E Using Oracle Fusion Middleware Accessibility Options

E.1	Install and Configure Java Access Bridge (Windows Only).....	E-1
E.2	Enabling Fusion Middleware Control Accessibility Mode.....	E-1
E.2.1	Making HTML Pages More Accessible	E-1
E.2.2	Viewing Text Descriptions of Fusion Middleware Control Charts.....	E-2
E.3	Fusion Middleware Control Keyboard Navigation.....	E-3

F Examples of Administrative Changes

F.1	How to Use This Appendix.....	F-1
F.2	Examples of Administrative Changes (by Component)	F-2

G Viewing Release Numbers

G.1	Release Number Format	G-1
G.2	Viewing the Software Inventory and Release Numbers	G-2
G.2.1	Viewing Oracle Fusion Middleware Installation Release Numbers	G-2
G.2.2	Viewing Component Release Numbers	G-2
G.2.3	Viewing Oracle Internet Directory Release Numbers	G-3
G.2.4	Viewing Metadata Repository Release Numbers	G-4

H Oracle Wallet Manager and orapki

H.1	New orapki Features	H-2
H.1.1	orapki Usage Examples.....	H-2
H.1.2	New CRL Management Features	H-3
H.1.3	New Version 3 Certificate Support	H-3
H.1.4	Trust Chain Export	H-3
H.1.5	Wallet Password Change.....	H-3
H.1.6	Converting Between Oracle Wallet and JKS Keystore	H-3
H.2	Using the orapki Utility for Certificate Validation and CRL Management	H-5
H.2.1	orapki Overview	H-5
H.2.1.1	orapki Syntax	H-5
H.2.1.2	Environment Setup for orapki.....	H-6
H.2.2	Displaying orapki Help	H-6

H.2.3	Creating Signed Certificates for Testing Purposes	H-6
H.2.4	Managing Oracle Wallets with the orapki Utility.....	H-7
H.2.4.1	Creating and Viewing Oracle Wallets with orapki	H-7
H.2.4.2	Adding Certificates and Certificate Requests to Oracle Wallets with orapki	H-7
H.2.4.3	Exporting Certificates and Certificate Requests from Oracle Wallets with orapki.....	H-8
H.2.5	Managing Certificate Revocation Lists (CRLs) with orapki Utility	H-8
H.2.5.1	About Certificate Validation with Certificate Revocation Lists	H-9
H.2.5.1.1	What CRLs Should You Use?	H-9
H.2.5.1.2	How CRL Checking Works.....	H-9
H.2.5.2	Certificate Revocation List Management	H-10
H.2.5.2.1	Renaming CRLs with a Hash Value for Certificate Validation	H-10
H.2.5.2.2	Uploading CRLs to Oracle Internet Directory	H-11
H.2.5.2.3	Listing CRLs Stored in Oracle Internet Directory	H-12
H.2.5.2.4	Viewing CRLs in Oracle Internet Directory	H-12
H.2.5.2.5	Deleting CRLs from Oracle Internet Directory	H-13
H.2.6	orapki Utility Commands Summary	H-13
H.2.6.1	orapki cert create	H-14
H.2.6.1.1	Purpose	H-14
H.2.6.1.2	Syntax	H-14
H.2.6.2	orapki cert display	H-14
H.2.6.2.1	Purpose	H-14
H.2.6.2.2	Syntax	H-14
H.2.6.3	orapki crl create	H-14
H.2.6.3.1	Purpose	H-14
H.2.6.3.2	Syntax	H-14
H.2.6.4	orapki crl delete	H-15
H.2.6.4.1	Purpose	H-15
H.2.6.4.2	Syntax	H-15
H.2.6.5	orapki crl display	H-15
H.2.6.5.1	Purpose	H-15
H.2.6.5.2	Syntax	H-15
H.2.6.6	orapki crl hash.....	H-16
H.2.6.6.1	Purpose	H-16
H.2.6.6.2	Syntax	H-16
H.2.6.7	orapki crl list.....	H-16
H.2.6.7.1	Purpose	H-16
H.2.6.7.2	Syntax	H-16
H.2.6.8	orapki crl revoke.....	H-16
H.2.6.8.1	Purpose	H-16
H.2.6.8.2	Syntax	H-16
H.2.6.9	orapki crl status.....	H-17
H.2.6.9.1	Purpose	H-17
H.2.6.9.2	Syntax	H-17
H.2.6.10	orapki crl upload	H-17
H.2.6.10.1	Purpose	H-17
H.2.6.10.2	Syntax	H-17
H.2.6.11	orapki crl verify.....	H-18

H.2.6.11.1	Purpose	H-18
H.2.6.11.2	Syntax.....	H-18
H.2.6.12	orapki wallet add.....	H-18
H.2.6.12.1	Purpose	H-18
H.2.6.12.2	Syntax.....	H-18
H.2.6.13	orapki wallet change_pwd.....	H-19
H.2.6.13.1	Purpose	H-19
H.2.6.13.2	Syntax.....	H-19
H.2.6.14	orapki wallet create	H-19
H.2.6.14.1	Purpose	H-19
H.2.6.14.2	Syntax.....	H-19
H.2.6.15	orapki wallet display.....	H-19
H.2.6.15.1	Purpose	H-19
H.2.6.15.2	Syntax.....	H-19
H.2.6.16	orapki wallet export	H-20
H.2.6.16.1	Purpose	H-20
H.2.6.16.2	Syntax.....	H-20
H.2.6.17	orapki wallet export_trust_chain	H-20
H.2.6.17.1	Purpose	H-20
H.2.6.17.2	Syntax.....	H-20
H.3	Equivalent Features for Oracle Wallet Manager	H-20
H.4	Equivalent Features for orapki.....	H-22
H.5	Equivalent Features for the SSL Configuration Tool.....	H-23

I Troubleshooting Oracle Fusion Middleware

I.1	Diagnosing Oracle Fusion Middleware Problems	I-1
I.2	Common Problems and Solutions	I-1
I.2.1	Using a Different Version of Spring.....	I-1
I.2.2	ClassNotFoundException Errors When Starting Managed Servers	I-2
I.3	Troubleshooting Fusion Middleware Control.....	I-2
I.3.1	Troubleshooting the Display of Performance Metrics and Charts in Fusion Middleware Control.....	I-2
I.3.1.1	What Are Agent-Monitored Targets?	I-2
I.3.1.2	Setting Monitoring Credentials for All Agent-Monitored Targets in a Farm	I-3
I.3.1.3	Changing the Monitoring Credentials for a Specific Agent-Monitored Target ...	I-3
I.3.1.4	Verifying or Changing the Oracle Management Agent URL	I-4
I.3.2	Securing the Connection from Fusion Middleware Control to Oracle WebLogic Server Administration Console	I-4
I.4	Need More Help?.....	I-5
I.4.1	Using Remote Diagnostic Agent	I-5

Index

List of Figures

2-1	Oracle Fusion Middleware Environment.....	2-2
2-2	Oracle WebLogic Server Domain	2-3
6-1	SSL Handshake.....	6-5
6-2	SSL in Oracle Fusion Middleware	6-6
13-1	ADR Directory Structure for Oracle Fusion Middleware.....	13-4
13-2	Incident Creation Generated by Incident Log Detector	13-7
13-3	Incident Creation Generated by WLDF Watch Notification	13-8
13-4	Flow for Investigating a Problem	13-14
15-1	Simple Authentication with the IPv6/IPv4 Proxy	15-15
15-2	IPv6 with an Authenticating WebGate and Challenge Redirect	15-16
17-1	Decision Flow Chart for Type of Backup	17-2
G-1	Example of an Oracle Fusion Middleware Release Number.....	G-1

List of Tables

3-1	Environment Variables for Linux and UNIX.....	3-1
3-2	Environment Variables for Windows	3-3
3-3	Comparing Fusion Middleware Control and WebLogic Server Administration Console	3-4
3-4	Navigating Within Fusion Middleware Control.....	3-9
6-1	WLST Commands for SSL Configuration	6-38
6-2	WLST Commands for Oracle Wallet Management	6-38
6-3	WLST Commands for Java Keystore (JKS) Management	6-38
6-4	Parameters in Properties File	6-59
6-5	Default Values of Parameters.....	6-59
7-1	Main Scripts	7-2
7-2	Domain-Level Information Variables for SSL Automation Tool	7-2
7-3	Component-Specific Information Variables for SSL Automation Tool	7-3
7-4	Component Options to SSLServerConfig.sh.....	7-5
7-5	Component Options to SSLClientConfig.sh	7-11
9-1	Oracle JDeveloper Extensions	9-4
10-1	Tools to Deploy Applications.....	10-2
10-2	MDS Configuration Attributes for Deployed Applications	10-30
12-1	ODL Format Message Fields	12-2
12-2	Log File Location.....	12-4
12-3	Diagnostic Message Types and Level	12-17
12-4	Mapping of Log Levels Among ODL, Oracle WebLogic Server, and Java	12-18
13-1	DiagnosticConfig MBean Parameters for Diagnostic Framework	13-9
13-2	Diagnostic Dump Actions.....	13-17
14-1	MDS Operations and Required Roles	14-7
14-2	Purging Data Documentation	14-28
15-1	Support for IPv6.....	15-8
18-1	Recovery Procedures for Particular Components.....	18-7
18-2	Recovery Procedures for Loss of Host for Particular Components.....	18-24
19-1	Supported Domain Extensions	19-2
20-1	Options for the copyBinary Script	20-7
20-2	Options for the pasteBinary Script	20-8
20-3	Options for the copyConfig Script for Java Components	20-10
20-4	Options for the copyConfig Script for System Components	20-12
20-5	Options for the extractMovePlan Script	20-13
20-6	Options for the pasteConfig Script for Java Components	20-14
20-7	Options for the pasteConfig Script for System Components	20-16
20-8	Move Plan Properties for Components	20-25
20-9	Common Move Plan Properties for Java Components	20-26
20-10	Move Plan Properties for Oracle ADF Connections	20-29
20-11	Move Plan Properties for Oracle SOA Suite	20-32
20-12	Move Plan Properties for Oracle HTTP Server.....	20-33
20-13	Move Plan Properties for Oracle Internet Directory	20-34
20-14	Move Plan Properties for Oracle Virtual Directory	20-35
A-1	Oracle Fusion Middleware Command-Line Tools	A-1
B-1	URLs for Components.....	B-1
C-1	Port Numbers Sorted by Component	C-1
C-2	Port Numbers Sorted by Number	C-2
D-1	Metadata Schemas Created by Repository Creation Utility	D-1
D-2	Metadata Repository Tablespaces and Data Files.....	D-3
E-1	Keyboard Navigation for Common Tasks	E-3
F-1	Examples of Administrative Changes	F-2
H-1	Mapping for Oracle Wallet Manager Features for Wallets.....	H-21

H-2	Mapping for Oracle Wallet Manager Features for Certificates	H-21
H-3	Mapping for orapki Features for Wallets and CRLs.....	H-22
H-4	Mapping for orapki Features for Certificates	H-23
H-5	Equivalent Features for the SSL Configuration Tool.....	H-23

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

Our goal is to make Oracle products, services, and supporting documentation accessible to all users, including users that are disabled. To that end, our documentation includes features that make information available to users of assistive technology. This documentation is available in HTML format, and contains markup to facilitate access by the disabled community. Accessibility standards will continue to evolve over time, and Oracle is actively engaged with other market-leading technology vendors to address technical obstacles so that our documentation can be accessible to all of our customers. For more information, visit the Oracle Accessibility Program Web site at <http://www.oracle.com/accessibility/>.

Accessibility of Code Examples in Documentation

Screen readers may not always correctly read the code examples in this document. The conventions for writing code require that closing braces should appear on an otherwise empty line; however, some screen readers may not always read a line of text that consists solely of a bracket or brace.

Accessibility of Links to External Web Sites in Documentation

This documentation may contain links to Web sites of other companies or organizations that Oracle does not own or control. Oracle neither evaluates nor makes any representations regarding the accessibility of these Web sites.

Access to Oracle Support

Oracle customers have access to electronic support through My Oracle Support. For information, visit <http://www.oracle.com/support/contact.html> or visit <http://www.oracle.com/accessibility/support.html> if you are hearing impaired.

Related Documents

For more information, see the following documents in the Oracle Fusion Middleware 11g Release 1 (11.1.1) 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*
- *Oracle Fusion Middleware Administrator's Guide for Oracle HTTP Server*
- *Oracle Fusion Middleware Administrator's Guide for Oracle Web Cache*
- *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 Administrator's Guide for Oracle Identity Federation*
- *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.
<i>italic</i>	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
monospace	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

What's New in This Guide?

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 11g Release 1 (11.1.1.4)

Oracle Fusion Middleware 11g Release 1 (11.1.1.4) includes the following new and changed administrative features:

- Oracle Fusion Middleware now supports IBM WebSphere as an application server. See the *Oracle Fusion Middleware Third-Party Application Server Guide* for more information.
- MDS provides the following new and changed features:
 - Support for purging metadata labels using Fusion Middleware Control or the command line. See [Section 14.3.11](#).
 - Support for importing to and exporting from a remote location. See [Section 14.3.1](#).
 - Support for targeting the repository to additional Managed Servers. See [Section 14.3.2.2](#).
- Additions to the cloning scripts, allowing you to clone the configuration of a domain and components. See [Chapter 20](#).
- Revised procedures for moving from a test environment to a production environment. See [Chapter 21](#).
- Scripts to purge data from your database. See [Section 14.5](#).

New Features for Oracle Fusion Middleware 11g Release 1 (11.1.1.3)

Oracle Fusion Middleware 11g Release 1 (11.1.1.3) includes the following new and changed administrative features:

- MDS now supports DB2. See [Section 14.3.1.1](#).
- The cloning syntax has changed. See [Chapter 20](#).
- Oracle Enterprise Manager Fusion Middleware Control now provides an interface to create and modify data sources. See [Section 10.2](#).

New Features for Oracle Fusion Middleware 11g Release 1 (11.1.1.2)

Oracle Fusion Middleware 11g Release 1 (11.1.1.2) includes the following new administrative features:

- The Diagnostic Framework, which aids in detecting, diagnosing, and resolving problems. For more information, see [Chapter 13](#).
- An Oracle Common home, which contains the binary and library files required for the Oracle Enterprise Manager Fusion Middleware Control and Java Required Files (JRF). For more information, see [Section 2.6](#).
- Changes to cloning. For more information, see [Chapter 20](#).

New Features for Oracle Fusion Middleware 11g Release 1 (11.1.1)

Oracle Fusion Middleware 11g Release 1 (11.1.1) includes many new and changed administrative features, including the following:

- The inclusion of Oracle WebLogic Server in Oracle Fusion Middleware, replacing Oracle Containers for Java EE. Oracle WebLogic Server is an enterprise-ready Java application server that supports the deployment of mission-critical applications in a robust, secure, highly available, and scalable environment. For an overview of Oracle WebLogic Server and the Oracle Fusion Middleware environment, see [Section 2.1](#).
- New commands for many functions. Many components and services now use Oracle WebLogic Server Scripting Tool (WLST) commands. For example, commands to configure log files are WLST commands. See [Section 3.5.1](#) for general information about invoking WLST.
- The Oracle Metadata Services (MDS) Repository, a particular type of repository that 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 about the MDS Repository, see [Section 14.3](#).
- Wallet and Keystore Management: 11g Release 1 (11.1.1) provides new features for managing Oracle wallets and JKS keystores:
 - Password-protected wallets

When creating a wallet in prior releases, the administrator was always required to create a password-protected wallet. Once this wallet was created, the administrator could optionally create an auto-login wallet. Components needed the auto-login wallet at run time. Without an auto-login wallet, the password had to be specified in the component configuration file in an encrypted or obfuscated format.

In 11g Release 1 (11.1.1), this behavior has changed. Every time you create a password-protected wallet, an auto-login wallet is automatically created as well. This enables management tasks to be performed on the password-protected wallet, while components can use the auto-login wallet at run time. This eliminates the need to store passwords in configuration files.

To take advantage of this feature when creating a wallet with Fusion Middleware Control, you must uncheck the auto-login check box and enter the wallet password. Remember that this creates both the password-protected and auto-login wallets.

A new type of wallet has also been introduced, which is a standalone auto-login wallet. This wallet can be used for both management and run time without requiring a password. To create this wallet when creating a wallet with Fusion Middleware Control, check the auto-login check box. You do *not* need to provide a password for this type of wallet.

Note: The standalone auto-login wallet is the default choice for wallet creation.

– Wallet and Keystore Management Tools

In prior releases, Oracle Wallet Manager was the graphical interface tool and `orapki` the command-line tool to manage Oracle wallets.

In 11g Release 1 (11.1.1), you can use the Web-based interface Fusion Middleware Control or the WLST command-line tool to manage both Oracle wallets and JKS keystore files. One advantage of these new tools is that they allow you to manage keystores centrally across instances, because they work in the context of a management server.

While Oracle Wallet Manager is still available, its usage should be limited to PKCS#11 wallets (that is, Hardware Security Module integration).

You can still use `orapki` to manage both Oracle wallets and JKS keystores, but only local changes (on a per-instance basis) are possible. `orapki` allows you to manage PKCS#11 wallets and CRLs.

The following table shows the different tools and their capabilities:

Tool	Oracle Wallet	Java Keystore (JKS)	Local Updates	Distributed Updates	PKCS11	CRL	Graphical UI	Command Line
<code>orapki</code> (10g, 11g)	x	x	x		x	x		x
Oracle Wallet Manager (not in 11g)	x		x		x		x	
Fusion Middleware Control (new in 11g)	x	x		x			x	
WLST (new in 11g)	x	x		x				x

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"](#)

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:

- [What Is Oracle Fusion Middleware?](#)
- [Oracle Fusion Middleware Components](#)

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, 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 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 also provides an out-of-the-box, enterprise-ready customizable application, WebCenter 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*

- 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 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 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*

- Oracle Enterprise Content Management Suite, 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.

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

Understanding Oracle Fusion Middleware Concepts

This chapter provides information about Oracle Fusion Middleware concepts that are related to administering Oracle Fusion Middleware.

- [Understanding Key Oracle Fusion Middleware Concepts](#)
- [What Is an Oracle WebLogic Server Domain?](#)
- [What Is an Oracle Instance?](#)
- [What Is a Middleware Home?](#)
- [What Is a WebLogic Server Home?](#)
- [What Is an Oracle Home and the Oracle Common Home?](#)
- [What Is the Oracle Metadata Repository?](#)

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 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:

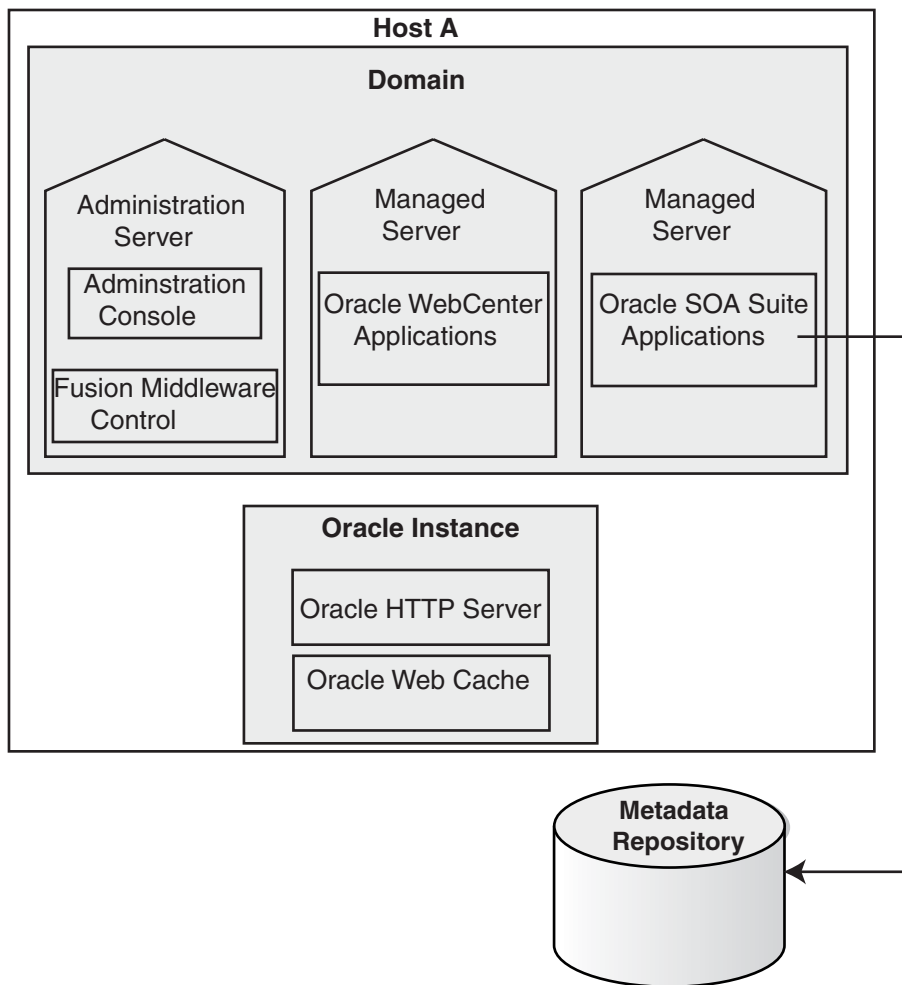
- An Oracle WebLogic Server domain, which contains one Administration Server and one or more Managed Servers. The Administration Server contains Oracle WebLogic Server Administration Console and Fusion Middleware Control. The Managed Servers contain components, such as Oracle WebCenter and Oracle SOA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

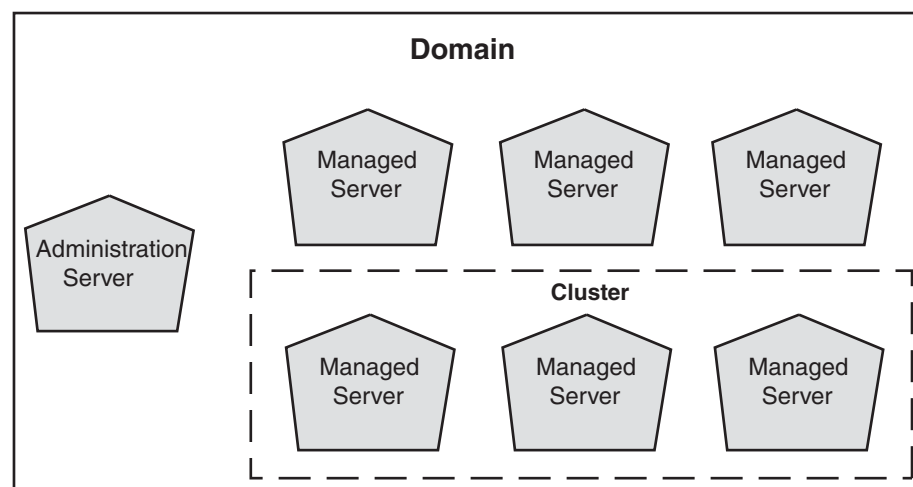
Managed Servers in a domain can be grouped together into a cluster.

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, 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, 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, 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 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.

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"](#)

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:

- [Setting Up Environment Variables](#)
- [Overview of Oracle Fusion Middleware Administration Tools](#)
- [Getting Started Using Oracle Enterprise Manager Fusion Middleware Control](#)
- [Getting Started Using Oracle WebLogic Server Administration Console](#)
- [Getting Started Using Command-Line Tools](#)
- [Getting Started Using the Fusion Middleware Control MBean Browsers](#)
- [Managing Components](#)
- [Changing the Administrative User Password](#)
- [Basic Tasks for Configuring and Managing Oracle Fusion Middleware](#)

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	<i>hostname:display_number.screen_number</i>
	Very few tools, such as <code>oidadmin</code> , require the DISPLAY variable.

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

Environment Variable	Value
LD_LIBRARY_PATH	<p>On Solaris, ensure that the value contains the following directory:</p> <p><code>\$ORACLE_HOME/lib32</code></p> <p>On Linux and HP-UX, ensure that the value contains the following directory:</p> <p><code>\$ORACLE_HOME/lib</code></p> <p>On IBM AIX, ensure that this environment variable is not set.</p>
(IBM AIX only) LIBPATH	<p>If the calling application is a 32-bit application, ensure that the value contains the following directory:</p> <p><code>\$ORACLE_HOME/lib32</code></p> <p>If the calling application is a 64-bit application, ensure that the value contains the following directory:</p> <p><code>\$ORACLE_HOME/lib</code></p>
(Solaris only) LD_LIBRARY_PATH_64	<p>Ensure that the value contains the following directory:</p> <p><code>\$ORACLE_HOME/lib</code></p>
(HP-UX only) SHLIB_PATH	<p>Ensure that the value contains the following directory:</p> <p><code>\$ORACLE_HOME/lib32</code></p>
MW_HOME	<p>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:</p> <p><code>/scratch/Oracle/Middleware</code></p>
ORACLE_HOME	<p>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:</p> <p><code>/scratch/Oracle/Middleware/ORACLE_HOME_SOA1</code></p>
ORACLE_INSTANCE	<p>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:</p> <p><code>/scratch/Oracle/Middleware/WebTier/instances/instance1</code></p>
PATH	<p>Ensure that the value contains the following directories, which contains basic commands used by all installations:</p> <p><code>\$ORACLE_COMMON_HOME/bin</code> <code>\$ORACLE_COMMON_HOME/common/bin</code></p> <p>When you start to work with specific components, you may want to add additional directories to your path, as recommended by the component documentation.</p>
JAVA_HOME	<p>Ensure that the value contains the following directory:</p> <p><code>MW_HOME/jdkn</code></p>
CLASSPATH	<p>Ensure that the value contains the following directories:</p> <p><code>\$ORACLE_HOME/lib:MW_HOME/jdkn/lib</code></p>

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\jdkn
CLASSPATH	Ensure that the value contains the following directories: ORACLE_HOME\lib:MW_HOME\jdkn\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 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
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

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

Task	Tool to Use
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: <i>DOMAIN_HOME/servers/server_name/logs/access.log</i> <i>DOMAIN_HOME/servers/server_name/data/ldap/log/EmbeddedLDAP.log</i> <i>DOMAIN_HOME/servers/server_name/data/ldap/log/EmbeddedLDAPAccess.log</i>
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.
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.

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).

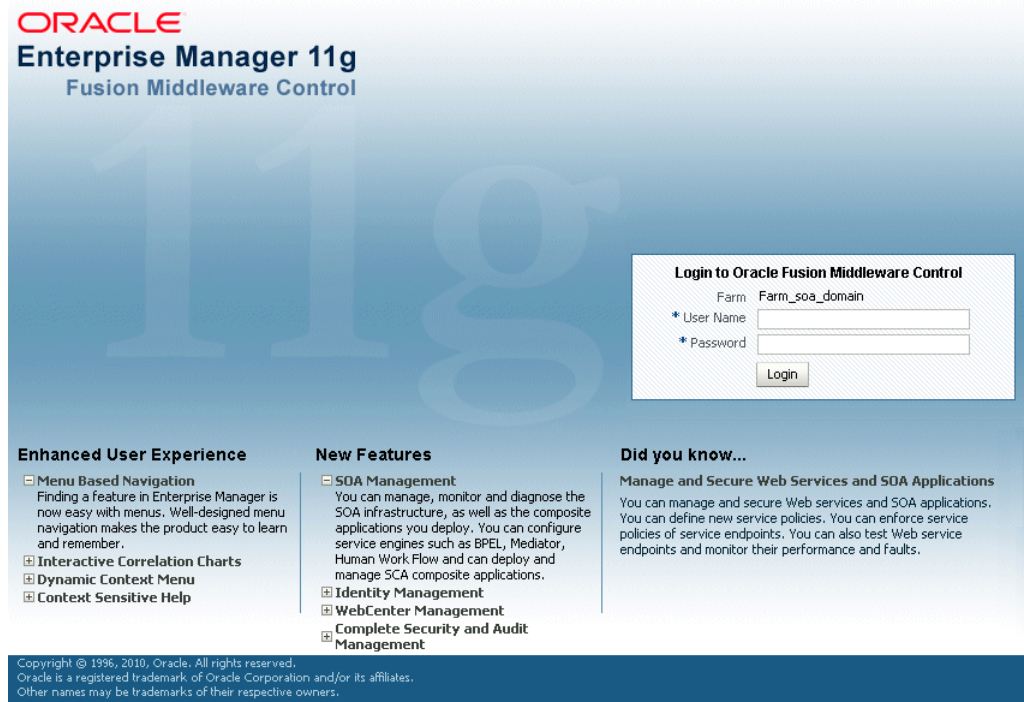
For other installation types, the information is displayed on the Create Domain screen of the Configuration Wizard when the configuration completes.

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.

The screenshot shows the Oracle Enterprise Manager Fusion Middleware Control interface. The interface is divided into a Target Navigation Pane on the left and a Content Pane on the right. The Target Navigation Pane shows a tree view of the farm structure, including Farm_soa_domain, Application Deployments, SOA, WebLogic Domain, soa_domain, AdminServer, soa_server1, Metadata Repositories, mds-soa, and User M... A right-click context menu is open over the 'soa_server1' target, showing options like Home, Administration, and General Information. The Content Pane displays the 'soa_server1' summary page, which includes a 'Response and Load' graph showing Request Processing Time (ms) and Requests (per minute) over time. The graph shows a spike in request processing time around 08:05. Below the graph is a table of 'Application Deployments' with columns for Name, Status, Active Sessions, Request Processing Time (ms), and Bean Access r.

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. For information about configuring Single Sign-On between Fusion Middleware Control and the WebLogic Server Administration Console, see "Configuring Single Sign-On for Administration Consoles" in the *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle Identity Management*.

[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: <i>logfile_name</i>
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.

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

To:	Take This Action:
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 screenshot shows the Oracle Enterprise Manager 11g Fusion Middleware Control interface. The main content area is titled "Farm_soa_domain" and shows a green 100% status indicator. Below this, there are two tables: "Deployments" and "Fusion Middleware".

Deployments Table:

Name	Status	Target
Application Deployments		
Internal Applications		
Resource Adapters		
BPMComposer	Up	soa_server1
BPMComposerServices	Up	soa_server1
composer	Up	soa_server1
DefaultToDoTaskFlow	Up	soa_server1
mdsappdb	Up	soa_server1
oracle-bam(11.1.1)	Up	bam_server1
OracleBPMComposerR	Up	soa_server1
OracleBPMProcessRole	Up	soa_server1
OracleBPMWorkspace	Up	soa_server1
SimpleApprovalTaskFlc	Up	soa_server1
worklistapp	Up	soa_server1
SOA		
soa-infra	Up	soa_server1
default	Up	
SimpleApproval	Up	

Fusion Middleware Table:

Name	Status	Host	CPU Usage (%)
WebLogic Domain			
soa_domain			
AdminServer	Up	example.com	21.19
bam_server1	Up	example.com	3.85
soa_server1	Up	example.com	28.75
BAM			
OracleBamServer (b	Up	example.com	
OracleBamWeb (bar	Up	example.com	
Metadata Repositories			
mds-mds_repos_file		example.com	
mds-owsm		example.com	
mds-soa		example.com	
User Messaging Service			
usermessagingdriver	Up	example.com	
usermessagingdriver	Up	example.com	
usermessagingserve	Up	example.com	

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:

soa-infra | Logged in as weblogic | host .
Page Refreshed Oct 20, 2010 2:21:31 PM GMT

Dashboard | Deployed Composites | Instances | Faults and Rejected Messages

Recent Composite Instances
Show Only Running Instances Running 0 Total 8

Instance ID	Composite	Start
8	SRDemoComposite [2.0]	Sep 4, 2009 10:35:3
7	Composite3Events [1.0]	Sep 4, 2009 10:35:2
6	SRDemoComposite [2.0]	Sep 3, 2009 5:20:3
5	Composite3Events [1.0]	Sep 3, 2009 5:20:3
4	Composite3Events [1.0]	Sep 3, 2009 5:18:1
3	SRDemoComposite [2.0]	Sep 3, 2009 5:18:0
2	SRDemoComposite [2.0]	Sep 3, 2009 12:11:5
1	SRDemoComposite [2.0]	Sep 3, 2009 12:10:2

» Show All

Deployed Composites

Composite	Status	Mode	Instances	Faults
SRDemoComposi	Active	Active	0	0
sdpmessagingssc	Active	Active	0	0

» Show All (2)

Recent Faults and Rejected Messages
Show only system faults:

Error Message	Recovery	Fault Time	Composite	Fault Location	Composite Instance ID
No faults found					

» Show All

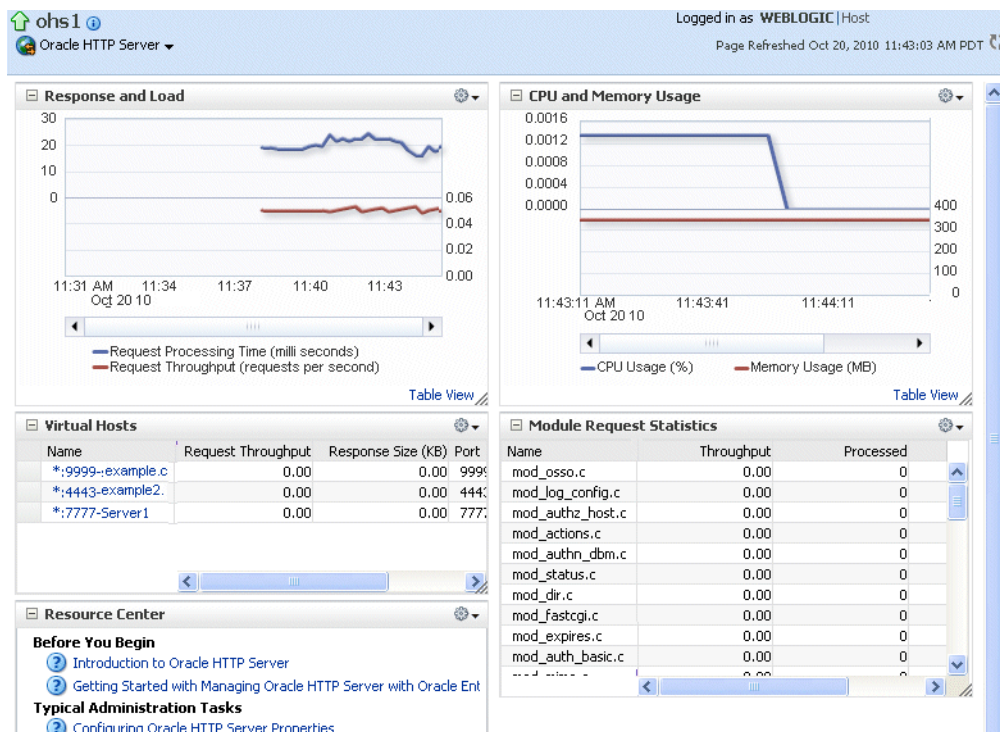
Service Engines | Composite Instances and Faults

- 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:

- In the target navigation pane, expand the farm, then **Web Tier**.
- 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:



- 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:

- From the target navigation pane, expand the farm and then **Application Deployments**.
- 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:

- From the target navigation pane, expand the farm, then **SOA**, and then **soa-infra**.
- 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:

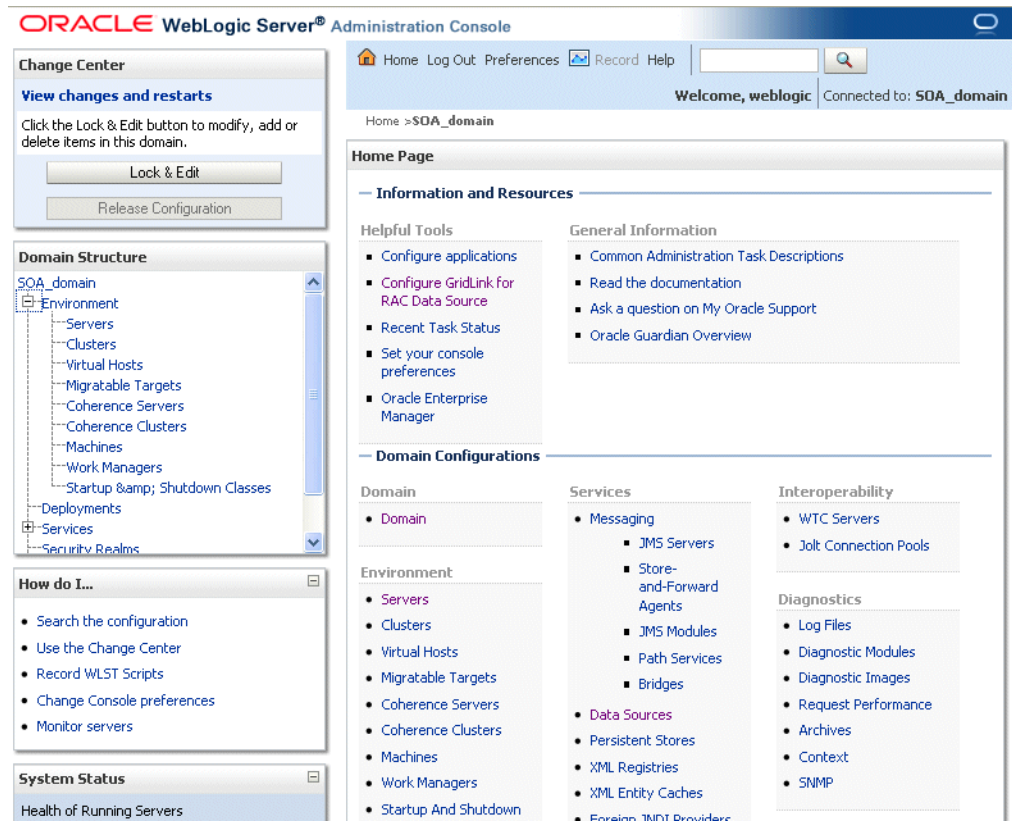
```
http://hostname:port_number/console
```

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.

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, 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 directory, which you should add to your PATH environment variable:

```
(UNIX) ORACLE_HOME/opmn/bin
(Windows) ORACLE_HOME\opmn\bin
```

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 you to 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, 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**.

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.

Starting and Stopping Oracle Fusion Middleware

This chapter describes procedures for starting and stopping Oracle Fusion Middleware.

It contains the following topics:

- [Overview of Starting and Stopping Procedures](#)
- [Starting and Stopping Oracle WebLogic Server Instances](#)
- [Starting and Stopping Components](#)
- [Starting and Stopping Fusion Middleware Control](#)
- [Starting and Stopping Oracle Management Agent](#)
- [Starting and Stopping Applications](#)
- [Starting and Stopping Your Oracle Fusion Middleware Environment](#)
- [Starting and Stopping: Special 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:

- [Starting and Stopping Administration Servers](#)
- [Starting and Stopping Managed Servers](#)
- [Enabling Servers to Start Without Supplying Credentials](#)
- [Configuring Node Manager to Start Managed Servers](#)

4.2.1 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]
```

4.2.2 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](#)

4.2.2.1 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, or Oracle JRF, you must first configure Node Manager, as described in [Section 4.2.4](#).

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**.

4.2.2.2 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
```

4.2.3 Enabling Servers to Start Without Supplying Credentials

You can enable the Administration Server and Managed Servers to start without prompting you for the administrator username 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 4.2.1](#) and [Section 4.2.2](#).

Note: When you start the Administration Server or Managed Server, the username 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

4.2.4 Configuring Node Manager to Start Managed Servers

If a Managed Server contains other Oracle Fusion Middleware products, such as Oracle SOA Suite, Oracle WebCenter, 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 domain directory.

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.

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

4.3 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_HOME/opmn/bin
(Windows) ORACLE_HOME\opmn\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 4.2.1](#).

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 4.2.1](#).
3. If you have not already done so, configure Node Manager, as described in [Section 4.2.4](#).
4. Ensure the Node Manager is running. If the 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:


```
opmnctl startall
```
6. Start the Oracle WebLogic Server Managed Servers as described in [Section 4.2.2.2](#). 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:


```
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:


```
opmnctl stopall
```
- 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:


```
opmnctl stopall
```
- 5. Stop the Administration Server as described in [Section 4.2.1](#).
- 6. If you want to stop the 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 11g documentation library

Managing Ports

This chapter describes how to view and change Oracle Fusion Middleware port numbers.

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')
```

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	0:0:0:0:0:0:1	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	0:0:0:0:0:0:1	WLS_Services	Default[iiop][2]	iiop
8888	139.185.136.176	WLS_Spaces	Default[ldap]	ldap
8890	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	0:0:0:0:0:0:1	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	0:0:0:0:0:0:1	AdminServer	Default[iiop][2]	iiop
8889	0:0:0:0:0:0:1	WLS_Portlet	Default[t3][2]	t3
8888	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](#).

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
```

5.3.5 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 10g and Oracle Database 11g 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 10g or Oracle Database 11g 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" 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"
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"
- 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

If you are using Oracle Single Sign-On, 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 `PlsqlDatabaseConnectionString`.

- 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:

- [Chapter 6, "Configuring SSL in Oracle Fusion Middleware"](#)
- [Chapter 7, "Using the SSL Automation Tool"](#)
- [Chapter 8, "Managing Keystores, Wallets, and Certificates"](#)

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 11g Release 1 (11.1.1) due to security concerns; components or applications that used SSL version 2 in pre-11g Release 1 (11.1.1) 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

Private, or 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, 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.

Other 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.

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 using the server's certificate.
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

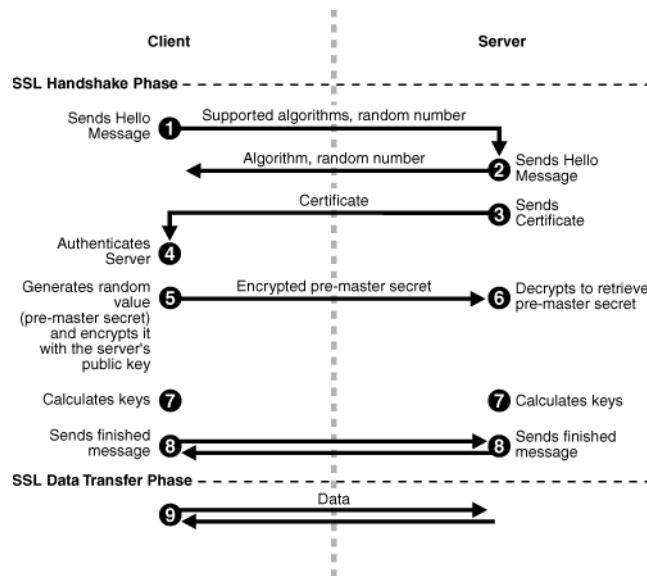
The encryption keys are symmetric, that is, the same key is used to encrypt and decrypt the data.

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



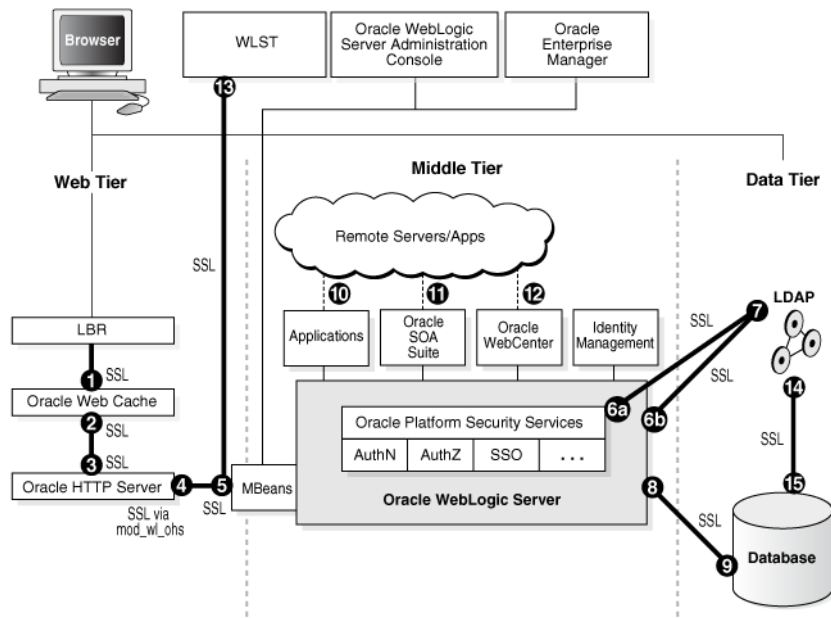
6.2 About SSL in Oracle Fusion Middleware

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"](#)
- 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 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 11g Release 1 (11.1.1), 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, "Keystores and Wallets"](#).

See Also: [Section 8.1, "Key and Certificate Storage in Oracle Fusion Middleware"](#) 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.
- 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.

Advanced Tools

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

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

See Also: [Section 8.1, "Key and Certificate Storage in Oracle Fusion Middleware"](#) 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.

For a list of all the configuration tools, see [Section 6.2.4, "Tools for SSL Configuration."](#)

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.

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](#)

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, "Enable Outbound SSL for Oracle Web Cache Using Fusion Middleware Control"](#).

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, "Managing Keystores, Wallets, and Certificates"](#).

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

The SSL Configuration page contains two sets of information:

Web Cache Page Refreshed Feb 6, 2009 2:12:21 PM PST

SSL Configuration
All ports configured for this system component are shown. SSL can be configured for a port during edit operation.

Port	Port Type	Host Name	IP Address	SSL Enal
7778	NORM	localhost	ANY	
7782	NORM	localhost	ANY	✓
7779	ADMINISTRATION	localhost	ANY	
7781	INVALIDATION	localhost	ANY	
7780	STATISTICS	localhost	ANY	

SSL Communication Between WebCache and Oracle HTTP Server (OHS)
Select the wallet that would be used for communication between this WebCache instance and Oracle HTTP Servers (OHS).
Client Wallet Name: default

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, "Enable Outbound SSL for Oracle Web Cache Using Fusion Middleware Control"](#).

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

Web Cache Page Refreshed Apr 17, 2009 2:55:17 PM PDT

Ports Configuration > Edit Port

Information
All changes made in this page require a server restart to take effect.

Edit Port : ANY:8090 OK Cancel

Edit attributes of a port for this system component. Ports created can listen on local IP Address of associated host or any of available network interfaces.

Endpoint Attributes

Port Type: NORM

Endpoint Name: ANY:8090

IP Address: ANY

* Port: 8090

SSL Configuration
Server side SSL attributes for listening endpoint.

Enable SSL

Server Wallet Name: default

Advanced SSL Settings

Server SSL properties

SSL Authentication: Server Authentication

* SSL Protocol Version: All

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. 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:

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 `CACHE.index1.LISTEN.index1`.

2. Configure the listening endpoint with SSL properties:

```
configureSSL('inst1',
            'wcl',
            '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.
-
-

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**.

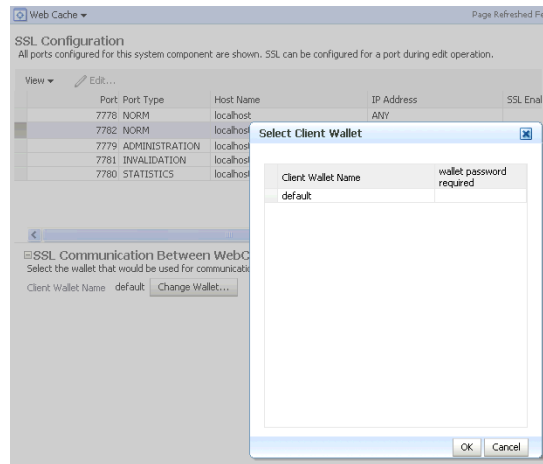
Web Cache Page Refreshed Feb 6, 2009 2:12:21 PM PST

SSL Configuration
All ports configured for this system component are shown. SSL can be configured for a port during edit operation.

View	Port	Port Type	Host Name	IP Address	SSL Enal
	7778	NORM	localhost	ANY	
	7782	NORM	localhost	ANY	✓
	7779	ADMINISTRATION	localhost	ANY	
	7781	INVALIDATION	localhost	ANY	
	7780	STATISTICS	localhost	ANY	

SSL Communication Between WebCache and Oracle HTTP Server (OHS)
Select the wallet that would be used for communication between this WebCache instance and Oracle HTTP Servers (OHS).
Client Wallet Name: default

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



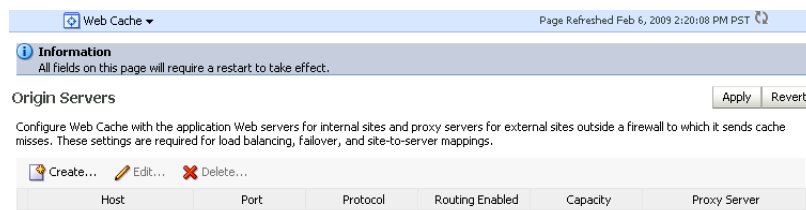
3. Select the desired wallet for outbound SSL and click **OK**.

SSL Configuration

Take these steps:

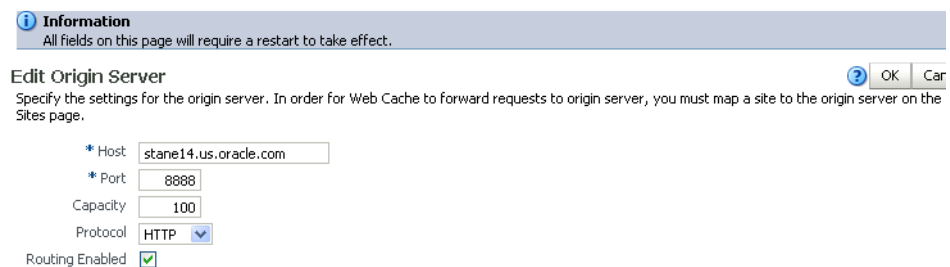
1. Navigate to the Oracle Web Cache instance, then **Administration**, then **Origin Servers**.

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**. 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

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.

For inbound traffic:

- [Section 6.4.3.1, "Enable SSL for Inbound Requests to Oracle HTTP Server Virtual Hosts Using Fusion Middleware Control"](#)
- [Section 6.4.3.2, "Enable SSL for Inbound Requests to Oracle HTTP Server Virtual Hosts Using WLST"](#)

For outbound traffic:

- [Section 6.4.3.3, "Enable SSL for Outbound Requests from Oracle HTTP Server"](#)

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

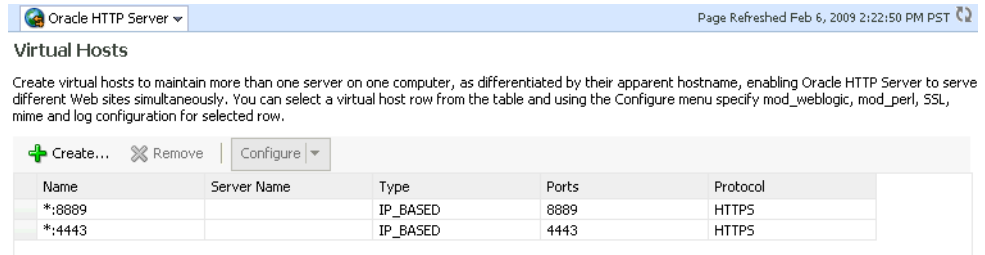
You can SSL-enable inbound traffic to Oracle HTTP Server virtual hosts using these steps:

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**.

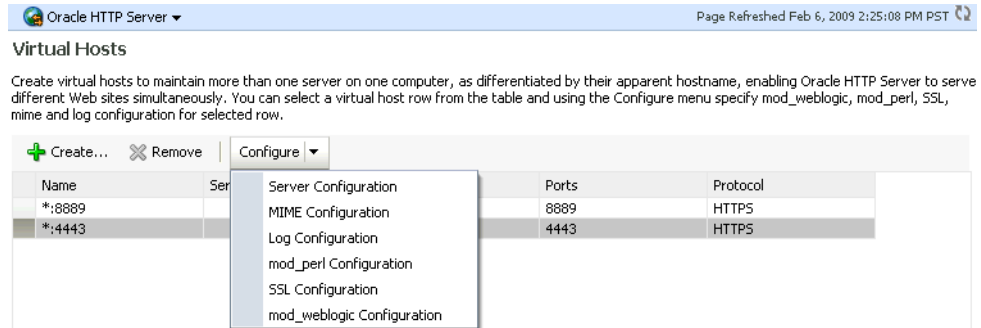
For details about wallet creation and maintenance, see [Chapter 8, "Managing Keystores, Wallets, and Certificates"](#).

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**.

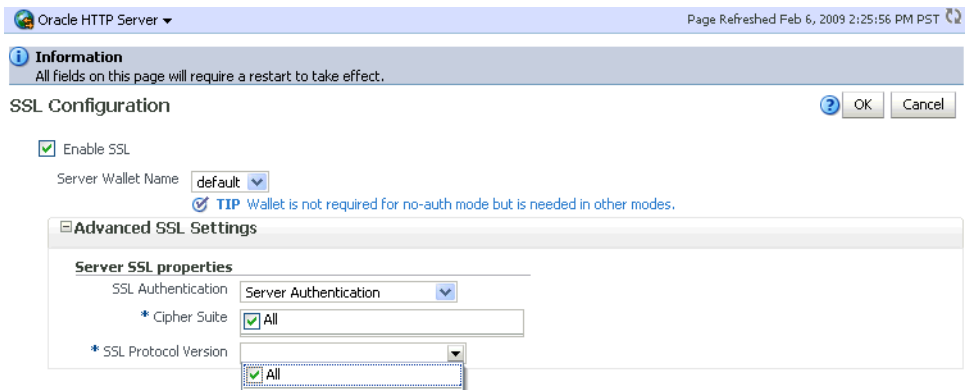


The SSL Configuration page appears.

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: 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, "Properties Files for SSL"](#).

6. Click **OK** to apply the changes.
7. Restart the Oracle HTTP Server instance by navigating to **Oracle HTTP Server**, then **Control**, then **Restart**.
8. 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.

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.
 - We could also specify a properties file as a parameter to `configureSSL`; see [Table 6-4](#) for details.
-
-

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`.

The steps are as follows:

1. Generate a custom keystore for Oracle WebLogic Server (see [Section 6.5.1, "Configuring SSL for Oracle WebLogic Server"](#)) 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.
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
WLSSEWallet "${ORACLE_INSTANCE}/config/OHS/ohs1/keystores/default"
</IfModule>
```

Save the file and exit. Restart Oracle HTTP Server to activate the changes.

4. 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.

`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**.

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](#)
- [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 for policy stores and credential stores connections to an LDAP directory. Anonymous and one-way SSL is supported.

Anonymous SSL (Server-side)

Start the LDAP server in anonymous authentication mode.

For Oracle Internet Directory, see [Section 6.6.1.1, "Enable Inbound SSL on an Oracle Internet Directory Listener Using Fusion Middleware Control"](#).

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 Java EE 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, "SSL-Enable Oracle Database."](#)

See Also: Configuring Secure Sockets Layer Authentication in the *Oracle Database Advanced Security Administrator's Guide* 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, "SSL-Enable a Data Source."](#)

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
- How to enable SSL for a Web service

For these and related topics, see *Oracle Fusion Middleware Securing 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

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

- Securing the WebCenter Spaces Connection to Oracle Content Server with SSL
- Securing the Browser Connection to WebCenter 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 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 Identity Federation

See "Configuring SSL for Oracle Identity Federation" in the *Oracle Fusion Middleware Administrator's Guide for Oracle Identity Federation* for details.

Note: Use Sun Microsystems' `keytool` utility to manage keystores and certificates required for SSL configuration in Oracle Identity Federation.

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, "Enable Inbound SSL for Oracle Web Cache Using Fusion Middleware Control."](#)
- Inbound SSL for Oracle HTTP Server
See [Section 6.4.3.1, "Enable SSL for Inbound Requests to Oracle HTTP Server Virtual Hosts Using Fusion Middleware Control."](#)
- Inbound SSL for Oracle WebLogic Server
See [Section 6.5.1.1, "Inbound SSL to Oracle WebLogic Server."](#)
- SSL between Oracle Web Cache and Oracle HTTP Server
See [Section 6.4.2.3, "Enable Outbound SSL for Oracle Web Cache Using Fusion Middleware Control."](#)
- SSL between Oracle HTTP Server and Oracle WebLogic Server
See [Section 6.4.3.3, "Enable SSL for Outbound Requests from Oracle HTTP Server."](#)

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, "Enable Inbound SSL for Oracle Web Cache Using Fusion Middleware Control."](#)
- Inbound SSL for Oracle HTTP Server

See [Section 6.4.3.1, "Enable SSL for Inbound Requests to Oracle HTTP Server Virtual Hosts Using Fusion Middleware Control."](#)

- Inbound SSL for Oracle WebLogic Server
See [Section 6.5.1.1, "Inbound SSL to Oracle WebLogic Server."](#)
- SSL between Oracle Web Cache and Oracle HTTP Server
See [Section 6.4.2.3, "Enable Outbound SSL for Oracle Web Cache Using Fusion Middleware Control."](#)
- SSL between Oracle HTTP Server and Oracle WebLogic Server
See [Section 6.4.3.3, "Enable SSL for Outbound Requests from Oracle HTTP Server."](#)

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 installing a security certificate on the Discoverer Plus client machine, importing certificate details into the Java Plug-in certificate store, and related tasks.

The *Oracle Fusion Middleware Configuration Guide for Oracle Business Intelligence Discoverer* provides details about configuring SSL for Oracle Discoverer in these sections:

- About Running Discoverer over HTTPS
- About Discoverer and the Oracle Fusion Middleware Security Model

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, "Best Practices for Application Developers."](#)

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

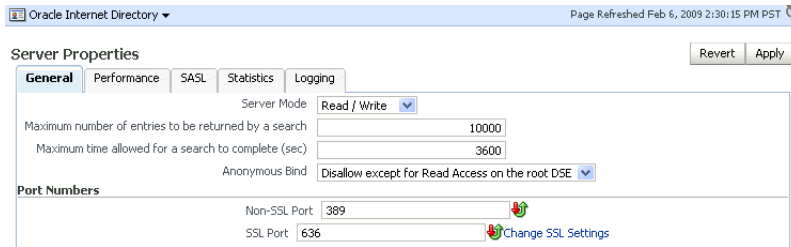
Out of the box, Oracle Internet Directory nodes are SSL-enabled in no-auth mode.

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

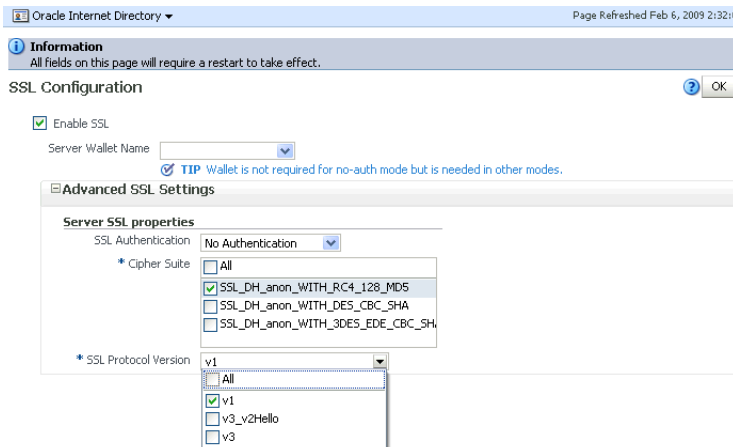
6.6.1.1 Enable Inbound SSL on an Oracle Internet Directory Listener Using Fusion Middleware Control

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.

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.
-

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, "SSL-Enable Oracle Database."](#)

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, "Creating a Keystore Using Fusion Middleware Control"](#) 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**.

Oracle Virtual Directory Administration - Listeners

Oracle Virtual Directory provides services to clients through connections known as listeners. There two types of Listeners: LDAP and HTTP, and both can be protected using SSL. This page allows you to configure listeners.

Name	Enabled	Type	Threads	Listening Port
LDAP Endpoint	✓	LDAP	10	6501
LDAP SSL Endpoint	✓	LDAPS	10	7501
Admin Gateway	✓	ADMINS	10	8899
DSML Gateway	✗	HTTP	10	8080

The Edit Listener page appears:

Oracle Virtual Directory Administration - Edit Listener - LDAP SSL Endpoint

Information: All fields on this page will require a restart to take effect.

Listener Type: LDAP SSL Endpoint

Basic Configuration:

- Listener Name: LDAP SSL Endpoint
- Listener Port: 7501
- Threads: 10
- SSL Configuration Status: Enabled
- Listener Enabled:

5. Click **Change SSL Settings**.
6. On the SSL Settings dialog:

Oracle Virtual Directory Administration - SSL Configuration

Information: All fields on this page will require a restart to take effect.

SSL Configuration

Enable SSL

Server Keystore Name: test

* Server Keystore Password: [masked]

Server Truststore Name: test

* Server Truststore Password: [masked]

Advanced SSL Settings

Server SSL properties

SSL Authentication: Server Authentication

* Cipher Suite: All

- SSL_RSA_WITH_RC4_128_MD5
- SSL_RSA_WITH_RC4_128_SHA
- TLS_RSA_WITH_AES_128_CBC_SHA
- TLS_DHE_RSA_WITH_AES_128_CBC_SHA
- SSL_RSA_WITH_3DES_EDE_CBC_SHA

* SSL Protocol Version: v1:v2Hello

- All
- v1
- v3
- v2Hello

- Select **Enable SSL**.
- For Server Keystore Name, select the keystore you created in step 3, for example, `OVDtestJKs`.
- For Server Keystore Password, type the keystore password you specified in step 3.
- For Server Truststore Name, select the keystore you created in step 3, for example, `OVDtestJKs`.
- For Server Truststore Password, type the keystore password you specified in step 3.
- 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.
 - 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 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 this instance; 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.

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, "Changing a Self-Signed Wallet to a Third-Party Wallet."](#)

```
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 =
(AADDRESS = (PROTOCOL = IPC) (KEY = EXTPROC1))
(AADDRESS = (PROTOCOL = TCP) (HOST = mynode.mycorp.com) (PORT = 1521))
(AADDRESS = (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 =
(AADDRESS = (PROTOCOL = IPC) (KEY = EXTPROC1521))
)
(DESCRIPTION =
(AADDRESS = (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)
    )
    (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 data source, or a new data source. See *Creating a JDBC Data Source in Oracle Fusion Middleware Configuring and Managing JDBC 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:

```
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")))
```

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](#)

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.
 - 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, "Manage CRLs on the File System"](#) 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, "Managing Certificate Revocation Lists \(CRLs\) with orapki Utility."](#)

CRL Renaming to Hashed Form

If specifying a fleshiest directory, 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 wldir -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.txt -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, "Manage CRLs on the File System."](#)
3. Configure your component to use this CRL. Follow the steps outlined in [Section 6.7.2.1, "Configuring CRL Validation for a Component."](#)
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.

See [Section 6.9.28, "Properties Files for SSL"](#) 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. Ensure that the keys used in FIPS mode are at least 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, "Managing Keystores, Wallets, and Certificates."](#)
- 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 11g Release 1 (11.1.1), 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, "Command-Line Interface for Keystores and Wallets"](#) 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

Table 6–3 (Cont.) WLST Commands for Java Keystore (JKS) Management

Use this command...	To...	Use with WLST...
<code>exportKeyStoreObject</code>	Export an object (for example, a certificate) from a JKS keystore to a file.	Online
<code>generateKey</code>	Generate a keypair in a JKS keystore.	Online
<code>getKeyStoreObject</code>	Display a certificate or other object present in a JKS keystore.	Online
<code>importKeyStore</code>	Import a JKS keystore from a file.	Online
<code>importKeyStoreObject</code>	Import a certificate or other object from a file to a JKS keystore.	Online
<code>listKeyStoreObjects</code>	List all objects (for example, certificates) present in a JKS keystore.	Online
<code>listKeyStores</code>	List all JKS keystores configured for a component instance.	Online
<code>removeKeyStoreObject</code>	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
<code>instName</code>	Specifies the name of the application server instance.
<code>compName</code>	Specifies the name of the component instance.
<code>compType</code>	Specifies the type of component. Valid values are 'ohs', 'oid', and 'webcache'.
<code>walletName</code>	Specifies the name of the wallet file.
<code>password</code>	Specifies the password of the wallet.
<code>DN</code>	Specifies the Distinguished Name of the key pair entry.
<code>keySize</code>	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
<code>instName</code>	Specifies the name of the application server instance.
<code>compName</code>	Specifies the name of the component instance.
<code>compType</code>	Specifies the type of component. Valid values are 'ohs', 'oid', and 'webcache'.
<code>walletName</code>	Specifies the name of the wallet file.
<code>password</code>	Specifies the password of the wallet.
<code>DN</code>	Specifies the Distinguished Name of the key pair entry.
<code>keySize</code>	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.
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, "Properties Files for SSL."](#)

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, "Location of Keystores."](#)

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.

Argument	Definition
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, "Managing Keystores, Wallets, and Certificates."](#)

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.
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 <code>base64.txt</code> .
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
<code>instName</code>	Specifies the name of the application server instance.
<code>compName</code>	Specifies the name of the component instance.
<code>compType</code>	Specifies the type of component. Valid values are 'oid', 'ohs', and 'webcache'.
<code>walletName</code>	Specifies the name of the wallet file.
<code>password</code>	Specifies the password of the wallet.
<code>path</code>	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
<code>instName</code>	Specifies the name of the application server instance.
<code>compName</code>	Specifies the name of the component instance.
<code>compType</code>	Specifies the type of component. Valid values are 'ohs', 'oid', and 'webcache'.
<code>walletName</code>	Specifies the name of the wallet file.
<code>password</code>	Specifies the password of the wallet.
<code>type</code>	Specifies the type of wallet object to be exported. Valid values are 'CertificateRequest', 'Certificate', 'TrustedCertificate' or 'TrustedChain'.
<code>path</code>	Specifies the absolute path of the directory under which the object is exported as a file <code>base64.txt</code> .
<code>DN</code>	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
<code>instName</code>	Specifies the name of the application server instance.
<code>compName</code>	Specifies the name of the component instance.
<code>compType</code>	Specifies the type of component. Valid value is 'ovd'.
<code>keystoreName</code>	Specifies the name of the keystore.
<code>password</code>	Specifies the password of the keystore.
<code>DN</code>	Specifies the Distinguished Name of the key pair entry.
<code>keySize</code>	Specifies the key size in bits.
<code>alias</code>	Specifies the alias of the key pair entry in the keystore.
<code>algorithm</code>	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 <code>listKeyStoreObjects</code> 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.
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 <code>listWalletObjects</code> 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 server instance `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
<code>instName</code>	Specifies the name of the application server instance.
<code>compName</code>	Specifies the name of the component instance.
<code>compType</code>	Specifies the type of component. Valid value is 'ovd'.
<code>keystoreName</code>	Specifies the name of the keystore being imported. This name must be unique for this component instance.
<code>password</code>	Specifies the password of the keystore.
<code>filePath</code>	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.
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'.

Argument	Definition
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
<code>instName</code>	Specifies the name of the application server instance.
<code>compName</code>	Specifies the name of the component instance
<code>compType</code>	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.

Argument	Definition
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 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

Using the SSL Automation Tool

This chapter contains the following sections:

- [Introduction to the SSL Automation Tool](#)
- [Prerequisites](#)
- [Generating the CA Certificate](#)
- [Configuring a Component Server](#)
- [Configuring a Client](#)

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 [adc2100651.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 11g, Oracle Identity Manager, and Oracle 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	<code>WLS_SSL_Server_Config.sh</code>	Oracle WebLogic Server and Java EE components
oid	<code>OID_SSL_Server_Config.sh</code>	Oracle Internet Directory server
ovd	<code>OVD_SSL_Server_Config.sh</code>	Oracle Virtual Directory server
oam	<code>OAM_SSL_Server_Config.sh</code>	Oracle Access Manager 10g server

Provide information when prompted.

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 [adc2100651.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 certificate 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) [adc2100651.example.com]:
The subject DN is
  cn=adc2100651.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 [adc2100651.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 certificate 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) [adc2100651.example.com]:
The subject DN is
cn=adc2100651.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 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
>>> Invoking /mw784/im7335/bin/ldapbind -h adc2100651.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 [adc2100651.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 certificate 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) [adc2100651.example.com]:
The subject DN is
cn=adc2100651.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 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
/mw784/im7335/bin/ldapbind -h adc2100651.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 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 [adc123.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) []:adc123.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 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 OIM or OAM 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 [adc2100651.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 certificate 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 certificate 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 [adc2100651.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 certificate 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 certificate 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 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 [adc123.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 [adc123.example.com]:
```

```
>>>Enter the Access Server Port [6021]:
```

```
>>>Enter Access Server ID: aa1
```

```
>>>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) []:adc123.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.                                                    ***  
*****
```

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 11g Release 1 (11.1.1), all Java components and Java EE applications use the JKS-based keystore and truststore.

You use a JKS-based keystore for the following:

- Oracle Virtual Directory
- Applications deployed on Oracle WebLogic Server, including:
 - Oracle SOA Suite
 - Oracle WebCenter

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

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

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 <code>orapki</code> for PKCS#11 or 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	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-11g wallets (corresponding to 10g Release 10.1.2 and 10.1.3 formats) are supported in 11g Release 1 (11.1.1).

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 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-11g keystore or wallet that you wish to use must be imported using either Fusion Middleware Control or the WLST utility.

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 and for applications deployed in Oracle WebLogic Server.

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.

Oracle Virtual Directory ▼ Page Refreshed Feb 6, 2009 2:40:38 PM PST

Keystores > Create JKS Keystore

Create JKS Keystore

To create a keystore, enter a keystore name and password. The keystore name should be unique within a component. Passwords have a minimum length of eight characters, and contain alphabetic characters combined with numeric or special characters.

Keystore Details

* Keystore Name

* Keystore Password

* Confirm Password

Add Self-Signed Certificate

Add a self-signed certificate that becomes part of the keystore. Alias must be unique within a keystore.

Create Keystore with Self-signed Certificate

* Alias

* Common Name

Organizational Unit

Organization

City

State

Country

Key Size

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.

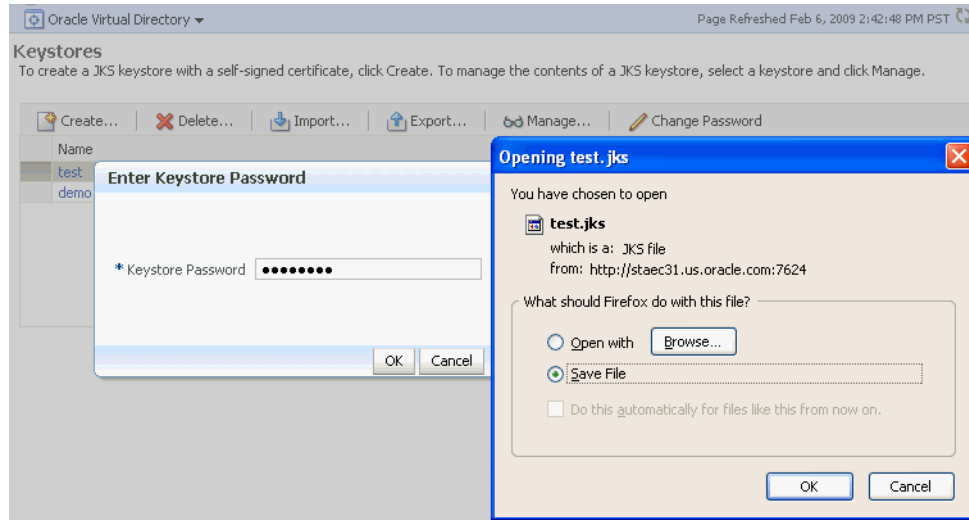
See Also: [Section 6.9.6, "createKeystore"](#).

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.

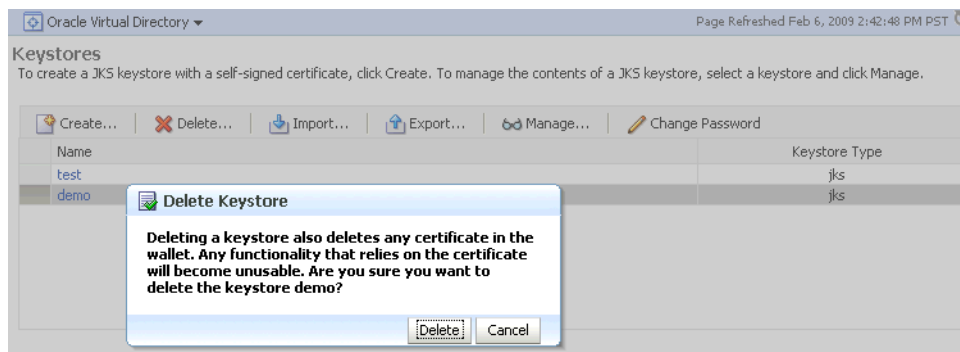
This command exports the keystore into a file named `test` under the directory `/tmp`.

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.

Oracle Virtual Directory Page Refreshed Feb 6, 2009 2:46:41 PM PST

Keystores > Import JKS Keystore

Import JKS Keystore OK Cancel

Click "Browse" to pick the keystore from the local file system. Keystores usually have a ".jks" extension. Ensure the keystore name is unique for the component. Specify the password for the keystore you are importing.

File:

* Keystore Name:

* Keystore Password:

6. Click **OK**.
7. The imported keystore appears in the list of Java keystores.

Oracle Virtual Directory Page Refreshed Feb 6, 2009 2:48:27 PM PST

Keystores

To create a JKS keystore with a self-signed certificate, click Create. To manage the contents of a JKS keystore, select a keystore and click Manage.

Name	Keystore Type
test	jks
demojks	jks
demo	jks

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.

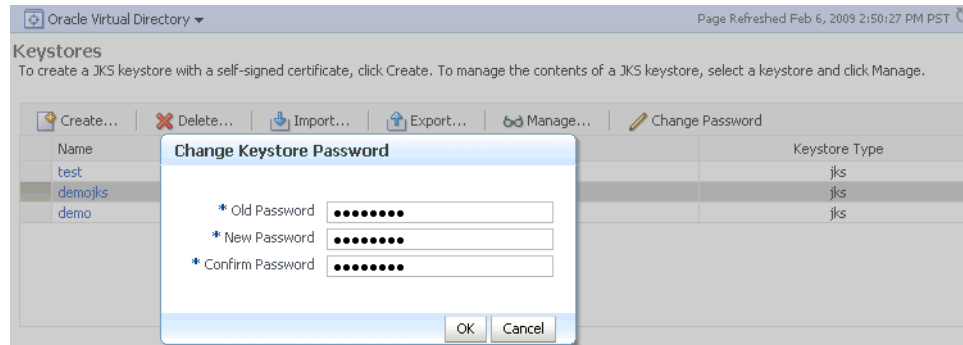
See Also: [Section 6.9.18, "importKeystore"](#).

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](#)

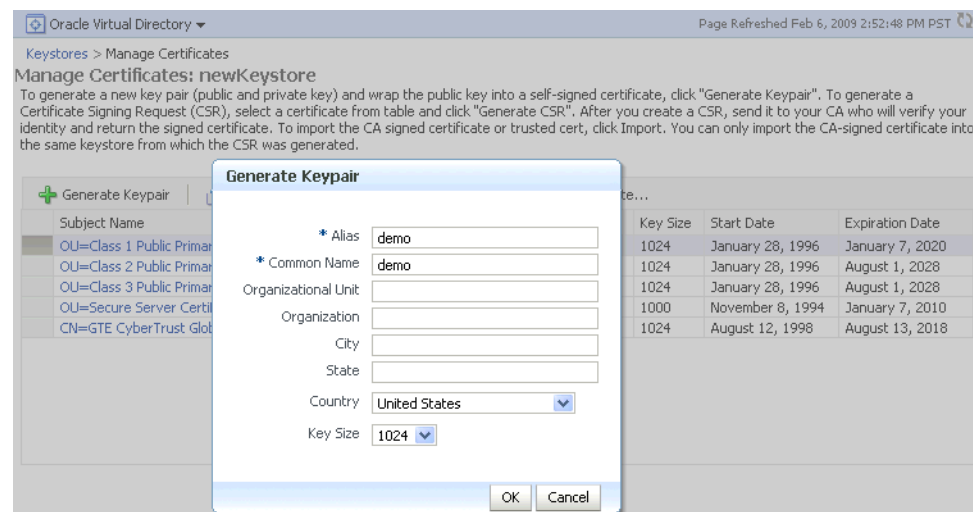
- [Exporting a Certificate or Trusted Certificate from the Keystore Using Fusion Middleware Control](#)
- [Exporting a Certificate or Trusted Certificate from the Keystore Using WLST](#)
- [Deleting a Certificate or Trusted Certificate from the Keystore Using Fusion Middleware Control](#)
- [Deleting 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:

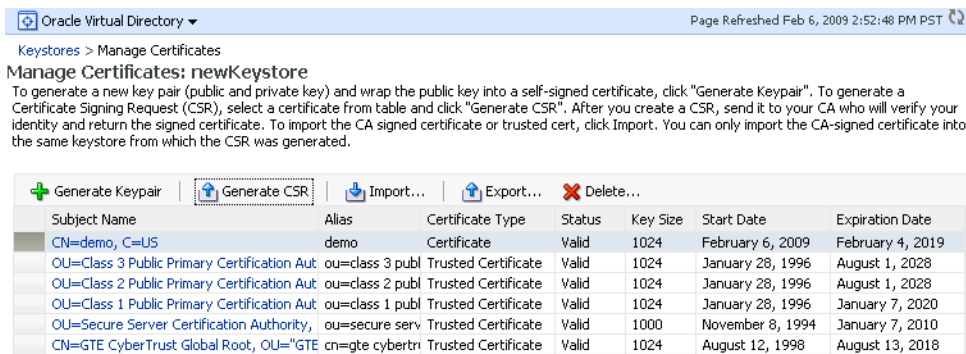
```
generateKey('inst1', 'ovd5', 'ovd', 'newKeystore', 'password', 'subject_dn', 'key_size', 'alias')
```

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.

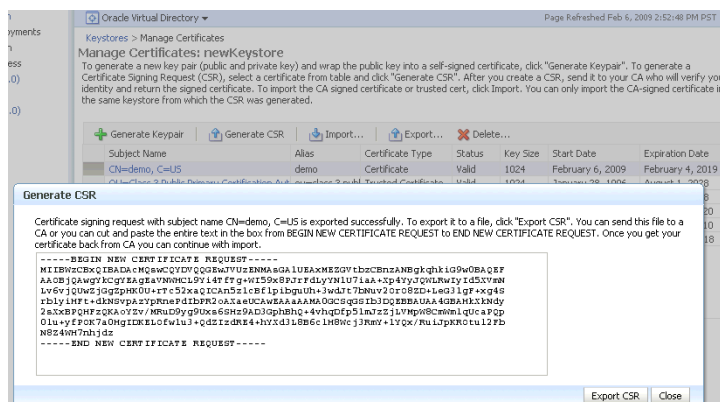
8.3.5.3 Generating a Certificate Signing Request Using Fusion Middleware Control

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 paste it to a file.
 - Click the **Export CSR** button to directly save it to a file.



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')
```

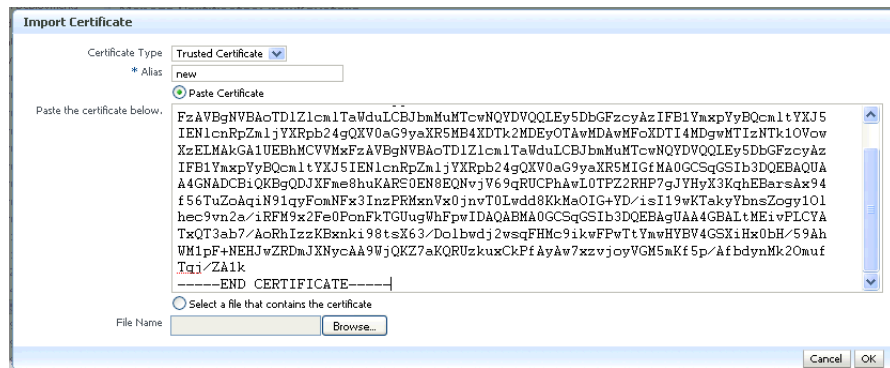
where `password` is the password for this keystore, `/tmp` is the path under which the certificate request is generated in BASE64 format in the file `base64.txt`, and `alias` is the alias of the key pair that is used to generate the certificate request.

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.

Oracle Virtual Directory Page Refreshed Feb 6, 2009 2:52:48 PM PST

Keystores > Manage Certificates

Manage Certificates: newKeystore

To generate a new key pair (public and private key) and wrap the public key into a self-signed certificate, click "Generate Keypair". To generate a Certificate Signing Request (CSR), select a certificate from table and click "Generate CSR". After you create a CSR, send it to your CA who will verify your identity and return the signed certificate. To import the CA signed certificate or trusted cert, click Import. You can only import the CA-signed certificate into the same keystore from which the CSR was generated.

Subject Name	Alias	Certificate Type	Status	Key Size	Start Date	Expiration Date
CN=demo, C=US	demo	Certificate	Valid	1024	February 6, 2009	February 4, 2019
OU=Class 2 Public Primary Certification Authority, CN=class 2 publ	ou=class 2 publ	Trusted Certificate	Valid	1024	January 28, 1996	August 1, 2028
OU=Class 1 Public Primary Certification Authority, CN=class 1 publ	ou=class 1 publ	Trusted Certificate	Valid	1024	January 28, 1996	January 7, 2020
OU=Secure Server Certification Authority, CN=secure serv	ou=secure serv	Trusted Certificate	Valid	1000	November 8, 1994	January 7, 2010
OU=Class 3 Public Primary Certification Authority, CN=class 3 publ	new	Trusted Certificate	Valid	1024	January 28, 1996	August 1, 2028
CN=GTE CyberTrust Global Root, OU="GTE CyberTrust Global Root"	cn=gte cybertr	Trusted Certificate	Valid	1024	August 12, 1998	August 13, 2018

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:

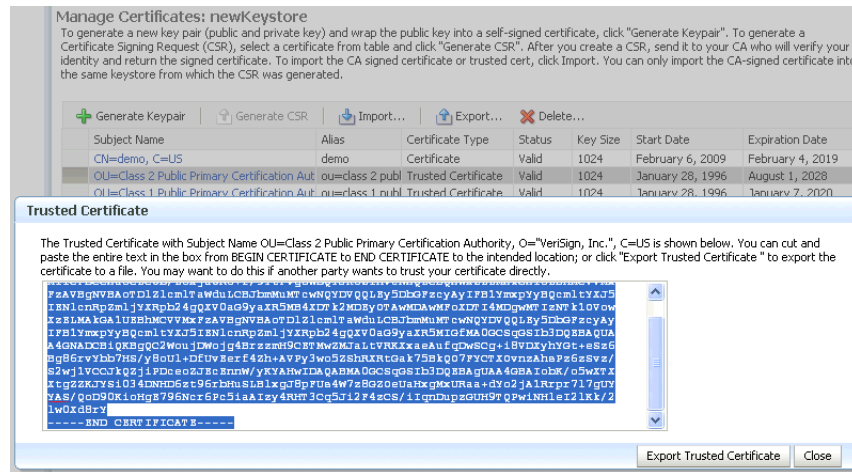
```
importKeystoreObject('inst1', 'ovd5', 'ovd', 'newKeystore', 'password',
'Certificate', '/tmp/cert.txt', 'alias')
```

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.

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:

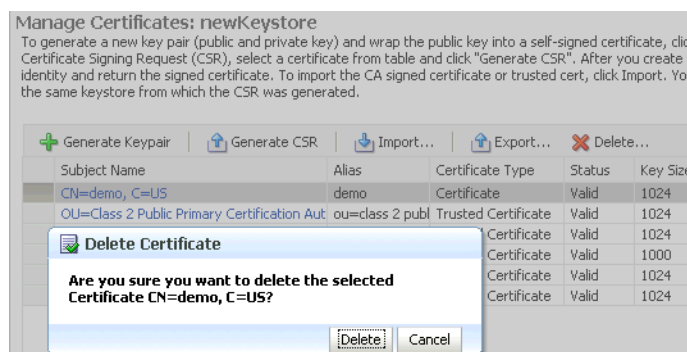
```
exportKeyStoreObject('inst1', 'ovd5', 'ovd', 'newKeystore', 'password',
'Certificate', '/tmp', 'alias')
```

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.

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:

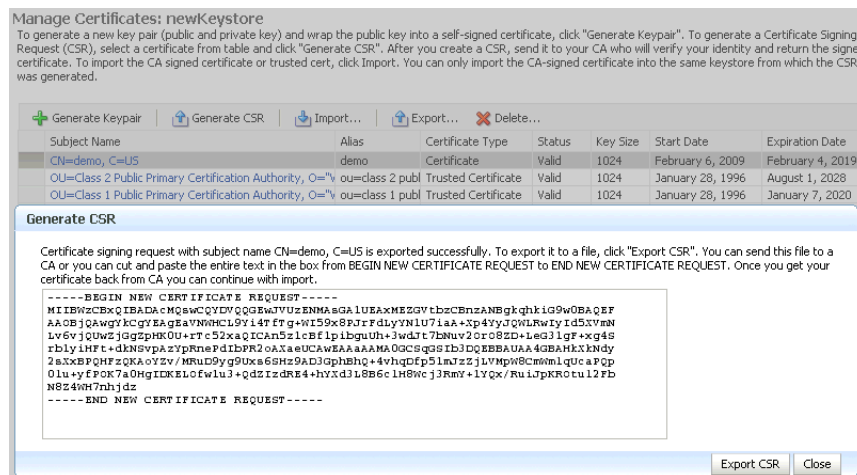
```
removeKeystoreObject('inst1', 'ovd5', 'ovd', 'newKeystore', 'password',
'Certificate', 'alias')
```

where `password` is the password for this keystore and `alias` is the alias of the certificate being deleted.

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`.



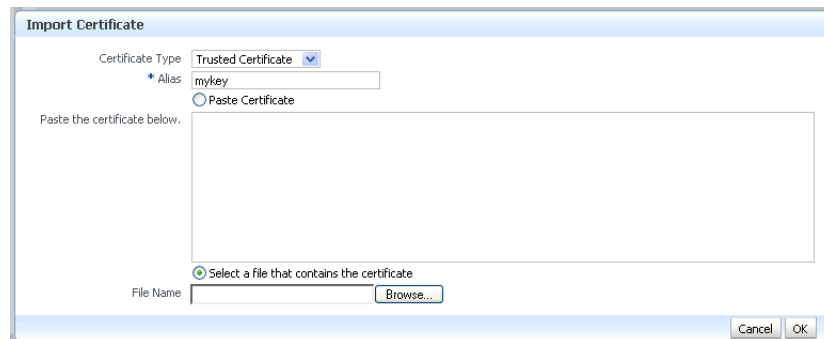
The certificate request is displayed.

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:

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.

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).
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.

For a Self-signed Keystore

The steps are:

1. Delete the existing keystore.
2. Create a new keystore with a key-pair using the new DN (based on the new host name).

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 11g Release 1 (11.1.1), 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 10g 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.

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.

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:

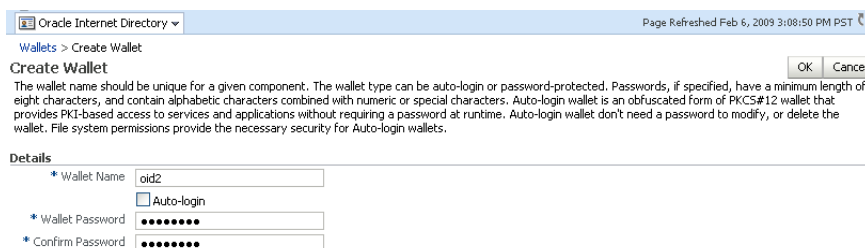
- [Creating a Wallet Using Fusion Middleware Control](#)
- [Creating a Wallet Using WLST](#)
- [Creating a Self-Signed Wallet Using Fusion Middleware Control](#)
- [Creating a Self-Signed Wallet Using WLST](#)
- [Changing a Self-Signed Wallet to a Third-Party Wallet Using Fusion Middleware Control](#)
- [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.



Oracle Internet Directory Page Refreshed Feb 6, 2009 3:08:50 PM PST

Wallets > Create Wallet

Create Wallet

The wallet name should be unique for a given component. The wallet type can be auto-login or password-protected. Passwords, if specified, have a minimum length of eight characters, and contain alphabetic characters combined with numeric or special characters. Auto-login wallet is an obfuscated form of PKCS#12 wallet that provides PKI-based access to services and applications without requiring a password at runtime. Auto-login wallet don't need a password to modify, or delete the wallet. File system permissions provide the necessary security for Auto-login wallets.

Details

* Wallet Name

Auto-login

* Wallet Password

* Confirm Password

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:

The screenshot shows a web browser window titled 'Oracle Internet Directory'. The page content is 'Create Wallet : Add Certificate Request'. Below the title is a form with the following fields:

- * Common Name: oid1
- Organizational Unit: (empty)
- Organization: (empty)
- City: (empty)
- State: (empty)
- Country: United States (dropdown menu)
- Key Size: 1024 (dropdown menu)

 There are 'OK' and 'Cancel' buttons at the top right of the form area.

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"](#).

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

Oracle Internet Directory Page Refreshed Feb 6, 2009 3:13:56 PM PST

Wallets > Create Self-Signed Wallet

Create Self-Signed Wallet OK Cancel

A self-signed wallet is not signed by a well known CA. A self-signed wallet is not recommended in a production environment. The wallet name should be unique for a given component. The wallet type can be auto-login or password-protected. Passwords, if specified, have a minimum length of eight characters, and contain alphabetic characters combined with numeric or special characters. Auto-login wallet is an obfuscated form of PKCS#12 wallet that provides PKI-based access to services and applications without requiring a password at runtime. Auto-login wallet don't need a password to modify, or delete the wallet. File system permissions provide the necessary security for Auto-login wallets.

Self-Signed Wallet Details

* Wallet Name

Auto-login

Wallet Password

Confirm Password

Add Self-Signed Certificate
Add a self-signed certificate that becomes part of the wallet.

* Common Name

Organizational Unit

Organization

City

State

Country

Key Size

4. Click **Submit**.
5. A confirmation message is displayed and the new wallet appears in the list of wallets.

Oracle Internet Directory Page Refreshed Feb 6, 2009 3:16:12 PM PST

Confirmation
Self-signed wallet selfsigned successfully created

Wallets
A Wallet is a Keystore that stores X.509 certificates and private keys in industry-standard, PKCS #12 format. To create a wallet, click Create. To create a wallet with a self-signed certificate, click Create Self-Signed Wallet. To manage the contents of a wallet, select a wallet and click Manage.

Name	Auto-login
oid2	
selfsigned	✓

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"](#)

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."](#)

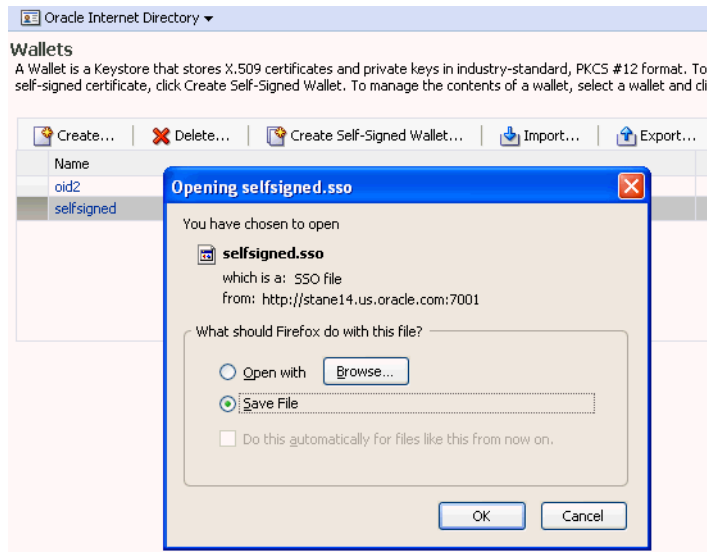
8.4.4.7 Exporting a Wallet Using Fusion Middleware Control

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 `"`.

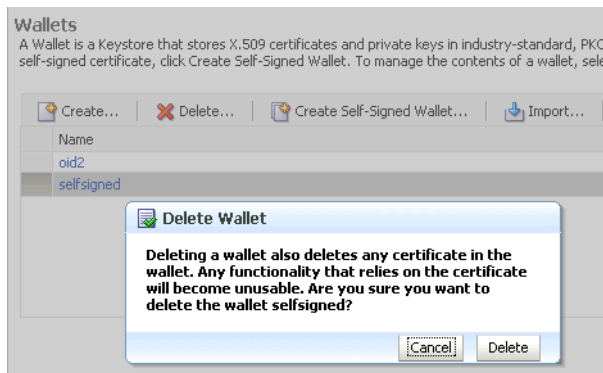
See Also: [Section 6.9.20, "importWallet"](#).

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:

- Log into Fusion Middleware Control using administrator credentials.
- Select the domain of interest.

Note: You can use Setup to discover a specific Oracle WebLogic Server domain to work with.

- Use the navigation pane to locate the instance (for example, an Oracle HTTP Server instance) that will use the wallet.

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

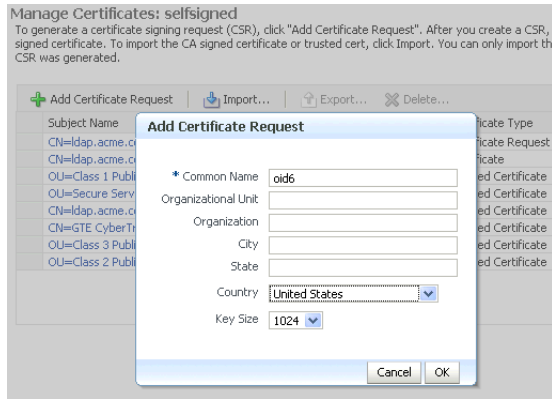
This section describes the following common certificate operations:

- [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

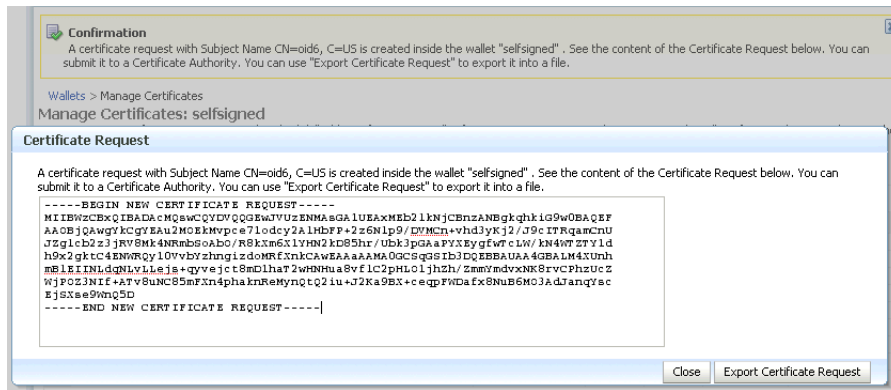
It is possible to add a certificate request at the time you create the wallet, but if it was not done at that time, you can do so using the following steps:

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.

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.

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`.

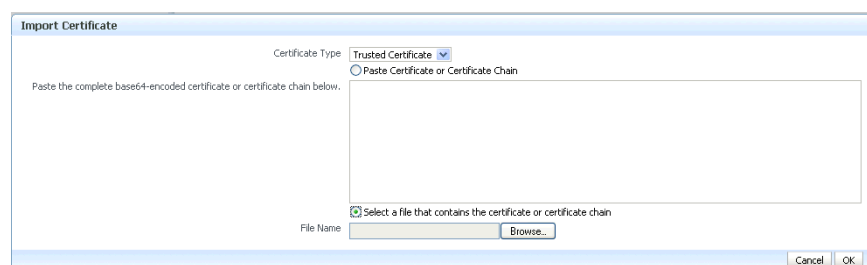
8.4.7.5 Importing a Certificate or a Trusted Certificate Using Fusion Middleware Control

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')
```

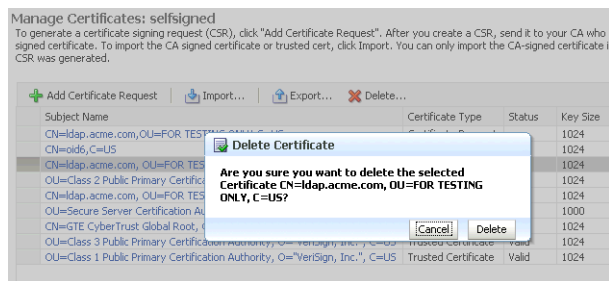
where `password` is the password for this wallet and `/tmp/cert.txt` is the file that contains BASE64 encoded certificate.

To import a trusted certificate, replace `Certificate` in the above command with `TrustedCertificate`.

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.

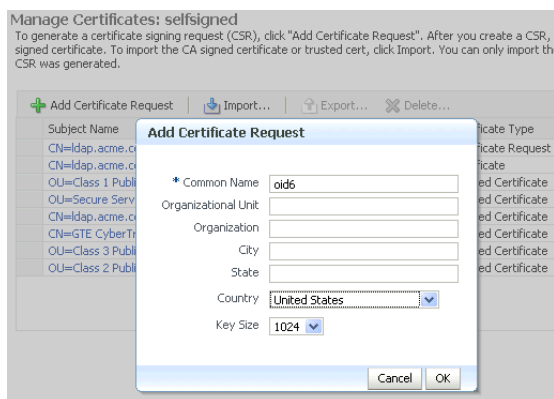
To delete a certificate request or trusted certificate, replace `Certificate` in the above command with `CertificateRequest` or `TrustedCertificate`.

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:

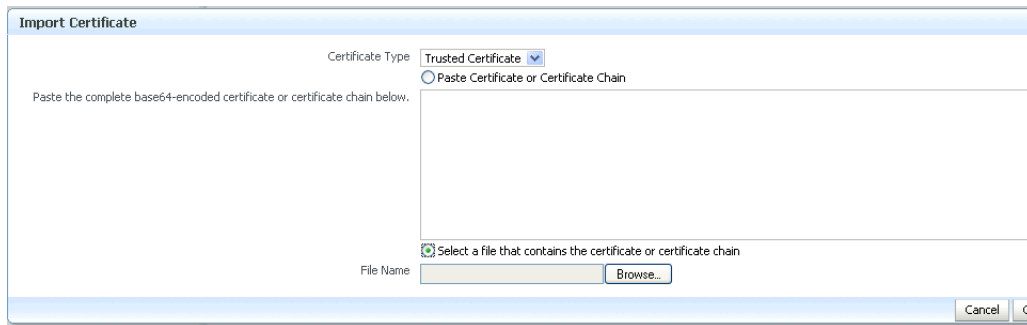


6. Enter the certificate request (CR) details and click **OK**.
The CR is generated. You can either:
 - Copy and paste the Base64-encoded certificate request to a file.
 - Export it directly to a file with the **Export Certificate Request** button.
7. Submit the certificate request file to a certificate 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
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

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, 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, 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, a sample structure could look like:

```
$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.

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).

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.

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.

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:

- [Chapter 9, "Understanding the Deployment Process"](#)
- [Chapter 10, "Deploying Applications"](#)

Understanding the Deployment Process

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 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*
- *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, see the *Oracle Fusion Middleware Administrator's Guide for Oracle WebCenter*
- 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 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: "Packaging 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: "Packaging a WebCenter Portal Application" in the *Oracle Fusion Middleware Developer's Guide for Oracle WebCenter*
 - 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 Fusion Developer's Guide for Oracle Application Development Framework*
 - For Oracle WebCenter: *Oracle Fusion Middleware Installation Guide for Oracle WebCenter*
 - For Oracle SOA Suite: *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 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 for Oracle WebLogic Server*
 - For Oracle ADF: "How to Create a JDBC Data Source for Oracle WebLogic Server" in the *Oracle Fusion Middleware Fusion Developer's Guide for Oracle Application Development Framework*
 - For Oracle WebCenter: "Configuring the JDBC Data Source" in the *Oracle Fusion Middleware Administrator's Guide for Oracle WebCenter*
 - 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: "Creating a WebLogic Managed Server Connection" in the *Oracle Fusion Middleware Developer's Guide for Oracle WebCenter*
- 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: "Configuring Security" in the *Oracle Fusion Middleware Administrator's Guide for Oracle WebCenter*
 - 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, 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: "Creating Deployment Profiles" in the *Oracle Fusion Middleware Developer's Guide for Oracle WebCenter*
 - 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. [Table 9–1](#) shows the extensions and where they are documented:

Table 9–1 Oracle JDeveloper Extensions

Component	Extension	See
Oracle WebCenter	WebCenter extensions	"Creating and Provisioning a WebLogic Managed Server Instance" in the <i>Oracle Fusion Middleware Administrator's Guide for Oracle WebCenter</i>
Oracle SOA Suite	SOA extensions	"Installing Additional Oracle Fusion Middleware Design Time Components" in the <i>Oracle Fusion Middleware Installation Guide for Oracle JDeveloper</i>

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: "Deploying the Application to a WebLogic Managed Server" in the *Oracle Fusion Middleware Administrator's Guide for Oracle WebCenter*
- 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: "Creating and Provisioning a WebLogic Managed Server Instance" in the *Oracle Fusion Middleware Administrator's Guide for Oracle WebCenter*
- 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.

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:

- [Overview of Deploying Applications](#)
- [Understanding and Managing Data Sources](#)
- [Deploying, Undeploying, and Redeploying Java EE Applications](#)
- [Deploying, Undeploying, and Redeploying Oracle ADF Applications](#)
- [Deploying, Undeploying, and Redeploying SOA Composite Applications](#)
- [Deploying, Undeploying, and Redeploying WebCenter Applications](#)
- [Managing Deployment Plans](#)
- [About the Common Deployment Tasks in Fusion Middleware Control](#)
- [Changing MDS Configuration Attributes for Deployed Applications](#)

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:

- [What Types of Applications Can You Deploy?](#)
- [Understanding Deployment, Redeployment, and Undeployment](#)

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 application. WebCenter 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 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 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

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 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, 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 multi data sources. To do so, 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.

See Also: *Oracle Fusion Middleware Configuring and Managing JDBC for Oracle WebLogic Server* for more information about configuring multi data sources

10.2.1 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 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 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](#)

10.2.1.1 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:

SOA_domain Logged in as weblogic
 WebLogic Domain Page Refreshed Aug 10, 2010 10:45:52 AM PDT

JDBC Data Sources

This table lists the JDBC system data sources that have been created in this domain. You can create, configure, test, control or delete the system data sources from this page.

Name	JNDI Name	Targets
BAMDataSource	jdbc/oracle/bam/adc	bam_server1
EDNDataSource	jdbc/EDNDataSource	soa_server1
EDNLocalTxDataSource	jdbc/EDNLocalTxDataSource	soa_server1
OraSDPMDDataSource	jdbc/OraSDPMDDataSource	bam_server1,soa_server1
SOADataSource	jdbc/SOADataSource	soa_server1
SOALocalTxDataSource	jdbc/SOALocalTxDataSource	soa_server1
mds-owsm	jdbc/mds/owsm	AdminServer,bam_server1,soa_server1
mds-soa	jdbc/mds/MDS_LocalTxDataSource	AdminServer,soa_server1,bam_server1

4. Click **Create** to open the Creating New JDBC Data Source wizard.
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 for Oracle WebLogic Server* for more information about the data source properties.

10.2.1.2 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 for Oracle WebLogic Server* for more information about the data source properties.

10.2.1.3 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*.

10.2.1.4 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 for Oracle WebLogic Server* for more information about the JDBC data source control options.

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:

Select Archive ? Cancel Step 1 of 4 Next

Specify the application or the exploded directory. Optionally you can specify a deployment plan.

Archive or Exploded Directory
 Java EE archive, Web Modules (WAR files), EJB Modules (EJB JAR files) and Resource Adapter Modules (RAR files) can be deployed. You can also deploy an exploded archive that is present on the server where Enterprise Manager is running.

Archive is on the machine where this web browser is running.
 Archive or exploded directory is on the server where Enterprise Manager is running.

Deployment Plan
 The deployment plan is a file that contains the deployment settings for an application. You can use a previously saved deployment plan for this application. Later in the deployment process, you can optionally edit the deployment plan and save it for a future deployment of this application. If you do not have a deployment plan, one will be created automatically during the deployment process when deployment configuration is done.

Create a new deployment plan when deployment configuration is done.
 Deployment plan is on the machine where this web browser is running.
 Deployment plan is on the server where Enterprise Manager is running.

Information
 Use this page to deploy Java EE applications that require Oracle Metadata Services (MDS) or that take advantage of the Oracle Application Development Framework (Oracle ADF).
 If your application is a SOA composite, use the SOA Composite deployment wizard.
 If your application is not a SOA composite or it does not require an MDS repository or ADF connections, then you can deploy your application using this wizard or the Oracle WebLogic Server Administration Console.

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.

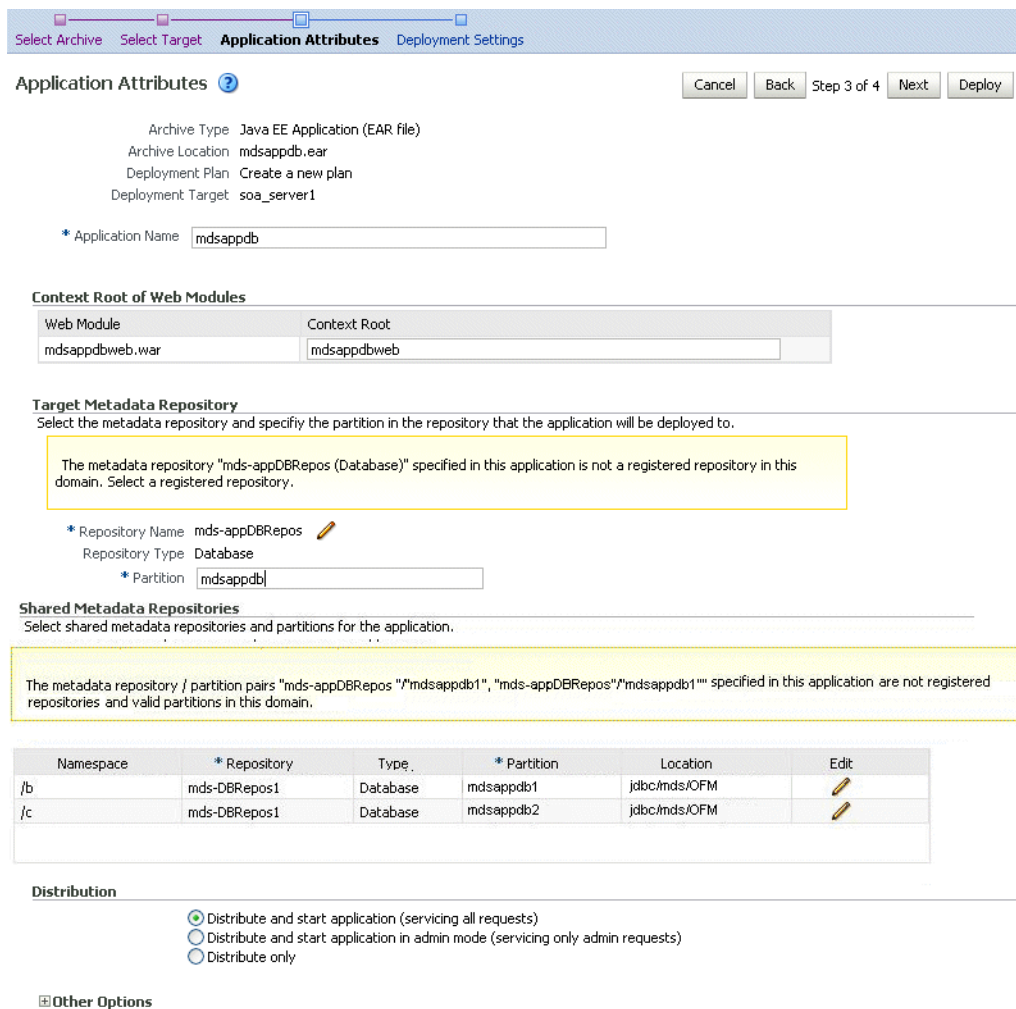
10.4.1.1 Deploying ADF Applications Using Fusion Middleware Control

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: *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: *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: *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 Applications

WebCenter 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 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 application. The following topics describe using Fusion Middleware Control to accomplish these tasks:

- [Deploying WebCenter Applications](#)
- [Undeploying WebCenter Applications](#)
- [Redeploying WebCenter Applications](#)

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

10.6.1 Deploying WebCenter 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 MAR is a compressed archive of selected metadata. 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*.

To deploy an Oracle WebCenter 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, 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.

For more information about setting these attributes, see the *Oracle Fusion Middleware Administrator's Guide for Oracle WebCenter*.

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 Applications

To undeploy a WebCenter 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 Applications

To redeploy a WebCenter Application:

1. From the navigation pane, expand the farm, then **WebLogic Domain**, and then the domain.
The server home page is displayed.
2. Select the server in which you want to deploy 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. If the partition or repository specified in the application is not valid in the domain, Fusion Middleware Control displays a message.

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-ebj-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/docs/cd/E17477_01/javasee/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 will 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.

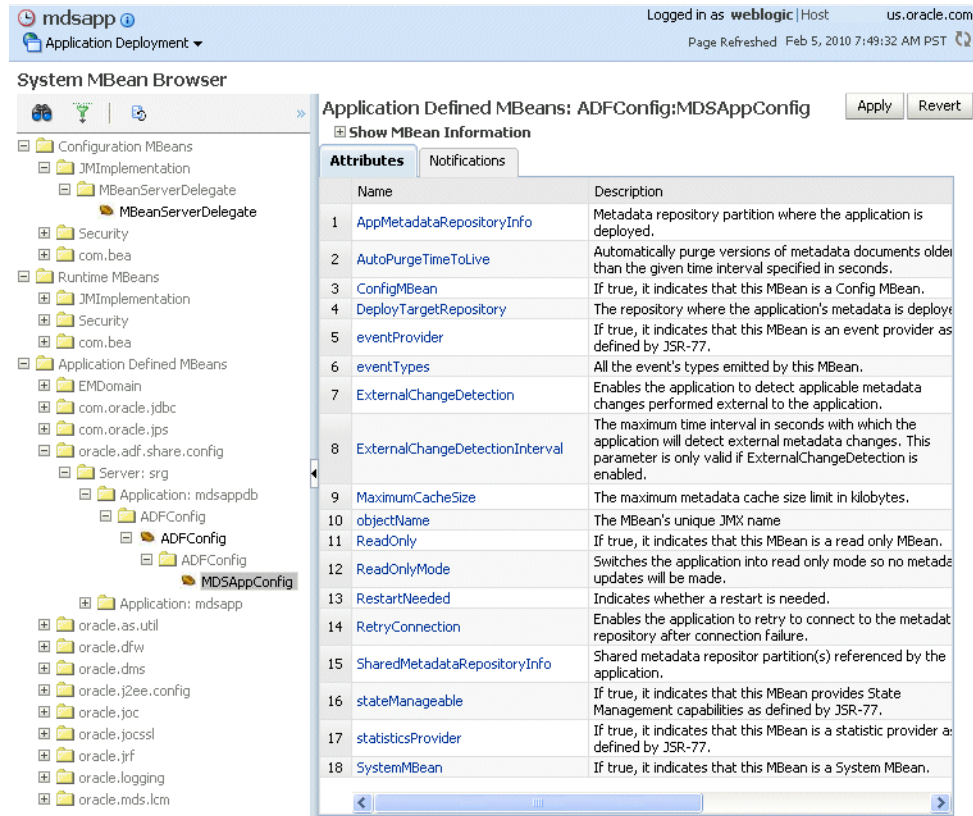
2. From the Application Deployment menu, choose **System MBean Browser**.

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	<p>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.</p> <p>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.</p> <p>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.</p> <p>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.</p> <p>This attribute is applicable only to database-based repositories. The default is true.</p>
ExternalChangeDetectionInterval	<p>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.</p> <p>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.</p> <p>The default is 30 seconds.</p>
MaximumCacheSize	<p>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.</p>
ReadOnlyMode	<p>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.</p>
RetryConnection	<p>Enables the application to retry the connection to the metadata repository after connection failure.</p>
SharedMetadataRepositoryInfo	<p>Read only. Specifies the MDS Repository partition used by the application. Note that an application can use more than one shared metadata repository.</p>

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` will be 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:

- [Chapter 11, "Monitoring Oracle Fusion Middleware"](#)
- [Chapter 12, "Managing Log Files and Diagnostic Data"](#)
- [Chapter 13, "Diagnosing Problems"](#)

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 with the command line, use the WLST `state` command, using 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 with the command line, use the `opmnctl status` command, using 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 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:

Farm_soa_domain Logged in as weblogic
Page Refreshed Sep 2, 2010 7:44:10 AM PDT

Deployments

Up (34)

Name	Status	Target
Application Deployments		
Internal Applications		
Resource Adapters		
BPMComposer	↑	soa_server1
BPMComposerServices	↑	soa_server1
composer	↑	soa_server1
DefaultToDoTaskFlow	↑	soa_server1
mdsappdb	↑	soa_server1
oracle-bam(11.1.1)	↑	bam_server1
OracleBPMComposerRC	↑	soa_server1
OracleBPMProcessRole:	↑	soa_server1
OracleBPMWorkspace	↑	soa_server1
SimpleApprovalTaskFlo	↑	soa_server1
worklistapp	↑	soa_server1
SOA		
soa-infra	↑	soa_server1
default	↑	
SimpleApproval [↑	

Fusion Middleware

Up (9)

Name	Status	Host	CPU Usage (%)
WebLogic Domain			
soa_domain			
AdminServer	↑	example.com	16.08
bam_server1	↑	example.com	18.37
soa_server1	↑	example.com	22.67
BAM			
OracleBamServer (b	↑	example.com	
OracleBamWeb (bam	↑	example.com	
Metadata Repositories			
mds-mds_repos_file		example.com	
mds-owsm		example.com	
mds-soa		example.com	
User Messaging Service			
usermessagingdriver	↑	example.com	
usermessagingdriver	↑	example.com	
usermessagingserve	↑	example.com	

Farm Resource Center

Before You Begin

- Introduction to Oracle Fusion Middleware
- Understanding Key Oracle Fusion Middleware Farm Concepts
- Overview of Oracle Fusion Middleware Administration Tools

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:

The screenshot displays the Oracle WebLogic Server Administration Console for a domain named 'soa_domain'. The interface is organized into several panels:

- Summary:** Provides general information about the domain, including the Administration Server (AdminServer) and its host (example.com). It includes a link to the Oracle WebLogic Server Administration Console.
- Servers:** Shows the status of the servers in the domain. A green pie chart indicates that 100% of the servers are up. A table lists the servers: AdminServer, bam_server1, and soa_server1, along with their hostnames, listen ports, active sessions, and request processing times.
- Clusters:** Shows 'No Clusters found'.
- Deployments:** Shows the status of various application deployments. A green pie chart indicates that 100% of the deployments are up. A table lists the deployments, including Application Deployments, Internal Applications, Resource Adapters, and SOA.
- Oracle WebLogic Domain Resource Center:** Provides links for 'Before You Begin' and 'Typical Administration Tasks'.

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.

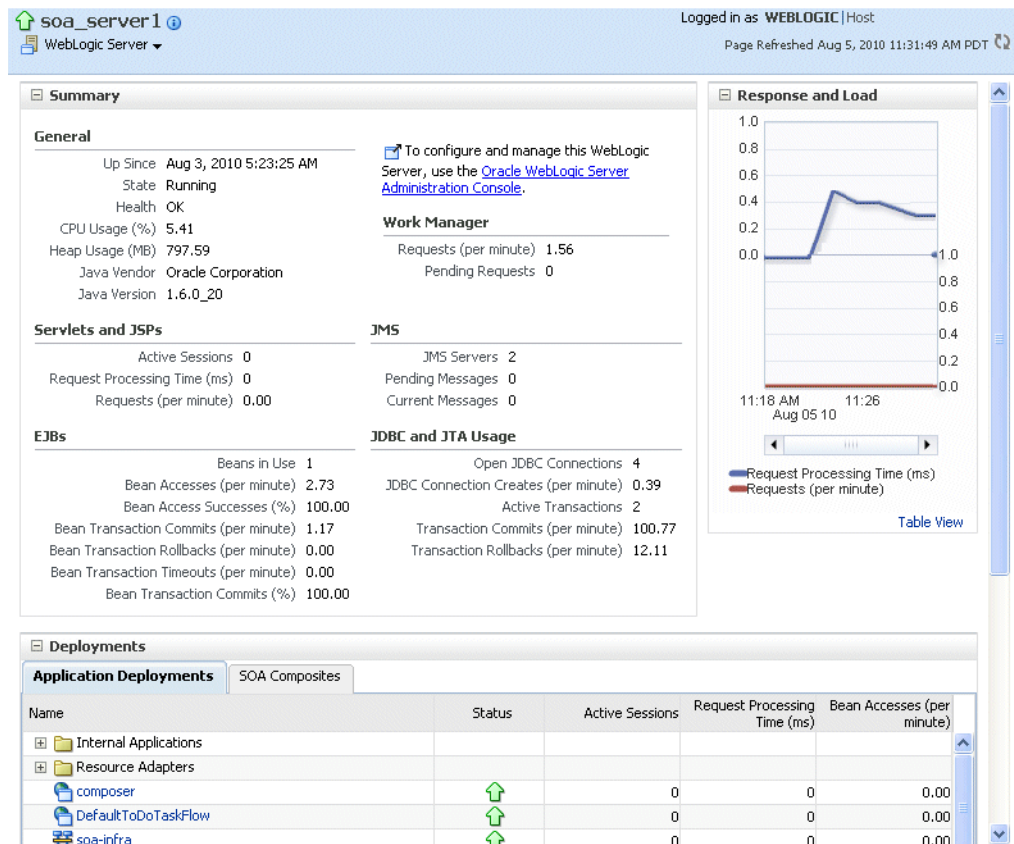
11.1.3 Monitoring an Oracle WebLogic Server Administration or Managed Server

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.

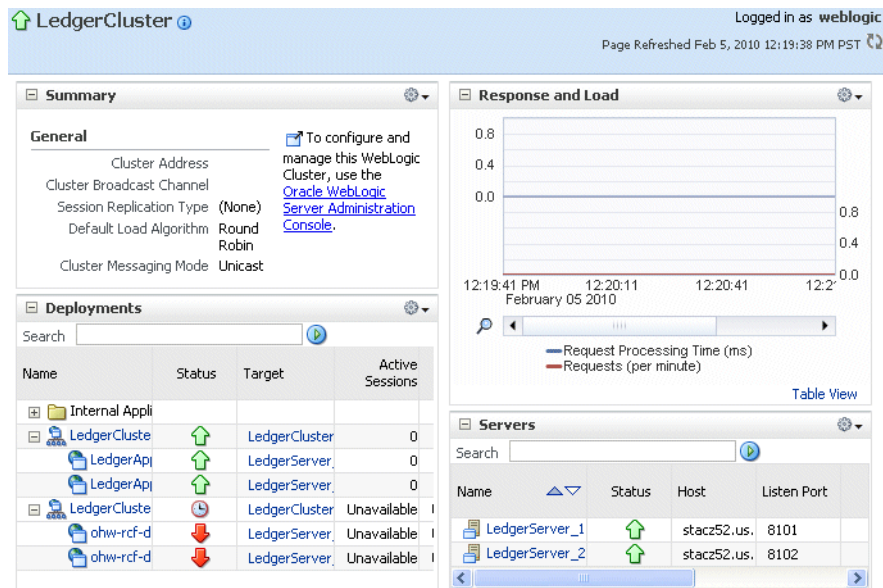
11.1.4 Monitoring a Cluster

You can view the status of a cluster, including the servers and deployments in the cluster using Fusion Middleware Control.

To monitor a cluster:

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.

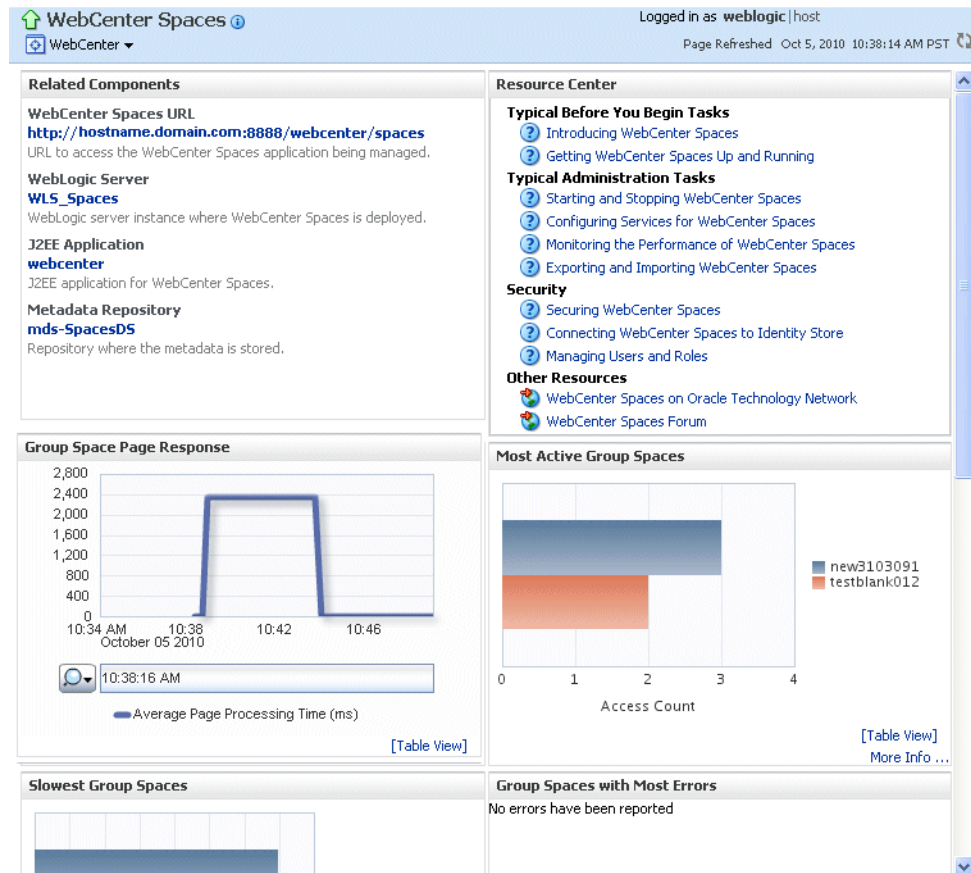
11.1.5 Monitoring a Java Component

You can view the status of a component, including whether the component is started, in the component home page in Fusion Middleware Control.

To monitor a Java component, such as WebCenter Spaces:

1. From the navigation pane, expand the farm, then the type of component, such as WebCenter, then the component, such as WebCenter Spaces.
2. Select the component. For example, select **WebCenter Spaces**.

The component home page is displayed, as shown in the following figure:



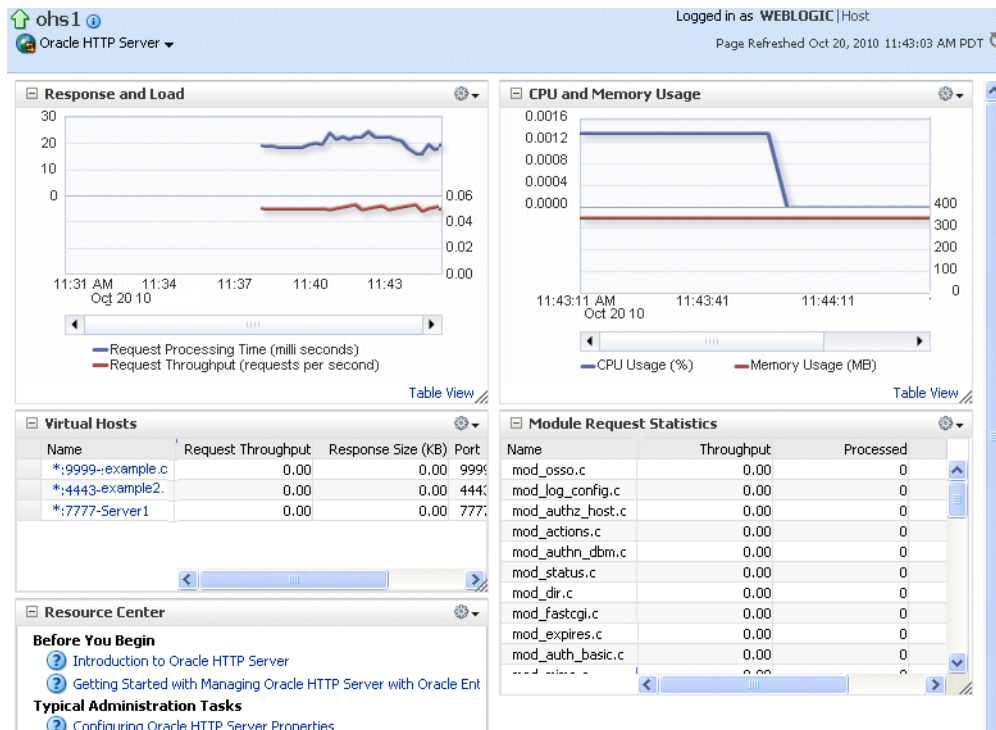
See Also: "Overview of the Administration Console" in the *Oracle Fusion Middleware Introduction to Oracle WebLogic Server* for information about using the Oracle WebLogic Server Administration Console to monitor Java components

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:



11.1.7 Monitoring Java EE Applications

To monitor a Java EE application using Fusion Middleware Control:

1. From the navigation pane, 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:

The screenshot shows the Oracle WebLogic Server Administration Console interface for the BPMComposer application. The page is titled "BPMComposer" and "Application Deployment". It is logged in as "weblogic" on a host. The page was refreshed on Sep 2, 2010 at 8:10:56 AM PDT.

The main content area is divided into several sections:

- Summary:**
 - General:** State: Active, Application Type: ear, Deployed On: soa_server1. A link is provided to configure and manage the application using the Oracle WebLogic Server Administration Console.
 - Servlets and JSPs:** Active Sessions: 1, Request Processing Time (ms): 48, Requests (per minute): 303.41.
 - Work Manager:** Requests (per minute): 26.30, Pending Requests: 1.
 - EJBs:** Beans in Use: 0, Bean Accesses (per minute): 0.00, Bean Access Successes (%): 0.00, Bean Transaction Commits (per minute): 0.00, Bean Transaction Rollbacks (per minute): 0.00, Bean Transaction Timeouts (per minute): 0.00, Bean Transaction Commits (%): 0.00.
- Entry Points:**
 - Web Modules:** A table with columns Name and Test Point. The entry is "/bpm/composer" with test point "http://dadvmn0623.us.oracle.com:8001/bpm/composer".
 - Web Services:** No Web Services Found.
- Most Requested:**
 - Servlets and JSPs:** A table with columns Name, Web Module, Requests Processed, Average Client Processing Time (ms), Requests (per minute), and Total Client Processing Time (ms).

Name	Web Module	Requests Processed	Average Client Processing Time (ms)	Requests (per minute)	Total Client Processing Time (ms)
jsp	/bpm/composer	1,502	24	277.64	36,359
Login	/bpm/composer	2	376	0.37	751
adfibResources	/bpm/composer	0	0	0.00	0
- Response and Load:** A line graph showing Request Processing Time (ms) and Requests (per minute) over time. The x-axis shows time from 07:57 AM to 08:07 AM on Sep 02 10. The y-axis ranges from 0 to 2,000. The graph shows a sharp spike in request processing time around 08:05 AM.

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 **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.

11.1.9 Monitoring SOA Composite Applications

To monitor a SOA composite application:

1. From the navigation pane, 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:

The screenshot shows the SOA composite home page for 'OrderBookingComposite [1.0]'. The page is logged in as 'weblogic' and shows a refresh time of 'Page Refreshed Oct 5, 2009 1:27:50 PM PDT'. The dashboard has several tabs: 'Dashboard', 'Instances', 'Faults and Rejected Messages', 'Unit Tests', and 'Policies'. The 'Instances' tab is selected, showing a table of recent instances. Below this, there is a section for 'Recent Faults and Rejected Messages' which currently shows 'No faults found'. At the bottom, there is a 'Component Metrics' section with a table showing metrics for various components.

Instance ID	Name	Conversation ID	State	Start Time
20006		med:DB8195201034	---	Oct 3, 2009 6:10:07 PM
20005			---	Oct 3, 2009 5:52:50 PM
20004		med:144BCA101021	---	Oct 3, 2009 3:48:34 PM
20003			---	Oct 3, 2009 3:47:40 PM
20002			---	Oct 3, 2009 3:47:37 PM

Name	Component Type	Total Instances	Running Instances	Faulted Instances	
				Recoverable	Non Recoverable
FulfillOrder	Mediator	3	0	0	0
PartnerSupplierM...	Mediator	3	0	0	0

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 Applications

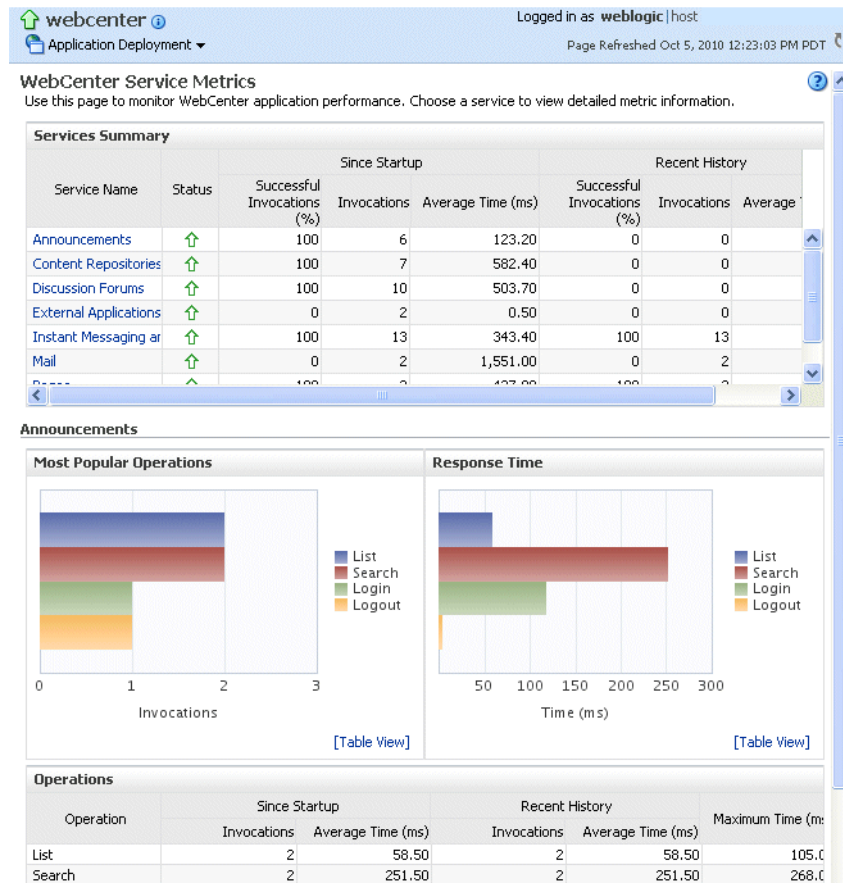
To monitor an Oracle WebCenter application:

1. From the navigation pane, 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:



See Also: "Understanding WebCenter Performance Metrics" in *Oracle Fusion Middleware Administrator's Guide for Oracle WebCenter* 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 **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.

- Expand the cluster application to show each instance of the application, as shown in the following figure:



- 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:

- From the navigation pane, expand the farm, then **WebLogic Domain**, and then the domain.
- Select the server to monitor.

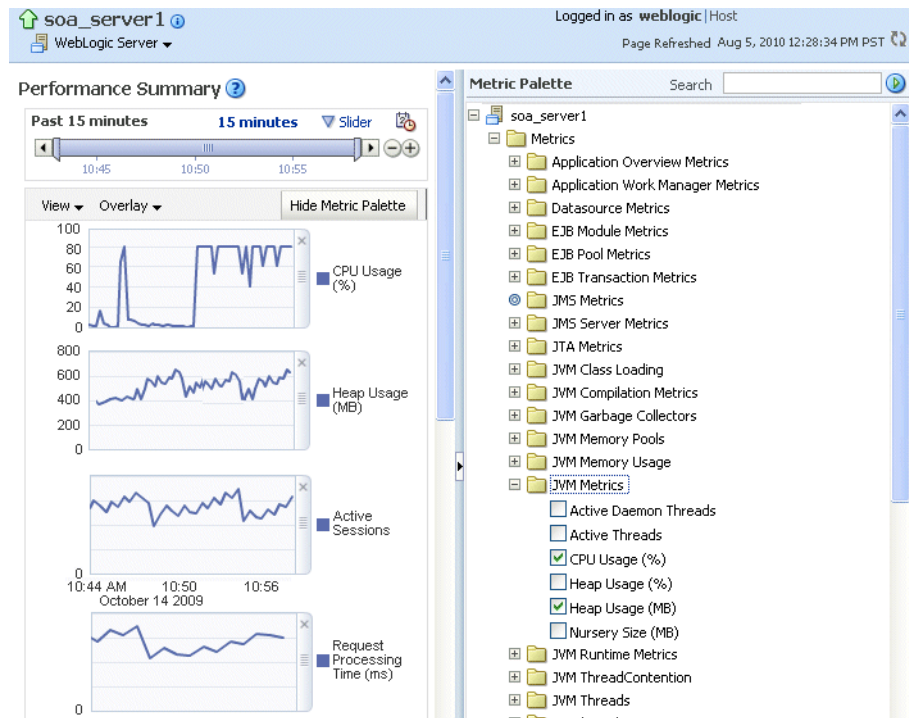
The Managed Server home page is displayed.

- 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.

- 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**.

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.

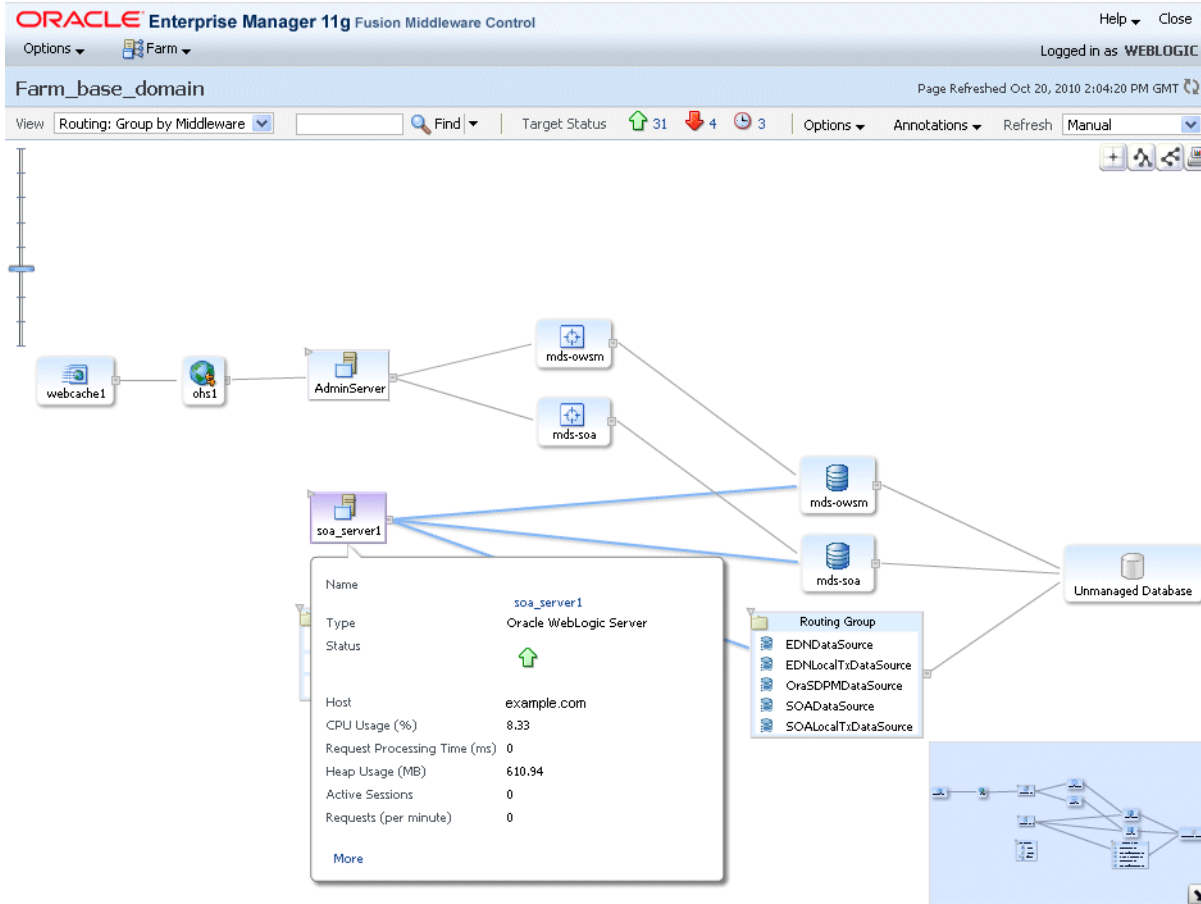
To view the topology:

1. Click **Topology**.

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.
- 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. When you click the target name, the target is highlighted. 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.

- View the targets by status. Choose **Up**, **Down**, or **Unknown** from the Target Status at the top of the page.

- Reposition the topology and change its orientation:
 - To center the topology, click the Center button
 - To change the orientation, click the top-to-bottom button or the left-to-right button.
 - 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, 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.
- 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.)

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 workaroud 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.
-

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](#)

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.
- 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. The ID is <code>oracle</code> for all Oracle components.	No
INSTANCE_ID (INST_ID)	The name of the Oracle instance to which the component that originated the message belongs.	No
COMPONENT ID (COMP)	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 error 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

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	<i>DOMAIN_HOME</i> /sysman/log/emoms.log <i>DOMAIN_HOME</i> /sysman/log/emoms.trc
Oracle Application Development Framework	<i>DOMAIN_HOME</i> /servers/ <i>server_name</i> /logs/ <i>server-name</i> -diagnostic.log
Oracle Business Activity Monitoring	<i>DOMAIN_HOME</i> /servers/ <i>server_name</i> /logs/bam-diagnostic.log
Oracle Business Intelligence Discoverer	<i>DOMAIN_HOME</i> /servers/ <i>server_name</i> /logs/discoverer/ <i>server_name</i> -diagnostic.log <i>DOMAIN_HOME</i> /servers/ <i>server_name</i> /logs/discoverer/ <i>server_name</i> -diagnostic.log
Oracle Business Process Management	<i>DOMAIN_HOME</i> /servers/ <i>server_name</i> /logs/ <i>server-name</i> -diagnostic.log
Oracle Directory Integration Platform	<i>DOMAIN_HOME</i> /servers/ <i>server_name</i> /logs/ <i>server-name</i> -diagnostic.log
Oracle Forms Services	<i>DOMAIN_HOME</i> /servers/ <i>server_name</i> /logs/ <i>server-name</i> -diagnostic.log <i>ORACLE_HOME</i> /j2ee/DevSuite/application-deployments/forms/application.log
Oracle Fusion Middleware Audit Framework	<i>DOMAIN_HOME</i> /servers/ <i>server_name</i> /logs/ <i>server-name</i> -diagnostic.log
Oracle HTTP Server	<i>ORACLE_INSTANCE</i> /diagnostics/logs/OHS/ <i>component_name</i> /*.log
Oracle Identity Federation	<i>DOMAIN_HOME</i> /servers/ <i>server_name</i> /logs/ <i>server-name</i> -diagnostic.log
Oracle Imaging and Process Management	<i>DOMAIN_HOME</i> /servers/ <i>server_name</i> /logs/ <i>server-name</i> -diagnostic.log
Oracle Information Rights Management	<i>DOMAIN_HOME</i> /servers/ <i>server_name</i> /logs/ <i>server-name</i> -diagnostic.log
Oracle Internet Directory	<i>ORACLE_INSTANCE</i> /diagnostics/logs/OID/oid*.log <i>ORACLE_INSTANCE</i> /diagnostics/logs/OID/tools/*.log
Oracle Platform Security Services	<i>DOMAIN_HOME</i> /servers/ <i>server_name</i> /logs/ <i>server-name</i> -diagnostic.log
Oracle Portal	<i>DOMAIN_HOME</i> /servers/ <i>server_name</i> /logs/ <i>server-name</i> -diagnostic.log
Oracle Reports	<i>ORACLE_INSTANCE</i> /diagnostics/logs/ReportsServerComponent <i>ORACLE_INSTANCE</i> /diagnostics/logs/ReportsBridgeComponent <i>ORACLE_INSTANCE</i> /diagnostics/logs/ReportsToolsComponent
Oracle SOA Suite	<i>DOMAIN_HOME</i> /servers/ <i>server_name</i> /logs/ <i>server-name</i> -diagnostic.log
Oracle TopLink	<i>DOMAIN_HOME</i> /servers/ <i>server_name</i> /logs/ <i>server-name</i> -diagnostic.log
Oracle Virtual Directory	<i>ORACLE_INSTANCE</i> /config/OVD/ <i>component_name</i> /diagnostic.log <i>ORACLE_INSTANCE</i> /diagnostics/logs/OVD/ <i>component_name</i> /diagnostic.log
Oracle Web Cache	<i>ORACLE_INSTANCE</i> /diagnostics/logs/WebCache/ <i>component_name</i> *.log

Table 12–2 (Cont.) Log File Location

Component	Log File Location
Oracle Web Services Manager	<i>DOMAIN_HOME</i> /servers/ <i>server_name</i> /logs/owsm/msglogging <i>DOMAIN_HOME</i> /servers/ <i>server_name</i> /logs/owsm-diagnostic.log
Oracle WebCenter	<i>DOMAIN_HOME</i> /servers/ <i>server_name</i> /logs/component-diagnostic.log
Oracle WebLogic Server	<i>DOMAIN_HOME</i> /servers/ <i>server_name</i> /logs/ <i>server_name</i> -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: <ol style="list-style-type: none"> 1. <i>ORACLE_HOME</i>/rcu/log/<i>timestamp</i> 2. /tmp/logdir.<i>timestamp</i>

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 commands:

```
./wlst.sh
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.

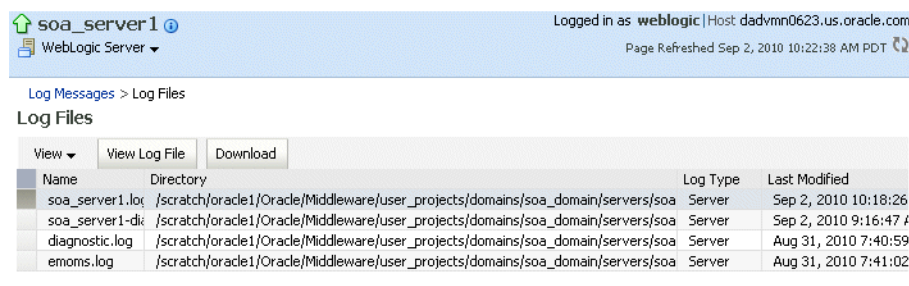
For example, to view the log files and their messages for a Managed Server:

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 the **Target Log Files** icon.

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:



Name	Directory	Log Type	Last Modified
soa_server1.log	/scratch/oracle1/Oracle/Middleware/user_projects/domains/soa_domain/servers/soa	Server	Sep 2, 2010 10:18:26
soa_server1-di	/scratch/oracle1/Oracle/Middleware/user_projects/domains/soa_domain/servers/soa	Server	Sep 2, 2010 9:16:47
diagnostic.log	/scratch/oracle1/Oracle/Middleware/user_projects/domains/soa_domain/servers/soa	Server	Aug 31, 2010 7:40:59
emoms.log	/scratch/oracle1/Oracle/Middleware/user_projects/domains/soa_domain/servers/soa	Server	Aug 31, 2010 7:41:02

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:

soa_server1 | Logged in as: weblogic | Host

WebLogic Server | Page Refreshed Sep 2, 2010 10:25:01 AM PDT

Log Messages > Log Files > View Log File: soa_server1.log

View Log File: soa_server1.log | View: Manual Refresh

Name: /scratch/oracle1/Oracle/Middleware/user_projects/domains/soa_domain/servers/soa_server1/logs/soa_server1.log | Download | Log Type: Server | Size (KB): 1,439.05

Last Modified: Sep 2, 2010 10:18:26 AM PDT

Date Range: Time Interval | Start Date: 9/1/10 1:11 PM | End Date: 9/2/10 10:18 AM | Search

Time	Message Type	Message ID	Message
Sep 1, 2010 1:11:28 PM PDT	Notification	BEA-000628	Created "1" resources for pool "mds-soa", out of which "1" are available and "0" are unavailable.
Sep 1, 2010 1:11:49 PM PDT	Notification	BEA-000628	Created "1" resources for pool "OraSDPMDDataSource", out of which "1" are available and "0" are unavailable.
Sep 1, 2010 1:15:01 PM PDT	Notification	BEA-000628	Created "1" resources for pool "SOALocalTxDataSource", out of which "1" are available and "0" are unavailable.
Sep 1, 2010 1:23:20 PM PDT	Notification	BEA-000628	Created "1" resources for pool "SOADDataSource", out of which "1" are available and "0" are unavailable.
Sep 1, 2010 1:26:06 PM PDT	Notification	BEA-001128	Connection for pool "OraSDPMDDataSource" closed.
Sep 1, 2010 1:26:06 PM PDT	Notification	BEA-001128	Connection for pool "SOALocalTxDataSource" closed.
Sep 1, 2010 1:26:06 PM PDT	Notification	BEA-001128	Connection for pool "SOADDataSource" closed.
Sep 1, 2010 1:26:06 PM PDT	Notification	BEA-001128	Connection for pool "SOADDataSource" closed.
Sep 1, 2010 1:26:06 PM PDT	Notification	BEA-001128	Connection for pool "SOADDataSource" closed.
Sep 1, 2010 1:26:06 PM PDT	Notification	BEA-001128	Connection for pool "mds-soa" closed.
Sep 1, 2010 1:26:09 PM PDT	Notification	BEA-000628	Created "1" resources for pool "SOADDataSource", out of which "1" are available and "0" are unavailable.
Sep 1, 2010 1:26:19 PM PDT	Notification	BEA-000628	Created "1" resources for pool "mds-owsm", out of which "1" are available and "0" are unavailable.
Sep 1, 2010 1:26:49 PM PDT	Notification	BEA-000628	Created "1" resources for pool "OraSDPMDDataSource", out of which "1" are available and "0" are unavailable.

Rows Selected: 1 | Columns Hidden: 18 | Total Rows: 1000

Sep 1, 2010 1:11:28 PM PDT (Notification)

Message ID	BEA-000628	Module	Common
Message Level	1	Host	dadvmn0623
ECID	d62829346c7e9879;7eebcf0a;12ac89813f7;-8000-00000000000000002	Host IP Address	10.229.149.27
Relationship ID	0	User	<anonymous>
Component	soa_server1	Thread ID	MDSPollingThread-[soa-infra, jdbc/mds /MDS_LocalTxDataSource]

Message: Created "1" resources for pool "mds-soa", out of which "1" are available and "0" are unavailable.

By default, the messages are sorted by time, in ascending order. You can sort the messages by any of the columns, such as Message Type, by clicking the column name.

- 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 "SOADDataSource", 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 "SOADDataSource", 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
"SOADDataSource" 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:

The screenshot displays the 'Log Messages' page for the 'soa_domain' (WebLogic Domain). It includes a search section with the following details:

- Selected Targets (39)**
- Date Range:** Most Recent, 1 Hours
- Message Types:** Incident Error, Error, Warning, Notification, Trace, Unknown (all checked)
- Search Criteria:** contains
- Buttons:** Search, Add Fields
- View Options:** Messages, View Related Messages, Export Messages to File

Time	Message Type	Message ID	Message	Target
Sep 7, 2010 8:24:15 AM PDT	Error		targetName Farm_soa_domain	em (A...
Sep 7, 2010 8:24:23 AM PDT	Error		targetName /Farm_soa_domain/soa_domain	em (A...
Sep 7, 2010 8:24:23 AM PDT	Error		targetName Farm_soa_domain	em (A...

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.

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:

```
2010-09-30T12:00:00-08:00
```

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')
```

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:

- From the navigation pane, select the target, such as the domain.
- From the dynamic target menu, choose **Logs**, then **View Log Messages**.
The Log Messages page is displayed.
- Search for particular types of messages as described in [Section 12.3.2.1](#).
- 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.

- 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:

- 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'], export='download_log.txt')
```

The messages are written to the file `download_log.txt`.

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: You cannot configure options for log files of system components, which are listed in [Section 3.5.2](#), 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:

```
./wlst.sh
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.

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**.

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 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.
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. 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.
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.

Table 12–3 (Cont.) Diagnostic Message Types and Level

Message Type	Level	Description
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
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.

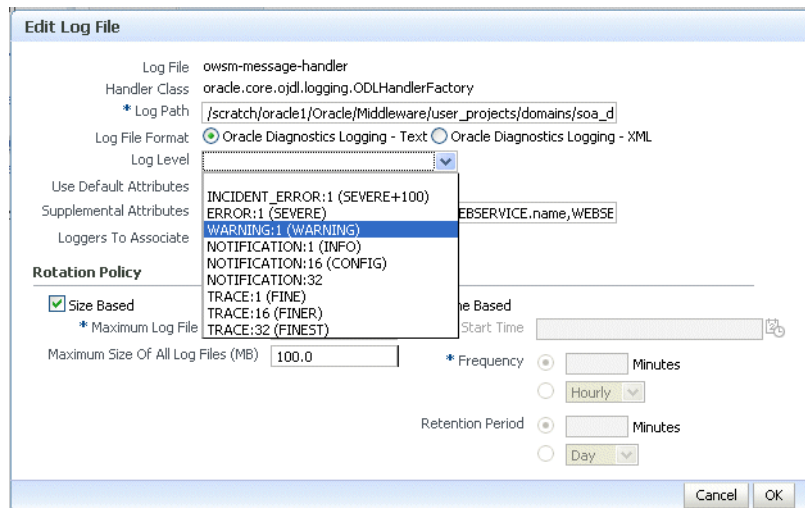
To set the message level for a component log file:

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.

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.

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.
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='0000H19TwKUCs1T6uBi8UH181kWX000002')
```


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:

soa_domain | Logged in as weblogic
WebLogic Domain | Page Refreshed Sep 7, 2010 10:41:24 AM PDT

Log Messages > Related Messages by ECID: d62829346c7e9879;-2cd59f15:12ac8870eb2;-8000-0000000000000148

Related Messages by ECID: d62829346c7e9879;-2cd59f15:12ac8870e... | Broaden Target Scope

Selected Targets (39)

Time	Message Type	Message ID	Message	Target
Aug 31, 2010 7:40:41 AM PDT	Warning	BEA-090171	Loading the identity certificate and private key stored under the alia	Ad
Aug 31, 2010 7:40:42 AM PDT	Warning	BEA-090169	Loading trusted certificates from the jks keystore file /scratch/oracle	Ad
Aug 31, 2010 7:40:42 AM PDT	Warning	BEA-090169	Loading trusted certificates from the jks keystore file /scratch/oracle	Ad
Aug 31, 2010 7:40:42 AM PDT	Incident Error	BEA-090152	Demo trusted CA certificate is being used in production mode: [[Veri	Ad
Aug 31, 2010 7:40:42 AM PDT	Warning	BEA-090898	Ignoring the trusted CA certificate "CN=thawte Primary Root CA - G	Ad
Aug 31, 2010 7:40:42 AM PDT	Warning	BEA-090898	Ignoring the trusted CA certificate "CN=T-TeleSec GlobalRoot Class	Ad
Aug 31, 2010 7:40:42 AM PDT	Warning	BEA-090898	Ignoring the trusted CA certificate "CN=GlobalSign,O=GlobalSign,OI	Ad
Aug 31, 2010 7:40:42 AM PDT	Warning	BEA-090898	Ignoring the trusted CA certificate "CN=T-TeleSec GlobalRoot Class	Ad
Aug 31, 2010 7:40:42 AM PDT	Warning	BEA-090898	Ignoring the trusted CA certificate "OU=Security Communication Ro	Ad
Aug 31, 2010 7:40:42 AM PDT	Warning	BEA-090898	Ignoring the trusted CA certificate "CN=VeriSign Universal Root Cerl	Ad
Aug 31, 2010 7:40:42 AM PDT	Warning	BEA-090898	Ignoring the trusted CA certificate "CN=KEYNECTIS ROOT CA,OU=f	Ad
Aug 31, 2010 7:40:42 AM PDT	Warning	BEA-090898	Ignoring the trusted CA certificate "CN=GeoTrust Primary Certificati	Ad
Aug 31, 2010 7:40:42 AM PDT	Notificatic	BEA-000307	Exportable key maximum lifespan set to 500 uses.	Ad

5. Trace the ECID to the earliest message. (You may need to increase the scope to view the first message with the ECID.)

Diagnosing Problems

This chapter describes how to use the 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:

- [Understanding the Diagnostic Framework](#)
- [How the Diagnostic Framework Works](#)
- [Configuring the Diagnostic Framework](#)
- [Investigating, Reporting, and Solving a Problem](#)

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. 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](#)
- [ADCRI 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:

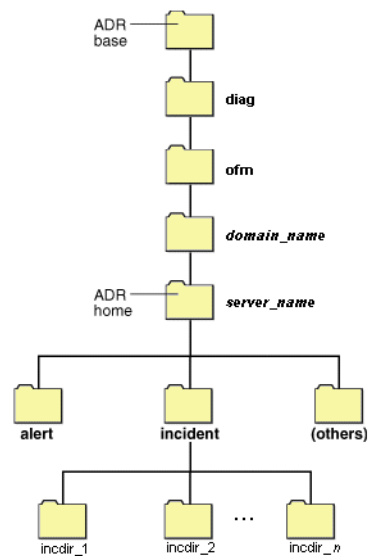
```
DOMAIN_HOME/servers/server_name/adr
```

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.

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 13.4.3.1, "Listing Diagnostic Dumps"](#)
- [Section 13.4.3.2, "Viewing a Description of a Diagnostic Dump"](#)
- [Section 13.4.3.3, "Executing Dumps"](#)
- [Section 13.4.4.1, "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 ADCRI Command-Line Utility

The ADR Command Interpreter (ADCRI) 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. ADCRI 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.

ADCRI 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 ADCRI command-line utility:

- [Packaging an Incident](#)
- [Purging Incidents](#)

See Also:

- The chapter "ADRCI: ADR Command Interpreter" in the *Oracle Database Utilities*
- The chapter "Managing Diagnostic Data" in the *Oracle Database Administrator's Guide*

Both manuals are located at:

<http://www.oracle.com/technology/documentation/database.html>

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.

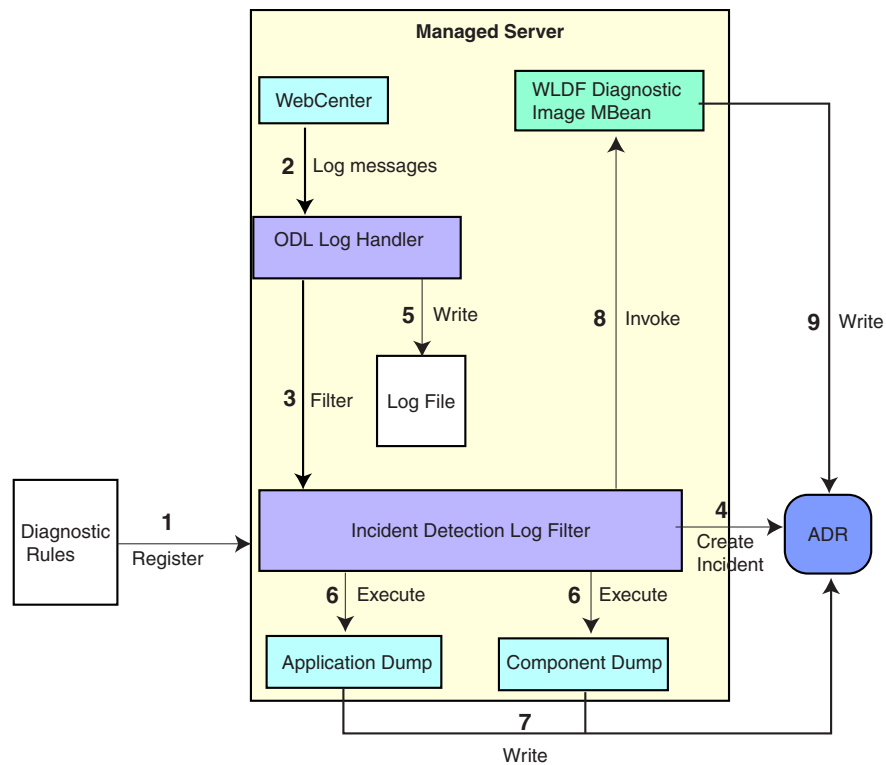
Incidents are automatically detected in two ways:

- 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 13.3.2](#).

- 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) 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.

When an incident is created, a message, similar to the following, is written to the log file:

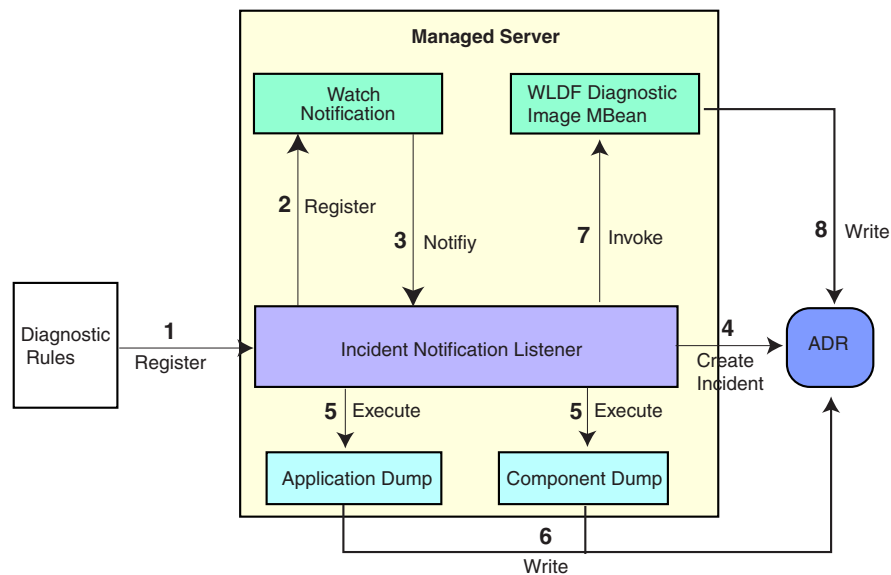
```
[2010-09-16T06:37:59.264-07:00] [dfw] [NOTIFICATION] [DFW-40104] [oracle.dfw]
[tid: 10] [ecid: 0000IF34gtMC8xT6uBf9EH1AgEck000000,0] [errid: 6]
[detailLoc: /middleware/user_projects/base_
domain/servers/AdminServer/adr/diag/ofm/base_domain/AdminServer]
[probKey: MDS-123456 [testComponent][testModule]] 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 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 parameters for the `DiagnosticConfig` MBean and a description of each parameter.

Table 13–1 *DiagnosticConfig MBean Parameters for Diagnostic Framework*

Parameter	Description
<code>floodControlEnabled</code>	Enables or disables flood control. Specify <code>true</code> for enabled or <code>false</code> for disabled. The default is <code>true</code> . Note that flood control does not apply to manually created incidents.
<code>floodControlIncidentCount</code>	Sets the number of incidents with the same problem key that can be created within the time period, specified by <code>floodControlIncidentTimeoutPeriod</code> , 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 <code>WARNING</code> level to the log file.
<code>floodControlIncidentTimeoutPeriod</code>	Sets the time period in which the number of incidents, as specified by <code>floodControlIncidentCount</code> , with the same Problem Key can be created before they are controlled by flood control. The default is 60 minutes.
<code>incidentCreationEnabled</code>	Enables or disables incident creation. Specify <code>true</code> for enabled or <code>false</code> for disabled. The default is <code>true</code> .
<code>logDetectionEnabled</code>	Enables or disables the detection of incidents through the log files. Specify <code>true</code> for enabled or <code>false</code> for disabled. The default is <code>true</code> .

Table 13–1 (Cont.) DiagnosticConfig MBean Parameters for Diagnostic Framework

Parameter	Description
maxTotalIncidentSize	<p>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.</p> <p>The default is 500MB. The limit may be exceeded during the creation of an incident, but when the incident creation completes, the oldest incidents are purged.</p>
reservedMemoryKB	<p>The amount of reserved memory that is released when OutOfMemoryError is detected.</p> <p>When the Diagnostic Framework starts, it allocates 512KB 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.</p> <p>The default is 512KB.</p>
uncaughtExceptionDetectionEnabled	<p>Enables the Java-based uncaught exception handler. When enabled and an uncaught exception is detected, an incident is created. Specify <code>true</code> for enabled or <code>false</code> for disabled.</p> <p>The default is <code>true</code>.</p>
useExternalCommands	<p>Indicates whether external JVM commands should be used to perform thread dumps. Specify <code>true</code> for enabled or <code>false</code> for disabled. The default is <code>true</code>.</p>

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:

SOA_domain Logged in as weblogic
 WebLogic Domain Page Refreshed Nov 4, 2010 12:13:48 PM PDT

System MBean Browser

Application Defined MBeans: oracle.dfw.jmx.DiagnosticsConfigMBean... Apply Revert

Hide MBean Information
 MBean Name oracle.dfw.name=DiagnosticsConfig,type=oracle.dfw.jmx.DiagnosticsConfigMBean,
 ServerName=bam_server1
 Description This configuration MBean provides attributes for configuring the Diagnostic Framework

Attributes Notifications

Name	Description	Access	Value
1 ConfigMBean	If true, it indicates that this MBean is a Config MBean.	R	true
2 eventProvider	If true, it indicates that this MBean is an event provider as defined by JSR-77.	R	true
3 eventTypes	All the event's types emitted by this MBean.	R	jmx.attribute.change
4 FloodControlEnabled	Incident flood control enabled/disabled	RW	true
5 FloodControlIncidentCount	The number of incidents that can occur with the same problem key within the time period specified by the 'flood control incident time period' setting	RW	5
6 FloodControlIncidentTimePeriod	The time span of flood control in minutes	RW	60
7 IncidentCreationEnabled	Incident creation enabled/disabled	RW	true
8 LogDetectionEnabled	Incident log filter detection enabled/disabled	RW	true
9 MaxTotalIncidentSize	Maximum disk space to set aside for all incidents under the active ADR Base	RW	500
10 objectName	The MBean's unique JMX name	R	oracle.dfw.name=Di
11 ReadOnly	If true, it indicates that this MBean is a read only MBean.	R	false
12 ReservedMemoryKB	Reserved memory, in KB, that will be released when an OutOfMemoryError occurs.	RW	512
13 RestartNeeded	Indicates whether a restart is needed.	R	false
14 SystemMBean	If true, it indicates that this MBean is a System MBean.	R	false
15 UncaughtExceptionDetectionEnabled	Uncaught exception detection enabled/disabled	RW	true
16 UseExternalThreadDumpCommand	Use of external thread dump command enabled/disabled	RW	true

- To change the values for the attributes listed in [Table 13-1](#), enter or select the value in the **Value** field.
- Click **Apply**.

13.3.2 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:

- 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:

Settings for Module-FMWDFW

Configuration Targets

General Collected Metrics **Watches and Notifications** Instrumentation

Save

Use this page to create and configure watches and notifications for this diagnostic module.

Enabled Specifies whether the Watch Notification component is enabled. [More Info...](#)

Severity: Notice The default notification severity level for all watches. When a watch triggers, the severity level is delivered with the notification. [More Info...](#)

Log Watch Severity: Warning The threshold severity level of log messages evaluated by log watch rules. Messages with a lower severity than this value will be ignored and not evaluated against the watch rules. [More Info...](#)

Save

Watches Notifications

Use this page to add watches to the current diagnostic module and to configure those watches. Click the name of an existing watch to configure that watch.

[Customize this table](#)

Watches

New Delete Showing 1 to 3 of 3 Previous | Next

<input type="checkbox"/>	Name	Type	Enabled	Alarm Type
<input type="checkbox"/>	Deadlock	Server Log	true	Automatic Reset
<input type="checkbox"/>	StuckThread	Server Log	true	Automatic Reset
<input type="checkbox"/>	UncheckedException	Server Log	true	Automatic Reset

New Delete Showing 1 to 3 of 3 Previous | Next

6. Select the Watches tab and click **New**.
The Create Watch page is displayed.
7. For **Name**, enter a name for the watch.
8. For **Watch Type**, select a type.
9. Click **Next**.
10. For **Current Watch Rule**, construct an expression. For example, (SEVERITY = 'Error') AND (MSGID = 'BEA-000337').
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

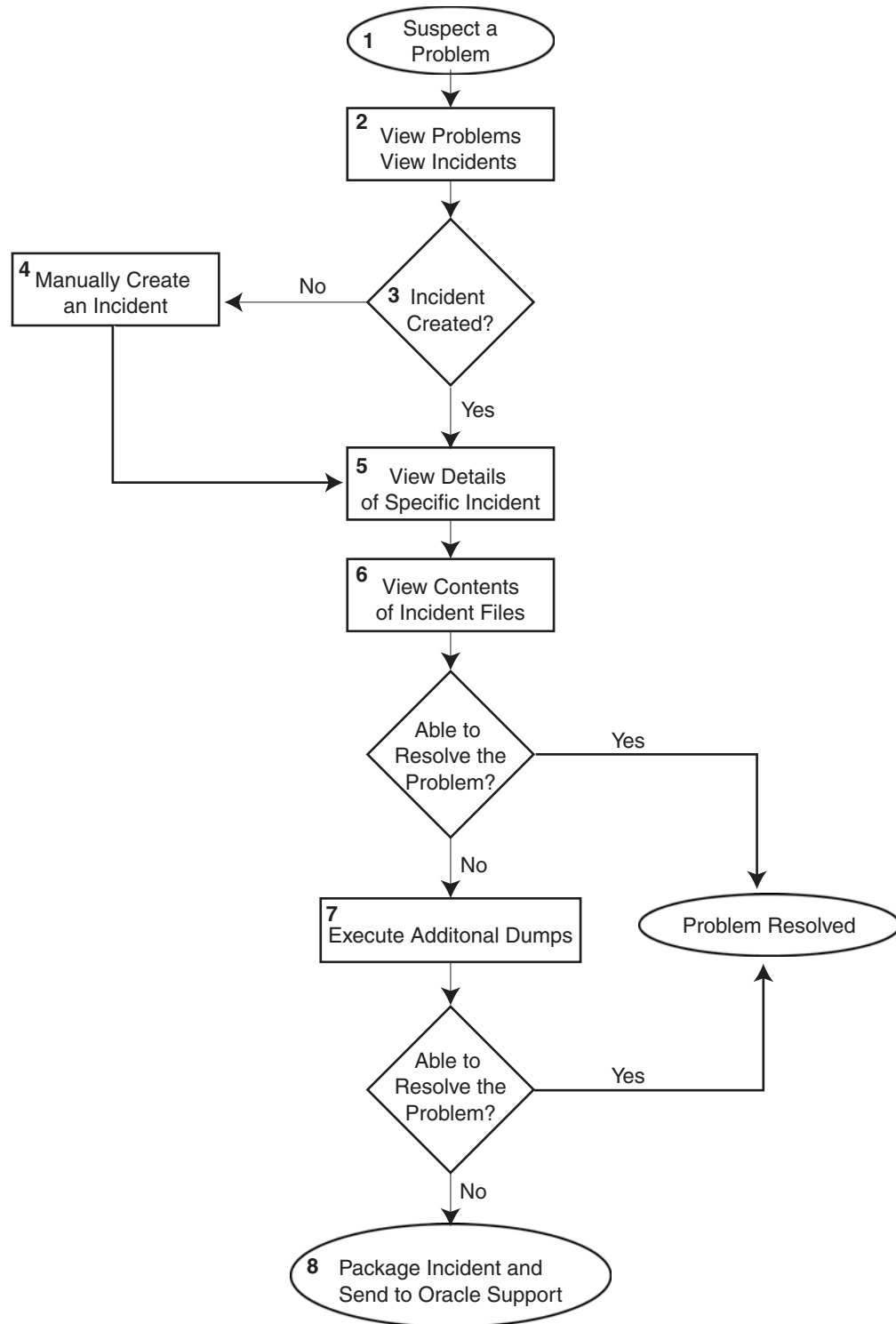
This section describes how to use WLST and ADRCI commands to investigate and report a problem (critical error), and in some cases, resolve the problem. The section begins with a roadmap that summarizes the typical set of tasks that you must perform. It describes the following topics:

- [Roadmap—Investigating, Reporting, and Resolving a Problem](#)
- [Viewing Problems and Incidents](#)
- [Working with Diagnostic Dumps](#)
- [Managing Incidents](#)

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 13.4.4.1](#) 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 13.4.3](#) 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 13.4.4.2](#) and [Section 13.4.4.3](#) 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 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

13.4.3.1 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 13–2](#) lists the diagnostic dump actions that are defined by Oracle Fusion Middleware and their descriptions.

Table 13–2 Diagnostic Dump Actions

Dump Action	Description
<code>dms.ecidctx</code>	The data associated with a specific Execution Context ID (ECID), if specified. Otherwise, the data associated with all available ECIDs.
<code>dms.metrics</code>	Dynamic Monitoring Service (DMS) metrics. For information about these metrics, see "DMS Internal Metrics" in the <i>Oracle Fusion Middleware Performance and Tuning Guide</i> .
<code>http.requests</code>	A summary of the currently active HTTP requests.
<code>jvm.classhistogram</code>	A JVM class histogram, the output of which varies depending on the JVM vendor.

Table 13–2 (Cont.) Diagnostic Dump Actions

Dump Action	Description
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.

13.4.3.2 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
```

13.4.3.3 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 13.4.3.2](#).

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`.

13.4.4 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](#)

13.4.4.1 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 13.4.3](#).

13.4.4.2 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*, which is located at:

<http://www.oracle.com/technology/documentation/database.html>

13.4.4.3 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
```

13.4.4.4 Purging Incidents

By default, incidents are purged when the total size of all incidents exceed 500MB. 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*, which is located at:

<http://www.oracle.com/technology/documentation/database.html>

Part VI

Advanced Administration

This part describes advanced administration tasks that involve reconfiguring Oracle Fusion Middleware.

It contains the following chapters:

- [Chapter 14, "Managing the Metadata Repository"](#)
- [Chapter 15, "Changing Network Configurations"](#)

Managing the Metadata Repository

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. 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.)

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.

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. 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 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 Test System to a Production 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, such as SQL Server and DB2. For more information about the supported versions of these databases, see:

http://www.oracle.com/technology/software/products/ias/files/fusion_certification.html

Note that when you use an Oracle Database, the MDS database user created by Repository Creation Utility (RCU) requires EXECUTE privilege on DBMS_OUTPUT and DBMS_LOB. When you create a metadata repository using RCU, if PUBLIC does not have EXECUTE privilege on DBMS_OUTPUT and DBMS_LOB, the RCU user must have the privilege to grant EXECUTE privilege on DBMS_OUTPUT and DBMS_LOB to the MDS user.

To ensure that you have the correct privileges, login to RCU as a SYSDBA or as a DBA user who has EXECUTE privilege with GRANT OPTION on DBMS_OUTPUT and DBMS_LOB.

Note the following about using SQL Server as the database for MDS:

- To create a metadata repository in SQL Server, set READ_COMMITTED_SNAPSHOT to ON for the hosting database. This enables the needed row versioning support. This feature can be enabled using the SQL command ALTER DATABASE, as in the following example:

```
ALTER DATABASE mds SET READ_COMMITTED_SNAPSHOT ON
```

- You should use case-sensitive collation to support the case-sensitive semantics in the metadata repository. For example, if Latin1_General is used, select the SQL_Latin1_General_CP1_CS_AS collation using the following SQL command:

```
ALTER DATABASE mds COLLATE SQL_Latin1_General_CP1_CS_AS
```

In the example, the substring CS denotes case sensitivity.

- When installed on SQL Server, Oracle Content Server 10g requires a database instance with case-insensitive collation. As a result, it cannot exist in the same SQL Server database instance as MDS. If you want to install both MDS and Oracle Content Server 10g, you must create a separate database instance with case-insensitive collation for installing the Oracle Content Server 10g schema.
- There are some minor differences between an Oracle schema and a SQL Server schema. The length of the certain text fields are shorter for a SQL Server schema. For example, the full path name of the metadata in SQL Server is limited to 400 characters.

Note the following about using DB2 as the database for MDS:

- Set the lock timeout parameter of the database to a low value. Unlike Oracle databases, DB2 does not support row versioning during update. If one user is updating a row, under some conditions, another user may be blocked when updating a different row and must wait until the transaction is committed or rolled back by the first user. To facilitate better concurrency, do not specify -1, which sets the lock timeout to infinity.

To query the lock timeout value for your DB2 database, use the following command:

```
db2 'get database config for database_alias' | grep -i timeout
```

If the value is too high, change it. For example, to change the lock timeout value to 180 seconds, use the following command:

```
db2 'update database config for database_alias using locktimeout 180'
```

- Set the DB2 registry variables `DB2_EVALUNCOMMITTED`, `DB2_SKIPINSERTED`, and `DB2_SJIPDELETED` to `OFF` to avoid deadlock and locking issues. By default, they are set to `OFF`. To view the current registry variables setting, use the `db2set -all` command.

If they are not set to `OFF`, use the following commands:

```
db2set DB2_EVALUNCOMMITTED=OFF
db2set DB2_SKIPINSERTED=OFF
db2set DB2_SKIPDELETED=OFF
```

Restart the database server using the using `db2stop` and `db2start` commands.

- DB2 may escalate a row lock to a table lock due to memory stress or lock usage. As the result, a user's transaction may be rolled back as a victim of deadlock or lock timeout. To reduce lock escalation, you can increase the size of the `MAXLOCKS` and `LOCKLIST` configuration parameters. Use the following commands:

```
db2 'update database config for database_alias using locklist value'
db2 'update database config for database_alias using maxlocks value'
```

- The isolation level must be set to `Currently Committed`. To verify the setting, use the following command:

```
db2 'get database config for database_alias' | grep -i commit
```

To set the isolation level to `Currently Committed`, use the following command:

```
db2 'update database config for database_alias using CUR_COMMIT ON'
```

- If the database transaction log is often full, increase the database configuration parameter to allow for a larger log file. A larger log file requires more space, but it reduces the need for applications to retry the operation. You should set the log file size to at least 10000 and the number of primary log files to at least 50. Use the following commands:

```
db2 'update database config for database_alias using LOGFILESIZ 10000'
db2 'update database config for database_alias using LOGPRIMARY 50'
```

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:

The screenshot shows the 'Metadata Repositories' page in Fusion Middleware Control. At the top, it indicates 'SOA_domain' and 'WebLogic Domain'. The page is titled 'Metadata Repositories' and includes a help icon. Below the title, there is a descriptive paragraph: 'You create most Fusion Middleware component schema repositories in a database using the Repository Creation Utility. Metadata Services (MDS) repositories can be created in a database with the Repository Creation Utility or created on disk as file-based repositories. You must register an MDS repository before you can deploy application metadata to the repository.'

The main content area is divided into two sections: 'Database-Based Repositories' and 'File-Based Repositories'. The 'Database-Based Repositories' section has 'Register...' and 'Deregister' buttons above a table. The table has the following data:

Repository Name	Database Type	Database Name	Schema Name
mds-soa	Oracle	orcl3.us.oracle.com	DEV3_MDS
mds-owsm	Oracle	orcl3.us.oracle.com	DEV3_MDS

The 'File-Based Repositories' section also has 'Register...' and 'Deregister' buttons above a table with the following structure:

Repository Name	Directory

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:

SOA_domain Logged in as weblogic
 WebLogic Domain Page Refreshed Nov 10, 2010 12:31:39 PM PST

Metadata Repositories > Register Metadata Repository

Register Database-Based Metadata Repository OK Cancel

A repository stores information used by Application Server components and other applications. A metadata repository must be registered to be operational. A database-based repository is created using the Repository Creation Utility. To register, input database connection information and click Query, then select one of the Metadata Repository and click OK button.

Database Connection Information

Database Type: Oracle SQL Server IBM DB2

* Host Name: * User Name:

* Port: * Password:

* Service Name: Role:

Metadata Repository	Is Registered?	Schema Name	Version	Status	Modified Time
MDS	false	OFM1_MDS	11.1.1.4.0	VALID	Oct 5, 2010 8:...
MDS	true	DEV_MDS	11.1.1.4.0	VALID	Sep 9, 2010 10:...

Selected Repository

The selected schema can be registered only if it has not already been registered.

Repository Name:

Schema Password:

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')
```

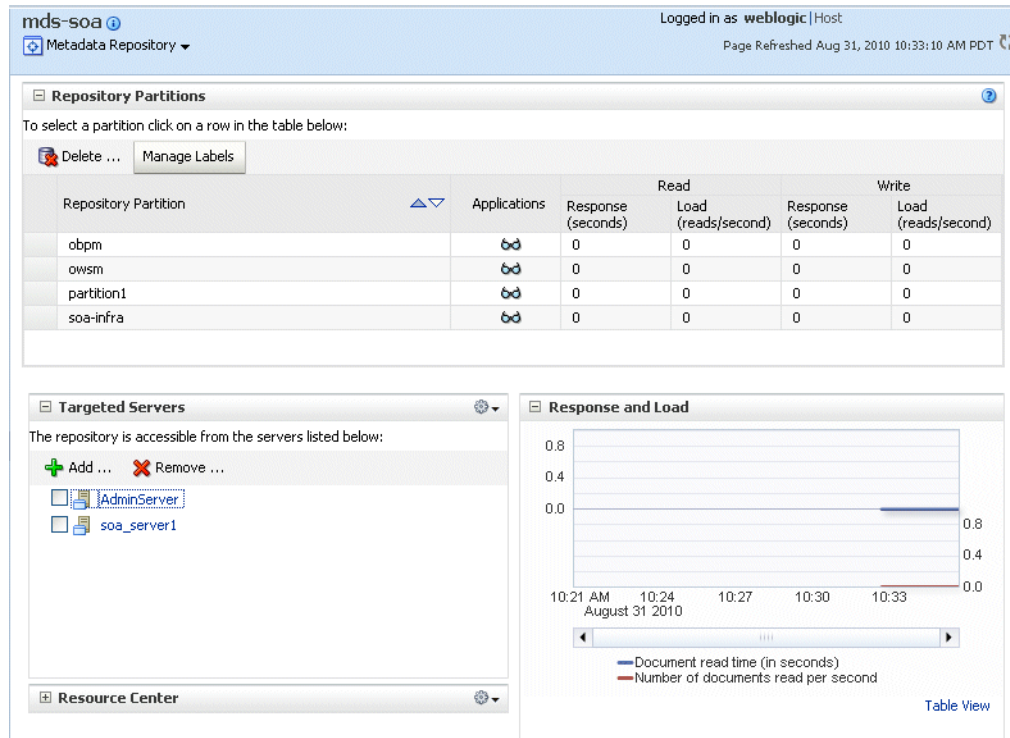
14.3.2.2 Adding or Removing Servers Targeted to the MDS Repository

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:

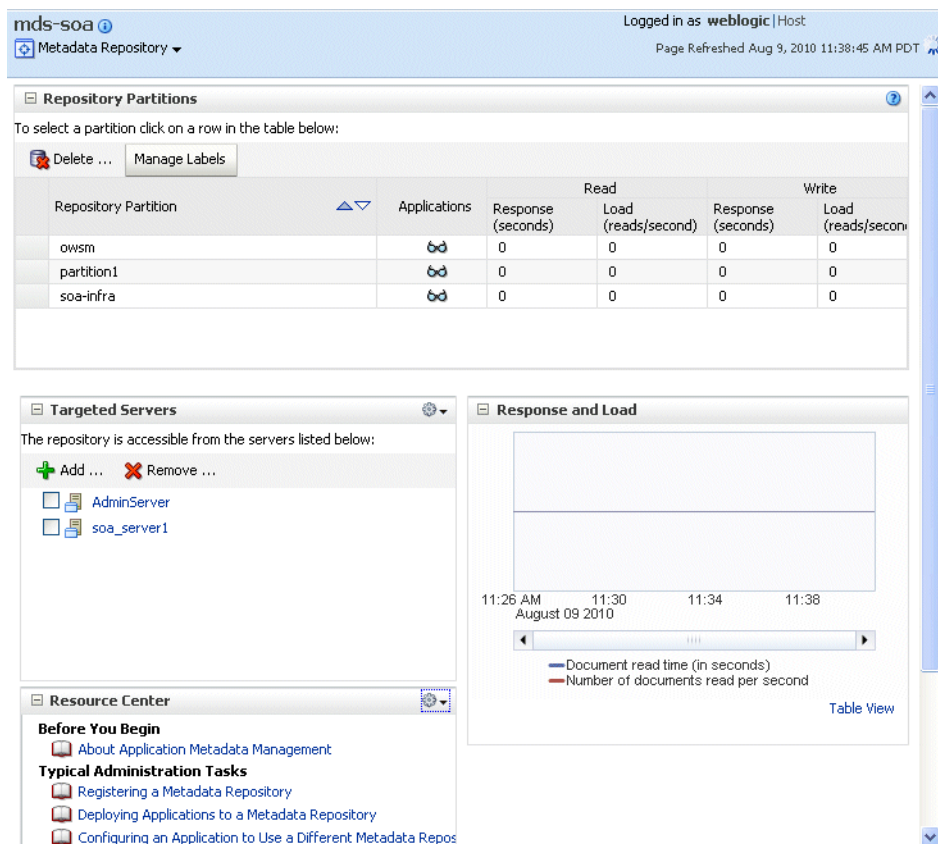
- [Viewing Information About an MDS Repository Using Fusion Middleware Control](#)
- [Viewing Information About an MDS Repository Using System MBeans](#)

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:

soa_server1 | WebLogic Server | Logged in as weblogic | Host | Page Refreshed Aug 31, 2010 10:43:54 AM PDT

System MBean Browser

Application Defined MBeans: MDSDomainRuntime:MDSDomainRuntime

Show MBean Information

Attributes | **Operations** | Notifications

Name	Description	F
1 cloneMetadataPartition	Clones the given repository partition.	F
2 createMetadataPartition	Creates a new metadata partition in the specified repository.	
3 deleteMetadataLabels	Delete metadata labels	
4 deleteMetadataPartition	Deletes the specified repository partition and all the documents within the partition.	
5 deregisterMetadataDBRepository	Deregisters DB metadata repository with the Domain.	
6 deregisterMetadataFileRepository	Deregisters File metadata repository with the Domain.	
7 isRepositoryTargeted	Is the repository targeted on the specified server or cluster?	
8 listMetadataLabels	List metadata labels	
9 listPartitions	Lists all metadata partitions in the specified repository.	
10 listRepositories	Lists all metadata repositories registered with the domain and the detail information for each repository.	
11 listRepositoryDetails	Lists all metadata partitions in the specified repository.	
12 purgeMetadataLabels	Delete metadata labels	
13 registerMetadataDBRepository	Registers DB metadata repository with the Domain.	
14 registerMetadataDBRepository	Registers DB metadata repository with the Domain.	
15 registerMetadataDBRepositoryWAS	Registers DB metadata repository with the Domain.	
16 registerMetadataDBRepositoryWAS	Registers DB metadata repository with the Domain.	
17 registerMetadataDBRepositoryWAS	Registers DB metadata repository with the Domain.	
18 registerMetadataFileRepository	Registers File metadata repository with the	

d. Select **cloneMetadataPartition**.

The Operation: cloneMetadataPartition 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 Test System to a Production 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, 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:

The screenshot shows the MDS Configuration page. At the top, it says 'mdsappdb' and 'Application Deployment'. The user is logged in as 'weblogic|Host'. The page was refreshed on Aug 31, 2010 10:50:07 AM PDT. The main section is 'MDS Configuration' with a help icon. Under 'Target Metadata Repository', the Repository is 'mds-soa', Type is 'DB', and Partition is 'partition1'. There are three main sections: 'Export', 'Import', and 'Purge'. The 'Export' section has a description: 'Export a versioned stripe of metadata documents from a metadata repository partition to a file system directory or archive. Only the tip version will be exported for a file repository.' It has an 'Export' button. There are two radio buttons: the first is selected and says 'Export metadata documents to an archive on the machine where this web browser is running.', the second says 'Export metadata documents to a directory or archive on the machine where this application is running.' Below the radio buttons is a text input field. There is also an 'Exclude base documents' checkbox. The 'Import' section has a description: 'Import metadata documents from a file system directory or archive to a metadata repository partition. If the target metadata repository is a database repository, the documents will be imported as new tip versions.' It has an 'Import' button. There are two radio buttons: the first is selected and says 'Import metadata documents from an archive on the machine where this web browser is running.', the second says 'Import metadata documents from a directory or archive on the machine where this application is running.' Below the radio buttons is a text input field with a 'Browse...' button. The 'Purge' section has a description: 'Purge the version history of unlabeled documents from the Application's repository partition that are older than the selected time period. The tip version of a document will not be purged even if it is not labeled.' It has a 'Purge' button. Below the description is a text input field for 'Purge all unlabeled past versions older than' and a dropdown menu set to 'Days'.

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 production application is on a different system, copy the exported metadata to that system.
 5. From the navigation pane for the production 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/mypartition', 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**.
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.

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 recreated.

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 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:

Repository Partition: mds-ds

Use this page to find and delete metadata labels that are no longer in use within the selected partition.

By default, the table lists all the metadata labels created more than one year ago. To show newer labels in the partition, select a new value from the controls above the Creation Time column and press Return.

For more information, click the online help icon at the top of the page.

Name	Description	Age	Creation Time
<input checked="" type="checkbox"/> Creation_test3_04:27:22		50 days, 18 hr, 9 min	15 Sep 2010 16:27 PDT
<input checked="" type="checkbox"/> PostMerge_test1_04:10:45	post-merge label for sandbox: test1	50 days, 18 hr, 26 min	15 Sep 2010 16:10 PDT
<input checked="" type="checkbox"/> test1label		50 days, 18 hr, 54 min	15 Sep 2010 15:42 PDT

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. 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.
5. To filter the labels displayed by name, enter the filter criteria in the box above **Name**. 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.
6. To filter the labels by those older than a particular time, select the time in **Older Than**. For example, if you specify 2 months, the labels older than 2 months are returned.
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,partition=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 **JDBC**, and select **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/technology/documentation/database.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 will ultimately save you time and money as well as give 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
Oracle Application Development Framework	See "Cleaning Up Temporary Storage Tables" in the <i>Oracle Fusion Middleware Fusion Developer's Guide for Oracle Application Development Framework</i> .
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: <code>ORACLE_COMMON_HOME/common/sql/adfbc_purge_statesnapshots.sql</code> The PS_TXN table is automatically purged.
Oracle SOA Suite	See "Managing Database Growth" in the <i>Oracle Fusion Middleware Administrator's Guide for Oracle SOA Suite and Oracle Business Process Management Suite</i> .
Oracle WebLogic Server: Oracle Infrastructure Web Services	Use the following script to purge data if WS-RM uses a database store: <code>ORACLE_COMMON_HOME/common/sql/ows_purge_wsrmsg.sql</code>

Table 14–2 (Cont.) Purging Data Documentation

Component	Description
Oracle WebLogic Server: JAXWS Web Services	Clean up the Web Service persistence store, as described in "Cleaning Up Web Service Persistence" in <i>Oracle Fusion Middleware Programming Advanced Features of JAX-WS Web Services for Oracle WebLogic Server</i> . Use the defaultMaximumObjectLifetime field of the WebServicePersistenceMBean to set the maximum lifetime of the objects. See "Understanding WebLogic Server MBeans" in <i>Oracle Fusion Middleware Developing Custom Management Utilities With JMX for Oracle WebLogic Server</i> .
Oracle WebLogic Server: Stateful EJBs	No configuration required. Automatically purges data.
Oracle WebLogic Server: JMS	See "Configuring Basic JMS System Resources" and "Managing JMS Messages" in <i>Oracle Fusion Middleware Configuring and Managing JMS for Oracle WebLogic Server</i> . Also see "Tuning WebLogic JMS" in <i>Oracle Fusion Middleware Performance and Tuning for Oracle WebLogic Server</i> .
Oracle WebLogic Server: Session persistence for JDBC or file-based data sources	No configuration required. Automatically purges data.
MDS Repository	See Section 14.3.11 for information on automatically and manually purging data.
Oracle Web Services Manager	No configuration required. Automatically purges data.
Oracle Universal Content Management	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 <i>Oracle Fusion Middleware System Administrator's Guide for Content Server</i> .
Oracle WebCenter Spaces and Lists	Purge MDS metadata, as described in Section 14.3.11 .
Oracle WebCenter Activity Stream	See Section 14.5.2.1 .
Oracle WebCenter Analytics	See Section 14.5.2.2 .
Oracle Real-Time Decisions	No configuration required. Automatically purges data.
Oracle BI Enterprise Edition	No configuration required. Automatically purges data.
Oracle Business Intelligence Publisher	Delete job history, as described in "Deleting a Job History" in the <i>Oracle Fusion Middleware Report Designer's Guide for Oracle Business Intelligence Publisher</i> .
Oracle Internet Directory	No configuration required. Automatically purges data.
Oracle Identity Manager	No configuration required. Automatically purges data.
Oracle Identity Federation	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_wsrmsg.sql
```

14.5.2 Purging Oracle WebCenter Data

The following topics describe purging Oracle WebCenter data:

- [Purging Oracle WebCenter Activity Stream Data](#)
- [Purging Oracle WebCenter Analytics Data](#)
- [Partitioning Oracle WebCenter Analytics Data](#)

14.5.2.1 Purging Oracle WebCenter Activity Stream Data

Oracle WebCenter Activity Streaming provides a set of WLST commands for purging database records in a non-partitioned 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 decreases because of the large volume of data.

To purge Oracle WebCenter 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.

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 Analytics Data

Oracle WebCenter Analytics provides a script for purging database records in a non-partitioned environment. Purging is necessary when a database contains records that are not needed for analysis in reports or when the performance of Oracle WebCenter 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 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 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 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 will purge 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 will purge 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 will purge 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 will purge 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 process will run. When the process reaches this time, it will stop, 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 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 uses the native partitioning of the database to automatically create partitions.

Oracle WebCenter 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_Uilities 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;
```

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:

- [Changing the Network Configuration of Oracle Fusion Middleware](#)
- [Changing the Network Configuration of a Database](#)
- [Moving Between On-Network and Off-Network](#)
- [Changing Between a Static IP Address and DHCP](#)
- [Using IPv6](#)

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 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:

- [Changing the Network Configuration of a Managed Server](#)
- [Changing the Network Configuration of Web Tier Components](#)

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 the Help.

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 hostname, domain name, or IP address to ensure that everything is resolving properly.

Task 4 Change References to the Network Configuration

You need to 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 non-default 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 `PlsqlDatabaseConnectionString` 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 `PlsqlDatabaseConnectionString` 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 4.2.1](#).
2. Go to the Administration Console.
3. In the Change Center, click **Lock & Edit**.
4. In the Domain Structure section, expand **Services**, then **JDBC**, and select **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 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.
- 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 Manager	Yes	Yes	To configure for IPv6, see Section 15.5.5 .
Oracle Application Development Framework	Yes	Yes	
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.

Table 15–1 (Cont.) Support for IPv6

Component	IPv6 Only	Dual Stack	Notes
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 Identity Federation	No	Yes	Requires a dual stack because Oracle Database requires IPv4 addresses.
Oracle Imaging and Process Management	No	Yes	Requires a dual stack, but the client (the browser) can be on a host configured for IPv6
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 <i>Oracle Fusion Middleware Administrator's Guide for Oracle Internet Directory</i> .
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 <i>Oracle Fusion Middleware Administrator's Guide for Oracle Portal</i> 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.
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 .

Table 15–1 (Cont.) Support for IPv6

Component	IPv6 Only	Dual Stack	Notes
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 <i>Oracle Fusion Middleware Administrator's Guide for Oracle Virtual Directory</i> .
Oracle Web Cache	Yes	Yes	Enabled by default. To disable, see Section 15.5.3 .
Oracle WebCenter	No	Yes	Requires a dual stack because Oracle Database requires IPv4 addresses.
Oracle WebLogic Server	Yes	Yes	Most Oracle WebLogic Server plug-ins do not support IPv6. The mod_wl_ohs plug-in for Oracle HTTP Server does support IPv6.

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 Manager 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, 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, 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, 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 `WLogFile` 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 `MULTIPOINT` 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" />
  </MULTIPOINT>
  <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_HOME/opmn/bin/opmnctl restartproc process-type=HTTP_Server
ORACLE_HOME/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 11g Release 1 (11.1.1) to use as a reverse proxy for IPv6.
3. Change the Oracle HTTP Server 11g Release 1 (11.1.1) 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 Manager Support for IPv6

Oracle Access Manager supports Internet Protocol Version 4 (IPv4). Oracle Fusion Middleware supports Internet Protocol Version 4 (IPv4) and Internet Protocol Version 6 (IPv6). IPv6 is enabled with Oracle HTTP Server with the mod_wl_ohs plug-in.

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.

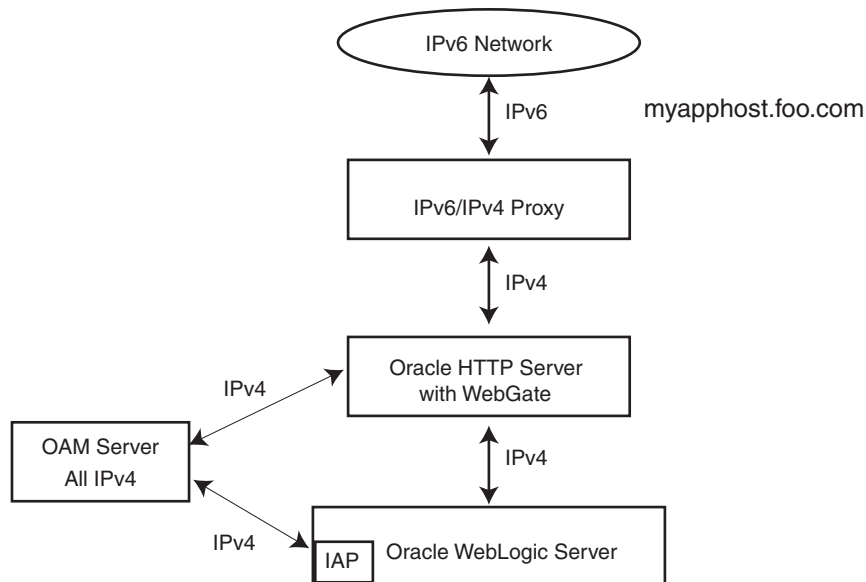
This section describes configuring Oracle Access Manager 10g for IPv6. For information about configuring Oracle Access Manager 11g for IPv6, see "Configuring OAM 11g for IPv6 Clients" in the *Oracle Fusion Middleware Administrator's Guide for Oracle Access Manager*.

15.5.5.1 Simple Authentication with IPv6

[Figure 15–1](#) 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 15–1 Simple Authentication with the IPv6/IPv4 Proxy

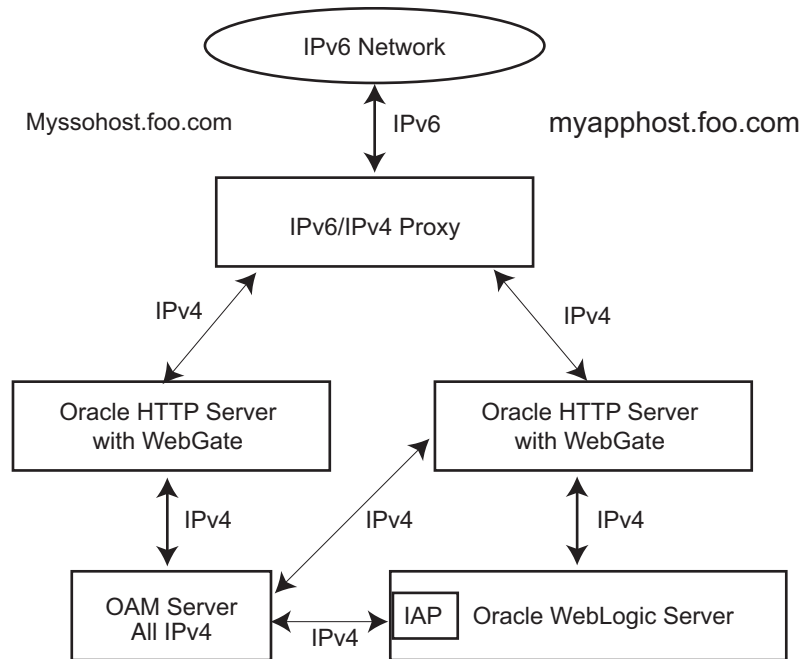


As illustrated in [Figure 15–1](#), the IPv6 network communicates with the IPv6/IPv4 proxy, which in turn communicates with the Oracle HTTP Server and WebGate using IPv4. WebGate, Oracle Access Manager servers, and Oracle WebLogic Server with the Authentication provider all communicate with each other using IPv4.

15.5.5.2 Configuring IPv6 with an Authenticating WebGate and Challenge Redirect

[Figure 15–2](#) 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 15–2 IPv6 with an Authenticating WebGate 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, 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*.

15.5.5.3 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.

15.5.5.4 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*

15.5.5.5 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 15-1](#) 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 11g 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**.

15.5.5.6 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 15–2](#) 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 11g 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 /oamssso/login.html http://myssohostv4.foo.com/oamssso/login.html  
ProxyPassReverse /oamssso/login.html http://myssohostv4.foo.com/oamssso/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**.

15.5.5.7 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 11g 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:

- [Chapter 16, "Introducing Backup and Recovery"](#)
- [Chapter 17, "Backing Up Your Environment"](#)
- [Chapter 18, "Recovering Your Environment"](#)

Introducing Backup and Recovery

This chapter provides an introduction to backing up and recovering Oracle Fusion Middleware.

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:

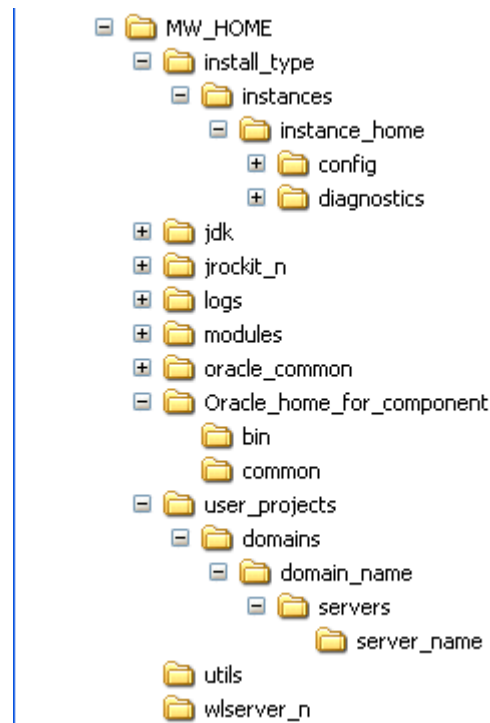
- On each Managed Server restart, the latest configuration is retrieved from the Administration Server. This happens even when the Node Manager is down on the node where the Managed Server is running. If the Administration Server is unavailable during the Managed Server restart and if the MSI (Managed 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 will fail. 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.

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.

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
- OraInst.loc and oratab files, which are located in the following directory:

```
(Linux and IBM AIX) /etc  
(Other UNIX systems) /var/opt/oracle
```

- The beahomelist file, which is located at:

(UNIX) `user_home/boa/beahomelist`
 (Windows) `C:\boa\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. However, note the limitation described in [Section 17.2](#).

16.3.2 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 or cluster.
- After making configuration changes to a component or cluster.
- 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 an offline backup of static files and directories:** This involves backing up the static files and directories described in [Section 16.3.1](#). Perform an offline backup of static files and directories at the following times:
 - After patching your Oracle Fusion Middleware environment. This backup serves as the basis for subsequent online backups.
 - After upgrading your Oracle Fusion Middleware environment. This backup serves as the basis for subsequent online backups.

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
- Metadata repository 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 will fail. 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
- A domain
- The WebLogic Server Administration Server
- A Managed Server
- An Oracle home
- An Oracle instance home
- A component, such as Oracle SOA Suite or Oracle Web Cache
- A 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](#)
- [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 Data Integrator](#)
- [Backup and Recovery Recommendations for Oracle Enterprise Content Management Suite](#)

These topics include information about configuration files for particular components. Note that the list of files is 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, Managed Server, 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*, which is available at:

<http://www.oracle.com/technology/documentation/database.html>

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

If a JMS uses a JDBC accessible store, back up the database.

Backup Recommendations

Back up the domain and the JMS file persistent store if it is not located within the domain.

Back up the schema in the database if JDBC-based persistent store is configured. Note the following:

- All JMS data should be backed up using an offline backup.
- 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 (in the case of database-based persistence) or using a highly available RAID-backed storage device (for example, SAN/NAS).
- It is currently not possible to take consistent backup of persistent stores for a system that uses JMS and transaction logs. This is because the transaction logs can only be file-based and the JMS can be either file-based or it can reside in the database. For highest reliability, use a highly available fault-tolerant storage (for example, SAN) for JMS and transaction log file stores.
- 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. If the persistent store is a custom store that is dedicated to JMS use, then you can delete the entire store.

See additional information and limitations in [Section 17.2](#).

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.

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.

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 Identity Federation](#)
- [Backup and Recovery Recommendations for Oracle 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](#)

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/nxli7i/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 Identity Federation

This section describes the Oracle 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 Oracle 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 Manager

This section describes the Oracle 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 OES for the Oracle Access Manager policy store.

Backup Recommendations

Back up the Middleware home and the domain home for the Oracle 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 OES for the Oracle Access Manager policy store.

Recovery Recommendations

Recover the Middleware home and the domain home for the Oracle 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 Oracle 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](#).

16.5.3 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.

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](#).

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.

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

The following sections describe backup and recovery recommendations for Oracle WebCenter:

- [Backup and Recovery Recommendations for Oracle WebCenter](#)
- [Backup and Recovery Recommendations for Oracle WebCenter Portlet](#)
- [Backup and Recovery Recommendations for Oracle WebCenter Discussions Server](#)
- [Backup and Recovery Recommendations for Oracle WebCenter Wiki and Blog Server](#)
- [Backup and Recovery Recommendations for Oracle WebCenter Activity Graph](#)
- [Backup and Recovery Recommendations for Oracle WebCenter Analytics](#)
- [Backup and Recovery Recommendations for Oracle Content Server](#)

16.5.4.1 Backup and Recovery Recommendations for Oracle WebCenter

This section describes the Oracle WebCenter 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 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 Portlet

This section describes the Oracle WebCenter Portlet 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 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 Discussions Server

This section describes the Oracle WebCenter Discussions 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

JIVE schema

Backup Recommendations

Back up the Administration Server domain and the database containing the JIVE schema.

Recovery Recommendations

Recover the Oracle WebCenter 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 Wiki and Blog Server

This section describes the Oracle WebCenter Wiki and Blog Server data that must be backed up and restored.

Configuration Files

Configuration files are bundled in the WAR file, which is located in the domain.

Database Repository Dependencies

WIKI schema

Backup Recommendations

Back up the Administration Server domain and the database containing the WIKI schema.

Recovery Recommendations

Recover the Oracle WebCenter 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](#).

16.5.4.5 Backup and Recovery Recommendations for Oracle WebCenter Activity Graph

This section describes the Oracle WebCenter 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 Activity Graph, see [Section 18.2.7.8](#).

16.5.4.6 Backup and Recovery Recommendations for Oracle WebCenter Analytics

This section describes the Oracle WebCenter 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 Analytics, see [Section 18.2.7.9](#).

16.5.4.7 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

Database Repository Dependencies

OCSERVER 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

None

Backup Recommendations

Back up the Administration Server domain.

Recovery Recommendations

Restore the jps-config.xml file.

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](#) and [Section 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_
plsqli.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, and the Oracle instance containing Oracle Portal, as well as 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/jdbcpsds.conf
ORACLE_INSTANCE/config/ReportsServer/server_name/xmlpsds.conf
ORACLE_INSTANCE/config/ReportsServer/server_name/textpsds.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_REPORTSn` 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/jdbcpsds.conf
DOMAIN_HOME/config/fmwconfig/servers/server_name/applications/reports_
version/configuration/xmlpsds.conf
DOMAIN_HOME/config/fmwconfig/servers/server_name/applications/reports_
version/configuration/textpsds.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/jdbcpsds.conf
ORACLE_INSTANCE/config/ReportsTools/xmlpsds.conf
ORACLE_INSTANCE/config/ReportsTools/textpsds.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 Enterprise Edition 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 NQSCfg.INI configuration file points to the Repository Publishing Directory (RPD) name. The NQSCfg.INI file *must* exist in the following location:

```
ORACLE_INSTANCE/bifoundation/OracleBIServerComponent/comp_instance
name/repository/
```

Database Repository Dependencies

UserStats, BISERVER, SCHEDULER, and BIPUBLISHER 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 BIP_URL
-login username -pwd password
```

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

BIPUBLISHER 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 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 ODI 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 18.2.7.13](#). For the steps specific to recovering from loss of host, see [Section 18.3.4.5](#).

16.5.10 Backup and Recovery Recommendations for Oracle Enterprise Content Management Suite

The following sections describe backup and recovery recommendations for Oracle Enterprise Content Management Suite:

- [Backup and Recovery Recommendations for Oracle Information Rights Management](#)
- [Backup and Recovery Recommendations for Oracle Imaging and Process Management](#)
- [Backup and Recovery Recommendations for Oracle Universal Content Management](#)
- [Backup and Recovery Recommendations for Oracle Universal Records Management](#)

16.5.10.1 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 18.2.7.14](#). You use the same procedure to recover from loss of host.

16.5.10.2 Backup and Recovery Recommendations for Oracle Imaging and Process Management

This section describes the Oracle Imaging and Process Management 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 Imaging and Process Management, see [Section 18.2.7.15](#). You use the same procedure to recover from loss of host.

16.5.10.3 Backup and Recovery Recommendations for Oracle Universal Content Management

This section describes the Oracle Universal Content Management 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 Universal Content Management, see [Section 18.2.7.16](#). For the steps specific to recovering from loss of host, see [Section 18.3.4.13.1](#).

16.5.10.4 Backup and Recovery Recommendations for Oracle Universal Records Management

Because Oracle Universal Records Management depends on Oracle Universal Content Management and has no additional backup and recovery artifacts, see the backup and recovery recommendations for Oracle Universal Content Management in [Section 16.5.10.3](#).

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](#).

- File systems and files can only be restored as of the last *good* backup. There is no support for roll-forward recovery to the current point in time.
- All of the files required for recovery are maintained within the Middleware home, Oracle instance home, and Oracle Inventory (for loss of host use cases) directories. Generally, if no administration changes, such as configuration changes, deployments, redeployments, or patching, have been done during the backup, it is always safe to restore the file system pertaining to a particular component to a previous point in time using the last *good* backup. Thus, new backups must always be performed after any administration changes.
- 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.
- If multiple Managed Servers run on different hosts (not in a cluster), the domain should be configured to use an external LDAP for policy store instead of using file-based policy store.
- If the Administration Server is on a different host than the Managed Servers, the domain should be configured to use an external LDAP for the policy store instead of using a file-base policy store.

See Also: If you are using Cold Failover Cluster or Disaster Recovery, refer to the *Oracle Fusion Middleware High Availability Guide* for additional information.

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 16.3.2](#), 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/bin/nodemanager.properties  
MW_HOME/wlserver_n/common/bin/wlsifconfig.sh  
MW_HOME/wlserver_n/common/bin/setPatchEnv.sh  
MW_HOME/wlserver_n/common/bin/commEnvg.sh
```

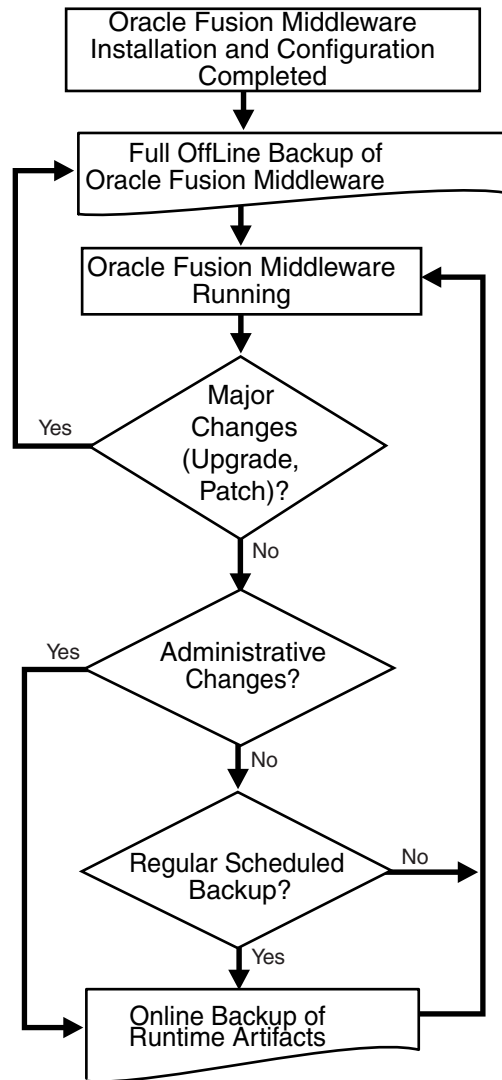
See [Section 17.3.1](#) for information on performing a full backup.

- 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 March 20, 2010, name the backup:

```
mw_home_backup_092010.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.
- Persistent stores: A persistent store provides a built-in, high-performance storage solution for WebLogic Server subsystems and services that require persistence. For example, it can store persistent JMS (Java Messaging Service) messages or durable subscriber information, as well as temporarily store messages sent to an unavailable destination using the Store-and-Forward feature. The persistent store

supports persistence to a file-based store (File Store) or to a JDBC-enabled database (JDBC Store). The default store maintains its data in the `data\store\default` directory inside the *servername* subdirectory of a domain's root directory.

It is currently not possible to take consistent backup of persistent stores for a system that uses JMS and transaction logs. This is because the transaction logs can only be file-based and the JMS can be either file-based or it can reside in the database. For highest reliability, use a highly available, fault-tolerant storage (for example, SAN) for JMS and transaction log file stores.

For clustered servers, you can migrate a failing server, including the Transaction Recovery Service, to a new system. When the server migrates to another system, it must be able to locate the transaction log records to complete or recover transactions. Transaction log records are stored in the default persistent store for the server.

If you plan to migrate clustered servers in the event of a failure, you must set up the default persistent store so that it stores records in a shared storage system that is accessible to any potential system to which a failed migratable server might be migrated. For highest reliability, use a shared storage solution (for example, SAN) or an Oracle database that is highly available and supports a point-in-time recovery. This solution is also recommended for all the JMS modules.

- **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](#)

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. 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 will fail. 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.

If you use a third-party backup solution, ensure that the tool meets the requirements listed previously.

- On Linux and UNIX, for online and offline backups, use tar.
- Oracle Recovery Manager (RMAN) to back up database-based metadata repositories.

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:

```
tar -cf mw_home_backup_092010.tar 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.

For example:

```
tar -cf domain_home_backup_092010.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:

```
tar -cf sc_home_backup_092010.tar ORACLE_INSTANCE/*
```

5. If a Managed Server is not located within the domain, back up the Managed Server directory. For example:

```
tar -cf mg1_home_backup_092010.tar server_name/*
```

6. Back up the OraInventory directory. For example:

```
tar -cf Inven_home_backup_092010.tar /scratch/oracle/OraInventory
```

7. Back up oraInst.loc and oratab files, which are located in the following directory:

```
/etc
```

8. Back up the database repositories using the Oracle Recovery Manager (RMAN). For detailed steps, see the *Oracle Database Backup and Recovery User's Guide*, which is available at:

<http://www.oracle.com/technology/documentation/database.html>

9. On Windows, export the following registry key:

```
HKEY_LOCAL_MACHINE\Software\Oracle
```

In addition, for system components, such as Oracle Web Cache, export the following Windows Registry key:

```
HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services
```

To export a key, use the following command:

```
regedit /E FileName Key
```

For example:

```
regedit /E C:\oracleregistry.reg HKEY_LOCAL_MACHINE/oracle
```

You can also use the Registry Editor to export the key. See the Registry Editor Help for more information.

10. For Oracle BI EE on Windows, you should export Windows Registry entries, as described in [Section 17.3.3](#).
11. 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 on a regular basis and at the times described in [Section 16.3.2](#).

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. For example:

```
tar -cf domain_home_backup_092010.tar 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.

3. Back up the Oracle instance home. This backs up the system components, such as Oracle HTTP Server. For example:

```
tar -cf sc_home_backup_092010.tar ORACLE_INSTANCE/*
```

4. Back up the database repositories using the Oracle Recovery Manager (RMAN). For detailed steps, see the *Oracle Database Backup and Recovery User's Guide*, which is available at:

<http://www.oracle.com/technology/documentation/database.html>

5. Create a record of your Oracle Fusion Middleware environment. See [Section 17.4](#).

17.3.3 Exporting Oracle BI EE Registry Entries On Windows

On Windows, export the Oracle BI EE Registry entries. In the case of loss of host, you can import those entries into the new host. Take the following steps:

1. Execute the following command:

```
regedit
```

2. In the Registry Editor, locate the node HKEY_LOCAL_MACHINE\SOFTWARE\Oracle.
3. Right-click the node and click **Export**. Enter a filename and click **Save**.
4. Locate the node locate the node HKEY_LOCAL_MACHINE\SYSTEM\ControlSet001\Services.
5. Export each node within this node that begins **Oracle**. (Right-click the node and click **Export**. Enter a filename and click **Save**.)
6. Locate the node locate the node HKEY_LOCAL_MACHINE\SYSTEM\ControlSet002\Services.
7. Export each node within this node that begins **Oracle**. (Right-click the node and click **Export**. Enter a filename and click **Save**.)
8. Locate the node locate the node HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services.
9. Export each node within this node that begins **Oracle**. (Right-click the node and click **Export**. Enter a filename and click **Save**.)

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

Recovering Your Environment

This chapter describes recommended recovery strategies and procedures for recovering Oracle Fusion Middleware from different types of failures and outages.

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:

- 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 will fail. 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.

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](#)

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_092010.tar
(Windows) jar xtf mw_home_backup_092010.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_092010.tar
(Windows) jar xtf domain_backup_092010.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_092010.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_092010.tar
(Windows) jar xtf Instance_home_backup_092010.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_092010.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 -wlsServerHome 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 or if the file system is lost, take the following steps:

- a. Recover the Middleware home from the backup, if required. For example:

```
tar -xf mw_home_backup_092010.tar
```

- b. 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.

- c. Unpack the domain template jar file, using the unpack utility:

```
unpack.sh -template=/scratch/aim1/ms.jar
-domain=MW_HOME/user_projects/domains/domain_name
-log=/scratch/logs/new.log -log_priority=info
```

- d. 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 mode applications.

- e. 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_092010.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/aim1/ms.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_092010.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 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 Real-Time Decisions	Section 18.2.7.12

Table 18–1 (Cont.) Recovery Procedures for Particular Components

Component	Procedure
Oracle Data Integrator	Section 18.2.7.13
Oracle Identity Manager	Section 18.2.7.3
Oracle Identity Navigator	Section 18.2.7.4
Oracle Imaging and Process Management	Section 18.2.7.15
Oracle Information Rights Management	Section 18.2.7.14
Oracle Universal Content Management	Section 18.2.7.16
Oracle Universal Records Management	Section 18.2.7.17
Oracle WebCenter Activity Graph	Section 18.2.7.8
Oracle WebCenter 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 "Reconciliation Configuration" in *Oracle Fusion Middleware User's Guide for Oracle Identity Manager*.

Note: Oracle recommends that you make sure 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, make sure 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 Manager

To recover Oracle Access Manager:

1. Restore the Middleware home and the domain home for the Oracle 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 Oracle 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 Activities Graph

To recover Oracle WebCenter Activities 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 Analytics

To recover Oracle WebCenter 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 Web 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. Restore the database containing the Oracle BI EE schemas, if necessary. See [Section 18.2.10](#).
3. Reconcile the LDAP Database with the Web Catalog and the Oracle BI EE repository (RPD), as described in [Section 18.2.7.10.3](#).

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 non-clustered environment. Recover the following, depending on the extent of the failure:

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. Restore the database containing the Oracle BI EE schemas, if necessary. See [Section 18.2.10](#).
 5. Reconcile the LDAP Database with the Web Catalog and the Oracle BI EE repository (RPD), as described in [Section 18.2.7.10.3](#).

18.2.7.10.3 Reconciling the LDAP Database with the Web Catalog and RPD You must reconcile the LDAP database with the Web Catalog and 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.

To enable synchronization, edit the following file:

```
INSTANCE_HOME/config/OracleBIServerComponent/coreapplication_obis1/NQSConfig.INI
```

Set the flag `FMW_UPDATE_ROLE_AND_USER_REF_GUIDS` to `yes`. Then, restart the servers. The information in the LDAP database and the Web Catalog and RPD will be synchronized.

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*.

18.2.7.11 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. Restore the database containing the Oracle Business Intelligence Publisher schemas, if necessary. See [Section 18.2.10](#).

If backup artifacts are restored from different times, then user accounts, user reports, and user permissions will 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 different time, the analytic models will have missed a period of learning, but their intelligence will be unaffected.

18.2.7.13 Recovering Oracle Data Integrator

To recover Oracle Data Integrator:

1. If the database must be restored, 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](#).

```
cd ORACLE_HOME
tar -xf oracle_home_backup_092010.tar
```

3. Recover the domain directory from backup:

```
cd DOMAIN_HOME
tar -xf domain_backup_092010.tar
```

18.2.7.14 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. Restore the database, if necessary. See [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.

18.2.7.15 Recovering Oracle Imaging and Process Management

Oracle Imaging and Process Management stores data in the following locations:

- A database for Oracle I/PM configuration data
- A database that functions as a document repository
- JMS persistent queues

When you recover Oracle I/PM, you should ensure that all data is restored from the same point-in-time.

To recover Oracle I/PM:

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](#).

18.2.7.16 Recovering Oracle Universal Content Management

To recover Oracle Universal Content Management:

1. If the database must be restored, restore it, 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_092010.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 IDCAlyse 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 IDCAlyse.

18.2.7.17 Recovering Oracle Universal Records Management

Because Oracle Universal Records Management depends on Oracle Universal Content Management and has no additional backup and recovery artifacts, see the recovery procedure for Oracle Universal Content Management in [Section 18.2.7.16](#).

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*, which is available at:

<http://www.oracle.com/technology/documentation/database.html>

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 an Oracle WebLogic Server Domain](#)
- [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 an Oracle WebLogic Server Domain

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_092010.tar
(Windows) jar xtf domain_backup_092010.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. 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.

2. 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_092010.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_092010.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 the 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_092010.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_092010.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 the 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 the 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 the 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_092010.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_092010.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/aime1/ms.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` or `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 the 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_092010.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_092010.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/aim1/ms.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` or `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 the 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 the Node Manager running in the new host with the Administration Server:

```
connect('username', 'password', 'http://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 Manager	Section 18.3.4.5.7
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

Table 18–2 (Cont.) Recovery Procedures for Loss of Host for Particular Components

Component	Procedure
Oracle Business Process Management	No additional steps needed for loss of host. See Section 18.2.7.7 for information about recovering Oracle Business Process Management.
Oracle Data Integrator	Section 18.3.4.12
Oracle Directory Integration Platform	Section 18.3.4.5.3
Oracle Forms Services	Section 18.3.4.8.2
Oracle HTTP Server	Section 18.3.4.7.1
Oracle Identity Federation	Section 18.3.4.5.4
Oracle Identity Manager	Section 18.3.4.5.5
Oracle Identity Navigator	Section 18.3.4.5.6
Oracle Imaging and Process Management	No additional steps needed for loss of host. See Section 18.2.7.15 for information about recovering Oracle Imaging and Process Management.
Oracle Information Rights Management	No additional steps needed for loss of host. See Section 18.2.7.14 for information about recovering Oracle Information Rights Management.
Oracle Internet Directory	Section 18.3.4.5.1
Oracle Portal	Section 18.3.4.8.1
Oracle Real-Time Decisions	Section 18.3.4.11
Oracle Reports	Section 18.3.4.8.3
Oracle SOA Suite	No additional steps needed if recovering to the same host. See Section 18.3.4.6 for information about recovering to a different host.
Oracle Universal Content Management	Section 18.3.4.13.1
Oracle Universal Records Management	Section 18.3.4.13.2
Oracle Virtual Directory	Section 18.3.4.5.2
Oracle Web Cache	Section 18.3.4.7.2
Oracle WebCenter Activity Graph	No additional steps needed for loss of host. See Section 18.2.7.8 for information about recovering Oracle WebCenter Activity Graph.
Oracle WebCenter Analytics	No additional steps needed for loss of host. See Section 18.2.7.9 for information about recovering Oracle WebCenter Analytics.

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 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 Manager to a Different Host](#)
- [Recovering Oracle Adaptive Access Manager to a Different 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 Identity Federation to a Different Host Because Oracle Identity Federation provides SSO functionality, if the host name on which Oracle 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 Oracle Identity Federation server.

If a load balancer is part of the environment and the host where Oracle 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 Oracle 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 Oracle Identity Federation, then the host name must be updated manually for Oracle Identity Federation and remote partners.

To recover Oracle 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 Oracle Identity Federation instance.
 - c. From the Oracle 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 Oracle Identity Federation is deployed:

```
DOMAIN_HOME/bin/startManagedWebLogic.sh managed_server_name
admin_url
```

6. If Oracle Identity Federation is acting as an SSL server, you must replace the SSL certificate presented by Oracle 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 Manager to a Different Host To recover Oracle 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 Oracle Access Manager console.
4. Select the Oracle Access Manager proxy server.
5. Modify **Host**, specifying the new host name.
6. If you have protected pages, you must reregister Oracle Single Sign-On or WebGate as partners with Oracle Access Manager, using the oamreg tool, described in "About the Remote Registration Tool" in the *Oracle Fusion Middleware Administrator's Guide for Oracle Access Manager*. Also see "OSSO Remote Registration Request" 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 Oracle 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](#).

18.3.4.6 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 `OsoConfigFile` 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>
  OsoIpCheck off
  OsoSecureCookies off
  OsoIdleTimeout off
  OsoConfigFile /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 need to 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
                        -wlsServerHome wlsServer_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
```

- `rwsvrlet.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>
  OsoIpCheck off
  OsoSecureCookies off
  OsoIdleTimeout off
  OsoConfigFile /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
  -wlsServerHome wlsServer_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 OsoConfigFile 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>
    OsoIpCheck off
    OsoSecureCookies off
    OsoIdleTimeout off
    OsoConfigFile /tmp/path_of_file_created
```

18.3.4.9 Recovering Oracle BI Enterprise Edition to a Different Host

You can recover Oracle BI Enterprise Edition to a different host, but the new host must have the same name as the original host.

Note: You cannot recover Oracle BI EE or any other component which is co-located with Oracle BI EE to a new host that has a new host name. You can only recover Oracle BI EE and other components to a new host with the same host name as the original host. For example, if Oracle BI EE is co-located with Oracle Real-Time Decisions and Oracle Business Intelligence Publisher, none of these components can be recovered to a new host with different host name.

The following topics describe how to move Oracle BI EE to a different host with the same name:

- [Recovering Oracle BI Enterprise Edition to a Different Host in a Non-Clustered Environment](#)
- [Recovering Oracle BI Enterprise Edition to a Different Host in a Clustered Environment](#)

18.3.4.9.1 Recovering Oracle BI Enterprise Edition to a Different Host in a Non-Clustered Environment The steps you take to recover Oracle BI EE to a different host depends on the operating system:

- On UNIX, take the following steps:
 1. Restore the Middleware home, 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\vc80 redistrib_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 Enterprise Edition 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 must be replaced for some reason, such as losing the original operating system, and you must recover to Host C.

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\vc80 redistrib_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 APHOST2" in the *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle Business Intelligence*.

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.

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*.

Because instance1 on Host A is no longer available, you must modify its count of BI Servers, Presentation Servers, and JavaHosts to be 0.

Then, instance2 on Host B automatically becomes the primary instance. The new instance on Host C, instance3, becomes the secondary instance.

8. The Oracle BI Cluster Controllers and Oracle BI Scheduler are singleton components that operate in active/passive mode. Configure a secondary instance of these components so that they are distributed for high availability, as described

- in "Configuring Secondary Instances of Singleton System Components" in the *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle Business Intelligence*.
9. Set the listen address of the `bi_servern` 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_servern` 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 HTTP Host and Port" 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/ClusterConfig.xml
```

For example, change the instance number in the `NodeId` element and change the host name or IP address in the `HostNameOrIP` element;

```
<Node>
  <NodeType>Server</NodeType>
  <!--This Configuration setting is managed by Oracle Business
Intelligence Enterprise Manager-->
  <MasterServer>>true</MasterServer>
  <!--This Configuration setting is managed by Oracle Business
Intelligence Enterprise Manager-->
  <NodeId>instance2:coreapplication_obis1</NodeId>
  <!--This Configuration setting is managed by Oracle Business
Intelligence Enterprise Manager-->
<HostNameOrIP>host1.example.com</HostNameOrIP>
  <!--This Configuration setting is managed by Oracle Business
Intelligence Enterprise Manager-->
  <ServicePort>9703</ServicePort>
```

```

        <!--This Configuration setting is managed by Oracle
Business
Intelligence Enterprise Manager--><MonitorPort>9701</MonitorPort>
</Node>

```

- 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](#).

18.3.4.9.4 Importing Oracle BI Enterprise Edition Registry Entries On Windows, you must import the Oracle BI Enterprise Edition 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 will 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 will have missed a period of learning, but their intelligence will be unaffected.

18.3.4.12 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.

18.3.4.13 Recovering Oracle Enterprise Content Management Suite to a Different Host

The following sections describe how to recover Oracle Universal Content Management and Oracle Universal Records Management to a different host:

- [Recovering Oracle Universal Content Management to a Different Host](#)
- [Recovering Oracle Universal Records Management After Loss of Host](#)

18.3.4.13.1 Recovering Oracle Universal Content Management to a Different Host To recover Oracle Universal Content Management 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_092010.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`.

18.3.4.13.2 Recovering Oracle Universal Records Management After Loss of Host Because Oracle Universal Records Management depends on Oracle Universal Content Management and has no additional backup and recovery artifacts, see the recovery procedure for Oracle Universal Content Management in [Section 18.3.4.13.1](#).

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](#)
- [Modify the mod_wl_ohs.conf File](#)
- [Creating a New Machine for Certain 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
- Oracle 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 Modify 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 an Listen address,) you must create a new machine with the new host name before you start the Administration Server:

- Oracle Access Manager
- Oracle Adaptive Access Manager
- Oracle Identity Manager
- Oracle Identity Navigator

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('DomainHome')
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/sampldomain> updateDomain()
wls:/offline/sampldomain> exit()
```

- For the Administration Server, set the machine with the new host name, using the following WLST command, in offline mode:

```
wls:/offline> readDomain('DomainHome')
wls:/offline/sampldomain> cd ('/Machine/newhostname')
wls:/offline/sampldomain> machine = cmo
wls:/offline/sampldomain> cd ('/Server/AdminServer')
wls:/offline/sampldomain> set('Machine', machine)
wls:/offline/sampldomain> updateDomain()
wls:/offline/sampldomain> exit()
```

18.3.5.6 Reassociate 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 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/boa/beahomelist
(Windows) C:\boa\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 recover the following Windows Registry key.

```
HKEY_LOCAL_MACHINE\Software\Oracle
```

In addition, when you recover system components, such as Oracle Web Cache, you must recover the following Windows Registry key:

```
HKEY_LOCAL_MACHINE\System\CurrentControlSet\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*, which is available at:

<http://www.oracle.com/technology/documentation/database.html>

Part VIII

Advanced Administration: Expanding Your Environment

This part describes how to expand your Oracle Fusion Middleware environment.

It contains the following chapters:

- [Chapter 19, "Scaling Your Environment"](#)
- [Chapter 20, "Cloning Oracle Fusion Middleware"](#)
- [Chapter 21, "Moving from a Test to a Production Environment"](#)

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, cloning existing Middleware homes or existing Oracle Fusion Middleware components such as Oracle SOA Suite or Oracle HTTP Server, as described by the following topics:

- [Overview of Scaling Your Environment](#)
- [Extending a Domain to Support Additional Components](#)
- [Adding Additional Managed Servers to a Domain](#)
- [Creating Clusters](#)
- [Cloning a Middleware Home or Component](#)

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, 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 component. Extend with Oracle WebCenter 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:

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, as described in the *Oracle Fusion Middleware Installation Guide for Oracle WebCenter*.
3. From an Oracle home that was installed for the component you want to add (for example, for Oracle WebCenter), 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.2.1](#) to use a different version of Spring than that which is supplied with JRF

19.4 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

19.5 Cloning a Middleware Home or Component

You can clone a Middleware home or many Oracle Fusion Middleware components. **Cloning** is the process of copying an existing entity to a different location while preserving its configuration. Some situations in which cloning Oracle Fusion Middleware is useful are:

- Creating a Middleware home that is a copy of a production, test, or development environment. Cloning enables 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 how to clone a Middleware home or component, see [Chapter 20](#).

Cloning Oracle Fusion Middleware

You can clone a Middleware home and certain Oracle Fusion Middleware components, such as Oracle SOA Suite, Oracle HTTP Server, Oracle Internet Directory and Oracle Virtual Directory.

This chapter includes the following topics:

- [Introduction to Cloning](#)
- [What You Can Clone](#)
- [Understanding the Cloning Process](#)
- [Cloning Syntax](#)
- [Cloning Oracle Fusion Middleware Entities](#)
- [Recovering from Cloning Errors](#)
- [Considerations and Limitations for Cloning](#)

20.1 Introduction to Cloning

Cloning is the process of copying an existing entity to a different location while preserving its state. Some situations in which cloning Oracle Fusion Middleware is useful are:

- Creating a Middleware home that is a copy of a production, test, or development environment. Cloning enables you to 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.
- Preparing a "gold" image of a patched Middleware home and deploying it to many hosts.

The cloned entity behaves the same as the source entity. For example, a cloned Middleware home can be deinstalled or patched using the installer. It can also be used as the source for another cloning operation.

20.2 What You Can Clone

You can clone the following, on the same host or a different host. The clone must be on the same operating system as the source.

- **Middleware home:** You can clone the Middleware home, the Oracle WebLogic Server home, and all of the Oracle homes within the Middleware home. (You can clone a Middleware home that contains no Oracle homes, but you need to have the

cloning jar file and cloning scripts that are compatible with the version of the Middleware home you want to clone.)

You can apply the clone of the Middleware home to the same host or a different host.

- **Java components:** You can clone 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, on the same host or a different host.
- **System components:** You can clone the configuration of a domain containing system components, such as Oracle HTTP Server and Oracle Internet Directory, to the same or a different Oracle instance, to the same or a different Middleware home, on the same host or a different host.

See [Section 20.7](#) for details of considerations and limitations affecting specific components.

20.3 Understanding the Cloning Process

When you clone an entity of Oracle Fusion Middleware, the cloning process takes a snapshot of the information required for cloning. The following topics describe the cloning process:

- [Understanding Cloning a Middleware Home](#)
- [Understanding Cloning Components](#)

20.3.1 Understanding Cloning a Middleware Home

When you clone a Middleware home, you create an archive of the source Middleware home and use the archive to create the cloned Middleware home:

1. At the source, you run the `copyBinary` script, specifying the Middleware home that you want to clone. The script prepares the source for cloning 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 all of the Oracle homes and Oracle WebLogic Server home 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 WebLogic Server home with the Middleware home.

The clone program then restores the file permissions and relinks any files if that is necessary.

See [Section 20.5.1](#) for detailed information about these steps.

20.3.2 Understanding Cloning Components

You can clone certain Oracle Fusion Middleware components, as described in [Section 20.2](#). You create an archive of the source component's configuration and use the archive to create the cloned component.

For Java components, such as Oracle SOA Suite, you use the `copyConfig`, `extractMovePlan`, and `pasteConfig` scripts to clone the configuration, including the domain, the Administration Server, and the Managed Servers.

For some system components, such as Oracle HTTP Server, you use the `copyConfig`, `extractMovePlan`, and `pasteConfig` scripts to clone the configuration, including the Oracle instance.

Note: When you clone 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.)

To clone components, you take the following steps:

1. You clone the Middleware home, as described in [Section 20.3.1](#).
2. At the source, make sure that the Administration Server and all Managed Servers are started.
3. At the source, you run the `copyConfig` script, specifying the source component that you want to clone. The script creates a configuration archive file that contains a snapshot of the configuration of an Oracle WebLogic Server domain or system component instance.
4. You extract a move plan from the source using the `extractMovePlan` script. A **move plan** contains configuration settings of the source environment. You can edit the move plan, specifying properties for the target environment.
5. At the target, you run the `pasteConfig` script, specifying the destination for the component and the move plan. 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.

See [Section 20.5.2](#) and [Section 20.5.3](#) for detailed information about these steps.

20.4 Cloning Syntax

Cloning uses the following jar file to execute the scripts necessary to clone binary and configuration files:

```
(UNIX) ORACLE_COMMON_HOME/jlib/cloningclient.jar
(Windows) ORACLE_COMMON_HOME\jlib\cloningclient.jar
```

You use the following scripts to clone a Middleware home or component:

- To copy the binary files of the source Middleware home:

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

See [Section 20.4.1.1](#) for the syntax of the script.

- To apply the copied Middleware home to the target:

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

See [Section 20.4.1.2](#) for the syntax of the script.

- To copy the source component configuration:

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

See [Section 20.4.1.3](#) and [Section 20.4.1.4](#) for the syntax of the script.

- To extract a move plan from the source component:

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

See [Section 20.4.1.5](#) for the syntax of the script.

- To apply the copied component configuration to the target:

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

See [Section 20.4.1.6](#) and [Section 20.4.1.7](#) for the syntax of the script.

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: If you are applying the clone 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.
-
-

20.4.1 Cloning Scripts

The following topics describe the syntax of the cloning scripts. The options are described in the tables that follow the syntax.

- [copyBinary Script](#)
- [pasteBinary Script](#)
- [copyConfig Script for Java Components](#)
- [copyConfig Script for System Components](#)
- [extractMovePlan Script](#)
- [pasteConfig Script for Java Components](#)
- [pasteConfig Script for System Components](#)

Note:

- All cloning 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
```

20.4.1.1 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 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/jrocket_160_20_D1.1.0-18
              -archiveLoc /tmp/mw_clone.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 32 bit or 64 bit. The operation does not support a mix of 32-bit and 64-bit Oracle homes.

When you execute the command, 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 20-1 describes the options for the `copyBinary` script.

Table 20–1 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: <pre>setenv T2P_JAVA_OPTIONS "-d64 -Djava.io.tmpdir=/home/t2p/temp"</pre>	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
-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 present in the inventory. 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 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

20.4.1.2 pasteBinary Script

Applies the clone to the target destination, by pasting the binary files of the source Middleware home. You can apply the clone to the same host or a different host. 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 clone to the directory /scratch/oracle/MW_Home_clone, on Linux:

```
pasteBinary.sh -javaHome /scratch/Oracle/Middleware1/jrockit_160_20_D1.1.0-18
               -archiveLoc /tmp/mw_clone.jar
               -targetMWHomeLoc /scratch/oracle/MW_Home_clone
```

Table 20–2 describes the options for the pasteBinary script.

Table 20–2 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. Ensure that the archive location is not within the Middleware home structure. 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, but that name is now deprecated. In previous releases, the shortcut was -tl, but that shortcut is 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 20–2 (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 present in the inventory. 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 clone operation operates silently. That is, it does not prompt for confirmation. The default is that the operation prompts for confirmation. 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

20.4.1.3 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 persist their configuration information in different data stores, such as a file system, Oracle Metadata Service (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 up and running when you run the script.

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}]
           [-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/p.txt
-silent true

```

Note that the Administration Server and Managed Servers must be started when you run the copyConfig script.

Table 20–3 describes the options for the copyConfig script for Java components.

Table 20–3 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.	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, but that name is now deprecated. In previous releases, the shortcut was -domainport, but that shortcut is 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
-domainAdminPassword	-dap	The absolute path of a secure file containing the password for the administrative user for the domain on the target 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

Table 20–3 (Cont.) Options for the copyConfig Script for Java Components

Options	Shortcut	Description	Mandatory or Optional
-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 clones 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
-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 specify that it does not prompt for confirmation, specify this option with the value of true.	Optional

20.4.1.4 copyConfig Script for System Components

Creates a configuration archive that contains the snapshot of the configuration of an Oracle instance. The underlying components of the Oracle instance, such as Oracle HTTP Server or Oracle Internet Directory, persist their configuration information in different data stores, such as a file system, Oracle Metadata Service (MDS), LDAP, or a database.

You must run the copyConfig script for Oracle instance in the source environment. A configuration archive is created for each Oracle instance.

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
```

Note that the Administration Server and Managed Servers must be started when you run the copyConfig script.

[Table 20–4](#) describes the options for the copyConfig script for system components.

Table 20–4 Options for the copyConfig Script for 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: <pre>setenv T2P_JAVA_OPTIONS "-d64 -Djava.io.tmpdir=/home/t2p/temp"</pre>	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
-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 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 cloned. For example, if your Oracle Internet Directory component is named oid1, specify oid1.	Mandatory
-sourceInstanceHomeLoc	-sih	The absolute path of the Oracle instance for the source component.	Mandatory

20.4.1.5 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 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 cloning, 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  
/scratch/Oracle/t2p_plans/composites/configplan1.xml  
/scratch/Oracle/t2p_plans/composites/configplan2.xml
```

```

/scratch/Oracle/t2p_plans/adapters
/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 20–5 describes the options for the `extractMovePlan` script:

Table 20–5 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: <pre>setenv T2P_JAVA_OPTIONS "-d64 -Djava.io.tmpdir=/home/t2p/temp"</pre>	Mandatory
-archiveLoc	-al	The absolute path of the archive location. Use this option to specify the location of the archive file created by the <code>copyConfig</code> 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 20.5.4](#).

20.4.1.6 `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 Oracle WebLogic Server domain and the modified move plan. The `pasteConfig` script recreates 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 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 clone 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_cl.jar
-targetDomainLoc /scratch/oracle/MW_home1/user/projects/domains/dom_cl
-targetMWHomeLoc /scratch/oracle/MW_home1
-movePlanLoc /scratch/oracle/java_ee/move_plan.xml
-domainAdminPassword /scratch/pwd_dir/pass.txt
-logDirLoc /tmp/log

```

Table 20–7 describes the options for the pasteConfig script for Java components.

Table 20–6 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 cloned.	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.
-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 specify that it does not prompt for confirmation, specify this option with the value of true.	Optional

20.4.1.7 pasteConfig Script for System Components

Applies the copied configurations from the source environment into target environment. Inputs for the script include the location of the configuration archive created with the copyConfig script for the Oracle instance and the modified move plan. The pasteConfig script iterates and recreates the configuration information for the Oracle instance 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
            -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 clone to the Oracle instance im_2 and to name the cloned 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
```

Table 20-7 describes the options for the pasteConfig script for system components.

Table 20–7 Options for the pasteConfig Script for 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: <pre>setenv T2P_JAVA_OPTIONS "-d64 -Djava.io.tmpdir=/home/t2p/temp"</pre>	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. In previous releases, this option was named archiveLocation, but that name is now deprecated.	Mandatory
-movePlanLoc	-mpl	The absolute path of the move plan extracted from the source.	Mandatory
-targetComponentName	-tcn	The name of the target component to be cloned. The name must be unique in the instance.	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. 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.
-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 cloning.	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 specify that it not prompt for confirmation, specify this option with the value of true.	Optional
Domain-Detail Options			
-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.

Table 20–7 (Cont.) Options for the pasteConfig Script for System Components

Options	Shortcut	Description	Mandatory or Optional
-domainPortNum	-dpn	<p>The port number of the domain.</p> <p>Use this option if you want to register the component with the domain.</p> <p>The domain port number is listed in the following file as the adminPort:</p> <p><i>ORACLE_</i> <i>INSTANCE/config/OPMN/opmn/instance.properties</i></p> <p>For example: adminPort=7001</p> <p>In previous releases, this option was named domainPortNo, but that name is now deprecated.</p> <p>In previous releases, the shortcut was -domainport, but that shortcut is now deprecated.</p>	Optional, if you do not want to register the component with the domain.
-domainAdminUserName	-dau	<p>The name of the administrative user for the domain.</p> <p>Use this option if you want to register the component with the domain.</p> <p>In previous releases, the shortcut was -domainuser, but that shortcut is now deprecated.</p>	Optional, if you do not want to register the component with the domain.
-domainAdminPassword	-dap	<p>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.</p> <p>Use this option if you want to register the component with the domain.</p> <p>In previous releases, the shortcut was -domainpass, but that shortcut is now deprecated.</p>	Optional, if you do not want to register the component with the domain.

20.5 Cloning Oracle Fusion Middleware Entities

The general steps for cloning, whether you are cloning Middleware home or a component, are similar. The general steps are described in [Section 20.3](#).

The following topics describe how to clone Oracle Fusion Middleware entities and how to customize move plans for components:

- [Cloning a Middleware Home](#)
- [Cloning Java Components](#)
- [Cloning System Components](#)
- [Customizing Move Plans When Cloning Components](#)

20.5.1 Cloning a Middleware Home

You can clone a Middleware home, which can contain one or more Oracle homes and an Oracle WebLogic Server home. (You can also clone a Middleware home that contains zero Oracle homes, but you need to have the cloning jar file and cloning scripts that are compatible with the version of the Middleware home you want to clone.)

Note:

- The cloning operation archives only those Oracle homes that lie within a Middleware home. It does not clone Oracle homes that are located outside of the Middleware home.
- You can clone only one Middleware home at a time.
- If Oracle WebLogic Server is not located in the Middleware home, the Oracle WebLogic Server home is not archived and cloned, but any Oracle homes in the Middleware home are archived and cloned.
- On Windows, ensure that no Oracle WebLogic Server processes are running in the source Middleware home.
- On UNIX, if the target host does not contain the file oraInst.loc, you must create the file (you must have Super User or root privileges). By default, the clone operation looks for the file in the /etc directory. If you create it in another location, use the -invPtrLoc option to the pasteBinary script to specify the location.

The file must contain the following:

```
inventory_loc=oraInventory_location
inst_group=user_group
```

To clone a Middleware home:

1. On Windows, at the source Middleware home, stop the Administration Server and any Managed Servers running in the Middleware home.
2. 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 clone 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_clone.jar
               -sourceMWHomeLoc /scratch/Oracle/Middleware1
               -invPtrLoc /scratch/oracle/oraInst.loc
```

3. If you are cloning the Middleware home to a different host, copy the archive file to that system.
4. Copy the pasteBinary scripts and the cloningclient.jar file to the target system and ensure that they have execute permission. See [Section 20.4](#) 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 5.

5. At the target, extract the files from the archive using the pasteBinary script.

For example, to apply the clone to the directory /scratch/oracle/MW_Home_clone, use the following command:

```
pasteBinary.sh -javaHome /scratch/Oracle/Middleware1/jrockit_160_20_D1.1.0-18
               -archiveLoc /tmp/mw_clone.jar
```

```
-targetMWHomeLoc /scratch/oracle/MW_Home_clone
```

The Middleware home is extracted to `/scratch/oracle/MW_Home_clone` 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.

20.5.2 Cloning Java Components

You can clone some types of Oracle Fusion Middleware Java components, such as Oracle SOA Suite. When you do, you copy the configuration of the source domain.

When you clone Java components, you customize the properties in the Java components move plan, as described in [Table 20–9](#). If you are using Oracle ADF, you modify the properties shown in [Table 20–10](#).

To clone a Java component:

1. At the source Middleware home, make sure that the Administration Server and all Managed Servers are started.
2. At the source Middleware home, execute the `copyConfig` script to copy the domain.

For example, to clone 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 admin_username
              -domainAdminPassword /scratch/admin/passwd.txt
              -logDirLoc /tmp/logs
```

3. If you are cloning 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. 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 listed in [Table 20–9](#) to reflect the values for the target environment. In addition, you may need to modify additional properties, depending on the component. For example, for Oracle SOA Suite, you modify the properties in [Table 20–11](#). In addition, you may need to edit Oracle B2B channel properties, as described in the text following [Section 20.5.2.1](#).
6. At the target, extract the files from the archive using the `pasteConfig` script. For example, to apply the clone 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
```

For considerations about cloning Oracle SOA Suite, see [Section 20.5.2.1](#).

20.5.2.1 Considerations for Cloning Oracle SOA Suite

You can clone the domain containing Oracle SOA Suite to the same or a different Middleware home, on the same host or a different host.

To clone Oracle SOA Suite, use the procedure described in [Section 20.5.2](#). When you run the `extractMovePlan` script, you modify the move plan properties described in [Table 20–9](#) and [Table 20–11](#).

The Oracle SOA Suite domain can be configured with multiple components. When you clone Oracle SOA Suite, the following components are also cloned along with the SOA Service Infrastructure:

- Oracle BPEL Process Manager
- Oracle B2B
- Oracle Business Activity Monitoring
- Oracle Mediator
- Oracle Adapters
- Oracle Business Rules
- Oracle Web Services Manager Policy Manager
- Oracle User Messaging Service
- Fusion Middleware Control
- Any Java EE application deployed to the Oracle SOA Suite domain
- Any SOA Composites deployed to the Oracle SOA Suite domain

Along with the domain configuration of the preceding Oracle SOA Suite components, the component metadata of the following components is also cloned:

- Oracle Web Services Manager Policy Manager
- Oracle Human Workflow
- Oracle B2B
- Composites and applications that host their metadata in the MDS Repository

When you clone Oracle SOA Suite, you edit the properties in the move plan for Java components and the move plan for Oracle SOA Suite. The move plan properties for Java components are described in [Table 20–9](#). The move plan properties for Oracle SOA Suite are described in [Table 20–11](#).

In addition to the properties listed in [Table 20–9](#) and [Table 20–11](#), the properties for Oracle B2B channels are in the B2B Channels group. However, because B2B channels can be in various forms, there are no standard properties for a channel. When you extract a move plan, that move plan contains the properties for the current channels in the source environment. Edit the properties to specify values for the target environment.

The following example shows a portion of a move plan for B2B channels:

```

<movableComponent>
  <componentType>B2B</componentType>
  <moveDescriptor>
    <configGroup>
      <type>B2B Channels</type>
      <configProperty id="Channel1">
        <configProperty>
          <name>file-param-is_binary</name>
          <value></value>
          <itemMetadata>
            <dataType>STRING</dataType>
            <scope>READ_WRITE</scope>
          </itemMetadata>
        </configProperty>
        <configProperty>
          <name>file-param-marker</name>
          <value></value>
          <itemMetadata>
            <dataType>STRING</dataType>
            <scope>READ_WRITE</scope>
          </itemMetadata>
        </configProperty>
      </configGroup>
    </moveDescriptor>
  </movableComponent>

```

20.5.3 Cloning System Components

You can clone the system components, such as Oracle HTTP Server or Oracle Virtual Directory.

To clone a system component:

1. At the source Middleware home, make sure that the Administration Server and all Managed Servers are started.
2. At the source Middleware home, execute the copyConfig script.

For example, to clone 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

```

3. If you are cloning 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. 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

```

5. Edit the move plan, modifying the properties for the particular component to reflect the values for the target environment:
 - For Oracle HTTP Server, see [Table 20–12](#).
 - For Oracle Internet Directory, see [Table 20–13](#).
 - For Oracle Virtual Directory, see [Table 20–14](#).

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 cloning the Oracle instance to the same domain, use the `-targetInstanceName` and `-targetComponentName` properties in the move plan to specify a different name for the instance and component.

6. At the target, extract the files from the archive using the `pasteConfig` script. For example, to apply the clone to the Oracle instance `webtier_2` and name the cloned 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
```

The following topics provide information specific to the types of system components you can clone:

- [Cloning Oracle HTTP Server](#)
- [Cloning Oracle Internet Directory](#)
- [Cloning Oracle Virtual Directory](#)

20.5.3.1 Cloning Oracle HTTP Server

You can clone Oracle HTTP Server to the same or a different Oracle instance, to the same or a different Middleware home, on the same host or a different host.

You must apply the clone to an Oracle home that contains the binaries for Oracle HTTP Server.

To clone Oracle HTTP Server, use the procedure described in [Section 20.5.3](#).

If Oracle HTTP Server is configured with WebGate, 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
```

Modify the properties of the move plan listed in [Table 20–12](#).

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`.

20.5.3.2 Cloning Oracle Internet Directory

You can clone Oracle Internet Directory to the same or a different Oracle instance, to the same or a different Middleware home, on the same host or a different host.

You must apply the clone to an Oracle home that contains the binaries for Identity Management.

To clone Oracle Internet Directory, use the procedure described in [Section 20.5.3](#).

[Table 20–13](#) describes the properties that you can change for the move plan for Oracle Internet Directory.

Note that under certain conditions, you may see the following errors when you clone Oracle Internet Directory:

```
OID Cloning: Error cleaning replication agreements
OID Cloning: Error deleting replication dn
OID Cloning: Error updating orclreplicaid
```

If you do, take the following steps:

1. Run the following command:

```
ORACLE_HOME/ldap/bin/remtool -pcleanup
```

When prompted, enter the OID host, non-SSL port, and the ODS schema password.

2. 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
```

3. Using the value in obtained in Step 2, 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"
```

4. 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
```

5. Restart Oracle Internet Directory.

20.5.3.3 Cloning Oracle Virtual Directory

You can clone Oracle Virtual Directory to the same or a different Oracle instance, to the same or a different Middleware home, on the same host or a different host.

You must apply the clone to an Oracle home that contains the binaries for Identity Management.

To clone Oracle Virtual Directory, use the procedure described in [Section 20.5.3](#).

[Table 20–14](#) describes the properties that you can change for the move plan for Oracle Virtual Directory.

20.5.4 Customizing Move Plans When Cloning Components

When you clone Oracle Fusion Middleware components, you run the `extractMovePlan` script to create a move plan for the component that you are cloning. The `extractMovePlan` script extracts configuration information from the archive into a move plan. It also extracts any needed configuration plans. Before you apply the clone to the target, you must edit the move plan to reflect the values of the target environment.

The following shows an excerpt of a move plan for Java components:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<movePlan>
  <movableComponent>
    <componentType>J2EEDomain</componentType>
    <moveDescriptor>
      <configProperty>
        <name>Startup Mode</name>
        <value>PRODUCTION</value>
        <itemMetadata>
          <dataType>STRING</dataType>
          <scope>READ_WRITE</scope>
        </itemMetadata>
      </configProperty>
      <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>
          <configProperty>
            <name>Listen Address</name>
            <value>example.com</value>
            <itemMetadata>
              <dataType>STRING</dataType>
              <scope>READ_WRITE</scope>
            </itemMetadata>
          </configProperty>
        </configGroup>
      </moveDescriptor>
    </movableComponent>
  </movePlan>
```

You can modify properties with the scope of `READ_WRITE`. Do not modify the properties with the scope of `READ_ONLY`.

20.5.4.1 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 20-9](#), 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>
    </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>
    </configProperty>
    .
    .
  </configGroup>

```

20.5.4.2 Move Plans for Components

The tables in this section describe the move plan properties you can customize for Oracle Fusion Middleware components.

The properties that you edit differ depending on the type of component. [Table 20–8](#) provides pointers to the appropriate list of properties for each component.

Table 20–8 *Move Plan Properties for Components*

Component	Where to find the list of properties
Java components	Table 20–9
Oracle ADF connections	Table 20–10
Oracle SOA Suite	Table 20–9 and Table 20–11
Oracle HTTP Server	Table 20–12
Oracle Internet Directory	Table 20–13

Table 20–8 (Cont.) Move Plan Properties for Components

Component	Where to find the list of properties
Oracle Virtual Directory	Table 20–14

[Table 20–9](#) describes the properties you can customize for Java components.

Table 20–9 Common Move Plan Properties for Java Components

Properties	Description	Sample Value
Startup Mode	The startup mode of an Oracle WebLogic Server domain. Valid values are: <ul style="list-style-type: none"> ■ 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
Common Java Properties	The following properties are in the SERVER_CONFIG group.	
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
Oracle WebLogic Server Cluster Configuration Properties	The following properties are in the CLUSTER_CONFIG group.	
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
Machine Configuration Properties	The following properties are in the MACHINE_CONFIG group.	
Machine Name	The name of the machine	example.com
Node Manager Listen Address	The Listen address of the machine running the Node Manager	examplehost

Table 20–9 (Cont.) Common Move Plan Properties for Java Components

Properties	Description	Sample Value
Node Manager Listen Port	The port number of the Listen address of the machine running Node Manager	5556
Deployment Plans	The following properties are in the DEPLOYMENT_PLAN_CONFIG group.	
Deployment Plan	The location where an application's deployment plan will be extracted. The location is relative to the location of the move plan.	deployment_plans/helloWorldEar_plan.xml
Authenticators	The following properties are in the AUTHENTICATORS group.	
Host Name	The LDAP server host name	example.com
Port	The LDAP server port number	3060
Principal	The Admin 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/p.txt
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
Data Source Configuration	The following properties are in the DATASOURCE group.	
Driver Class	The driver class of the data source. Refer to "Selecting a JDBC Driver" in the <i>Oracle Fusion Middleware Configuring and Managing JDBC for Oracle WebLogic Server</i> 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/pass_ds.txt

Table 20–9 (Cont.) Common Move Plan Properties for Java Components

Properties	Description	Sample Value
LDAP-Based Policy and Credential Store Configuration	<p>The following properties are in the OPSS_SECURITY group. They are in the configProperty with the ID of LDAP.</p> <p>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 exposed. When you configure the move plan, you can change from a file-based to an LDAP or database-based store.</p> <p>If the source is LDAP-based, only the LDAP properties are exposed. You cannot change it to a different type, but you can change the LDAP endpoints.</p> <p>If the source is database-based, only the database properties are exposed. You cannot change it to a different type, but you can change the database-based endpoints.</p> <p>You can only use one type of store, LDAP-Based or Database-Based. To use one, uncomment the section in the move plan and ensure the other is commented.</p>	
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/pass_ldap.txt
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
Server Type	The type of server. Valid values are OID (Oracle Internet Directory) or OVD (Oracle Virtual Directory).	OID
LDAP Url	The URL of the LDAP connection. It contains the host name and port number of the LDAP store.	ldap://example.com:3060
Database-Based Policy and Credential Store Configuration	<p>The following properties are in the OPSS_SECURITY group. They are in the configProperty with the id of DB.</p> <p>If the source is a database-based store, these properties are exposed in the move plan. (The LDAP-based store is not exposed and you cannot move from a database-based to an LDAP-based store.)</p>	
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/pass_ldap.txt
DataSource Jndi Name	The name of the data source	opssds
Jps Root	The LDAP Server context root	cn=jpsRoot
Domain	The name of the domain	SOA_domain

Table 20–9 (Cont.) Common Move Plan Properties for Java Components

Properties	Description	Sample Value
Driver Class	The driver class of the data source. Refer to "Selecting a JDBC Driver" in the <i>Oracle Fusion Middleware Configuring and Managing JDBC for Oracle WebLogic Server</i> to choose the appropriate class.	oracle.jdbc.OracleDriver
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
RDBMS Security Store	The following properties are in the RDBMS Security Store group.	
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/orcl.us.oracle.com
Driver Class	The driver class of the RDBMS Security Store connection. Refer to "Selecting a JDBC Driver" in the <i>Oracle Fusion Middleware Configuring and Managing JDBC for Oracle WebLogic Server</i> 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/pass_rbms.txt
Resource Adapter Configuration	The following property is in the ADAPTER group.	
Deployment Plan	The path to the deployment plan to be used during cloning 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 20–10](#) describes the properties you can customize if you are using Oracle ADF connections.

Table 20–10 Move Plan Properties for Oracle ADF Connections

Properties	Description	Sample Value
Oracle Application Development Framework: Application		
Port	The port number used for the URL connections	7000
URL	The URL used for the connection	example.com
Oracle Application Development Framework Business Components		

Table 20–10 (Cont.) Move Plan Properties for Oracle ADF Connections

Properties	Description	Sample Value
ServiceEndpointProvider	Business Components service endpoint provider	ADFBC
JndiFactoryInitial	JNDI initial factory class	com.sun.java.jndi.InitialFactory
JndiProviderUrl	The URL of the JNDI provider	t3://example.com:7101
JndiSecurityPrincipal	JNDI security principal name	weblogic
FabricAddress	Fabric address string	http://example.com
WebServiceConnectionName	The Web Service connection name	test
Oracle Enterprise Scheduler		
NotificationServiceURL	The Oracle Enterprise Scheduler notification service URL	http://localhost:8001
BAM Connection Properties		
WEBTIER_SERVER	The Oracle BAM web server host	example.com
USER_NAME	The BAM user name	user
PASSWORD	The password for the BAM user name	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
Oracle Essbase		
SET_PROXY_HOST_DESCRIPTION	The host name for the Oracle Essbase server	example.com
SET_PROXY_PORT_DESCRIPTION	The listening port number of the Oracle Essbase server	1423
SET_USERNAME_DESCRIPTION	The user name	user3
SET_PASSWORD_DESCRIPTION	The password for the user	password
SET_CLUSTER_DESCRIPTION	The name of the cluster of which the Oracle Essbase server is a member	esbCluster
Oracle WebCenter Content Repository		
ServerHost	The host name of the machine where Content Server is running	example.com
ServerPort	The port number on which the Content Server listens	4444
ServerWebUrl	The Web server URL for the Oracle Content Server	http://example.com/cms/idcplg
Oracle WebCenter Announcements and Discussions		

Table 20–10 (Cont.) Move Plan Properties for Oracle ADF Connections

Properties	Description	Sample Value
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 users.	admin
Url	The URL of the Discussions server hosting Discussion forums and Announcements	http://example.com:8890/owc_discussions
Oracle WebCenter External Applications		
Url	The login URL for the external application	https://example.com/config/login?
Oracle WebCenter Instant Messaging and Presence		
BaseConnectionURL	The URL of the sever hosting instant messaging and presence services	http://example.com:8888
ExternalAppId	The external application ID associated with the Presence server connection. If specified, external application credential information is used to authenticate users against the Microsoft Live Communications Server, Microsoft Office Communications Server, or Oracle WebLogic Communications Server. This property is mandatory for Microsoft Live Communications Server and Microsoft Office Communications Server connections.	extApp
Oracle WebCenter Mail Server		
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	Specifies 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 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	Specifies whether the SMTP server is secured. Valid values are true and false. The default is false.	true
Oracle WebCenter Personal Events		
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

Table 20–10 (Cont.) Move Plan Properties for Oracle ADF Connections

Properties	Description	Sample Value
WebServiceURL	The URL of the Web service exposing the event application	http://example.com:80/ExchangeWS/PersonalEventsWebService.asmx
Oracle WebCenter Producers		
ProxyHost	The host name or IP address of the proxy server	example.com
ProxyPort	The port number of the proxy server	80
Oracle WebCenter URL Connection		
Host	The host name of the proxy server to be used for the Web Producers connection	example.com
Port	The port number to be used for the Web Producers connection	80
URL	The URL for the Web Producers connection	http://example.com:port
Oracle Secure Enterprise Search in Oracle WebCenter		
SoapURL	The Web Services URL that Oracle SES exposes to enable search requests	http://example.com:port/search/query/OracleSearch
Oracle WebCenter Worklists		
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		
WsdlUrl	The URL for the WSDL	http://example.com:port/MyWebService1?WSDL
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 20–11 describes the properties you can customize for Oracle SOA Suite.

Table 20–11 Move Plan Properties for Oracle SOA Suite

Properties	Description	Sample Value
SOA Composites Configuration		
Config Plan Location	The following property is in the Composite group: The location of the configuration plan to be used during cloning 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 20–12 describes the properties you can customize for Oracle HTTP Server.

Table 20–12 Move Plan Properties for Oracle HTTP Server

Properties	Description	Sample Value
Listen	The Listen address. It can include the host name and port or just the port.	8888 or orcl3.example.com:8888
User	The Oracle HTTP Server administration user	<i>admin_user</i>
Group	The group for the user	<i>admin_group1</i>
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
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/pass.txt
PlsqlDatabaseConnectionString	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
ORACConnectSN	Specific to the oradav module, the Oracle database to which to connect	<i>db_host:db_port:db_service_name</i>
ORAUser	Specific to the oradav module, the database user (schema) to use when connecting to the service specified by the ORACConnectSN 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/password.txt
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/"

Table 20–12 (Cont.) Move Plan Properties for Oracle HTTP Server

Properties	Description	Sample Value
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/oracle/cgi-bin/"
WebGateInstalldir	The location of the WebGate installation directory, as specified in the webgate.conf file	/scratch/oracle/mw_home/Oracle_OAMWebGate1/webgate/ohs

Table 20–13 describes the properties that you can change for the move plan for Oracle Internet Directory.

Table 20–13 Move Plan Properties for Oracle Internet Directory

Properties	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/pass_oid.txt
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/pass_ods.txt
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/pass_odssm.txt
DB Host Name	The host name 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 20–14 describes the properties that you can change for the move plan for Oracle Virtual Directory.

Table 20–14 Move Plan Properties for Oracle Virtual Directory

Properties	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
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

20.6 Recovering from Cloning 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:

- If the `pasteBinary` script returns an error during cloning 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.
- If the `pasteConfig` script returns an error during cloning 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 recreate 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 recreate it using RCU.
- If the `pasteConfig` script returns an error during cloning 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.

20.7 Considerations and Limitations for Cloning

Note the following important additional considerations about cloning:

- Cloning does 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 cloned home, because cloning proceeds 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 cloned home.

The Oracle home cloned as a part of the Middleware home contains only the binary files.

- When you clone a Middleware home, only the read-only portions of the Middleware home are cloned. Any user configuration files, such as the `user_projects` directory, are excluded from the cloned image. The WebLogic Server domain is not cloned.
- You must ensure that components, such as Oracle WebLogic Server and Oracle Coherence, are installed in the directory structure of the source Middleware home.
- You cannot clone a Middleware Home if its path is a symbolic link.
- If you created symbolic links to files or applications outside the source WebLogic Server or Oracle home, you must re-create the link manually in the cloned home for your applications to work properly.
- 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 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 clone 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 an Oracle Internet Directory component is cloned with the same database credentials as the source component, the name of the cloned OID component should be different than the source component to avoid conflicts in the OID schema.
- If an Oracle Internet Directory component is cloned with different database credentials from the source component, the name of the cloned Oracle Internet Directory component should be the same as the source component to avoid conflicts in the OID schema.
- You cannot clone Oracle Directory Integration Platform and Oracle Directory Service Manager. You must install them on the target.
- If there is not enough space in the temporary directory when you are cloning 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 20.4](#).

Moving from a Test to a Production Environment

This chapter describes how to move Oracle Fusion Middleware from a test environment to a production environment. You can develop and test applications in a test environment, and then eventually roll out the test applications and, optionally, test data to your production environment. You can also use this approach for testing and rolling out upgrades.

This chapter includes the following topics:

- [Overview of Procedures for Moving from a Test to a Production Environment](#)
- [Moving Identity Management Components to a Production Environment](#)
- [Moving Oracle SOA Suite to a Production Environment](#)
- [Moving Oracle WebCenter to a Production Environment](#)
- [Moving the Web Tier to a Production Environment](#)
- [Moving Oracle Portal, Oracle Forms Services, Oracle Reports, and Oracle BI Discoverer Components to a Production Environment](#)
- [Moving Oracle Business Intelligence Components to a Production System](#)
- [Moving Oracle Real-Time Decisions to a Production System](#)
- [Moving Oracle Enterprise Content Management to a Production System](#)
- [Moving Oracle Data Integrator to a Production Environment](#)
- [Considerations in Moving to and from an Oracle RAC Environment](#)

See: [Section 20.6](#) for information about recovering from errors when you use the scripts described in this chapter.

21.1 Overview of Procedures for Moving from a Test to a Production Environment

This chapter describes how to move installations from a test environment to a production environment.

Note: The target environment must have the superuser or administrative user, with the same credentials, as the user at the source environment.

After you complete the movement of the installation, you can modify the user on the target environment.

The general steps are:

1. If your environment uses a database, create a new database or copy the database from the test environment to the production environment.

Note that the database in the production environment must be the same type of database as in the test environment. For example, if the database in the test environment is an Oracle Database, the database in the production environment must be an Oracle Database.

2. Move a copy of the Middleware home from the test environment to the production environment using the `copyBinary` and `pasteBinary` commands, as described in [Section 20.5.1](#).

Note:

- The production environment must be on the same operating system as the test environment. Also, the operating system architecture must be the same in both environments. For example, both environment must be running 32-bit operating systems or 64-bit operating systems.
- The target environment must have the superuser or administrative user as the user at the source environment. The user's password can be different; you specify it on the command line when you use the `pasteBinary` command.

After you complete the movement of the installation, you can modify the user on the target environment.

3. Move a copy of the configuration of components, as described in [Section 20.5.2](#) and [Section 20.5.3](#).

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.)
 - When you move the configuration of components, the script `copyConfig` handles only global data sources defined in each Oracle WebLogic Server domain. For application-level data sources, you must deploy the ADF application configured with the application-level data sources to a server in the target domain, and manually configure the data sources on the target domain.
-
-

4. If an external LDAP was used in the test environment, move security information, such as users and groups, the identity and policy stores, and credentials. (The default security information is moved when you move the configuration of components.)
5. Move other data, such as UMS user messaging preferences, data for Oracle WebCenter applications, or Oracle Web Cache configuration files. Modify any information that is specific to the new environment such as host name or ports.

21.2 Moving Identity Management Components to a Production Environment

The following topics describe how to move Identity Management from a test environment to a production environment:

- [Moving Identity Management to a New Production Environment](#)
- [Moving Identity Management to an Existing Production Environment](#)

In both scenarios, you have performed the following in the test 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 Oracle Identity Federation.
- Created needed schemas using RCU.
- Installed and configured Identity Management, including the following components:
 - Oracle Internet Directory
 - Oracle Virtual Directory
 - Oracle Directory Integration Platform
 - Oracle Identity Federation
 - Oracle Directory Services Manager
 - Oracle Platform Security
 - Oracle Web Services Manager
 - SSL
 - Oracle Identity Manager
 - Oracle Identity Management Navigation
 - Oracle Adaptive Access Manager
 - Oracle Access Manager 11g or Oracle Access Manager 10g
- 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. In addition, you may have made other configuration changes such as adding ACLs, changing schemas, the Listener configuration, server configuration, plug-ins, mappings, auditing, logging, and keystores.

- 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 Oracle Identity Federation, configured various trusted identity providers and service providers.
- For Oracle Access Manager 11g, set up authentication with corresponding WebGates configured in the Web tier of the protected applications. The Oracle 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 Manager*.
- For Oracle Platform Security, created security policies and stored credentials in the Credential Store Framework (CSF). Oracle Platform Security is useful for applications such as ADF, WebCenter, and SOA Composite applications.
- 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 a production environment, you use trusted CA-signed certificates.)

21.2.1 Moving Identity Management to a New Production Environment

In this scenario, you have installed Identity Management components, such as Oracle Internet Directory, Oracle Virtual Directory, and Oracle Directory Integration Platform, in a test environment and you want to move them to a production environment that does not exist.

To move this environment to a production environment, perform the following tasks:

- [Task 1, "Copy the Database to a New Production Environment"](#)
- [Task 2, "Move the Middleware Homes and Domain Configuration to the New Production Environment"](#)
- [Task 3, "Move Oracle Internet Directory to the New Production Environment"](#)
- [Task 4, "Move Oracle Virtual Directory to a New Production Environment"](#)
- [Task 5, "Move Oracle Directory Integration Platform to a New Production Environment"](#)
- [Task 6, "Move Oracle Identity Federation to a New Production Environment"](#)
- [Task 7, "Move Oracle Identity Manager to a New Production Environment"](#)
- [Task 8, "Move Oracle Identity Navigator to a New Production Environment"](#)
- [Task 9, "Move Oracle Access Manager 11g to a New Production Environment"](#)
- [Task 10, "Move Oracle Access Manager 10g to a New Production Environment"](#)
- [Task 11, "Move Oracle Adaptive Access Manager to a New Production Environment"](#)
- [Task 12, "Move Audit Policies to a New Production Environment"](#)
- [Task 13, "Move Oracle Platform Security to a New Production Environment"](#)
- [Task 14, "Move Oracle Web Services Manager to a New Production Environment"](#)

Task 1 Copy the Database to a New Production Environment

Some components, such as Oracle Internet Directory, Oracle Directory Integration Platform (which depends on Oracle Internet Directory), and Oracle Identity Federation, require a database.

You can 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 move an Oracle Database, Release 11g, to the production system:

1. On the production system, 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 test environment, edit the tnsnames.ora file, adding an entry for the database on the production environment.

The following shows an example of the tnsnames.ora file. In the example, testDB is the database on the test environment and prodDB is the database on the production 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 test environment, edit the listener.ora file, adding an entry for the database on the production 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 production 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 test environment. The following command creates the password file:

```
orapwd password=password file=ORACLE_HOME/dbs/orapwproddb
```

5. In the production 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 production environment, set the `ORACLE_SID` environment variable to point to the production 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 test system to the production system, use `RMAN`.

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*, which is available at:

<http://www.oracle.com/technology/documentation/database.html>

See Also: The Oracle10g Database documentation to move a Release 10g Oracle Database

Task 2 Move the Middleware Homes and Domain Configuration to the New Production Environment

You move all Identity Management Middleware homes to the production environment. To move a copy of the Middleware home and the domain configuration from the test environment to the production environment:

1. Move a copy of the Middleware home containing the Identity Management components from the test environment to the production environment using the `copyBinary` and `pasteBinary` scripts. See [Section 20.5.1](#).

The Oracle WebLogic Server home and the Oracle homes in the Middleware home are also moved.

2. Move a copy of the configuration of each domain containing the Identity Management configuration, as described in [Section 20.5.2](#) and [Section 20.5.2.1](#). 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.
 - Moves Oracle Platform Security.
 - 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`.
 - Configures data sources.
 - Configures JMS resources.
 - Starts the Administration Server.

Task 3 Move Oracle Internet Directory to the New Production Environment

To move Oracle Internet Directory to a new production environment:

1. Move the Oracle Internet Directory configuration, as described in [Section 20.5.3.2](#).
2. If you have configured Oracle Internet Directory replication in the test environment, you must reconfigure it again in the production environment after moving. The replication configuration is not moved from the test to the production environment. See "Setting Up Replication" in the *Oracle Fusion Middleware Administrator's Guide for Oracle Internet Directory*.

Task 4 Move Oracle Virtual Directory to a New Production Environment

To move Oracle Virtual Directory to a new production environment:

1. Move the Oracle Virtual Directory configuration, as described in [Section 20.5.3.3](#).

Task 5 Move Oracle Directory Integration Platform to a New Production Environment

To move Oracle Directory Integration Platform to a new production system:

1. Move Oracle Internet Directory, as described in [Task 3, "Move Oracle Internet Directory to the New Production Environment"](#).

Oracle Directory Integration Platform profiles reside in Oracle Internet Directory. If you have correctly moved Oracle Internet Directory to the production system, the profiles are carried over to the production system.
2. If you configured SSL on the test environment, that configuration is not moved to the production environment. You must configure SSL on the production environment. See [Section 6.5.4.3](#).

Task 6 Move Oracle Identity Federation to a New Production Environment

An Oracle Identity Federation setup involves different modules, such as the Credential Store Framework (CSF) for credentials, JPS authorization rules to access CSF and Audit, that are configured, and that you would need to move. Because of that, Oracle recommends that you install a new Oracle Identity Federation server, then configure the new instance, as opposed to be moving settings from the test environment to the production environment.

See Also:

- *Oracle Fusion Middleware Installation Guide for Oracle Identity Management.*
- *Oracle Fusion Middleware Administrator's Guide for Oracle Identity Federation*, in the section titled "Host Connection Properties," for a discussion of the Oracle Identity Federation server hostname.

Task 7 Move Oracle Identity Manager to a New Production Environment

You can use the Oracle Identity Manager Deployment Manager to move most metadata from a test environment to a production environment. The following table lists the entities that you can move using Deployment Manager:

Entities	Deployment Manager Category
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 Adapters	Process
Resource Forms	Resource Form
Data Object Definitions	Data Object Definition
Rules	Rule
Notification Templates	Notification Template
GTC Providers	GTC Provider
Error Codes	Error Code
System Properties	System Property
EmailDef	Email Definition
EventHandler	Event Handlers
PasswordPolicy	Password Policy
GenericConnector	Generic Connector
ITResourceDef	IT Resource Definition
Request Templates	Request Template
Request Datasets	Request Dataset

Entities	Deployment Manager Category
Approval Policies	Approval Policy

To move Oracle Identity Manager to a new production environment:

1. On the test 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 System Administrator's Guide for Oracle Identity Manager* for information about how to export metadata.

The data is exported as .xml files.

2. On the production 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 System Administrator's Guide for Oracle Identity Manager* for information about how to import metadata.

The Deployment Manager does not manage resource bundles, jars and plug-ins, and 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 test environment to the production environment, using any standard file transfer method.
 - b. In the application, change any calls to external systems to point to the systems in the production environment. For example, if the workflow uses an LDAP server in the test environment, change references to point to an LDAP server in the production environment.
 - c. Use JDeveloper to build the sca jar file from the SOA composite.
 - d. Deploy the SOA composite application on the production environment, using the SOA Deployment wizard in Fusion Middleware Control (see [Section 10.5.1](#)) or JDeveloper.
4. In the test environment, export localization resource bundles, as well as the following sets of plug-in code from the test environment:
 - Scheduled Task jars
 - Adapter Java tasks
 - Third-party jars
 - Other plug-in code jars

Take the following steps:

- a. Edit the following script, which exports the entities into a zip file:

```
(UNIX) OIM_ORACLE_HOME/server/bin/exportMetadata.sh
(Windows) OIM_ORACLE_HOME\server\bin\exportMetadata.bat
```

Edit the script to specify the following values:

- CONTEXT: The URL of the application. For example, weblogic.jndi.WLInitialContextFactory.
- EXPORT_LOCATION: The full path to the directory where the zip file is to be created.

- TEMP_LOCATION_TO_EXTRACT: The full path to a directory where the files are to be stored temporarily before they are packaged into a zip file.
 - CONTROL_FILE: An XML file that controls what needs to be exported. You create the file, as described in Step b.
- b.** Create a control file to specify the types of entities to be exported. The following example shows a sample control file that specifies that all custom resource bundles, jar files, and plug-ins be exported:

```
<?xml version='1.0' encoding='UTF-8'?>
<MigrationDetails Operation = "Export">
  <entityDetails>
    <EntityType>CustomResourceBundles</EntityType>
    <FilteringCriteria>
      <Attribute>
        <Name>Resource_Type</Name>
        <Filter>*</Filter>
      </Attribute>
    </FilteringCriteria>
  </entityDetails>
  <entityDetails>
    <EntityType>Jars</EntityType>
    <FilteringCriteria>
      <Attribute>
        <Name>Jar_Type</Name><Filter>*</Filter>
      </Attribute>
    </FilteringCriteria>
  </entityDetails>
  <entityDetails>
    <EntityType>Plugins</EntityType>
    <FilteringCriteria>
      <Attribute>
        <Name>Type</Name><Filter>*</Filter>
      </Attribute>
    </FilteringCriteria>
  </entityDetails>
</MigrationDetails>
```

- c.** Execute the script, specifying the user name, password, and JNDI URL when prompted. (The JNDI URL is the URL to connect to the application. For example, `t3://hostname:port`).

The script creates a zip file named `exportPackage_timestamp.zip`, which is created in the directory `exportPackage_timestamp`.

- 5.** In the production environment, import localization resource bundles, as well as the sets of plug-in code from the test environment.

To import these entities, you use the following script, which exports the entities into a zip file:

```
(UNIX) OIM_ORACLE_HOME/server/bin/importMetadata.sh
(Windows) OIM_ORACLE_HOME\server\bin\importMetadata.bat
```

Take the following steps:

- a.** Edit the following script, which imports the entities from the zip file created by the export operation:

```
(UNIX) OIM_ORACLE_HOME/server/bin/importMetadata.sh
(Windows) OIM_ORACLE_HOME\server\bin\importMetadata.bat
```

Edit the script to specify the following values:

- CONTEXT: The URL of the application. For example, `weblogic.jndi.WLInitialContextFactory`.
 - IMPORT_LOCATION: The full path to the directory where the zip file created by the export operation is located.
 - TEMP_LOCATION_TO_EXTRACT: The full path to a directory where the files in the zip file are to be extracted before they are imported.
- b. Execute the script, specifying the user name, password, and JNDI URL when prompted. (The JNDI URL is the URL to connect to the application. For example, `t3://hostname:port`).

The script imports the data into the production environment.

6. Move any custom reconciliation profiles, as described in "Updating Reconciliation Profiles Manually" in the *Oracle Fusion Middleware System Administrator's Guide for Oracle Identity Manager*.

- a. Use the WLST command `exportMetadata` to export the custom reconciliation profiles from the test 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 production environment.

- c. Use the WLST command `importMetadata` to import the custom reconciliation profiles to the production environment:

```
connect('username','password',JNDI-URL')
importMetadata(application='OIM', server='server_name',
  fromLocation='directory', docs='/**')
```

7. 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 production 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 8 Move Oracle Identity Navigator to a New Production Environment

To move Oracle Identity Navigator to a new production environment:

1. In the test environment, use the WLST command `exportMetadata` to export the Oracle Identity Navigator metadata from the test environment:

```
connect('username','password',JNDI-URL')
exportMetadata(application='oinav', server='server_name',
  toLocation='directory')
```

The format of the JNDI URL is: `t3://admin_server_host:admin_server_port`.

2. In the production environment, use the WLST command `importMetadata` to import the Oracle Identity Navigator metadata to the production environment:

```
connect('username','password',JNDI-URL')
importMetadata(application='oinav', server='server_name',
  fromLocation='directory')
```

3. Restart Administration Server and the Managed Servers.

Task 9 Move Oracle Access Manager 11g to a New Production Environment

Note: The Administration Servers in both the test environment and the production environment must be started.

To replicate the policy configuration information from the test system into the production system:

1. Install and configure Oracle Access Manager, specifying the information for the production environment, as described in the *Oracle Fusion Middleware Installation Guide for Oracle Identity Management*.

2. Set the environment variable JAVA_HOME and add JAVA_HOME to the PATH.

3. Export the policies from the test system, using the following WLST command:

```
exportPolicy(pathTempOAMPolicyFile='path_of_Temp_PolicyFile')
```

4. Copy the policy file to the production environment.

5. Import the policies into the production environment, using the following command:

```
importPolicy(pathTempOAMPolicyFile='path_of_Temp_PolicyFile')
```

To replicate the configuration and the partner information from the test system to the production system, take the following steps:

1. Follow steps 1 through 5 in the preceding procedure.

2. Export the partner information from the test environment, using the following WLST command:

```
exportPartners(pathTempOAMPartnerFile='path_of_Temp_PartnerFile')
```

3. Copy the partner file to the production environment.

4. Import the partner information to the production environment, using the following WLST command:

```
importPartners(pathTempOAMPartnerFile='path_of_Temp_PartnerFile')
```

Task 10 Move Oracle Access Manager 10g to a New Production Environment

To move Oracle Access Manager 10g to a new production environment:

1. Move the Directory Server from the test environment to the production 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 production 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 test environment to the production environment), Oracle Access Manager inherits the policy and configuration data from the LDAP directory.

4. On the production system, 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 production 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 production 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 test system.
 - c. Click **Modify**. Then, add the host name and port for the production 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 production 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 production 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 test 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 test WebPass.
 - c. From the Identity System Console, select the System Configuration tab, then select **Identity Server** and delete the entry for the test Identity Server.
 - d. From the Access System Console, select the Access System Configuration tab, then select **Access Server Configuration**. Delete the entry for the test 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 production 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 11 Move Oracle Adaptive Access Manager to a New Production Environment

To move Oracle Adaptive Access Manager to a new production environment:

1. Install Oracle Adaptive Access Manager on the production environment, using the same installation and post-installation configuration steps that you used in the test environment.
2. Export snapshots from the test 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
3. Import snapshots into the production environment. Use the Oracle Adaptive Access Manager Administration console to import the contents of the zip file saved in step 2. See "System Snapshot Import/Export" in the *Oracle Fusion Middleware Administrator's Guide for Oracle Adaptive Access Manager* for more information.
 4. Manually update the production system 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 test environment and import them into the production 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 production 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 test system to production system. 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 Administration Console. See the *Oracle Fusion Middleware Administrator's Guide for Oracle Adaptive Access Manager*.
 - e. Copy the following from the test environment to the production 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 test environment to the production environment.
5. Validate that the move was successful:
- a. Log in to OAAM Admin console, as described in "OAAM Admin Console and Controls" in the *Oracle Fusion Middleware Administrator's Guide for Oracle Adaptive Access Manager*.
 - b. Navigate to Policies and check that the rules and groups from the test environment exist in the production environment.
 - c. Navigate to the KBA module and check that the challenge questions in the test environment exist in the production environment.
 - d. Test the Web applications that have been configured for Oracle Adaptive Access Manager. The user should be redirected to registration and challenge flow.

Task 12 Move Audit Policies to a New Production Environment

To move audit policies to a new production environment, see the following topics in the *Oracle Fusion Middleware Application Security Guide*:

- "Migrating Audit Policies"
- "Managing Audit Policies"

Task 13 Move Oracle Platform Security to a New Production Environment

To move Oracle Platform Security to a new production environment, you migrate the policy store and credential store:

1. If the policy store on the test environment is not file-based, migrate the policy store, using the WLST command `migrateSecurityStore`, as described in "Migrating Policies with the Command `migrateSecurityStore`" in the *Oracle Fusion Middleware Application Security Guide*.
2. If the credential store on the test environment is not file-based, migrate the credential store, using the WLST command `migrateSecurityStore`, as described in "Migrating Credentials with the Command `migrateSecurityStore`" in the *Oracle Fusion Middleware Application Security Guide*.
3. 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 14 Move Oracle Web Services Manager to a New Production Environment

To move Oracle Web Services Manager to a new production environment:

1. Move Oracle Platform Security to the production system, as described in [Task 13, "Move Oracle Platform Security to a New Production Environment"](#).

Oracle Web Services Manager depends on Oracle Platform Security and Oracle Fusion Middleware Audit Framework. Oracle Web Services Manager uses Oracle Platform Security for the following:

- Credential store: Oracle Web Services Manager stores client policy user name and password credentials and keystore passwords in the credential store.
 - Policy store: Oracle Web Services Manager permission-based authorization policies use Oracle Platform Security policy store to look up permissions.
 - Login modules: Oracle Web Services Manager uses Oracle Platform Security login modules for all of its authentication.
 - Keystore configuration. However, the keystores in the test and production environments are typically different.
2. Migrate audit policies, as described in "Migrating Audit Policies" in the *Oracle Fusion Middleware Application Security Guide*.
 3. Move policies that are not stored in the MDS Repository:

- a. If you have custom-built policies, move those by copying the jar files from the test to the production environment. The jar files are located in the following directory:

```
DOMAIN_HOME/lib
```

- b. For ADF BC and Oracle WebCenter 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.

4. 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 test environment to the production environment:

```
ORACLE_HOME/modules/oracle.wsm.policies_11.1.1/wsm-seed-policies.jar
```

5. Move the keystore from the test environment to the production environment.
 - a. Because private keys differ between test and production environments, you do not need to move them.
 - b. Public keys, intermediate certificates, and root certificates can be moved. Use the Sun Microsystems `java keytool export` and `import` commands to move them.
 - c. After migration, review the certificates to see if they are applicable in the production environment based on the clients invoking the services.
 - d. If the encryption key alias for the production keystore is different than the test environment keystore, then you must update the `rcpt-key-alias` for all the policies that perform message protection in the policy configuration.

From Fusion Middleware Control, select the domain. Then, from the WebLogic Domain menu, choose **Web Services**, then **Policies**. Select the policy and click **Edit**. Update the alias.

See Also: "Managing Horizontal Policy Migration" in the *Oracle Fusion Middleware Security and Administrator's Guide for Web Services*

21.2.2 Moving Identity Management to an Existing Production Environment

In this scenario, you have installed Identity Management components, such as Oracle Internet Directory, Oracle Directory Integration Platform, and Oracle Web Services Manager, in a test environment and you want to move them to a production environment that already exists.

On the existing production system, you have installed and configured the components. You want to move an application from the test environment to the production environment while retaining its security-related configuration. This requires migrating application-specific data from the test Identity Management environment to the production Identity Management environment.

To move Identity Management to an existing production environment, perform the following tasks:

- [Task 1, "Move Oracle Internet Directory to an Existing Production Environment"](#)
- [Task 2, "Move Oracle Access Manager 11g to an Existing Production Environment"](#)
- [Task 3, "Move Oracle Access Manager 10g to an Existing Production Environment"](#)
- [Task 4, "Move Oracle Adaptive Access Manager to an Existing Production Environment"](#)
- [Task 5, "Move Oracle Identity Manager to an Existing Production Environment"](#)
- [Task 6, "Move Oracle Identity Navigator to an Existing Production Environment"](#)
- [Task 7, "Move Oracle Platform Security to an Existing Production Environment"](#)
- [Task 8, "Move Oracle Web Services Manager to an Existing Production Environment"](#)

Task 1 Move Oracle Internet Directory to an Existing Production Environment

To move Oracle Internet Directory to an existing production environment:

1. You may have configured Oracle Platform Security to use the users and groups in the test environment. To move the users and groups from the test environment, take the following steps:
 - a. Identify the Default Subscriber for the test Oracle Internet Directory instance by running the following command from the test 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 test Oracle Internet Directory instance by running the following command from the test 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 > users.ldif
```

- c. Move the users into the production Oracle Internet Directory instance by running the following command from the production 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 test environment is set up as a staging environment to mimic the production environment, Oracle recommends that you set up one-way replication from the production Oracle Internet Directory to the test Oracle Internet Directory to ensure that any users or groups that exist in the production 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 test Oracle Internet Directory synchronized with the production and to replicate any users or groups that are added into production 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 a One-Way, Two-Way, or Multimaster LDAP-Based Replication Agreement by Using the Replication Wizard in Fusion Middleware Control"
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 test environment to the production environment, as described in Step 1. It also assumes that the orclGUIDs of the users at the test Oracle Internet Directory are identical to those in the production Oracle Internet Directory.

Take the following steps:

- a. Identify the Default Subscriber as described in Step 1a.
- b. Retrieve the RADs from the test 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 > rads.ldif
```

- c. Move the RADs into the production 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 Oracle Access Manager 11g to an Existing Production Environment

In this scenario, you move incremental changes that you have made in the test environment to the production environment.

Note: The Administration Servers in both the test environment and the production environment must be started.

To replicate the policy configuration information from the test system into the production system:

1. Set the environment variable JAVA_HOME and add JAVA_HOME to the PATH.
2. Export the policies from the test system, using the following WLST command:

```
exportPolicy(pathTempOAMPolicyFile='path_of_Temp_PolicyFile')
```

3. Copy the policy file to the production environment.
4. Import the policies into the production environment, using the following WLST command:

```
importPolicy(pathTempOAMPolicyFile='path_of_Temp_PolicyFile')
```

Task 3 Move Oracle Access Manager 10g to an Existing Production Environment

To move Oracle Access Manager 10g to an existing production environment:

1. In the production environment, use the Oracle Access Manager OAMCfgTool to create the same policy domain for the application. Ensure that the following specify values for the production 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 test environment, make the same types of changes in the production environment.

Task 4 Move Oracle Adaptive Access Manager to an Existing Production Environment

To move Oracle Adaptive Access Manager to an existing production environment:

1. Export the necessary delta data from the test system to one or more zip files. You can export the following types of items: policies, rule conditions, patterns, configurable actions, transactions, entities, KBA questions, KBA validations, all group types including alert and action groups, and black list and white list groups used in rules. See Step 2 in [Task 11, "Move Oracle Adaptive Access Manager to a New Production Environment"](#) in [Section 21.2.1](#).
2. Import the zip files created in Step 1 in the production system. See Step 3 in [Task 11, "Move Oracle Adaptive Access Manager to a New Production Environment"](#) in [Section 21.2.1](#).
3. Manually update the production system 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 test environment and import them into the production 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 production 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 test system to production system. 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 test environment to the production 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 test environment to the production environment.
 - g. Copy the following from the test environment to the production environment: properties files, resource bundles, VAD images, and end user JSP screens.
4. Validate that the move was successful:
 - a. Login to OAAM Admin console, as described in "OAAM Admin Console and Controls" in the *Oracle Fusion Middleware Administrator's Guide for Oracle Adaptive Access Manager*.
 - b. Navigate to Policies and check the existing group linking and check that newly added rules and groups from the test environment exist in the production environment.
 - c. Navigate to the KBA module and check that newly added challenge questions in the test environment exist in the production environment.
 - d. Test the Web applications that have been configured for Oracle Adaptive Access Manager. The user should be redirected to registration and challenge flow. The behavior should be the same as in the test environment.

Task 5 Move Oracle Identity Manager to an Existing Production Environment

To move Oracle Identity Manager to an existing production environment, follow the steps described in [Section 21.2.1, Task 7, "Move Oracle Identity Manager to a New Production Environment"](#).

Task 6 Move Oracle Identity Navigator to an Existing Production Environment

To move Oracle Identity Navigator to an existing production environment, follow the steps in [Section 21.2.1, Task 8, "Move Oracle Identity Navigator to a New Production Environment"](#).

Task 7 Move Oracle Platform Security to an Existing Production Environment

You must move all of the Oracle Platform Security policy and credential store information from the test environment to an existing production environment:

1. If the policy store on the test environment is not file-based, migrate the policy store, using the WLST command `migrateSecurityStore`, as described in "Migrating Policies with the Command `migrateSecurityStore`" in the *Oracle Fusion Middleware Application Security Guide*.
2. If the credential store on the test environment is not file-based, migrate the credential store, using the WLST command `migrateSecurityStore`, as described in "Migrating Credentials with the Command `migrateSecurityStore`" in the *Oracle Fusion Middleware Application Security Guide*.
3. Users and groups in the production 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 production 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 8 Move Oracle Web Services Manager to an Existing Production Environment

To move Oracle Web Services Manager to an existing production environment:

1. Move policies for SOA Composite applications, WebCenter, or ADF applications, which are stored in the MDS Repository.
To do so using Fusion Middleware Control:
 - a. On the test 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 test 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 production environment.
 - f. On the production 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 test 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**.

- i. 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 production 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 test 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 test environment to the production environment.
- c. In the production 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 test to the production environment. The jar files are located in the following directory:

```
DOMAIN_HOME/lib
```

See Also: "Managing Horizontal Policy Migration" 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 test environment to the production 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 application if they have changed since the application was first deployed in the production environment. For example, policy A was initially configured in the test environment with the BASIC 128 algorithm and attached to the HelloWorld application. The application was deployed to the production environment. Then, on the test environment, you changed policy A to use the Basic 129 algorithm.
4. Move any policy attachments for a JAX-WS application if they have changed since the application was first deployed.

21.3 Moving Oracle SOA Suite to a Production Environment

The following topics describe how to move Oracle SOA Suite from a test environment to a production environment:

- [Moving Oracle SOA Suite to a New Production Environment](#)
- [Moving Oracle SOA Suite to an Existing Production Environment](#)

In both scenarios, you have performed the following in a test environment:

- Installed Oracle WebLogic Server and created the Middleware home.
- Created the required schemas in the test 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 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 a test environment.

Note: The Oracle User Messaging Service (UMS) is used in SOA and BAM scenarios. The functionality and actions in both scenarios are similar, but there are small differences. In particular, for BAM, only the e-mail driver is supported, so the reconfiguration steps for UMS only apply to the e-mail 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 6](#) for details on moving UMS from a test to a production system.

21.3.1 Moving Oracle SOA Suite to a New Production Environment

In this scenario, you have installed Oracle SOA Suite in a test environment as described in [Section 21.3](#) and you want to move it to a production environment, which does not yet exist.

To move this environment to a new production environment, perform the following tasks:

- [Task 1, "Move the Middleware Home and Oracle SOA Suite and Perform the Initial Configuration"](#)

- [Task 2, "Configure Security in the New Production Environment"](#)
- [Task 3, "Move Human Workflow to the New Production Environment"](#)
- [Task 4, "Move Oracle Business Activity Monitoring Data to the New Production Environment"](#)
- [Task 5, "Move Oracle Business Process Management to the New Production Environment"](#)
- [Task 6, "Move UMS-Related Details to the New Production Environment"](#)
- [Task 7, "Enable SSL and Create Custom Keystores"](#)

Task 1 Move the Middleware Home and Oracle SOA Suite and Perform the Initial Configuration

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 the Middleware home and perform the initial configuration on the production system:

1. Create a database and create the required schemas in the production database using RCU. See *Oracle Fusion Middleware Repository Creation Utility User's Guide*.
2. Move a copy of the Middleware home to the production environment using the `copyBinary` and `pasteBinary` scripts, as described in [Section 20.5.1](#).

The Oracle WebLogic Server home and the Oracle homes in the Middleware home are also moved.

3. Move a copy of the domain containing the Oracle SOA Suite configuration, as described in [Section 20.5.2](#) and [Section 20.5.2.1](#). This step moves the configuration, including the domain, Administration Server, and Managed Servers. Moving the configuration also:
 - 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 production 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.

4. Install and configure Identity Management components, such as Oracle Internet Directory.

For information about installing Identity Management components, see the *Oracle Fusion Middleware Installation Guide for Oracle SOA Suite and Oracle Business Process Management Suite*.

For information about configuring users and groups in Oracle Internet Directory, see "Configuring Users and Groups in the Oracle Internet Directory and Oracle Virtual Directory Authentication Providers" in the *Oracle Fusion Middleware Securing Oracle WebLogic Server*.

5. Create any custom schemas used by your applications. For example, if your application uses a custom schema in the test environment, create the schema in the production environment.
6. 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 production 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 2 Configure Security in the New Production Environment

You must configure security. The steps you take depends on the configuration of your environment and application. The following steps assume that you are using Oracle Internet Directory, JKS certificates, Oracle Web Services Manager, and Oracle Platform Security:

1. If necessary, move users and groups to the production environment. For example, if you are using the Oracle Business Activity Monitoring or Human Workflow demo, move those users:

- a. Export the users and groups from LDAP identity store on the test environment, using the `ldapsearch` command. This produces an `ldif` file that you later import into the LDAP identity store in the production 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"
```

- b. Import the `ldif` file that you exported from the test environment into the production 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
```

2. Export any JKS certificates for B2B endpoints from the test environment to the production environment. Then, import them to the production environment. For information about exporting and importing JKS certificates, see [Section 8.3.3](#).
3. If the security policies are stored in an external LDAP or database-based store, import the security policies, for example those that are related to the Human Workflow application roles, from the test environment to the production

environment, as described in [Task 7, "Move Oracle Platform Security to an Existing Production Environment"](#) in [Section 21.2.2](#).

4. If the credential policies are stored in an external LDAP or database-based store, import the credential store information from the test environment to the production environment, as described in [Task 7, "Move Oracle Platform Security to an Existing Production Environment"](#) in [Section 21.2.2](#).

Task 3 Move Human Workflow to the New Production Environment

When you moved a copy of the domain from the test environment to the production environment, the scripts moved the following Human Workflow entities:

- Attribute labels
- Flex field mappings
- Approval groups
- Standard views

The scripts do not move the following:

- User views
- Rules

To move Human Workflow user views and roles to a new production environment:

1. Move Human Workflow user metadata, such as user views or vacation rules, from the test environment to the production environment, using the Data Migrator. The Data Migrator is available as an ant target that can be executed in the command line. It calls a properties file, `migration.properties`, that you create specifying the input parameters for the migration of data.

The `migration.properties` file contains the following input parameters:

```
operationType = {EXPORT | IMPORT}
objectType = {VIEW | RULE | APPROVAL_GROUP | TASK_PAYLOAD_FLEX_FIELD_MAPPING}
name = name of VIEW or APPROVAL_GROUP or TASK_PAYLOAD_FLEX_FIELD_MAPPING
user = username of VIEW or RULE
group = groupname for RULE
grantPermission = {true | false}
migrateAttributeLabel = {true | false}
override = {true | false}
skip = {true | false}
migrateToActiveVersion = {true | false}
```

You use the following script:

```
ORACLE_HOME/bin/ant-t2p-worklist.xml
```

The command has the following format:

```
ant -f ant-t2p-worklist.xml
  -Dbea.home=BEA_HOME
  -Dsoa.home=SOA_HOME
  -Dmigration.properties.file=MIGRATION_PROPERTY_FILE_PATH
  -Dsoa.hostname=SOA_HOSTNAME
  -Dsoa.rmi.port=SOA_RMI_PORT
  -Dsoa.admin.user=SOA_ADMIN_USER
  -Dsoa.admin.password=SOA_ADMIN_PASSWORD
  -Drealm=REALM
  -Dmigration.file=MIGRATION_FILE
  -Dmap.file=MAP_FILE
```

For additional information about the migration utility, see "Using the User Metadata Migration Utility" in the *Oracle Fusion Middleware Modeling and Implementation Guide for Oracle Business Process Management*.

Take the following steps:

- a. 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.
- b. Create a migration.properties file to export user metadata for the worklist application (for example rules, user views, vacation rules) from the test environment. You can create the migration.properties file in any location. Note the following:
 - You can only export one type of data at a time.
 - When you are exporting data for a particular user or group, you must migrate them in separate operations.

For example, to export all rules for a given user, the migration.properties file would contain the following:

```
operationType = EXPORT
objectType = RULE
name = ALL
user = username
group =
grantPermission = true
migrateAttributeLabel = false
override = true
skip = true
migrateToActiveVersion = false
```

Note that the parameter group is left blank when you export rules for a given user.

To export all rules for a given group, the migration.properties file would contain the following:

```
operationType = EXPORT
objectType = RULE
name = ALL
user =
group = LoanAgentGroup
grantPermission = true
migrateAttributeLabel = false
override = true
skip = true
migrateToActiveVersion = false
```

Note that the parameter user is left blank when you export rules for a given group.

- c. Export the data. The following example shows how to invoke the command and specify the parameters:

```
ant -f ant-t2p-worklist.xml
-Dbea.home=/scratch/oracle/MW_HOME
-Dsoa.home=/scratch/oracle/MW_HOME/AS11gR1SOA
-Dmigration.properties.file=migration.properties
-Dsoa.hostname=hostname -Dsoa.rmi.port=7001
```



```
-Dsoa.admin.user=weblogic
-Dsoa.admin.password=password
-Drealm=jazn.com
-Dmigration.file=/tmp/export_all_userRules.xml
-Dmap.file=/tmp/export_all_userRules_mapper.xml
```

- d. Ensure that the application is deployed to the production system.
- e. Create the migration.properties file to import user metadata for the worklist application to the production environment. Note the following:
 - You can only import one type of data at a time.
 - When you are importing data for a particular user or group, you must import them in separate operations.

For example, to import all rules for a given user, the migration.properties file would contain the following:

```
operationType = IMPORT
objectType = RULE
name = ALL
user = username
group =
grantPermission = true
migrateAttributeLabel = false
override = true
skip = true
migrateToActiveVersion = false
```

Note that the parameter `group` is left blank when you import rules for a given user.

To import all rules for a given group, the migration.properties file would contain the following:

```
operationType = IMPORT
objectType = RULE
name = ALL
user =
group = LoanAgentGroup
grantPermission = true
migrateAttributeLabel = false
override = true
skip = true
migrateToActiveVersion = false
```

Note that the parameter `user` is left blank when you import rules for a given group.

- f. Import the data to the production environment from the file `export_all_userRules.xml`, which you created in the previous steps. The following example shows how to invoke the command and specify the parameters:

```
ant -f ant-t2p-worklist.xml
-Dbea.home=/scratch/oracle/MW_HOME
-Dsoa.home=/scratch/oracle/MW_HOME/AS11gR1SOA
-Dmigration.properties.file=migration.properties
-Dsoa.hostname=hostname
-Dsoa.rmi.port=7001
-Dsoa.admin.user=weblogic
-Dsoa.admin.password=password
-Drealm=jazn.com
```

```
-Dmigration.file=/tmp/export_all_userRules.xml
-Dmap.file=/tmp/export_all_userRules_mapper.xml
```

Note that if the data, such as rules and views, are attached to the user, then the user should be an available user in the production SOA server.

2. Deploy J2EE Human Task Forms, as you would deploy any .ear file. See [Section 10.3.1](#) for more information.
3. If necessary, update the workflow notification configuration with production mail server and inbound and outbound e-mail accounts. See "Configuring Oracle User Messaging Service" in the *Oracle Fusion Middleware Administrator's Guide for Oracle SOA Suite and Oracle Business Process Management Suite*.

Task 4 Move Oracle Business Activity Monitoring Data to the New Production Environment

To move Oracle Business Activity Monitoring to the new production environment:

1. On the test environment, 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*, which is available at:

<http://www.oracle.com/technology/documentation/database.html>

The Oracle BAM objects, such as reports, alerts, and data definitions from the test environment are exported.

2. Install and configure Oracle Internet Directory as the LDAP provider for BAM applications on the production environment, as described in [Task 1](#), if you have not already done so.
3. Set up the Oracle Internet Directory Authenticator, if it was not set up in the test environment. (If it was set up in the test environment, moving the configuration moves the configuration to the production environment.)
 - a. From the Oracle WebLogic Server Administration Console, select **Security Realms**, then **myrealm**, then **Providers**.
A default Authenticator is configured for the realm.
 - b. Click **New** to add a new authenticator.
 - c. Enter a name for the provider, such as `OIDAuthenticator` for a provider that authenticates the user to the Oracle Internet Directory.
 - d. For **Type**, select **OracleInternetDirectoryAuthenticator**.
 - e. Click **OK**.
 - f. On the Providers tab, click the newly created `OIDAuthenticator`.

- g. For **Control Flag**, select **Sufficient** to indicate that if a user can be authenticated successfully by this authenticator, then it should accept that authentication and should not continue to invoke any additional authenticators.
 - h. Select the Provider Specific tab.
 - i. Enter the details of the LDAP provider.
 - j. Click **Save**.
 - k. In the Providers tab, reorder the authenticators so that the newly created authenticator is first.
4. Restart the Administration Server and the Managed Server.
 5. Move BAM data and artifacts to the production environment:
 - a. Create the BAM JPS root context by importing the ldif file. The following shows a sample ldif file:


```
dn: cn=jpsroot_bam_test,dc=us,dc=oracle,dc=com
cn: jpsroot_bam_test
objectclass: top
objectclass: orclcontainer
```
 - b. Move the BAM application policy and roles to LDAP using Fusion Middleware Control:
 - From the navigation pane, right-click the domain that contains Oracle Business Activity Monitoring and choose **Security**, then **Security Provider Configuration**.
 - Follow the steps in "Reassociating Domain Stores with Fusion Middleware Control" in the *Oracle Fusion Middleware Application Security Guide*.
 - c. Import the ORACLEBAM database schema that you exported from the test 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.
 - d. Modify the e-mail server configuration on the production environment, as described in "Configuring Oracle User Messaging Service" in the *Oracle Fusion Middleware Administrator's Guide for Oracle SOA Suite and Oracle Business Process Management Suite*.
 - e. Restart the Oracle Business Activity Monitoring Managed Server.

Task 5 Move Oracle Business Process Management to the New Production Environment

To move Oracle Business Process Management to the new production environment, you move Oracle Business Process Management user metadata, such as organizations and dashboards, from the test environment to the production environment, using the 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.

Note that the migration tool does not move any user-specific configuration because users in the test and production 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
```

For Organizations, the following objects are moved to the production environment: Organizational Units, Roles, Calendars, Organization Role, and Extended User Properties.

For Dashboards, data with the BAM_WIDGET data type in the BPMUserApplicationData table is moved to the production environment.

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 Organizations and Dashboard:
 - a. Create a configuration file to export Organizations. (You pass that file to the ant command.)

The following shows a sample configuration file that exports Organizations:

```
<?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://hostname:port</serverURL>
      <adminUserLogin>admin_username</adminUserLogin>
      <adminUserPassword>admin_password</adminUserPassword>
      <realm>jazn.com</realm>
    </serverEndPoint>
  </sourceEndPoint>
  <targetEndPoint>
    <fileEndPoint>
      <migrationFile>/tmp/bpm_organization.xml</migrationFile>
    </fileEndPoint>
  </targetEndPoint>
</operation>EXPORT</operation>
```

```

<object>ORGANIZATION</object>
<objectDetails>
  <login>username</login>
  <password>password</password>
  <identityContext>jazn.com</identityContext>
  <organization/>
</objectDetails>
</testToProductionMigrationConfiguration>

```

In the configuration file, you must specify the values for the test environment in the following elements:

- migrationFile: This element specifies the file that was generated by the export operation.
- serverURL: The SOA server URL
- adminUserLogin
- adminUserPassword
- objectDetails: Update the login and password elements.

b. Export Organizations, using the following command:

```

ant -f ant-t2p-workspace.xml
  -Dbea.home=BEA_HOME
  -Dbpm.home=BPM_HOME
  -Dbpm.t2p.migration.config=ORG_MIGRATION_CONFIG_FILE

```

c. 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://hostname: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 test environment in the following elements:

- serverURL: The SOA server URL
- adminUserLogin
- adminUserPassword
- migrationFile. Note that this element specifies the file that was generated by the export operation.
- objectDetails: Update the login and password elements.
- userApplicationData: Update the ownerID element.

d. 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 Organization and Dashboards:

a. Create a configuration file to import Organizations. (You pass that file to the ant command.)

The following shows a sample configuration file that imports Organizations:

```
<?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_organization.xml</migrationFile>
    </fileEndPoint>
  </sourceEndPoint>
  <targetEndPoint>
    <serverEndPoint>
      <serverURL>t3://hostname:port</serverURL>
      <adminUserLogin>admin_username</adminUserLogin>
      <adminUserPassword>admin_password</adminUserPassword>
      <realm>jazn.com</realm>
    </serverEndPoint>
  </targetEndPoint>
  <operation>IMPORT</operation>
  <object>ORGANIZATION</object>
  <objectDetails>
    <login>username</login>
    <password>password</password>
    <identityContext>jazn.com</identityContext>
    <organization/>
  </objectDetails>
</testToProductionMigrationConfiguration>
```

In the configuration file, you must update the following elements with the values for the production environment:

- migrationFile: This element specifies the file that was generated by the export operation.
- serverURL: The SOA server URL
- adminUserLogin
- adminUserPassword

- objectDetails: Update the login and password elements.

b. Import Organizations, using the following command:

```
ant -f ant-t2p-workspace.xml
    -Dbea.home=BEA_HOME
    -Dbpm.home=BPM_HOME
    -Dbpm.t2p.migration.config=ORG_MIGRATION_CONFIG_FILE
```

c. Create a configuration file to import Dashboards. The format is the same as for Organizations, except that you substitute the following lines:

```
<sourceEndPoint>
  <fileEndPoint>
    <migrationFile>/tmp/bpm_dashboard.xml</migrationFile>
  </fileEndPoint>
</sourceEndPoint>
<targetEndPoint>
  <serverEndPoint>
    <serverURL>t3://hostname: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 production environment:

- serverURL: The SOA server URL
- adminUserLogin
- adminUserPassword
- migrationFile. Note that this element specifies the file that was generated by the export operation.
- objectDetails: Update the login and password elements.
- userApplicationData: Update the ownerID element.

d. 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 6 Move UMS-Related Details to the New Production Environment

To move UMS details to the new production environment:

1. Configure the required UMS drivers in the production environment.

- Use Fusion Middleware Control to configure the User Messaging Service drivers with production driver information.
- Use the WLST command `deployUserMessagingDriver` to deploy multiple drivers similar to the test environment.

Note: To see different options for deploying additional drivers, execute `help('deployUserMessagingDriver')` at the `wls:/offline>` prompt.

- Re-create any custom-created *business terms* in the production environment. This step is essential in order to use the same set of *User Preferences* filter settings in the production environment, and to ensure that filters built with custom business terms are functional.
 - Restart the production environment to apply the changes.
2. Move the User Messaging preferences from the test environment to the production environment:

- a. In the test 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')
```

Note: 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 production environment.
- c. In the production 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')
```

Note: In the 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 test and production 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 7 Enable SSL and Create Custom Keystores

During the pasteConfig operation, SSL is disabled. In addition, custom keystores are not created in the production environment. Take the following steps:

1. Enable SSL, as described in [Section 6.5](#).
2. Create custom keystores, as described in [Section 8.3.3.1](#).

21.3.2 Moving Oracle SOA Suite to an Existing Production Environment

In this scenario, you have a working production environment and want to test changes in your applications or configuration before rolling those changes into the production environment. In the test environment, you have the same environment as described in [Section 21.3](#).

To move Oracle SOA Suite to an existing production system:

- [Task 1, "Move Oracle SOA Suite Changes"](#)
- [Task 2, "Move Oracle B2B Changes"](#)
- [Task 3, "Move Oracle Business Process Management Changes"](#)
- [Task 4, "Move BAM Data"](#)
- [Task 5, "Move Oracle User Messaging Service Data"](#)

Task 1 Move Oracle SOA Suite Changes

Move any changes that you have made to Oracle SOA Suite:

1. If you have added users and groups in the test environment or modified security policies or credentials, follow the steps in [Task 2](#) in [Section 21.3.1](#) to move them to the production environment.
2. If you have modified EJBs or Plain Old Java Objects (POJOs) in the test environment that support the composite references, move them to the production 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 test environment, move them to the production 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 production environment. For more information about configuration plans, see "Moving SOA Composite Applications to and from Development, Test, and Production Environments" in the *Oracle Fusion Middleware Developer's Guide for Oracle SOA Suite*.

Task 2 Move Oracle B2B Changes

If you have made any changes to Oracle B2B in the test environment, move those changes to the production environment.

Note that if you export selective agreements using the `tpanames` parameter, you must import each zip file individually.

1. Move Oracle B2B system configuration parameters by using the Oracle B2B interface to configure the properties. See "Configuring System Parameters" in the *Oracle Fusion Middleware User's Guide for Oracle B2B* for details.
2. Move other configuration properties by using the B2B command line, as described in "Setting Properties of b2b-config" in the *Oracle Fusion Middleware User's Guide for Oracle B2B*.
3. Move the B2B agreements and trading partners to the production environment:
 - a. Export the data from the test 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 production 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*.

4. Configure B2B agreement external endpoints with production locations and credentials, as described in "Configuring Channels" in the *Oracle Fusion Middleware User's Guide for Oracle B2B*.
5. 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*.
6. 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

If you have made any changes to Oracle Business Process Management in the test environment, move them to the production environment.

To move Oracle Business Process Management to the existing production environment, you move Oracle Business Process Management user metadata, such as organizations and dashboards, from the test environment to the production environment, using the 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.

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 Organizations and Dashboards:
 - a. Create a configuration file to export Organizations. (You pass that file to the ant command.)

The following shows a sample configuration file that exports Organizations:

```
<?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://hostname:port</serverURL>
      <adminUserLogin>admin_username</adminUserLogin>
      <adminUserPassword>admin_password</adminUserPassword>
      <realm>jazn.com</realm>
    </serverEndPoint>
  </sourceEndPoint>
  <targetEndPoint>
    <fileEndPoint>
      <migrationFile>/tmp/bpm_organization.xml</migrationFile>
    </fileEndPoint>
  </targetEndPoint>
  <operation>EXPORT</operation>
  <object>ORGANIZATION</object>
  <objectDetails>
    <login>username</login>
    <password>password</password>
    <identityContext>jazn.com</identityContext>
    <organization/>
  </objectDetails>
</testToProductionMigrationConfiguration>
```

In the configuration file, you must specify the values for the test environment in the following elements:

- migrationFile: This element specifies the file that was generated by the export operation.
- serverURL: The SOA server URL
- adminUserLogin
- adminUserPassword
- objectDetails: Update the login and password elements.

- b. Export Organizations, using the following command:

```
ant -f ant-t2p-workspace.xml
  -Dbea.home=BEA_HOME
  -Dbpm.home=BPM_HOME
  -Dbpm.t2p.migration.config=ORG_MIGRATION_CONFIG_FILE
```

c. 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://hostname: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 test environment in the following elements:

- migrationFile: This element specifies the file that was generated by the export operation.
- serverURL: The SOA server URL
- adminUserLogin
- adminUserPassword
- objectDetails: Update the login and password elements.
- userApplicationData: Update the ownerID element.

d. 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 Organization and Dashboards:**a. Create a configuration file to import Organizations. (You pass that file to the ant command.)**

The following shows a sample configuration file that imports Organizations:

```
<?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_organization.xml</migrationFile>
    </fileEndPoint>
  </sourceEndPoint>
  <targetEndPoint>
    <serverEndPoint>
      <serverURL>t3://hostname:port</serverURL>
      <adminUserLogin>admin_username</adminUserLogin>
      <adminUserPassword>admin_password</adminUserPassword>
      <realm>jazn.com</realm>
    </serverEndPoint>
  </targetEndPoint>
  <operation>IMPORT</operation>
  <object>ORGANIZATION</object>
  <objectDetails>
    <login>username</login>
    <password>password</password>
    <identityContext>jazn.com</identityContext>
    <organization/>
  </objectDetails>
</testToProductionMigrationConfiguration>

```

In the configuration file, you must update the following elements with the values for the production environment:

- migrationFile: This element specifies the file that was generated by the export operation.
- serverURL: The SOA server URL
- adminUserLogin
- adminUserPassword
- objectDetails: Update the login and password elements.

b. Import Organizations, using the following command:

```

ant -f ant-t2p-workspace.xml
  -Dbea.home=BEA_HOME
  -Dbpm.home=BPM_HOME
  -Dbpm.t2p.migration.config=ORG_MIGRATION_CONFIG_FILE

```

c. Create a configuration file to import Dashboards. The format is the same as for Organizations, except that you substitute the following lines:

```

<sourceEndPoint>
  <fileEndPoint>
    <migrationFile>/tmp/bpm_dashboard.xml</migrationFile>
  </fileEndPoint>
</sourceEndPoint>
<targetEndPoint>
  <serverEndPoint>
    <serverURL>t3://hostname: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 production environment:

- migrationFile: This element specifies the file that was generated by the export operation.
- serverURL: The SOA server URL
- adminUserLogin
- adminUserPassword
- objectDetails: Update the login and password elements.
- userApplicationData: Update the ownerID element.

d. 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 4 Move BAM Data

Move any BAM data that has changed:

1. Export BAM artifacts from the test 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 test environment, using the ldapsearch command. This produces an ldif file that you later import into the LDAP identity store in the production 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 production 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. If you have not already done so, set up the LDAP security provider and make it the default provider, as described in Step 2 and Step 3 in [Task 4, "Move Oracle Business Activity Monitoring Data to the New Production Environment"](#).
- c. Import the BAM users from the ldif file that you exported from the test environment into the LDAP provider, such as Oracle Internet Directory, on the production 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
```

- d. 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 Domain Stores with Fusion Middleware Control" in the *Oracle Fusion Middleware Application Security Guide*.
- e. 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

Move Oracle User Messaging Service data:

1. Configure the required UMS drivers in the production environment.

Note: While moving Oracle User Messaging Service to an existing production 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 test instance.

- Use Fusion Middleware Control to configure the User Messaging Service drivers with production driver information.
- Use the WLST command `deployUserMessagingDriver` to deploy multiple drivers similar to the test environment.

Note: To see different options for deploying additional drivers, execute `help('deployUserMessagingDriver')` at the `wls:/offline>` prompt.

- Re-create any custom-created business terms in the production environment. This step is essential in order to use the same set of User Preferences filter settings in the production environment, and to ensure that filters built with custom business terms are functional.
 - Restart the production environment to apply the changes.
2. Move the User Messaging preferences from the test environment to the production 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 test environment.
 - Newly created or modified user devices and filters in the test environment must be created or modified using the following URL in the production 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.

21.4 Moving Oracle WebCenter to a Production Environment

The following topics describe how to move Oracle WebCenter from a test environment to a production environment:

- [Moving Oracle WebCenter to a New Production Environment](#)
- [Moving Oracle WebCenter to an Existing Production Environment](#)

In both scenarios, you have performed the following in a test environment:

- Installed Oracle WebLogic Server.
- Installed Oracle WebCenter.
- Created the required schemas in the test database using RCU.
- Installed and configured Oracle SOA Suite.
- Configured Oracle WebCenter using the Configuration Wizard. You created a domain and Managed Servers and configured Oracle WebCenter Spaces, Oracle WebCenter Portlets, Oracle Discussions, Oracle WebCenter Wiki and Blog Server.
- Installed and configured Oracle Universal Content Management.

- Installed Identity Management components, such as Oracle Internet Directory, Oracle Identity Federation, and Oracle Access Manager.
- Configured Group Spaces.
- Configured Oracle WebCenter 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 and deployed custom WebCenter applications.
- Created instance data in the WebCenter Spaces application, including creating a group space based on a Community of Interest template, and provisioned services. In addition, created some roles and assigned new members to the roles.

21.4.1 Moving Oracle WebCenter to a New Production Environment

In this scenario, you have installed Oracle WebCenter in a test environment as described in [Section 21.4](#) and you want to move it to a production environment, which does not yet exist. You move WebCenter Spaces applications and custom WebCenter Framework applications.

To move Oracle WebCenter to a new production environment, perform the following tasks:

- [Task 1, "Install and Configure Oracle WebCenter in the Production Environment"](#)
- [Task 2, "Export WebCenter Spaces Applications and Required Data from the Test Environment"](#)
- [Task 3, "Export Custom WebCenter Framework Applications from the Test Environment"](#)
- [Task 4, "Import WebCenter Spaces Data and Application to the Production Environment"](#)
- [Task 5, "Import Custom WebCenter Framework Applications to the Production Environment"](#)
- [Task 6, "Enable SSL and Create Custom Keystores"](#)

Task 1 Install and Configure Oracle WebCenter in the Production Environment

Install and configure Oracle WebCenter in the production environment:

1. Create the required schemas in the production database using RCU. See *Oracle Fusion Middleware Repository Creation Utility User's Guide*.
2. Move a copy of the Middleware home to the production environment, as described in [Section 20.5.1](#). The Oracle WebLogic Server home and the Oracle homes in the Middleware home are also moved.
3. Because Oracle Universal Content Management requires a Web server, move the Oracle HTTP Server, as described in [Section 21.5.1.1](#).
4. Move copy of the domain containing the Oracle WebCenter configuration, as described in [Section 20.5.2](#). This step moves the configuration, including the domain, Administration Server, and Managed Servers. In addition, it:
 - Associates WebCenter Spaces with an external identity store.

- For WebCenter Spaces and WebCenter Framework, creates authenticators for Identity Management in Oracle WebLogic Server.
- For WebCenter Spaces and WebCenter Framework, reassociates the policy and credential store.
- For WebCenter Spaces, moves application producer data.
- For WebCenter Spaces, moves portlet customization, personalizations, and metadata from the test environment to the production environment.
- Moves the data for the policy and credential store from the test environment to the production environment.
- Moves the custom Oracle WebCenter Framework application metadata from the test environment to the production environment.
- Starts the Administration Server.

Task 2 Export WebCenter Spaces Applications and Required Data from the Test Environment

Export WebCenter Spaces applications and data required for the applications from the test environment:

1. If necessary, export the required data for WebCenter Spaces applications, including the LDAP identity store, the Content Server, and the Discussion Forum.

Typically, the production environment LDAP identity store may already be populated with users and groups. Take the following steps only if you need to export users, groups, and passwords from the test environment.

- a. Export the users, groups, and passwords from LDAP identity store, using the `ldapsearch` command. This produces an `ldif` file that you later import into the LDAP identity store in the production 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=ldap_user" -w test_ldap_passwd -b "cn=users,dc=example,dc=com"
-s subtree "objectclass=*" "*" orclguid -L > my_users.ldif
```

- b. Export Oracle Content Server, executing 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 user;
exit;
```

```
ORACLE_HOME/bin/expdp "sys/password@connect_id as sysdba"
schemas=prefix_OCSERVER directory=directory dumpfile=filename
```

See Also: The chapters on Oracle Data Pump in the *Oracle Database Utilities*, which is available at:

<http://www.oracle.com/technology/documentation/database.html>

- c. Export the Discussion Forum using the Oracle Database export utility (`ORACLE_HOME` is the Oracle home for the Oracle Database):

```
ORACLE_HOME/bin/export "sys/password@connect_id as sysdba"
OWNER=prefix_DISCUSSIONS FILE=/tmp/df.dmp statistics=none
```

Refer to "Exporting and Importing WebCenter Spaces Applications for Data Migration" in the *Oracle Fusion Middleware Administrator's Guide for Oracle WebCenter* for more information.

2. Move Oracle SOA Suite from the test environment to the production environment, as described in [Section 21.3](#).
3. Export Portlet customization, personalizations, and metadata from the test environment, using the following command:

```
exportPortletClientMetadata(appName='app_name', fileName='filename',
exportPersonalizations=1)
```

For detailed syntax, see "exportPortletClientMetadata" in the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

4. Export the WebCenter Spaces application by using the following WLST commands:

```
connect('username', 'password', 't3://hostname:port')
exportWebCenterApplication(appName, fileName,
    exportCustomizations=true, exportSecurity=true, exportData=true)
```

Refer to "Exporting and Importing Custom WebCenter Applications for Data Migration" in the *Oracle Fusion Middleware Administrator's Guide for Oracle WebCenter* for details.

Task 3 Export Custom WebCenter Framework Applications from the Test Environment

To export custom WebCenter Framework applications from the test environment:

1. Export Portlet customizations, personalizations, and metadata from the test environment, using the following command:

```
exportPortletClientMetadata(appName='app_name', fileName='filename',
exportPersonalizations=1)
```

For detailed syntax, see "exportPortletClientMetadata" in the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

2. Export the WebCenter Framework application data from the test database, 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 user;
exit;
```

```
ORACLE_HOME/bin/expdp "sys/password@connect_id as sysdba"
schemas=prefix_WEBCENTER directory=directory dumpfile=filename
```

For information about the CREATE OR REPLACE DIRECTORY command, see "CREATE DIRECTORY" in the *Oracle Database SQL Language Reference*.

For information about the expdp command, refer to the chapters on Oracle Data Pump in *Oracle Database Utilities*, which is available at:

<http://www.oracle.com/technology/documentation/database.html>

3. Export Oracle Content Server, executing 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 user;
exit;
```

```
ORACLE_HOME/bin/expdp "sys/password@connect_id as sysdba"
schemas=prefix_OCSERVER directory=directory dumpfile=filename
```

See Also: The chapters on Oracle Data Pump in the *Oracle Database Utilities*, which is available at:

<http://www.oracle.com/technology/documentation/database.html>

4. Export the Discussion Forum using the Oracle Database export utility (*ORACLE_HOME* is the Oracle home for the Oracle Database):

```
ORACLE_HOME/bin/export "sys/password@connect_id as sysdba"
OWNER=prefix_DISCUSSIONS FILE=/tmp/df.dmp statistics=none
```

5. Migrate the external LDAP-based policy store, using the WLST command `migrateSecurityStore`, as described in "Migrating Policies with the Command `migrateSecurityStore`" in the *Oracle Fusion Middleware Application Security Guide*.
6. Migrate the external LDAP-based credential store, using the WLST command `migrateSecurityStore`, as described in "Migrating Credentials with the Command `migrateSecurityStore`" in the *Oracle Fusion Middleware Application Security Guide*.

Task 4 Import WebCenter Spaces Data and Application to the Production Environment

To import the WebCenter Spaces data and application to the production environment:

1. If necessary, import users, groups, and passwords by importing the ldif file that you exported from the test environment in [Task 2](#) into the production environment.

Typically, the production environment LDAP identity store may already be populated with users and groups. Use the following command only if you need to export users, groups, and passwords from the test environment and want to import them into the production environment. (*ORACLE_HOME* is the Oracle home for Identity Management.)

```
ORACLE_HOME/bin/ldapaddmt -h production_oid_host
-p production_oid_port -D "cn=ldap_user"
-w "production_ldap_passwd" -r -f ldif_filename
```

Refer to "Exporting and Importing Custom WebCenter Applications for Data Migration" in the *Oracle Fusion Middleware Administrator's Guide for Oracle WebCenter* for more information.

2. Import Content Server:
 - a. Import the Oracle Content Server data to the production database, using the file you exported in [Task 2](#). Execute 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 user;
exit;
```

```
ORACLE_HOME/bin/impdp "sys/password@connect_id as sysdba"
  remap_schema=testprefix_OCSEVER:prod_prefix_OCSEVER
  DIRECTORY=directory dumpfile=filename
  TABLE_EXISTS_ACTION=REPLACE
```

- b. Copy the following directories from the test system to the production system. You can use `tar` to compress the files from the test system and restore them on the production system:

```
WebCenter_ORACLE_HOME/ucm/vault
WebCenter_ORACLE_HOME/ucm/weblayout
```

3. Import the Discussion Forum:

- a. Connect to the production database using SQLPlus. (*ORACLE_HOME* is the Oracle home for the production database.)

```
ORACLE_HOME/bin/sqlplus "sys/password as sysdba"
```

- b. Drop the target user:

```
drop user prefix_DISCUSSIONS cascade;
```

- c. Create the target user:

```
create user prefix_DISCUSSIONS identified by password
  default tablespace prefix_IAS_DISCUSSIONS
  temporary tablespace prefix_IAS_TEMP;
```

- d. Grant connect and resource privileges to the user and exit from SQLPlus:

```
grant connect,resource to prefix_DISCUSSIONS;
exit;
```

- e. Import the Discussion Forum data into the production database. You import the file that you exported from the test database in [Task 2](#). (*ORACLE_HOME* is the Oracle home for the production database.)

```
ORACLE_HOME/bin/imp "sys/password@connect_id as sysdba"
  FROMUSER=testprefix_DISCUSSIONS TOUSER=prod_prefix_DISCUSSIONS
  FILE=filename statistics=none
```

4. Import the WebCenter Spaces application by using the following WLST commands:

```
connect('username', 'password', 't3://hostname:port')
importWebCenterApplication(appName='appName', fileName='fileName')
```

Refer to "Exporting and Importing Custom WebCenter Applications for Data Migration" in the *Oracle Fusion Middleware Administrator's Guide for Oracle WebCenter* for details.

5. Import Portlet customization, personalizations, and metadata to the production environment, using the following command:

```
importPortletClientMetadata(appName='app_name', fileName='filename')
```

For detailed syntax, see "importPortletClientMetadata" in the *Oracle Fusion Middleware WebLogic Scripting Tool Command Reference*.

Note: If you are using the embedded Oracle WebLogic Server LDAP identity store, see "Managing Security" in the *Oracle Fusion Middleware Administrator's Guide for Oracle WebCenter*.

Although secure, the embedded LDAP identity store is not a production-class store and should be replaced with an external LDAP-based identity store, such as Oracle Internet Directory, for enterprise production environments.

See Also: "Exporting and Importing WebCenter Portal Applications for Data Migration" in the *Oracle Fusion Middleware Administrator's Guide for Oracle WebCenter* for more information

Task 5 Import Custom WebCenter Framework Applications to the Production Environment

To import custom WebCenter Framework applications to the production environment:

1. Import Portlet customizations and metadata from file you exported in [Task 3](#) to the production environment, using the following WLST command:

```
importPortletClientMetadata(appName='app_name', fileName='filename')
```

2. Import the data from the database, 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 user;
exit;
```

```
ORACLE_HOME/bin/impdp "sys/password@connect_id as sysdba"
  REMAP_SCHEMA=test_prefix_WEBCENTER:prod_prefix_WEBCENTER
  DIRECTORY=directory dumpfile=filename TABLE_EXISTS_ACTION=REPLACE
```

3. Import Content Server, if you have not already done so in [Task 4](#), Step 2.
4. Import the Discussion Forum, if you have not already done so in [Task 4](#), Step 3.
5. If you have not already done so, import the LDAP identity, policy, and credential stores. Import the ldif file that you exported from the test environment in [Task 2](#) into the production 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
```

6. Migrate the external LDAP-based policy store, using the WLST command `migrateSecurityStore`, as described in "Migrating Policies with the Command `migrateSecurityStore`" in the *Oracle Fusion Middleware Application Security Guide*.
7. Migrate the external LDAP-based credential store, using the WLST command `migrateSecurityStore`, as described in "Migrating Credentials with the Command `migrateSecurityStore`" in the *Oracle Fusion Middleware Application Security Guide*.

Task 6 Enable SSL and Create Custom Keystores

During the pasteConfig operation, SSL is disabled. In addition, custom keystores are not created in the production environment. Take the following steps:

1. Enable SSL, as described in [Section 6.5](#).
2. Create custom keystores, as described in [Section 8.3.3.1](#).

21.4.2 Moving Oracle WebCenter to an Existing Production Environment

In this scenario, you have a working production environment with Oracle WebCenter installed and configured and you want to test changes in your applications or configuration before rolling those changes into the production environment. For example, you have modified a WebCenter Spaces application, you want to deploy a newer version of a WebCenter Framework application, or you have modified existing security policies or configuration.

To move the changes to an existing production environment, perform the following tasks:

- [Task 1, "Export Oracle WebCenter Spaces Data from the Test Environment"](#)
- [Task 2, "Import Group Spaces Data to the Production Environment"](#)
- [Task 3, "Export WebCenter Group Space Template from the Test Environment"](#)
- [Task 4, "Import WebCenter Group Space Template from the Test Environment"](#)

Task 1 Export Oracle WebCenter Spaces Data from the Test Environment

To export Oracle WebCenter Spaces data from the test environment:

1. Export Oracle WebCenter Spaces data from Discussion Forum, using the Oracle Database export utility:

```
ORACLE_HOME/bin/export "sys/password@connect_id as sysdba"
  $discussions_schema/$discussions_password file=discussions_forumid.dmp
  log=jive_forumid.log
```

```
TABLES=jiveforum,jiveThread,jivemessage,jiveForumProp,jiveQuestion,jiveAnswer,
jiveGateway
  rows=y STATISTICS=None QUERY="\WHERE forumid \= $forumid\"
```

2. Export the Oracle WebCenter Spaces data from Wiki, using the owc_wiki_export.sql script. The script is located in the following directory:

```
ORACLE_HOME/wikiserver/owc-wiki/WEB-INF/classes
```

Use the following commands:

```
ORACLE_HOME/bin/sqlplus "sys/password as sysdba"
create or replace directory WC_PUMP_DIR as 'path';
grant read,write on DIRECTORY WC_PUMP_DIR to user;
@owc_wiki_export.sql
```

3. Export the Group Space from the test environment:
 - a. Login to WebCenter Spaces with administrative privileges.
 - b. Click the Administration link at the top of the application.
 - c. Click the Group Spaces tab.
 - d. Click the Group Spaces subtab.

- e. Select the group space required by highlighting the row in the table.
- f. From the Change State drop down, select **Offline**.
- g. Click **Save**.
- h. Click **Export** in the toolbar.

Task 2 Import Group Spaces Data to the Production Environment

To import Group Spaces data to the production environment:

1. Import Group Space data into the Discussion Forum data into the production database. You import the file that you exported from the test database in [Task 1](#). (*ORACLE_HOME* is the Oracle home for the production database.)

```
ORACLE_HOME/bin/impdp "sys/password@connect_id as sysdba"
  file=filename log=df_category.log ignore=y STATISTICS=None
  FROMUSER=test_prefix_DISCUSSIONS TOUSER=prod_prefix_DISCUSSIONS
```

```
imp sys/passwd@dbhost file=T2PTEST_category.dmp log=df_category.log ignore=y
  STATISTICS=None FROMUSER=TEST_DISCUSSIONS TOUSER=PROD_DISCUSSIONS
```

```
imp sys/passwd@dbhost file=T2PTEST_forumid.dmp log=df_forumid.log ignore=y
  STATISTICS=None FROMUSER=TEST_DISCUSSIONS TOUSER=PROD_DISCUSSIONS
```

```
imp sys/passwd@dbhost file=T2PTEST_forumid_perm.dmp log=df_forumid_perm.log
  ignore=y STATISTICS=None FROMUSER=TEST_DISCUSSIONS TOUSER=PROD_DISCUSSIONS
```

2. Import Group Space data into Wiki using the `owc_wiki_import.sql` script. Edit the script, adding the following line before the line `DBMS_DATAPUMP.start_job(dp_handle)`:

```
DBMS_DATAPUMP.METADATA_REMAP(dp_handle, 'REMAP_SCHEMA', 'source', 'target');
```

3. Create the directory `WC_PUMP_DIR` in the production environment.
4. Copy the file that you generated when you exported the data from the database in [Task 1](#), to the `WC_PUMP_DIR` in the production environment.
5. Run the script:

```
ORACLE_HOME/bin/sqlplus "sys/password as sysdba"
@owc_wiki_import.sql
```

6. Import the Group Space to the production environment:
 - a. Login to WebCenter Spaces with administrative privileges.
 - b. Click the Administration link at the top of the application.
 - c. Click the Group Spaces tab.
 - d. Click the Group Spaces subtab.
 - e. Select the group space required by highlighting the row in the table.
 - f. From the Change State drop down, select **Offline**.
 - g. Click **Save**.
 - h. Click **Import** in the toolbar and select the exported archive.
7. Import Group Space data from the Content Server, by using WebDAV to drag the folders under the Group Spaces in the test environment to Group Spaces in the production environment.

Task 3 Export WebCenter Group Space Template from the Test Environment

To export the Group Space template from the test environment:

1. Login to WebCenter Spaces with administrative privileges.
2. Click the Administration link at the top of the application.
3. Click the Group Spaces tab.
4. From the Group Spaces tab, select **Templates**.
5. Click **Export** in the toolbar.

Task 4 Import WebCenter Group Space Template from the Test Environment

To import the Group Space template to the production environment:

1. Login to WebCenter Spaces with administrative privileges.
2. Click the Administration link at the top of the application.
3. Click the Group Spaces tab.
4. From the Group Spaces tab, select **Templates**.
5. Click **Import** in the toolbar, and select the exported archive.

21.5 Moving the Web Tier to a Production Environment

In this scenario, you have installed Oracle HTTP Server and Oracle Web Cache in a test environment and you want to move them to a production environment.

The following topics describe how to move the Web tier from a test environment to a production environment:

- [Moving the Web Tier to a New Production Environment](#)
- [Moving the Web Tier to an Existing Production Environment](#)

21.5.1 Moving the Web Tier to a New Production Environment

The following topics describe how to move the Web tier to a new production environment:

- [Moving Oracle HTTP Server to a New Production Environment](#)
- [Moving Oracle Web Cache to a New Production Environment](#)

21.5.1.1 Moving Oracle HTTP Server to a New Production Environment

In this scenario, you have installed Oracle HTTP Server in a test environment and you want to move it to a production environment, which does not yet exist. In the test 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 Oracle Access Manager. In this scenario, the Oracle Access Manager Access Servers are not in the test environment. They reside on a separate production system. However, WebGate is running in the test environment.

To move this environment to a new production environment, perform the following tasks:

- [Task 1, "Move the Oracle HTTP Server and Oracle Instances"](#)
- [Task 2, "Start the Processes"](#)

Task 1 Move the Oracle HTTP Server and Oracle Instances

Note that if Oracle HTTP Server is used by WebGate, you must first move Oracle Access Manager to the production environment, as described in [Section 21.2.1, Task 9](#) or [Task 10](#), depending on your version of Oracle Access Manager.

In the production environment, move the binary files and create the Oracle instance and one or more Oracle HTTP Server component instances:

1. Move a copy of the Middleware home containing Oracle HTTP Server to the production environment, as described in [Section 20.5.1](#). The Oracle homes in the Middleware home are also moved.
2. Move the Oracle HTTP Server configuration, as described in [Section 20.5.3.1](#). This step moves the configuration, including the Oracle instances. In addition, it:
 - Updates the Listen address and the name of the virtual host.
 - Configures SSL, if it was configured in the test environment.
 - Updates the httpd.conf file with new values for the environment and topology directives, such as host name, 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 production environment.
 - Configures SSL for mod_wl_ohs, if SSL is configured for mod_wl_ohs.
 - Configures mod_osso, if it was configured in the test environment.
 - Configures PL/SQL, if it was configured in the test environment.
 - Configures mod_osso, if it was configured in the test environment.
 - Updates audit.config.xml, if any changes were made to it in the test environment.
 - Updates *component-log.xml*, if any changes were made to it in the test environment.
 - Configures WebGate if you are using Oracle Access Manager.

Task 2 Start the Processes

Start the processes in the Oracle instance:

```
ORACLE_INSTANCE/bin/opmnctl stopall
ORACLE_INSTANCE/bin/opmnctl startall
```

21.5.1.2 Moving Oracle Web Cache to a New Production Environment

In this scenario, you have installed Oracle Web Cache in a test environment and you want to move it to a production environment, which does not yet exist. In the test 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.

To move this environment to a new production 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 production environment, move the binary files and create the Oracle instances and Oracle Web Cache instances:

1. Move a copy of the Middleware home containing Oracle Web Cache to the production environment, as described in [Section 20.5.1](#). The Oracle homes in the Middleware home are also moved.
2. Create the Oracle instances and the Oracle Web Cache instances.

To create the Oracle instances and Oracle Web Cache instances using the Oracle Universal Installer, take the following steps:

- a. Run the following script:

```
(UNIX) ORACLE_HOME/common/bin/config.sh
(Windows) ORACLE_HOME\common\bin\config.bat
```

- b. Follow the instructions in the *Oracle Fusion Middleware Installation Guide for Oracle Web Tier*.

To create the instances and components using the command line, take the following steps:

- 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 -wlsServerHome 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 test 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 production environment is different from the password at the test environment:
 - Copy the value of the `PASSWORDHASH` attribute of the `<USER TYPE="INVALIDATION">` element from the `webcache.xml` file for the production 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 production 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 production environment.
 - For each `<CACHE>` element in `webcache.xml`, change the following, substituting the values that correspond to the host where the production 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 production 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 production 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 production system:
- (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 test environment to the corresponding production 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 test environment to the corresponding production 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 test environment to the production 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 test environment for either an SSL-enabled listen address or an OSwallet, or both, export the wallets from the test environment and import them at the production environment. For information about exporting and importing wallets, see [Section 8.4.4](#).

21.5.2 Moving the Web Tier to an Existing Production Environment

In this scenario, you have a working production environment and want to test changes in your applications or configuration before rolling those changes into the production environment.

- For Oracle HTTP Server, see [Section 21.5.2.1](#).
- For Oracle Web Cache, perform [Task 2](#) in [Section 21.5.1.2](#).

21.5.2.1 Moving Oracle HTTP Server to an Existing Production Environment

To move Oracle HTTP Server to an existing production 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 production 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 production

environment. Then, copy `auditconfig.xml` from the test environment to the corresponding production 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 production environment. Then, copy the file, which is located in the following directory, from the test environment to the production environment:

```
ORACLE_INSTANCE/diagnostics/logs/OHS/ohs_component_name
```

21.6 Moving Oracle Portal, Oracle Forms Services, Oracle Reports, and Oracle BI Discoverer Components to a Production Environment

In this scenario, you have installed Oracle Portal, Oracle Forms Services, Oracle Reports, and Oracle Business Intelligence Discoverer, in a test environment and you want to move it to a production environment.

The following topics describe how to move these components from a test environment to a production environment:

- [Moving Oracle Portal, Oracle Forms Services, Oracle Reports, and Oracle Business Intelligence Discoverer to a New Production Environment](#)
- [Moving Oracle Portal, Oracle Forms Services, Oracle Reports, and Oracle Business Intelligence Discoverer to an Existing Production Environment](#)

In both scenarios, you have performed the following in a test environment:

- Installed a database to be used for these components.
- Created schemas needed by the components, using RCU.
- 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 and new templates.
 - Created 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.

21.6.1 Moving Oracle Portal, Oracle Forms Services, Oracle Reports, and Oracle Business Intelligence Discoverer to a New Production Environment

In this scenario, you have installed Oracle Portal, Oracle Forms Services, Oracle Reports, and Oracle Business Intelligence Discoverer in a test environment and you want to move the components to a production environment that does not exist.

Although this section describes how to move all of the components to a production environment, you can choose to move only some of them.

To move this environment to a new production environment, perform the following tasks:

- [Task 1, "Copy the Database to the New Production Environment"](#)
- [Task 2, "Install and Configure the Components"](#)
- [Task 3, "Move Oracle Portal to the New Production Environment"](#)
- [Task 4, "Move Oracle Forms Services to the New Production Environment"](#)
- [Task 5, "Move Oracle Reports to the New Production System"](#)
- [Task 6, "Move Oracle Business Intelligence Discoverer to the New Production Environment"](#)

Task 1 Copy the Database to the New Production Environment

Move the database to the production system by using the Oracle Database RMAN utility.

For detailed steps, see the *Oracle Database Backup and Recovery User's Guide*, which is available at:

<http://www.oracle.com/technology/documentation/database.html>

See [Appendix D](#) for the schemas used by each component.

Task 2 Install and Configure the Components

To install and configure the components:

1. Move a copy of the Middleware home, as described in [Section 20.5.1](#). The Oracle WebLogic Server home and the Oracle homes in the Middleware home are also moved.
2. 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 3 Move Oracle Portal to the New Production Environment

To move Oracle Portal configuration to a new production environment:

1. Create a transport set on the test 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 test environment, as described in "Exporting Data" in the *Oracle Fusion Middleware Administrator's Guide for Oracle Portal*.
3. On the production environment, create a database link to the test 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 first 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, the contents of the transport set must first be moved to the transport set tables on the target system. You do this by acquiring the transport set from the test 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 test environment to the LDAP directory in the production 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 test system. 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 production system, 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 test environment file to the production 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 WebCenter or Oracle Portal is moved from test to production using export and import, portlet customizations are included in transport set. You do not need to take any additional steps.

Task 4 Move Oracle Forms Services to the New Production Environment

To move Oracle Forms Services to a new production environment:

1. Stop the processes running in the Oracle instance and stop the Managed Servers in the production 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, and PLX) from the test environment to the production 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 production environment. Instead, add the location to the default.env file.

3. Move the application-related data from the test environment to a database in the production environment using database migration tools.
4. Create entries in the SQL*Net configuration files to refer to the database in the production environment.
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 test environment to Oracle Internet Directory in the production environment. See Step 3 in [Task 3, "Move Oracle Internet Directory to the New Production Environment"](#) in [Section 21.2.1](#).
6. Copy any customizations in the following files that you want to maintain from the test environment file to the production environment file:

Type of File	Location
Forms application configuration	DOMAIN_HOME/config/fmwconfig/servers/WLS_FORMS/applications/formsapp_11.1.1/config/formsweb.cfg
Forms server configuration	DOMAIN_HOME/config/fmwconfig/servers/WLS_FORMS/applications/formsapp_11.1.1/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

If you modified the Oracle HTTP Server forms.conf file, restart Oracle HTTP Server:

```
ORACLE_INSTANCE/bin/opmnctl restartproc ias-component=ohs_name
```

7. Copy the following files from the test environment to the production 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/fttrace.cfg
Forms applications .ear	ORACLE_HOME/forms/j2ee/formsapp.ear
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/frmxm1sg.sh ORACLE_INSTANCE/bin/frmcmp_batch.sh ORACLE_INSTANCE/bin/frmf2xml.sh ORACLE_INSTANCE/bin/frmxm12f.sh ORACLE_INSTANCE/bin/frmxm1v.sh Windows: ORACLE_HOME\bin\frmbld.bat ORACLE_HOME\bin\frmcmp.bat ORACLE_INSTANCE\bin\frmplsqlconv.bat ORACLE_INSTANCE\bin\frmxm1sg.bat ORACLE_INSTANCE\bin\frmcmp_batch.bat ORACLE_INSTANCE\bin\frmf2xml.bat ORACLE_INSTANCE\bin\frmxm12f.bat ORACLE_INSTANCE\bin\frmxm1v.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 production environment.

8. 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
```

9. If you made customizations to the Forms Java EE application .ear file, such as overriding the default Forms servlet access URL, custom deploy the Forms Java EE application .ear file and create servlet aliases similar to test environment in the Forms Java EE application web.xml file.

Task 5 Move Oracle Reports to the New Production System

To move Oracle Reports to the production environment:

1. For the following Oracle Reports Server configuration files, merge changes made from the test environment to the production environment files. Note that you cannot just copy the files from the test environment to the production 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	<p><i>ORACLE_INSTANCE</i>/config/ReportsServerComponent/<i>server_name</i>/rwserver.conf</p> <p><i>ORACLE_INSTANCE</i>/config/ReportsServer/<i>server_name</i>/jdbcpds.conf</p> <p><i>ORACLE_INSTANCE</i>/config/ReportsServer/<i>server_name</i>/xmlpds.conf</p> <p><i>ORACLE_INSTANCE</i>/config/ReportsServer/<i>server_name</i>/textpds.conf</p> <p><i>ORACLE_INSTANCE</i>/config/ReportsServer/<i>server_name</i>/rwnetwork.conf</p> <p><i>ORACLE_INSTANCE</i>/config/ReportsServer/<i>server_name</i>/component-logs.xml</p> <p><i>ORACLE_INSTANCE</i>/config/ReportsServer/<i>server_name</i>/logging.xml</p>
Reports in-process server and servlet configuration	<p><i>DOMAIN_HOME</i>/config/fmwconfig/servers/<i>server_name</i>/applications/reports_<i>version</i>/configuration/cgicmd.dat</p> <p><i>DOMAIN_HOME</i>/config/fmwconfig/servers/<i>server_name</i>/applications/reports_<i>version</i>/configuration/rwservlet.properties</p> <p><i>DOMAIN_HOME</i>/config/fmwconfig/servers/<i>server_name</i>/applications/reports_<i>version</i>/configuration/rwserver.conf</p> <p><i>DOMAIN_HOME</i>/config/fmwconfig/servers/<i>server_name</i>/applications/reports_<i>version</i>/configuration/jdbcpds.conf</p> <p><i>DOMAIN_HOME</i>/config/fmwconfig/servers/<i>server_name</i>/applications/reports_<i>version</i>/configuration/xmlpds.conf</p> <p><i>DOMAIN_HOME</i>/config/fmwconfig/servers/<i>server_name</i>/applications/reports_<i>version</i>/configuration/textpds.conf</p> <p><i>DOMAIN_HOME</i>/config/fmwconfig/servers/<i>server_name</i>/applications/reports_<i>version</i>/configuration/rwnetwork.conf</p> <p><i>DOMAIN_HOME</i>/config/fmwconfig/servers/<i>server_name</i>/applications/reports_<i>version</i>/configuration/logging.xml</p> <p><i>DOMAIN_HOME</i>/config/fmwconfig/servers/<i>server_name</i>/applications/reports_<i>version</i>/configuration/logmetadata.xml</p> <p><i>DOMAIN_HOME</i>/config/fmwconfig/servers/<i>server_name</i>/applications/reports_<i>version</i>/configuration/jazn-data.xml</p>
Reports Tools configuration	<p><i>ORACLE_INSTANCE</i>/config/ReportsTools/rwbuilder.conf</p> <p><i>ORACLE_INSTANCE</i>/config/ReportsTools/rwnetwork.conf</p> <p><i>ORACLE_INSTANCE</i>/config/ReportsTools/jdbcpds.conf</p> <p><i>ORACLE_INSTANCE</i>/config/ReportsTools/xmlpds.conf</p> <p><i>ORACLE_INSTANCE</i>/config/ReportsTools/textpds.conf</p> <p><i>ORACLE_INSTANCE</i>/config/ReportsTools/component-logs.xml</p> <p><i>ORACLE_INSTANCE</i>/config/ReportsTools/logging.xml</p>
Reports Bridge configuration	<p><i>ORACLE_INSTANCE</i>/config/ReportsBridge/<i>bridge_name</i>/rwbridge.conf</p> <p><i>ORACLE_INSTANCE</i>/config/ReportsBridge/<i>bridge_name</i>/rwnetwork.conf</p> <p><i>ORACLE_INSTANCE</i>/config/ReportsBridge/<i>bridge_name</i>/component-logs.xml</p> <p><i>ORACLE_INSTANCE</i>/config/ReportsBridge/<i>bridge_name</i>/login.xml</p>
Reports shell scripts	<p>(UNIX) <i>ORACLE_INSTANCE</i>/config/reports/bin/rw*.sh</p> <p>(Windows) <i>ORACLE_INSTANCE</i>\config\reports\bin\rw*.bat</p> <p>(UNIX) <i>ORACLE_INSTANCE</i>/config/reports/bin/reports.sh</p> <p>(Windows) <i>ORACLE_INSTANCE</i>\config\reports\bin\reports.bat</p> <p>(UNIX) <i>ORACLE_INSTANCE</i>/config/reports/bin/namingservice.sh</p> <p>(Windows) <i>ORACLE_INSTANCE</i>\config\reports\bin\namingservice.bat</p>

2. For the following Oracle Fusion Middleware configuration files, which are related to Oracle Reports Server configuration files, merge changes made from the test environment to the production environment files. Note that you cannot just copy the files from the test environment to the production 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	<i>DOMAIN_HOME</i> /config/fmwconfig/jps-config.xml <i>DOMAIN_HOME</i> /config/fmwconfig/jps-config-jse.xml <i>DOMAIN_HOME</i> /config/fmwconfig/system-jazn-data.xml
Forms and Reports common files	Font setup, aliasing, subsetting, embedding: <i>ORACLE_INSTANCE</i> /config/FRComponent/frcommon/guicommon/tk/admin/UIFont.ali Printer configuration (UNIX only): <i>ORACLE_INSTANCE</i> /config/FRComponent/frcommon/guicommon/tk/admin/uiprint.txt Toolkit configuration, encoding (UNIX only): <i>ORACLE_INSTANCE</i> /config/FRComponent/frcommon/guicommon/tk/admin/uTk2Motif.rgb PPD files (UNIX only): <i>ORACLE_INSTANCE</i> /config/FRComponent/frcommon/guicommon/tk/admin/PPD/* AFM files (UNIX only): <i>ORACLE_INSTANCE</i> /config/FRComponent/frcommon/guicommon/tk/admin/AFM/*

3. If you created additional Oracle Reports Server component instances in the test environment, create these in the production environment using `opmnctl`.
4. For resources related to Oracle Reports Server, take the following actions:
 - Copy any fonts used in the test environment from the directory specified by environment variable `REPORTS_FONT_DIRECTORY` to production environment. By default, they are in *ORACLE_INSTANCE*/reports/fonts.
 - Move the Common UNIX Printing System (CUPS) printing configuration to the production 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 test environment to the production environment.
 - Deploy JSP Web reports to the production environment in the following location:

DOMAIN_HOME/servers/WLS_REPORTS/stage/reports/reports/web.war
 - Move Reports-specific data tables that are referred to in the RDF files to the database in the production 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 production environment:

ORACLE_INSTANCE/reports/cache
 - For Reports scheduled job information, copy the server data (*server_name.dat*) file to the following location in production 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 test environment and the production 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 production environment.

7. If the job repository or job status repository is configured in the database, you must create the same schemas in the production 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 test database for the schemas RW_JOBS, RW_SERVER_JOB_QUEUE, and RW_SERVER_QUEUE to production 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 test environment to Oracle Internet Directory in the production environment. See Step 3 in [Task 3, "Move Oracle Internet Directory to the New Production Environment"](#).
10. If you used JAZN-XML-based identity and policy store in the test environment, move them to the LDAP in the production 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 WLST command `migrateSecurityStore`, as described in "Migrating Credentials with the Command `migrateSecurityStore`" in the *Oracle Fusion Middleware Application Security Guide*.
12. Move any database proxy users to the production database using database cloning tools.
13. If any Reports plug-ins are registered, copy the corresponding .jar files to the production environment and add the path to the files to the environment variable `REPORTS_CLASSPATH`.

Task 6 Move Oracle Business Intelligence Discoverer to the New Production Environment

To move Oracle BI Discoverer to the new production environment:

1. If you have modified the default user preferences, copy the following files from the test environment to the production 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 test environment to the production environment:

```
DOMAIN_HOME/config/fmwconfig/servers/WLS_DISCO/applications/discoverer_11.1.1.3.0/configuration/configuration.xml  
DOMAIN_HOME/config/fmwconfig/servers/WLS_DISCO/applications/discoverer_11.1.1.3.0/configuration/configuration-preview.xml
```

In the configuration.xml file, change the values of the following elements to reflect the production environment:

- applicationURL
 - oracleInstance
 - discovererComponentName
3. If you have changed the server configuration files, copy the following file from the test environment to the production environment:

```
ORACLE_INSTANCE/config/OPMN/opmn/opmn.xml
```

4. Copy the following file from the test environment to the production environment:

```
ORACLE_INSTANCE/config/OHS/ohs_name/moduleconf/module_disco.conf
```

Change the values of the following elements to reflect the production environment:

- WebLogicCluster. Valid only if a cluster exists.
 - WebLogicHost
 - WebLogicPort
5. Copy the following files from the test environment to the production environment:

```
DOMAIN_HOME/servers/WLS_  
DISCO/stage/discoverer/11.1.1.1.0/discoverer/configuration/base-descktop.xss  
DOMAIN_HOME/servers/WLS_  
DISCO/stage/discoverer/11.1.1.1.0/discoverer/configuration/blstyles.xss  
DOMAIN_HOME/servers/WLS_  
DISCO/stage/discoverer/11.1.1.1.0/discoverer/configuration/dc-blaf-review.xss  
DOMAIN_HOME/servers/WLS_  
DISCO/stage/discoverer/11.1.1.1.0/discoverer/configuration/dc-blaf.xsd  
DOMAIN_HOME/servers/WLS_  
DISCO/stage/discoverer/11.1.1.1.0/discoverer/configuration/dc-blaf.xss  
DOMAIN_HOME/servers/WLS_  
DISCO/stage/discoverer/11.1.1.1.0/discoverer/configuration/minimal-desktop.xss  
DOMAIN_HOME/servers/WLS_  
DISCO/stage/discoverer/11.1.1.1.0/discoverer/configuration/minimal-pda.xss  
DOMAIN_HOME/servers/WLS_  
DISCO/stage/discoverer/11.1.1.1.0/discoverer/configuration/oracle-desktop.xss  
DOMAIN_HOME/servers/WLS_  
DISCO/stage/discoverer/11.1.1.1.0/discoverer/configuration/oracle-pda.xss  
DOMAIN_HOME/servers/WLS_  
DISCO/stage/discoverer/11.1.1.1.0/discoverer/configuration/pocketPC.xss  
DOMAIN_HOME/servers/WLS_  
DISCO/stage/discoverer/11.1.1.1.0/discoverer/configuration/simple-desktop.xss  
DOMAIN_HOME/servers/WLS_  
DISCO/stage/discoverer/11.1.1.1.0/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/11.1.1.1.0/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 test environment to the production environment:

```
ORACLE_HOME/network/admin/tnsnames.ora
```

8. Move the DISCOVERER schema from the test environment to the production 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 test and production, you do not need to move the data.

9. Move the EUL data from the test environment to the production environment:

- a. Create the EUL user and an empty EUL on the production 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 test database by using the Discoverer Administrator to export the schema from the test database and import it into the database in the production 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 test environment to the production environment:

- a. Install the catalog in the production OLAP database, using the following command:

```
java -classpath d4o.jar oracle.dss.d4o.administration.D40Command install  
-h hostname -po port -sid sid -su "sys as sysdba"  
-sp password -p d4osys-password -t users
```

- b. Authorize users in the production OLAP database, using the following command:

```
java -classpath d4o.jar oracle.dss.d4o.administration.D40Command  
authorize -h hostname -po port -sid sid -p d4osys-password -u user
```

- c. Export the Discoverer catalog from the test database and import it into the database in the production 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 test Discoverer metadata repository to the production 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 production 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 production environment as in the test environment.

21.6.2 Moving Oracle Portal, Oracle Forms Services, Oracle Reports, and Oracle Business Intelligence Discoverer to an Existing Production Environment

In this scenario, you have installed Oracle Portal, Oracle Forms Services, Oracle Reports, and Oracle Business Intelligence Discoverer in a test environment and you want to move the components to a production environment that already exists.

To move to an existing production environment, perform the following tasks:

- [Task 1, "Move Oracle Portal to an Existing Production Environment"](#)
- [Task 2, "Move Oracle Forms Services to an Existing Production Environment"](#)
- [Task 3, "Move Oracle Reports to an Existing Production Environment"](#)
- [Task 4, "Move Oracle Business Intelligence Discoverer to an Existing Production Environment"](#)

Task 1 Move Oracle Portal to an Existing Production Environment

This scenario assumes that you have made changes to Oracle Portal in the test 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 production environment, take the steps described in [Task 3, "Move Oracle Portal to the New Production Environment"](#) in [Section 21.6.1](#).

Task 2 Move Oracle Forms Services to an Existing Production Environment

To move Oracle Forms Services to the existing production environment:

1. Copy the Oracle Forms Services application files (FMX, MMX, and PLX) from the test environment to the production 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 production 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 Production Environment

To move Oracle Reports to an existing production environment, take the same steps as described in [Task 5, "Move Oracle Reports to the New Production System"](#) in [Section 21.6.1](#).

Task 4 Move Oracle Business Intelligence Discoverer to an Existing Production Environment

In this scenario, you primarily use the test environment to create EULs for developing a business area without compromising the performance of production systems.

To move Oracle BI Discoverer to an existing production environment:

1. Move the configuration files that are listed in Steps 1 and 5 in [Task 6, "Move Oracle Business Intelligence Discoverer to the New Production Environment"](#).
2. Move the DISCOVERER schema from the test environment to the production 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 test and production, you do not need to move the data.

3. Move the EUL schema from the test environment to the production environment by using the Oracle database export and import utilities to export the schema from the test database and import it into the database in the production environment.

Note that the user names and schema names must be the same in the production environment as in the test environment.

21.7 Moving Oracle Business Intelligence Components to a Production System

This section describes the steps for moving Oracle Business Intelligence from a test environment to a production 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 test to production considerations for the repository.

The following scenarios assume that you have already installed and configured Oracle Business Intelligence components in a test environment and that you want to move them to either a new or an existing production environment:

- [Moving Oracle Business Intelligence Components to a New Production Environment](#)
- [Moving Oracle Business Intelligence Components to an Existing Production Environment When There are Few Patches to Apply](#)

- [Moving Oracle Business Intelligence Components to an Existing Production Environment When There are Many Patches to Apply](#)

If you are applying patches to an existing production 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 21.7.2](#), 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 21.7.3](#), 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.

21.7.1 Moving Oracle Business Intelligence Components to a New Production Environment

This section describes the steps for moving Oracle Business Intelligence from a test environment to a new production environment.

This scenario assumes that you have already installed and configured Oracle Business Intelligence components in a test environment and that you want to move them to a new production environment. The steps for moving to a new production environment are identical to the migration process.

To move Oracle Business Intelligence components to a new production environment, perform the following tasks:

- [Task 1, "Patch and Move the Test Environment"](#)
- [Task 2, "Patch-Merge the Repository File"](#)
- [Task 3, "Extract the Archive into the New Production Environment"](#)
- [Task 4, "Create a BI Domain in the New Production Environment"](#)
- [Task 5, "Configure Security in the New Production Environment"](#)
- [Task 6, "Move the Repository File and Oracle BI Presentation Catalog from the Test Environment to the New Production Environment"](#)
- [Task 7, "Deploy the Test Repository File and Oracle BI Presentation Catalog in the New Production Environment"](#)
- [Task 8, "Copy and Scale Out to New Cluster Hosts in the Production Environment"](#)
- [Task 9, "\(Optional\) Refresh Global Unique Identifiers \(GUIDs\)"](#)
- [Task 10, "Enable New Agents and Oracle BI Publisher Scheduled Jobs"](#)
- [Task 11, "Update Links to External Systems"](#)
- [Task 12, "\(Optional\) Move Oracle Business Intelligence Related Applications"](#)
- [Task 13, "Validate the Production System"](#)

Task 1 Patch and Move the Test Environment

1. Patch the test environment, and test the environment until it is ready.
For information, see *Oracle Fusion Middleware Patching Guide*.
2. If you are trying new configuration settings, then note down the changes that you made so that they can be replicated.
3. Copy the Middleware home containing the Oracle BI EE components from the test environment using the copyBinary script, as described in [Section 20.3.1](#).

Task 2 Patch-Merge the Repository File

In the test environment, use the Administration Tool and the Oracle BI Server XML API to perform a patch merge of the test repository file (.rpd) with the production 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 Extract the Archive into the New Production Environment

1. Copy the archive file (created in [Task 1, "Patch and Move the Test Environment"](#)) from the test environment to the production environment.
2. Copy the following files to the production environment:
 - 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. Ensure that JDK 1.6.4 or later or JRockit (which is installed with the Oracle Business Intelligence installer) is installed in the production environment.
4. Use the pasteBinary script to recreate the Middleware home in the new production environment, as described in [Section 20.3.1](#).

Task 4 Create a BI Domain in the New Production Environment

For information, see *Oracle Fusion Middleware Installation Guide for Oracle Business Intelligence*.

Task 5 Configure Security in the New Production 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, policies, and roles), see the appropriate documentation for your authentication provider. The following list provides sources for various components:

- Oracle Internet Directory: See [Task 3, "Move Oracle Internet Directory to the New Production Environment"](#)
- Oracle WebLogic Server: See *Oracle Fusion Middleware Securing Oracle WebLogic Server*
- Oracle Platform Security Services: See *Oracle Fusion Middleware Application Security Guide*

Task 6 Move the Repository File and Oracle BI Presentation Catalog from the Test Environment to the New Production Environment

1. Copy the repository file (.rpd) to the production environment.
2. Zip the entire Oracle BI Presentation Catalog in the test environment (using zip).
3. Unzip the catalog in the new production environment.

For information, see *Oracle Fusion Middleware System Administrator's Guide for Oracle Business Intelligence Enterprise Edition*.

4. Use Fusion Middleware Control to set the catalog location in the production environment.

For information, see "Using Fusion Middleware Control to Upload a Repository and Set the Oracle BI Presentation Catalog Location" in *Oracle Fusion Middleware System Administrator's Guide for Oracle Business Intelligence Enterprise Edition*.

Task 7 Deploy the Test Repository File and Oracle BI Presentation Catalog in the New Production Environment

1. Use Fusion Middleware Control in the new production environment to upload the repository file (.rpd).

For information, see "Using Fusion Middleware Control to Upload a Repository and Set the Oracle BI Presentation Catalog Location" in *Oracle Fusion Middleware System Administrator's Guide for Oracle Business Intelligence Enterprise Edition*.

2. If necessary, use the Administration Tool or the Oracle BI Server XML API to update connection pool and database settings in the repository file. The file might contain data source connection information from the test environment that must be changed to the production environment connection settings.

See "Moving from Test to Production Environments" in *Oracle Fusion Middleware Integrator's Guide for Oracle Business Intelligence Enterprise Edition* for information about performing this step using the Oracle BI Server XML API.

3. (Optional) Make the production repository file read-only by selecting **Disallow Online RPD Updates** in the Performance tab of the Capacity Management page in Fusion Middleware Control.

Task 8 Copy and Scale Out to New Cluster Hosts in the Production Environment

1. Copy the archive file (created in [Task 1, "Patch and Move the Test Environment"](#)) 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 20.3.1](#).

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 9 (Optional) Refresh Global Unique Identifiers (GUIDs)

You do not normally refresh GUIDs in the LDAP directory (identity store users) between test and production environments, because the LDAP directories that contain the GUIDs should be fan-out replicas in both the test and the production environments. Possible scenarios for refreshing are described in the following list:

- Oracle Business Intelligence test servers and production servers are both configured against the corporate LDAP directory.

There is no need to refresh LDAP GUIDs.

- Oracle Business Intelligence test servers are configured against a test LDAP and the production servers against the corporate LDAP, but the test LDAP is a fan-out replica of the corporate LDAP directory.

There is no need to refresh LDAP GUIDs.

- Oracle Business Intelligence test servers are configured against a test LDAP and the production servers against the corporate LDAP, but the test LDAP is not a fan-out copy of the corporate LDAP directory.

A refresh of the LDAP GUIDs is needed. For information, see [Section 21.7.4](#).

Task 10 Enable New Agents and Oracle BI Publisher Scheduled Jobs

If new agents were created in the test environment, click each agent in the Oracle BI Presentation Services Catalog Manager (in the production environment) to enable it.

Oracle BI Publisher reports are stored in the Oracle BI Presentation Catalog and so existing reports and new reports that are created in the test environment should be available.

In a production environment, Oracle WebLogic Server administrators should create JNDI connections (to be used by Oracle BI Publisher reports), using the same names as in the test environment. The connections should point to the production databases instead of the test databases. In this way, all reports automatically point to the production environment databases, instead of test environment databases, without any modification.

Task 11 Update Links to External Systems

Ensure that you move the static content that relates to external systems to the production environment, including the following files: JPG files in dashboards, Oracle Web Services Manager policy files, Action Framework files, and other policy files that specify how to communicate with external systems.

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 12 (Optional) Move Oracle Business Intelligence Related Applications

Move Oracle Business Intelligence related applications (such as Oracle BI for Microsoft Office) to the new production environment. For information, see the appropriate sections for these applications (where available).

Task 13 Validate the Production System

Validate that the production system accurately represents the test system.

21.7.2 Moving Oracle Business Intelligence Components to an Existing Production Environment When There are Few Patches to Apply

This section describes the steps for moving Oracle Business Intelligence from a test environment to an existing production environment when there are few patches to apply. (See [Section 21.7.3](#) if you have many patches to apply.)

The following steps assume that you have already installed and configured Oracle Business Intelligence components in a test environment and that you want to move them to an existing production environment.

To move Oracle Business Intelligence components to an existing production environment when there are few patches to apply, perform the following tasks:

- [Task 1, "Patch the Test and Existing Production Environments"](#)
- [Task 2, "Deploy the Test Repository File to the Existing Production Environment"](#)
- [Task 3, "Deploy the Test Oracle BI Presentation Catalog to the Existing Production 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 8, "Validate the Production System"](#)

Task 1 Patch the Test and Existing Production Environments

A patch applies a collection of bug fixes to an existing environment and includes new binary files and metadata updates.

1. Patch the test environment as required, and test until ready.
2. Patch the existing production environment to the same level as the test 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 Test Repository File to the Existing Production Environment

1. In the test environment, use the Administration Tool and the Oracle BI Server XML API to perform a patch merge of the test repository file (.rpd) with the production file.

You must complete this task only if you are moving to an existing production environment and have made changes to the RPD file in the test 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 production environment to upload the RPD file to use.

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 test environment that must be changed to the production environment connection settings.

See "Moving from Test to Production Environments" in the *Oracle Fusion Middleware Integrator's Guide for Oracle Business Intelligence Enterprise Edition* for information about performing this step using the Oracle BI Server XML API.

4. (Optional) Make the production 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 Test Oracle BI Presentation Catalog to the Existing Production Environment

1. Drag and drop new or updated folders from the test catalog into the production catalog as follows:
 - a. Open two Catalog Manager windows: one with the test catalog and another with the production catalog.
 - b. Selectively copy and paste the folders that you want from the test catalog into the production catalog.

Note: If you copy and paste folders where the same content has been changed in the test or production environments, then the test content overwrites the production 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 production 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 test and production environments, because the LDAP directories that contain the GUIDs should be fan-out replicas in both the test and the production environments. Possible scenarios for refreshing are described in the following list:

- Oracle Business Intelligence test servers and production servers are both configured against the corporate LDAP directory.

There is no need to refresh LDAP GUIDs.

- Oracle Business Intelligence test servers are configured against a test LDAP and the production servers against the corporate LDAP, but the test LDAP is a fan-out replica of the corporate LDAP directory.

There is no need to refresh LDAP GUIDs.

- Oracle Business Intelligence test servers are configured against a test LDAP and the production servers against the corporate LDAP, but the test LDAP is not a fan-out copy of the corporate LDAP directory.

A refresh of the LDAP GUIDs is needed. For information, see [Section 21.7.4](#).

Task 5 Enable New Agents and Oracle BI Publisher Scheduled Jobs

If new agents were created in the test environment, then click each agent in the Oracle BI Presentation Services Catalog Manager (in the production environment) to enable it.

Oracle BI Publisher reports are stored in the Oracle BI Presentation Catalog and so existing reports and new reports created in the test environment should be available.

In a production environment, Oracle WebLogic Server administrators should create JNDI connections (to be used by Oracle BI Publisher reports), using the same names as in the test environment. The connections should point to the production databases instead of the test databases. In this way, all reports automatically point to the production environment databases, instead of test 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 production environment, including the following files: JPG files in dashboards, Oracle Web Services Manager policy files, Action Framework files, and other policy files that specify how to communicate with external systems.

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 Oracle BI for Microsoft Office) to the existing production environment. For information, see the appropriate sections for these applications (where available).

Task 8 Validate the Production System

Validate that the production system accurately represents the test system.

21.7.3 Moving Oracle Business Intelligence Components to an Existing Production Environment When There are Many Patches to Apply

This section describes the steps for moving Oracle Business Intelligence from a test environment to an existing production environment when there are many patches to apply.

The following scenarios assume that you have already installed and configured Oracle Business Intelligence components in a test environment and that you want to move them to an existing production environment.

Use one of the following strategies to move Oracle Business Intelligence components to an existing production environment when there are many patches to apply:

- [Moving to an Existing Production Environment When New Hardware Is Available](#)
- [Moving to an Existing Production Environment When New Hardware Is Not Available](#)

21.7.3.1 Moving to an Existing Production Environment When New Hardware Is Available

Perform the following tasks to move Oracle Business Intelligence components to an existing production environment when there are many patches to apply and new hardware is available:

- [Task 1, "Follow the Steps for Moving to a New Production Environment"](#)
- [Task 2, "Switch Users from the Existing Production Environment to the New One"](#)
- [Task 3, "Remove the Existing Production Environment and Prepare It for the Next Patch"](#)

Task 1 Follow the Steps for Moving to a New Production Environment

Complete the steps in [Section 21.7.1](#) for moving to a new production environment.

These steps include merging the new test RPD file and catalog with those in the existing production environment. Ideally you merge once and resolve the issues while users continue using the existing environment. When the files are correct, you lock the production environment and repeat the merge to access the latest changes.

Task 2 Switch Users from the Existing Production Environment to the New One

Use a load balancer such as Oracle Web Cache to redirect users from a standard URL to the new production environment.

Task 3 Remove the Existing Production 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.

21.7.3.2 Moving to an Existing Production Environment When New Hardware Is Not Available

Perform the following tasks to move Oracle Business Intelligence components to an existing production environment when there are many patches to apply and new hardware is not available:

- [Task 1, "Scale the Production Environment Back to One Host"](#)
- [Task 2, "Patch the Host in the Production Environment"](#)
- [Task 3, "Remove the Existing Software on the Cluster Hosts"](#)
- [Task 4, "Move the Production Environment and Then Copy to the Cluster Hosts"](#)

Task 1 Scale the Production Environment Back to One Host

Use the Capacity Management tab of the Scalability page in Fusion Middleware Control in the production 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 production environment.

For more information, see the Fusion Middleware Control Help system.

Task 2 Patch the Host in the Production Environment

Patch the host in the production 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 Production Environment and Then Copy to the Cluster Hosts

Complete the tasks beginning with [Task 8, "Copy and Scale Out to New Cluster Hosts in the Production Environment"](#) in [Section 21.7.1](#).

21.7.4 Refreshing the User GUIDs

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*.

Note: Refreshing the GUIDs requires that you stop and restart the system components from the command line and not Fusion Middleware Control. This includes the Administration Server and Managed Servers. After the Administration Server is stopped, you cannot start it from Fusion Middleware Control because it is not available during this time.

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 to add the last line in the following instruction.

```
<ps:Catalog xmlns:ps="oracle.bi.presentation.services/config/v1.1">  
<ps:UpgradeAndExit>>false</ps:UpgradeAndExit>  
<ps:UpdateAccountGUIDs>UpdateAndExit</ps:UpdateAccountGUIDs>
```

4. From a terminal window, stop and restart the managed processes using the `opmnctl` 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 `NQSCfg.INI` file to reset the `FMW_UPDATE_ROLE_AND_USER_REF_GUIDS = YES` to `NO` and restart the Oracle BI Servers.
6. Remove, set to `none`, or comment out the last line added to the `instanceconfig.xml` file (that instructs Presentation Services to refresh GUIDs on restart, as specified in Step 3).

```
<ps:Catalog xmlns:ps="oracle.bi.presentation.services/config/v1.1">
<ps:UpgradeAndExit>>false</ps:UpgradeAndExit>
<ps:UpdateAccountGUIDs>none</ps:UpdateAccountGUIDs>
```

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*.

21.8 Moving Oracle Real-Time Decisions to a Production System

The following topics describe how to move Oracle Real-Time Decisions (Oracle RTD) from a test environment to a new production environment:

- [Moving Oracle Real-Time Decisions to a New Production Environment](#)
- [Moving Oracle Real-Time Decisions to an Existing Production Environment](#)

21.8.1 Moving Oracle Real-Time Decisions to a New Production Environment

To move the environment to a production environment, perform the following tasks:

- [Task 1, "Move the Middleware Home and Perform the Initial Configuration"](#)
- [Task 2, "Install Oracle RTD Clients \(If Used\) on the Production Environment"](#)
- [Task 3, "Move Oracle RTD Inline Services"](#)
- [Task 4, "Edit Additional Oracle RTD Components for Production"](#)

Task 1 Move the Middleware Home and Perform the Initial Configuration

To move the Middleware home and Oracle RTD software and perform the initial configuration:

1. Create the required schemas in the production database using RCU. See the *Oracle Fusion Middleware Repository Creation Utility User's Guide*.

2. Move a copy of the Middleware home, as described in [Section 20.5.1](#). The Oracle WebLogic Server home and the Oracle homes in the Middleware home are also moved.

If your environment includes Oracle BI EE and you have already moved Oracle BI EE to a new production environment, as described in [Section 21.7.1](#), 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. Configure Oracle RTD and create a domain using the Configuration Wizard. See "Configuring Oracle RTD After Installation" in the *Oracle Fusion Middleware Administrator's Guide for Oracle Real-Time Decisions*. For the purposes of this configuration, follow the steps for configuring Oracle RTD after a Software Only install.

The following are important factors to consider in the setup and configuration of the production system:

1. When using the Repository Configuration Utility (RCU), connection pools must reflect database connections specific to the production environment.
2. During the configuration of custom roles and security settings, the setup parameters must be modified to reflect settings for the production environment.

For more details, see the Security chapter in *Oracle Fusion Middleware Administrator's Guide for Oracle Real-Time Decisions*.

3. Any performance tuning parameters defined in the test environment should also be re-created in the production environment. This includes performance parameters both at the application server and database levels.

Task 2 Install Oracle RTD Clients (If Used) on the Production Environment

If used for the integration of Oracle RTD to a customer's front-end applications, Oracle RTD clients must be installed in the production 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 production architecture.

Task 3 Move Oracle RTD Inline Services

Move Oracle RTD Inline Services that exist on the test environment to the production environment:

1. Moving Inline Services to the production 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, via 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 production 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 test and production 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 test and production 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 production environment should be configured to point to the production database.

Task 4 Edit Additional Oracle RTD Components for Production

Additional tasks that you may need to perform with Oracle RTD include the following:

1. Creating and configuring the model snapshot tables.

- a. The Oracle RTD model snapshot tables can be created in the production environment in two ways, RCU or the tool `sdexec/SDDDBTool`, which is provided with the installation.

RCU creates the necessary snapshot tables in the same schema as the Oracle RTD platform tables, while `sdexec/SDDDBTool` 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 production 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.

21.8.2 Moving Oracle Real-Time Decisions to an Existing Production Environment

After a production 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 production follows the same steps as outlined for a full product test to production move.

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*.

21.9 Moving Oracle Enterprise Content Management to a Production System

The following topics describe how to move Oracle Enterprise Content Management Suite to a production environment:

- [Moving Oracle Enterprise Content Management Suite to a New Production Environment](#)
- [Moving Oracle Enterprise Content Management Suite to an Existing Production Environment](#)

In both scenarios, you have performed the following in the test environment:

- Installed a database to be used for required schemas.
- Created needed schemas 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 Enterprise Content Management Suite.
- If Oracle Imaging and Process Management uses Workflow or Oracle Application Extension Framework (AXF), installed and configured Oracle SOA Suite.
- Configured Oracle Universal Content Management.
- Configured Oracle Imaging and Process Management.

If Oracle I/PM uses Oracle Universal Content Management 10g repository, ensure that the repository was manually configured for Oracle I/PM.

- Configured Oracle Universal Records Management.
- Defined several definitions, such as Connections, Applications, Searches, and Inputs for Oracle I/PM.
- Installed and configured Identity Management components, such as Oracle Internet Directory.

21.9.1 Moving Oracle Enterprise Content Management Suite to a New Production Environment

To move Oracle Enterprise Content Management Suite to a new production environment, perform the following tasks:

- [Task 1, "Create a New Database and Populate It with Schemas"](#)
- [Task 2, "Move the Middleware Home and the Oracle Enterprise Content Management Suite Configuration"](#)
- [Task 3, "Modify Oracle Information Rights Management Settings"](#)
- [Task 4, "Move Oracle Universal Content Management to a New Production Environment"](#)
- [Task 5, "Move Oracle Imaging and Process Management to a New Production Environment"](#)
- [Task 6, "Move Oracle Universal Records Management to a New Production Environment"](#)

Note that in a production system, Oracle Enterprise Content Management Suite 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 Create a New Database and Populate It with Schemas

To create the database and populate it with the necessary schemas:

1. Create a new database on the production system.
2. Create the required schemas in the production database using RCU. See the *Oracle Fusion Middleware Repository Creation Utility User's Guide*.

Task 2 Move the Middleware Home and the Oracle Enterprise Content Management Suite Configuration

Take the following steps to move the Middleware home and perform the initial configuration:

1. Move a copy of the Middleware home using the `copyBinary` and `pasteBinary` scripts, as described in [Section 20.5.1](#). The Oracle WebLogic Server home and the Oracle homes in the Middleware home are also moved.
2. Move the configuration of Oracle Enterprise Content Management Suite by using the `copyConfig`, `extractMovePlan`, and `pasteConfig` scripts as described in [Section 20.5.2](#).

This step copies the configuration of Oracle Enterprise Content Management Suite components. It copies the domain directory, the Administration Server, and Managed Servers. It also:

- Modifies information about the database that contains Oracle IRM metadata.
 - Copies the Oracle I/PM 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.
3. If you are using an Oracle UCM 10g repository, ensure that Full-Text is configured correctly on the production Oracle UCM system, if configured on the test Oracle UCM 10g system.

Note that if you are using an Oracle UCM 10g server, it is not configured when you move the Oracle Enterprise Content Management Suite. you must install it, similar to the way you installed it in the test environment, using the procedures described in the Oracle Universal Content Management page at:

<http://www.oracle.com/technology/products/content-management/ucm/index.html>

Task 3 Modify Oracle Information Rights Management Settings

You need to modify some Oracle IRM settings in the new production 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 test 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 Fusion Middleware Installation Guide for Oracle Enterprise Content Management Suite*.
3. If the production environment is not using the same LDAP store as the test environment, migrate the users from the test environment to the production environment. See "Reassociating the Identity Store with an External LDAP Authentication Provider" in the *Oracle Fusion Middleware Installation Guide for Oracle Enterprise Content Management Suite*.

Task 4 Move Oracle Universal Content Management to a New Production Environment

To move Oracle Universal Content Management to a new production environment:

1. Export the OCS database schema from the test 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 production database.

2. Import the OCS database schema that you exported from the test 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 a new database to hold the new search collection.

To set up an Oracle Secure Enterprise Search instance and configure it for Oracle UCM on the production system:

- a. Install Oracle Secure Enterprise Search as described in the *Oracle Secure Enterprise Search Installation and Upgrade Guide*.
- b. Create a new data source to connect to Oracle Secure Enterprise Search. See *Oracle Fusion Middleware Configuring and Managing JDBC for Oracle WebLogic Server*.

Oracle recommends that you use the same data source name as in the test environment.

- c. On the Oracle Universal Content Management 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 Fusion Middleware System Administrator's Guide for Content Server*.
4. If you configured the IntradocDir, WeblayoutDir, and VaultDir directories to be outside of the domain structure, copy the directories to the production environment.
5. In the production environment, modify the following file, updating the entries for IntradocDir, WeblayoutDir, VaultDir, and IdcHomeDir:

```
DOMAIN_HOME/bin/intradoc.cfg
```

This step is not necessary if the directory structures of the production environment is the same as the test environment.

6. On the production system, delete the following file:

```
IntradocDir/data/contenttracker/config/sct.cfg
```

The file is regenerated when you restart the server.

7. Modify the following file, updating the HttpServerAddress to reflect the correct address:

```
instance_dir/config/config.cfg
```

8. Restart the Administration Server and the Managed Server.

To use the new production system as a template and replicate it in multiple production systems, you follow the procedures in [Task 2, "Move Oracle Universal Content Management to an Existing Production Environment"](#), except that you must also change the parameters IDC_Name, InstanceMenuLabel, InstanceDescription, HttpServerAddress, and AutoNumberPrefix in the following file:

```
instance_dir/config/config.cfg
```

You may also need to change the parameters MailServer and SysAdminAddress.

Task 5 Move Oracle Imaging and Process Management to a New Production Environment

Note that no Oracle I/PM data is migrated in this procedure. Data migrated with Oracle UCM will not be accessible from production Oracle I/PM system.

Before you begin this procedure, if you use Workflow Integration or Oracle Application Extension Framework (AXF), you have:

- Installed and configured Oracle SOA Suite and moved its test environment to the production environment, as described in [Section 21.3.1](#).
- Configured Oracle I/PM, extending the SOA domain, as described in "Extending an Existing Domain" in the *Oracle Fusion Middleware Installation Guide for Oracle Enterprise Content Management Suite*.

To move Oracle I/PM to the new production environment:

1. Start the Administration Server and the Oracle I/PM Managed Server.
2. Ensure that the Oracle UCM 11g Managed Server is running, or, if you are using Oracle UCM 10g, that the Oracle UCM 10g services are running.
3. If you have moved Oracle Internet Directory to the production environment, export Oracle I/PM users and groups from the test environment.

The user who logs in first to an Oracle I/PM Managed Server is provisioned with full security throughout the server. It is easier to reassociate the identity store for Oracle I/PM with an external LDAP authentication provider before the first user logs in, completes the configuration of the Oracle I/PM Managed Server, and connects it to the Oracle UCM repository.

Export the users and groups from LDAP identity store on the test environment, using the `ldapsearch` command. This produces an `ldif` file that you later import into the LDAP identity store in the production 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"
```

4. Export the system definitions from the test environment.

Note the following:

- This procedure does not transfer any documents from the test environment to the production environment. This procedure only moves the structure defined by the applications, inputs, and searches.
 - The documents reside within the supporting Oracle UCM repository. Oracle I/PM does not recognize any documents that were transferred using Oracle UCM test-to-production procedures or database utilities. The documents will not be accessible from Oracle I/PM. Use Oracle I/PM to upload new documents into the Oracle UCM repository.
- a. Log into the test system as administrator to export the definitions, using the following URL:


```
http://hostname:16000/imaging
```
 - b. Expand Tools, then select **Export Definitions**.
 - c. For **Export Comments**, enter any comments, then click **Next**.
 - d. Select the applications to export, then click **Next**.

- e. Select the searches to export. All dependent applications for each search are included in the export. Click **Next**.
- f. Select the inputs to export. All dependent applications for each input are included in the export. Click **Next**.
- g. Review your selections for accuracy, then click **Next**.
If you need to make any changes, select the appropriate type of definition at the top of the page, such as Applications, and correct the selections. Then, select the Summary at the top of the page.
- h. Select **Create Export File**.
- i. Depending on your browser, a dialog box is displayed allowing you to open or save the file. The file is saved to the location you specify.

5. If the sample files are outside the domain directory, copy the files from the test environment to the production environment. The location of the files is specified in the Oracle I/PM MBean SampleDirectory.

For information about the Oracle I/PM MBeans, see "Oracle I/PM MBeans" in the *Oracle Fusion Middleware Administrator's Guide for Oracle Imaging and Process Management*. For more information about using Input sample files to create input definitions, see "Creating Input Definitions" in the *Oracle Fusion Middleware Administrator's Guide for Oracle Imaging and Process Management*.

6. If you have moved Oracle Internet Directory to the production environment, import users and groups into the LDAP identity store on the production environment by importing the ldif file that you exported from the test environment into the production 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
```

7. Enable SSL, as described in "System Security" in the *Oracle Fusion Middleware Administrator's Guide for Oracle Imaging and Process Management*.
8. If a non-default wallet was used at the test environment for either an SSL-enabled listen address or an OSwallet, or both, export the wallets from the test environment and import them at the production environment. For information about exporting and importing wallets, see [Section 8.4.4](#).
9. Configure connections on the production environment. Before you can import the applications into the production environment, you must set up the Oracle I/PM system connection objects that establish links to Oracle UCM repository and Workflow servers in the production environment. If the connections in the production environment are named the same as in the test environment, the imported definitions link correctly without further action.

If the names are different, you can select the desired connection name using the using the Oracle I/PM import tool.

10. Modify configuration settings. You use Fusion Middleware Control to modify the system MBean values, which control the execution of Oracle I/PM. If any MBean values need to be changed, follow the procedures in "Configuring the AgentUser and GDFontPath MBeans" in the *Oracle Fusion Middleware Installation Guide for Oracle Enterprise Content Management Suite*.

11. If the Work Manager configuration for the Input Agents does not already specify values for the production environment, update the configuration, as described in "Changing Oracle WebLogic Server Work Manager Settings" in the *Oracle Fusion Middleware Administrator's Guide for Oracle Imaging and Process Management*.
12. If the Work Manager configuration for the Workflow Agents does not already specify values for the production environment, update the configuration, as described in "Changing WebLogic Server Work Manager Settings" in the *Oracle Fusion Middleware Administrator's Guide for Oracle Imaging and Process Management*.
13. Note that User preferences are not migrated. Users must reconfigure their preferences on the production Oracle I/PM system.
14. Import the system definitions to the production environment.
 - a. Log into the production environment system as administrator to import the definitions, using the following URL:

```
http://hostname:16000/imaging
```
 - b. Expand **Tools**, then select **Import Definitions**.
 - c. For **File Location**, click **Browse**, and browse to the location of the file you exported from the test environment in step 4.

When you select the file, the File Date, and File Comments fields are populated.
 - d. Click **Next**.
 - e. From each table, select the applications, inputs, and searches, to be imported.

Selecting the Plus sign (+), shows the description for the definition. Selecting the pull down for the repository allows you to place each definition in any of the defined repository connections.
 - f. Click **Next**.
 - g. In the validation phase, you can see the status of each definition and you can select Document Security, Storage Policy, Workflow, and Full-Text options. See "Import Definitions: Validate Imports Page" in the *Oracle Fusion Middleware Administrator's Guide for Oracle Imaging and Process Management*.

If any of the definitions are marked by a red X, the definition cannot be imported. Some of the likely causes are:
 - For search and input definitions, a required application was not imported.
 - The security checks failed.
 - The connection specified by the application does not exist.
 - The Workflow validation failed.
 - h. Correct any problems. When all the definitions are valid, select **Submit**.

If the validation is successful, the changes are committed. If there are errors, the page shows new exceptions. Correct any errors and click **Submit**.
15. Move the Oracle Application Extension Framework (AXF) configuration database:
 - a. Export the following tables from the test database schema and insert them into the production database schema:
 - AXF_ACTIONS
 - AXF_ACTION_MENU

- AXF_ACTION_PARAMETERS
- AXF_COMMANDS
- AXF_ENUM_ITEMS
- AXF_ENUM_TYPES
- AXF_METADATA_ATTRIBUTES
- AXF_METADATA_BLOCKS
- AXF_SOLUTIONS
- AXF_SOLUTION_ATTRIBUTES
- AXF_SOLUTION_PARAMETERS
- AXF_XPATH_NAMESPACES
- AXF_XPATH_ATTRIBUTES

- b. Modify the Workflow Connection information to point the connection to the production Workflow system.

If you are using a different Workflow Connection name in the production environment, update the PARAMETER_VALUE column. Using the Workflow connection created in step 9, update the PARAMETER_VALUE column for the BPEL_CONNECTION row in the AXF_SOLUTION_ATTRIBUTES table using your preferred SQL utility:

```
UPDATE AXF_SOLUTION_ATTRIBUTES SET PARAMETER_VALUE = '<ConnectionName>'
WHERE PARAMETER_KEY = 'BPEL_CONNECTION'
```

- c. If you have created any SOA Composites for an AXF solution in the test environment, deploy those versions of composites in the production environment, as described in the *AXF11g Solution Template Guide*.

16. If you cannot retrieve documents from the IPM Viewer, you must change the permissions for the following file:

```
DOMAIN_HOME/oracle/imaging/imaging-server/ixTransformer
```

Task 6 Move Oracle Universal Records Management to a New Production Environment

To move Oracle Universal Records Management to a new production environment:

1. On the test environment, export the configuration settings, such as retention schedule, security classifications, and triggers, as described in "Exporting an Archive" in the *Oracle Fusion Middleware Administrator's Guide for Universal Records Management*.
2. Copy the archive to the production environment.
3. Import the archive to the production environment, as described in "Importing an Archive" in the *Oracle Fusion Middleware Administrator's Guide for Universal Records Management*.

21.9.2 Moving Oracle Enterprise Content Management Suite to an Existing Production Environment

In this scenario, you have installed Oracle Enterprise Content Management Suite components, such as Oracle Information Rights Management, in a test environment and you want to move them to a production environment that already exists.

On the existing production system, you have installed and configured the components. You want to move an application from the test environment to the production Oracle Enterprise Content Management Suite environment.

To move Oracle Enterprise Content Management Suite to an existing production environment, perform the following tasks:

- [Task 1, "Move Oracle Information Rights Management to an Existing Production Environment"](#)
- [Task 2, "Move Oracle Universal Content Management to an Existing Production Environment"](#)
- [Task 3, "Move Oracle Imaging and Process Management to an Existing Production Environment"](#)
- [Task 4, "Move Oracle Universal Records Management to an Existing Production Environment."](#)

Task 1 Move Oracle Information Rights Management to an Existing Production Environment

Organizations that run a proof of concept or pilot (test) deployment can copy the operational service into a production environment and continue to use all existing test content, contexts, and rights.

The IRM server URL (for example *protocol_schema:\hostname:port\irm_desktop*) is sealed into test content. Therefore, this value must not change on moving from test to production. For this reason, make sure you consider the following points when installing the test deployment:

- Configure SSL in the test deployment because switching from the HTTP protocol in the test system to the HTTPS protocol in the production system would prevent test-sealed content from working in a production system.
- Use a generic host name such as *irm.example.com* for the test deployment rather than a machine-specific host name such as *mytestdeploymachine.example.com*.

After the test to production installation has been completed, the DNS entries for domain name can be switched from the test server to the production system. If needed, you can use port redirection to ensure that the test deployment IRM Server URL points to the production environment deployment.

To move a test deployment into a production environment:

1. If the production database is different from the test database, you should back up the Oracle IRM schema. Restore the backup into the production database.
2. Copy the Oracle IRM keystore set up during the test installation to the production 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 test 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 "Setting Passwords for the Keystore" in the *Oracle Fusion Middleware Installation Guide for Oracle Enterprise Content Management Suite*.

5. Copy the Oracle IRM configuration file, `irm-config.xml`, which is usually located in the following directory, from the test environment to the production environment:

```
DOMAIN_HOME/config/fmwconfig
```

6. Because the test environment configuration may contain test-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 test environment keystore. The following is the suggested location of the file:

```
DOMAIN_HOME/config/fmwconfig
```

7. If the production environment is not using the same user store as the test environment, migrate the users from the test environment to the production environment. See "Reassociating the Identity Store with an External LDAP Authentication Provider" in the *Oracle Fusion Middleware Installation Guide for Oracle Enterprise Content Management Suite*.

Task 2 Move Oracle Universal Content Management to an Existing Production Environment

To move Oracle Universal Content Management to an existing production 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 test environment to the production environment, export archives from the test environment and then import them into the production environment, as described in "Managing Imports and Exports" in the *Oracle Fusion Middleware Administrator's Guide for Universal Records Management*.

Task 3 Move Oracle Imaging and Process Management to an Existing Production Environment

To move Oracle Imaging and Process Management from a test environment to an existing production environment, you use the same steps as described in [Task 5, "Move Oracle Imaging and Process Management to a New Production Environment"](#). However, note the following about updating definitions on the production environment:

- When you import a definition from the test environment to the existing production 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 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 test environment, you must manually delete it in the production 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 production environment, open the Managed Inputs folder and select the input that you want to import.
 2. Select **Toggle On-Line**.

Task 4 Move Oracle Universal Records Management to an Existing Production Environment.

To move Oracle URM to an existing production environment:

1. On the test environment, export any configuration settings that have changed, as described in "Exporting an Archive" in *Oracle Fusion Middleware Administrator's Guide for Universal Records Management*.
2. Copy the archive to the production environment.
3. Import the archive to the production environment, as described in "Importing an Archive" in *Oracle Fusion Middleware Administrator's Guide for Universal Records Management*.

21.10 Moving Oracle Data Integrator to a Production Environment

The following topics describe how to move Oracle Data Integrator from a test environment to a production environment:

- [Moving Oracle Data Integrator to a New Production Environment](#)
- [Moving Oracle Data Integrator Scenarios to an Existing Production Environment](#)

In both scenarios, you have performed the following in a test environment:

- Installed Oracle WebLogic Server and created the Middleware home for the Java components.
- Created the required schemas in the test database using RCU.
- Installed Oracle Data Integrator.
- Configured and deployed Oracle Data Integrator Java components using the Configuration Wizard. The Java components can connect and use the test repositories.
- Configured the standalone agents. The standalone agents can connect and use the test repositories.
- Configured the logical and physical architecture in the topology. The connection of the data servers must be tested from the Studio as well as from any Physical Agent.
- Configured the security by defining the users and the privileges.
- Declared the Open Tools used by the Scenarios.
- Deployed one or more development (packages) or run-time (scenarios) artifacts in the repositories.
- Changed some configuration settings. For example, you may have changed the data server configuration to match the test environment.

21.10.1 Moving Oracle Data Integrator to a New Production Environment

In this scenario, you have installed Oracle Data Integrator in a test environment and you want to move it to a production environment which does not yet exist.

To move Oracle Data Integrator to a new production environment, perform the following tasks:

- [Task 1, "Install the Software and Perform the Initial Configuration"](#)
- [Task 2, "Move the Topology to the Production Environment"](#)
- [Task 3, "Move the Work Repository Content to the Production Environment"](#)
- [Task 4, "Change the Settings for the Production Environment"](#)
- [Task 5, "Restart the Run-Time Agents"](#)

Task 1 Install the Software and Perform the Initial Configuration

To install the software and perform the initial configuration on the production system:

1. Create the required master and work repositories schemas in the production 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 production environment are created with unique IDs across your entire organization,

including your development and test repositories. Also make sure that the production work repository is created with the same type as the test repository (For example, if the test work repository is created as a development repository, the production work repository must also be created as a development repository).

2. Install Oracle Data Integrator:

- If you are using Java components, move a copy of the Middleware home from the test environment to the production environment, as described in [Section 20.5.1](#). The Oracle WebLogic Server home and the Oracle homes in the Middleware home are also moved.
- If you have performed a standalone installation of Oracle Data Integrator (that is, you did not install any Java components), install Oracle Data Integrator with the same components in the production environment.

3. Create a domain containing Oracle Data Integrator:

- If you are using Java components, move a copy of the domain containing Oracle Data Integrator from the test environment to the production environment, as described in [Section 20.5.2](#).
- If you have performed a standalone installation of Oracle Data Integrator, configure Oracle Data Integrator and create a domain using the Configuration Wizard, as described in "Configure a WebLogic Domain" in the *Oracle Fusion Middleware Installation Guide for Oracle Data Integrator*.

4. Configure the Oracle Data Integrator standalone agents required in the production environment. See "Configuring the Standalone Agent" in the *Oracle Fusion Middleware Installation Guide for Oracle Data Integrator*.

Task 2 Move the Topology to the Production Environment

To move the topology to the new production environment:

1. Export the topology from the test master repository using Oracle Data Integrator Studio. See "Export/Import the Topology and Security Settings" in the *Oracle Fusion Middleware Developer's Guide for Oracle Data Integrator*.
2. Import the topology into the production master repository using Oracle Data Integrator Studio. See "Export/Import the Topology and Security Settings" in the *Oracle Fusion Middleware Developer's Guide for Oracle Data Integrator*. Use the Synonym Mode INSERT_UPDATE for this import.

Task 3 Move the Work Repository Content to the Production Environment

To move the work repository content to the new production environment:

1. Export the work repository content from the test work repository using Oracle Data Integrator Studio. See "Exporting and Importing a Work Repository" in the *Oracle Fusion Middleware Developer's Guide for Oracle Data Integrator*.
2. Import the work repository content into the production work repository using Oracle Data Integrator Studio. See "Exporting and Importing a Work Repository" in the *Oracle Fusion Middleware Developer's Guide for Oracle Data Integrator*.

Task 4 Change the Settings for the Production Environment

Change the following settings in the production environment:

- Topology settings: It is usually recommended to use different contexts for test and production environments, and switch the context to schedule or run scenarios in a

given environment. Using a single context forces you to modify the physical architecture configuration (for example, the host name of the data servers) when moving to production.

If you are using different Oracle Data Integrator contexts for your test and production environments, and the physical architecture is already defined for the production environment, review this physical architecture before proceeding. See "Setting-up the Topology" in the *Oracle Fusion Middleware Developer's Guide for Oracle Data Integrator*.

If you are using a single context for both the environments, then you must review and change the following items in the physical architecture in the production master repository:

- Physical Agents: Change the host, port, and Web application context (for Java EE Agent) to match the configuration of the production environment.
- Data Servers: Change the data server connection information (JDBC, JNDI, data source name) to match the configuration of the production environment.
- Physical Schemas: The schemas (including file folder location) defined for the data servers must match the configuration of the production environment.
- Scheduling: If you are using different Oracle Data Integrator contexts for your test and production environments, and schedules are only defined to run in the test context, you must change these schedules in the production work repository to execute them in the production context. See "Scheduling Scenarios" in the *Oracle Fusion Middleware Developer's Guide for Oracle Data Integrator*.

If you are using a single context for both the environments, then existing schedules do not need to be modified.

- If scenario schedules are not defined in the test environment, you can create them in the production environment.

Task 5 Restart the Run-Time Agents

Restart the standalone and Java EE agents in the production environment. These agents start processing the scheduled scenarios.

21.10.2 Moving Oracle Data Integrator Scenarios to an Existing Production Environment

In this scenario, you have a number of new or regenerated scenarios in a test environment and you want to move them to a production environment that already exists.

Note: Scenarios already deployed in the production environment should be regenerated before being moved in the production environment. They should not be deleted then generated.

On the existing production system, you have installed and configured the components as described in [Section 21.10.1](#). You want to move the data integration scenarios from the test environment to the production Oracle Data Integrator environment.

To move Oracle Data Integrator scenarios to an existing production environment, perform the following tasks:

- [Task 1, "Move Topology Updates"](#)

- [Task 2, "Move New and Updated Scenarios"](#)

Task 1 Move Topology Updates

Perform this task if the topology in the test environment was modified or added for the new scenarios.

To move topology updates to an existing production system:

1. Export updated and new physical and logical topology items (data servers, schemas, agents) from the test master repository using Oracle Data Integration Studio. See "Exporting and Importing Objects" in the *Oracle Fusion Middleware Developer's Guide for Oracle Data Integrator*.
2. Import updated and new physical and logical topology items (data servers, schemas, agents) into the production master repository using Oracle Data Integration Studio. See "Exporting and Importing Objects" in the *Oracle Fusion Middleware Developer's Guide for Oracle Data Integrator*.
3. Update the Open Tools definitions and declare the new ones.

Task 2 Move New and Updated Scenarios

To move Oracle Data Integrator scenarios to an existing production system:

1. Export the new and regenerated scenarios from the test work repository using Oracle Data Integrator Studio. You can export them using single scenario export or multiple scenario export, or you can export all scenarios in a given project or folder. See "Exporting Scenarios" in the *Oracle Fusion Middleware Developer's Guide for Oracle Data Integrator*.
2. Import the new or regenerated scenarios in the production work repository using Oracle Data Integrator Studio. Use the Synonym Mode INSERT_UPDATE for this import. See "Importing Scenarios in Production" in the *Oracle Fusion Middleware Developer's Guide for Oracle Data Integrator*.
3. Change the schedules in the production work repository to execute the new scenarios in the production context. See "Scheduling Scenarios" in the *Oracle Fusion Middleware Developer's Guide for Oracle Data Integrator*.

21.11 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 test environment that is not an Oracle RAC environment to a production 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 test environment to the production environment.

Then, you configure your production environment for Oracle RAC, as described in the *Oracle Fusion Middleware High Availability Guide*, especially "Considerations for High Availability Oracle Database Access."

- If you are moving from a test environment that uses Oracle RAC to a production 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 production environment.

- If you are moving from a test environment that uses Oracle RAC to a production environment that uses Oracle RAC, but you have more Oracle RAC instances in the production environment, the move plan will have one entry for a multi data source (for example, mds-soa). In addition, it will have multiple entries for generic data sources. For example, if you have three Oracle RAC instances on the test 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 production environment. You update the move plan to point the generic data sources to the first three generic data sources in the production environment. Then, you can add an additional data source, as described in [Section 10.2.1.1](#).
- If you are moving from a test environment that uses Oracle RAC to a production environment that uses Oracle RAC, but you have fewer Oracle RAC instances in the production environment, the move plan will have one entry for a multi data source (for example, mds-soa). In addition, it will have multiple entries for generic data sources. For example, if you have four Oracle RAC instances on the test 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 production environment. You update the move plan to point the first three generic data sources to the three generic data sources in the production environment. You point the last generic data source to the third generic data source.

Part IX

Appendixes

This part contains the following appendixes:

- [Appendix A, "Oracle Fusion Middleware Command-Line Tools"](#)
- [Appendix B, "URLs for Components"](#)
- [Appendix C, "Port Numbers"](#)
- [Appendix D, "Metadata Repository Schemas"](#)
- [Appendix E, "Using Oracle Fusion Middleware Accessibility Options"](#)
- [Appendix F, "Examples of Administrative Changes"](#)
- [Appendix G, "Viewing Release Numbers"](#)
- [Appendix H, "Oracle Wallet Manager and orapki"](#)
- [Appendix I, "Troubleshooting Oracle Fusion Middleware"](#)

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: <i>MW_HOME</i> /wlsserver_ <i>n</i> /server/adr Windows: <i>MW_HOME</i> \wlsserver_ <i>n</i> \server\adr	Package incident and problem information into a zip file for transmission to Oracle Support.
bulkdelete	UNIX: <i>ORACLE_HOME</i> /ldap/bin/bulkdelete.sh Windows: <i>ORACLE_HOME</i> \ldap\bin\bulkdelete.bat	Delete a subtree efficiently in Oracle Internet Directory. See: <i>Oracle Fusion Middleware Reference for Oracle Identity Management</i>
bulkload	UNIX: <i>ORACLE_HOME</i> /ldap/bin/bulkload.sh Windows: <i>ORACLE_HOME</i> \ldap\bin\bulkload.bat	Create Oracle Internet Directory entries from data residing in or created by other applications. See: <i>Oracle Fusion Middleware Reference for Oracle Identity Management</i>
bulkmodify	UNIX: <i>ORACLE_HOME</i> /bin/bulkmodify Windows: <i>ORACLE_HOME</i> \bin\bulkmodify	Modify a large number of existing Oracle Internet Directory entries in an efficient way. See: <i>Oracle Fusion Middleware Reference for Oracle Identity Management</i>
catalog	UNIX: <i>ORACLE_HOME</i> /ldap/bin/catalog.sh Windows: <i>ORACLE_HOME</i> \ldap\bin\catalog.bat	Add and delete catalog entries in Oracle Internet Directory. See: <i>Oracle Fusion Middleware Reference for Oracle Identity Management</i>
chgiphost	UNIX: <i>ORACLE_HOME</i> /chgip/scripts/chpiphost.sh Windows: <i>ORACLE_HOME</i> \chgip\scripts\chpiphost.bat	Changes the network configuration of Oracle HTTP Server and Oracle Web Cache. See: Section 15.1.2
config	UNIX: <i>ORACLE_HOME</i> /common/bin/config.sh Windows: <i>ORACLE_HOME</i> \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: <i>ORACLE_HOME</i> /bin/eulbuilder.jar Windows: <i>ORACLE_HOME</i> \bin\eulbuilder.jar	Discoverer EUL Java command-line interface. Create and manipulate Discoverer EULs without installing Oracle Discoverer Administrator. See: <i>Oracle Business Intelligence Discoverer EUL Command Line for Java User's Guide</i>
iasua	UNIX: <i>ORACLE_HOME</i> /upgrade/iasua.sh Windows: <i>ORACLE_HOME</i> \upgrade\iasua.bat	Oracle Fusion Middleware Upgrade Assistant. See: <i>Oracle Fusion Middleware Upgrade Planning Guide</i>
frmcmp	UNIX: <i>ORACLE_HOME</i> /bin/frmcmp.sh Windows: <i>ORACLE_HOME</i> \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: <i>ORACLE_HOME</i> /bin/ldapadd Windows: <i>ORACLE_HOME</i> \bin\ldapadd	Add entries, their object classes, attributes, and values to Oracle Internet Directory. <i>See: Oracle Fusion Middleware Reference for Oracle Identity Management</i>
ldapaddmt	UNIX: <i>ORACLE_HOME</i> /bin/ldapaddmt Windows: <i>ORACLE_HOME</i> \bin\ldapaddmt	Add entries, their object classes, attributes, and values to Oracle Internet Directory. Like <i>ldapadd</i> , except supports multiple threads for adding entries concurrently. <i>See: Oracle Fusion Middleware Reference for Oracle Identity Management</i>
ldapbind	UNIX: <i>ORACLE_HOME</i> /bin/ldapbind Windows: <i>ORACLE_HOME</i> \bin\ldapbind	Determine if you can authenticate a client to a server. <i>See: Oracle Fusion Middleware Reference for Oracle Identity Management</i>
ldapcompare	UNIX: <i>ORACLE_HOME</i> /bin/ldapcompare Windows: <i>ORACLE_HOME</i> \bin\ldapcompare	Match attribute values you specify in the command line with the attribute values in the Oracle Internet Directory entry. <i>See: Oracle Fusion Middleware Reference for Oracle Identity Management</i>
ldapdelete	UNIX: <i>ORACLE_HOME</i> /bin/ldapdelete Windows: <i>ORACLE_HOME</i> \bin\ldapdelete	Remove entire entries from Oracle Internet Directory. <i>See: Oracle Fusion Middleware Reference for Oracle Identity Management</i>
ldapmoddn	UNIX: <i>ORACLE_HOME</i> /bin/ldapmoddn Windows: <i>ORACLE_HOME</i> \bin\ldapmoddn	Modify the DN or RDN of an Oracle Internet Directory entry. <i>See: Oracle Fusion Middleware Reference for Oracle Identity Management</i>
ldapmodify	UNIX: <i>ORACLE_HOME</i> /bin/ldapmodify Windows: <i>ORACLE_HOME</i> \bin\ldapmodify	Perform actions on attributes in Oracle Internet Directory. <i>See: Oracle Fusion Middleware Reference for Oracle Identity Management</i>
ldapmodifymt	UNIX: <i>ORACLE_HOME</i> /bin/ldapmodifymt Windows: <i>ORACLE_HOME</i> \bin\ldapmodifymt	Modify several Oracle Internet Directory entries concurrently. <i>See: Oracle Fusion Middleware Reference for Oracle Identity Management</i>
ldapsearch	UNIX: <i>ORACLE_HOME</i> /bin/ldapsearch Windows: <i>ORACLE_HOME</i> \bin\ldapsearch	Search and retrieve specific entries in Oracle Internet Directory. <i>See: Oracle Fusion Middleware Reference for Oracle Identity Management</i>
ldifmigrator	UNIX: <i>ORACLE_HOME</i> /bin/ldifmigrator Windows: <i>ORACLE_HOME</i> \bin\ldifmigrator.bat	Migrate data from application-specific repositories into Oracle Internet Directory. <i>See: Oracle Fusion Middleware Reference for Oracle Identity Management</i>
ldifwrite	UNIX: <i>ORACLE_HOME</i> /bin/ldifwrite Windows: <i>ORACLE_HOME</i> \bin\ldifwrite.bat	Convert to LDIF all or part of the information residing in an Oracle Internet Directory. <i>See: Oracle Fusion Middleware Reference for Oracle Identity Management</i>
oidcmprec	UNIX: <i>ORACLE_HOME</i> /bin/oidcmprec Windows: <i>ORACLE_HOME</i> \bin\oidcmprec	Compare one Oracle Internet Directory with another, detect conflicts or discrepancies, and optionally resolve them. <i>See: Oracle Fusion Middleware Reference for Oracle Identity Management</i>
oidcred	UNIX: <i>ORACLE_HOME</i> /bin/oidcred Windows: <i>ORACLE_HOME</i> \bin\oidcred	Add, update, or delete a credential that has been created in the Credential Store Framework. <i>See: Oracle Fusion Middleware Reference for Oracle Identity Management</i>

Table A-1 (Cont.) Oracle Fusion Middleware Command-Line Tools

Command	Path	Description
oidctl	UNIX: <i>ORACLE_HOME</i> /bin/oidctl Windows: <i>ORACLE_HOME</i> \bin\oidctl	Start and stop Oracle Internet Directory. See: <i>Oracle Fusion Middleware Reference for Oracle Identity Management</i>
oiddiag	UNIX: <i>ORACLE_HOME</i> /bin/oiddiag Windows: <i>ORACLE_HOME</i> \bin\oiddiag	Collects diagnostic information for Oracle Internet Directory. See: <i>Oracle Fusion Middleware Reference for Oracle Identity Management</i>
oidmon	UNIX: <i>ORACLE_HOME</i> /bin/oidmon Windows: <i>ORACLE_HOME</i> \bin\oidmon	Monitor Oracle Internet Directory processes. See: <i>Oracle Fusion Middleware Reference for Oracle Identity Management</i>
oidpasswd	UNIX: <i>ORACLE_HOME</i> /bin/oidpasswd Windows: <i>ORACLE_HOME</i> \bin\oidpasswd	Change the Oracle Internet Directory password and otherwise restricts access for Oracle Internet Directory See: <i>Oracle Fusion Middleware Reference for Oracle Identity Management</i>
oidprovtool	UNIX: <i>ORACLE_HOME</i> /bin/oidprovtool Windows: <i>ORACLE_HOME</i> \bin\oidprovtool.bat	Administer provisioning profile entries in Oracle Internet Directory. See: <i>Oracle Fusion Middleware Reference for Oracle Identity Management</i>
oidrealm	UNIX: <i>ORACLE_HOME</i> /bin/oidrealm Windows: <i>ORACLE_HOME</i> \bin\oidrealm.bat	Create multiple realms in Oracle Internet Directory. See: <i>Oracle Fusion Middleware Reference for Oracle Identity Management</i>
oidstats	UNIX: SQL command, oidstats.sql Windows: SQL command, oidstats.sql	Analyze the various database ods schema objects to estimate statistics. See: <i>Oracle Fusion Middleware Reference for Oracle Identity Management</i>
opmnctl	UNIX: <i>ORACLE_INSTANCE</i> /bin/opmnctl.exe Windows: <i>ORACLE_INSTANCE</i> \bin\opmnctl.exe	Start, stop, and get status on OPMN-managed processes. See: <i>Oracle Fusion Middleware Oracle Process Manager and Notification Server Administrator's Guide</i>
orapki	UNIX: <i>ORACLE_HOME</i> /bin/orapki Windows: <i>ORACLE_HOME</i> \bin\orapki.bat	Manages wallets and certificates. See Appendix H .
remtool	UNIX: <i>ORACLE_HOME</i> /ldap/bin/remtool Windows: <i>ORACLE_HOME</i> \ldap\bin\remtool	Search for problems and seek to rectify them in the event of an Oracle Internet Directory replication failure. See: <i>Oracle Fusion Middleware Reference for Oracle Identity Management</i>
rwbuilder	UNIX: <i>ORACLE_HOME</i> /bin/rwbuilder Windows: <i>ORACLE_HOME</i> \bin\rwbuilder	Invoke the Reports Builder. See: <i>Oracle Fusion Middleware Publishing Reports to the Web with Oracle Reports Services</i>
rwsgi	UNIX: <i>ORACLE_HOME</i> /bin/rwsgi Windows: <i>ORACLE_HOME</i> \bin\rwsgi	Like <i>rwservlet</i> , translate and deliver information between HTTP and the Reports Server. The <i>rwservlet</i> command is the recommended choice; <i>rwsgi</i> is maintained only for backward compatibility. See: <i>Oracle Fusion Middleware Publishing Reports to the Web with Oracle Reports Services</i>
rwclient	UNIX: <i>ORACLE_HOME</i> /bin/rwclient Windows: <i>ORACLE_HOME</i> \bin\rwclient	Parse and transfer a command line to the specified (or default) Reports Server. See: <i>Oracle Fusion Middleware Publishing Reports to the Web with Oracle Reports Services</i>

Table A-1 (Cont.) Oracle Fusion Middleware Command-Line Tools

Command	Path	Description
rwconverter	UNIX: <i>ORACLE_HOME</i> /bin/rwconverter Windows: <i>ORACLE_HOME</i> \bin\rwconverter	Convert one or more report definitions or PL/SQL libraries from one storage format to another. See: <i>Oracle Fusion Middleware Publishing Reports to the Web with Oracle Reports Services</i>
rwrun	UNIX: <i>ORACLE_HOME</i> /bin/rwrun Windows: <i>ORACLE_HOME</i> \bin\rwrun	Run a report using the Oracle Reports Services in-process server. See: <i>Oracle Fusion Middleware Publishing Reports to the Web with Oracle Reports Services</i>
rwserver	UNIX: <i>ORACLE_HOME</i> /bin/rwserver Windows: <i>ORACLE_HOME</i> \bin\rwserver.bat	Invoke the Reports Server. See: <i>Oracle Fusion Middleware Publishing Reports to the Web with Oracle Reports Services</i>
ssocfg	UNIX: sso/bin/ssocfg.sh Windows: sso\bin\ssocfg.bat	Update host, port, and protocol of Oracle Single Sign-On URL. See: <i>Oracle Fusion Middleware Administrator's Guide for Oracle Single Sign-On</i> , Release 10.1.3.4
ssoconf.sql	UNIX: <i>ORACLE_HOME</i> /portal/admin/plsql/sso/ssoconf.sql Windows: <i>ORACLE_HOME</i> \portal\admin\plsql\sso\ssoconf.sql	Script to point Oracle Single Sign-On server to a different Oracle Internet Directory. See: <i>Oracle Fusion Middleware Administrator's Guide for Oracle Single Sign-On</i> Release 10.1.3.4
wlst	UNIX: <i>WLS_HOME</i> /common/bin/wlst.sh Windows: <i>WLS_HOME</i> \common\bin\wlst.cmd UNIX: <i>ORACLE_HOME_for_component</i> /common/bin/wlst.sh Windows: <i>ORACLE_HOME_for_component</i> \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 <i>Oracle Fusion Middleware WebLogic Scripting Tool Command Reference</i>

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 Imaging and Process Management	http://host:16000/imaging	First user to log in to Oracle I/PM
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	<code>http://host:http_listen_port/reports/rwservlet</code>	<code>orcladmin</code> The default password is the same as the <code>weblogic</code> password of the <i>Infrastructure</i> instance used by Oracle Reports.
Oracle WebCenter Discussions Server	<code>http://host:8890/owc_discussions</code>	<code>weblogic</code>
Oracle WebCenter Spaces	<code>http://host:8888/webcenter</code>	<code>weblogic</code>
Oracle WebCenter Portlets	<code>http://host:8889/richtextportlet/info</code> <code>http://host:8889/wsrp-tools/info</code> <code>http://host:8889/portalTools</code>	<code>weblogic</code>
Oracle WebCenter Wiki and Blogs Server	<code>http://host:8890/owc_wiki</code>	<code>weblogic</code>
Oracle WebLogic Server Administration Console	<code>http://host:7001/console</code>	<code>weblogic</code>

Port Numbers

This appendix provides information about Oracle Fusion Middleware port numbers. It contains the following topics:

- [Port Numbers by Component](#)
- [Port Numbers \(Sorted by Number\)](#)

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 Business Activity Monitoring	9001	9000-9080
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
Oracle HTTP Server SSL Listen Port	4443	4443-4543
Oracle Imaging and Process Management	16000	16000

Table C-1 (Cont.) Port Numbers Sorted by Component

Component or Service	Default Port Number	Allotted Port Range
Oracle Identity Federation Server Managed Server	7499	7499-9000
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 Discussions Server	8890	8881-8890
Oracle WebCenter Portlets	8889	8881-8890
Oracle WebCenter Spaces	8888	8881-8890
Oracle WebCenter Wiki and Blog 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
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)

Table C-2 (Cont.) Port Numbers Sorted by Number

Default Port Number	Component or Service
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 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
8888	Oracle HTTP Server, Oracle WebCenter Spaces
8889	Oracle WebCenter Portlets
8890	Oracle WebCenter Discussions Server and Oracle WebCenter Wiki and Blog Server
9001	Oracle Business Activity Monitoring Managed Server
9001	Oracle Forms Services Managed Server
9001	Oracle Portal Managed Server
9001	Oracle Reports Managed Server
16000	Oracle Imaging and Process Management

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 Manager	OAM	Contains information for Oracle Access Manager.
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	BISERVER BISCHEDULER BISCORECARD BIPUBLISHER	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 11g.
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 Identity Federation	OIF	Contains metadata for Oracle Identity Federation.
Oracle Identity Manager	OIM	Contains metadata for applications that use Oracle Identity Manager.
Oracle Imaging and Process Management	IPM	Contains metadata for Oracle Imaging and Process Management.
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 Service	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 Universal Content Management	OCS OCSSEARCH	Contains metadata for Oracle Universal Content Management.
Oracle Universal Records Management	URMSERVER	Contains metadata for Oracle Universal Records Management
Oracle User Messaging	ORASDPM	Contains metadata related to User Messaging.

Table D-1 (Cont.) Metadata Schemas Created by Repository Creation Utility

Component	Schema	Description
Oracle Web Services Manager	MDS	Contains configuration information.
Oracle WebCenter	WEBCENTER MDS	Contains information for Oracle WebCenter.
Oracle WebCenter Discussions	DISCUSSIONS DISCUSSIONS_ CRAWLER	Contains information for WebCenter Discussions.
Oracle WebCenter Portlets	PORTLET	Contains information for WebCenter Portlet Producers.
Oracle WebCenter Spaces	WEBCENTER MDS	Contains information for WebCenter Services Links, Lists, Tags, and Events.
Oracle WebCenter Wiki and Blog Server	WIKI	Contains information for WebCenter Wiki and Blogs 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
OAM	OAM	oam.dbf
BIPUBLISHER	BIPUBLISHER	BIPUBLISHER.dbf
BISCHEDULER	BISCHEDULER	bischeduler.dbf
BISCORECARD	BISCORECARD	biscorecard.dbf
BISERVER	BISERVER	biserver.dbf
DISCOVERER	DISCO_PTMS_META	discoptm5meta.dbf
	DISCO_PTMS_CACHE	discoptm5cache.dbf
	DISCO_PSTORE	discopstore.dbf
DISCUSSIONS	IAS_DISCUSS	iasjive.dbf
DISCUSSIONS_CRAWLER	IAS_DSCRAWL	iasjivecrawl.dbf
IPM	IPM	ipm.dbf
MDS	MDS	iasmds.dbf
OAAM	BRSADATA	brsdatan.dbf
OAAM_PARTN	OAAM_DATA	partn_brsdatan.dbf
OAM	OAM	oam.dbf
OCS	OCS	ocs.dbf
OCSSEARCH	OCSSEARCH	ocssearch.dbf
ODI_REPO	ODI_USER	prefix_odi_user_n.dbf

Table D–2 (Cont.) Metadata Repository Tablespaces and Data Files

Schema	Tablespace	Default Data File
ODS	OLTS_DEFAULT	default1_oid.dbf
ODSERVER	ODSERVER	odserver.dbf
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	iasdpm.dbf
	IAS_ORASDPM_AQ	iasdpmmq.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
WIKI	IAS_WIKI	iaswiki.dbf

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://java.sun.com/javase/technologies/accessibility/accessbridge/>
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

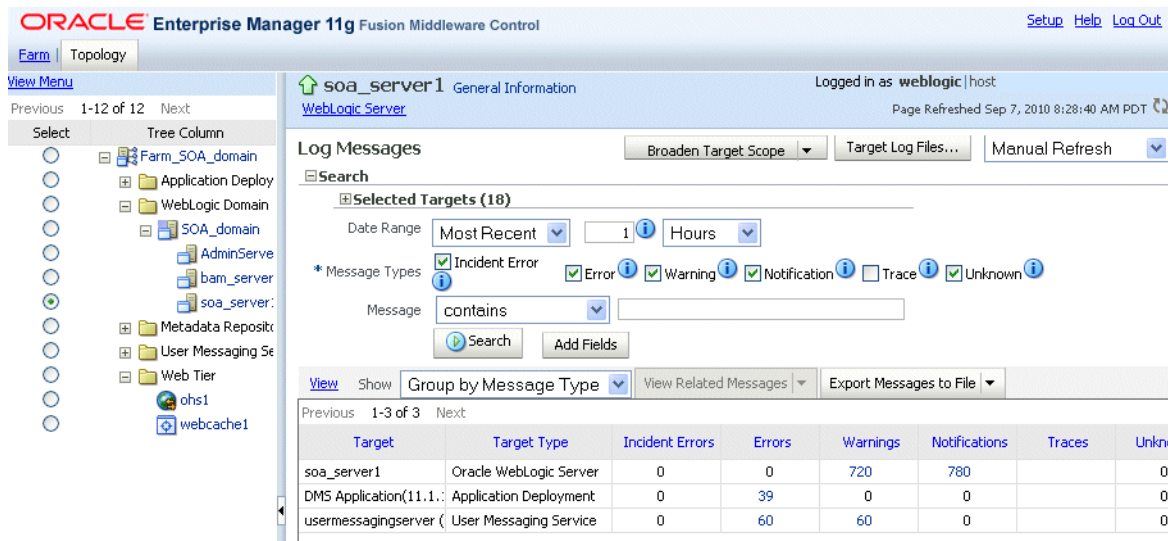
accessibility-specific constructs to the HTML, and by altering some navigation elements on the pages.

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

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:

- [How to Use This Appendix](#)
- [Examples of Administrative Changes \(by Component\)](#)

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 <code>ldapsearch</code> utility
Dynamic Monitoring Service (DMS)	Manual edits to DMS configuration files, such as <code>dms.conf</code>
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 <code>mod_wl_ohs</code> , 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 <code>opmnctl</code> utility
Oracle Internet Directory	Oracle Internet Directory administrative and configuration operations, such as running the <code>oidpasswd</code> 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 <code>ptlconfig</code> 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. <i>Note: Oracle recommends that you perform backup and file synchronization more frequently when running Oracle Reports Services.</i>
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

Viewing Release Numbers

This appendix describes how to view Oracle Fusion Middleware release numbers.

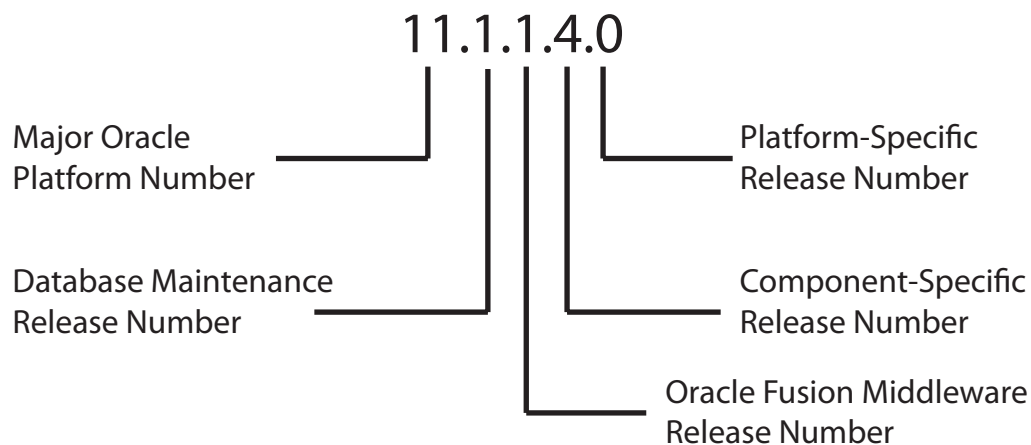
This appendix contains the following topics:

- [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 Oracle Universal Installer, as follows:

1. Launch Oracle Universal Installer:

```
(UNIX) ORACLE_HOME/oui/bin/runInstaller.sh  
(Windows) ORACLE_HOME\oui\bin\setup.exe
```

2. Click **Installed Products** to open the Inventory Page.
3. In the Inventory Page, expand **Oracle Homes**. The entries for all installations on your host are displayed.
4. Expand the Oracle home entry for the installation you are interested in.
5. An entry with the release number for your original installation, followed by entries for any patch sets that have been applied, are displayed.

G.2.2 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 in the following ways:

- [On the File System](#)
- [Using Oracle Universal Installer](#)

On the File System

You can view component release numbers as follows on UNIX:

```
cd ORACLE_HOME/inventory  
ls -d Components**/*
```

Using Oracle Universal Installer

If you installed Oracle Fusion Middleware using Oracle Universal Installer, you can view component release numbers as follows:

1. Launch Oracle Universal Installer:

```
(UNIX) ORACLE_HOME/oui/bin/runInstaller.sh
(Windows) ORACLE_HOME\oui\bin\setup.exe
```

2. Click **Installed Products** to open the Inventory Page.

3. In the Inventory Page, expand **Oracle Homes**. Entries for all installations on your host are displayed.

4. Expand the Oracle home entry for the installation you are interested in.

5. An entry with the release number for your original installation, followed by entries for any patch sets that have been applied, are displayed.

6. Expand the initial entry to view the component release numbers at installation time. If you have subsequent patch set entries, expand them to see the component release numbers updated for each patch set.

G.2.3 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.2.4 Viewing Metadata Repository Release Numbers

The Metadata Repository is an Oracle Database database that has a release number. This number is updated when you apply a patch set release or upgrade the database.

You can view the Metadata Repository release number 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

Oracle Wallet Manager and orapki

Oracle Application Server 10g 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 10g provided the SSL Configuration Tool.

Oracle Fusion Middleware 11g Release 1 (11.1.1) 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

Use this appendix to learn about `orapki` updates, and to help transition to the new certificate, wallet management, and SSL configuration tools provided in 11g Release 1 (11.1.1). The appendix contains these topics:

- [New orapki Features](#)
- [Using the orapki Utility for Certificate Validation and CRL Management](#)
- [Equivalent Features for Oracle Wallet Manager](#)
- [Equivalent Features for orapki](#)
- [Equivalent Features for the SSL Configuration Tool](#)

See Also: *Oracle Application Server Administrator's Guide* for Release 10g for details of Oracle Wallet Manager and `orapki` usage.

See Also: Doc ID 1226654.1, "How To Create a Wallet via ORAPKI in FMW 11g" at:

<https://support.us.oracle.com/oip/faces/secure/km/DocumentDisplay.jspx?id=1226654.1>

Note: The `orapki` utility is located in the binary directory of Oracle Common home.

H.1 New orapki Features

The `orapki` command-line utility contains these new features in Oracle Fusion Middleware 11g Release 1 (11.1.1):

- [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" at:

<https://support.us.oracle.com/oip/faces/secure/km/DocumentDisplay.jspx?id=1226654.1>

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 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" at:

<https://support.us.oracle.com/oip/faces/secure/km/DocumentDisplay.jspx?id=1226654.1>

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"](#)

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/nzcr1.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 `./nzcr1.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 11g Release 1 (11.1.1).

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 11g Release 1 (11.1.1).

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 <code>orapki</code> command-line tools
Importing a PKCS#7 certificate chain into the wallet	Not supported	Use WLST or <code>orapki</code> command-line tools
Exporting a PKCS#7 certificate chain from the wallet	Not supported	Use WLST or <code>orapki</code> 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 11g Release 1 (11.1.1).

Table H–3 Mapping for orapki Features for Wallets and CRLs

orapki Feature	How Implemented in 11gR1	Notes
Creating a standard PKCS#12 wallet	<code>createWallet()</code>	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 <code>orapki</code> command-line tool
Uploading a wallet to an LDAP Directory	Not supported	Use <code>orapki</code> command-line tool
Downloading a wallet from an LDAP directory	Not supported	Use <code>orapki</code> command-line tool
Deleting a wallet	<code>deleteWallet()</code>	
Changing the wallet password	<code>changeWalletPassword()</code>	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 <code>orapki</code> command line tool
Create, revoke, hash, verify, upload, list, display, delete CRLs	Not supported	Use <code>orapki</code> command line tool

[Table H–4](#) shows the features provided by the `orapki` utility for certificates, and the equivalent commands or options in 11g Release 1 (11.1.1).

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-11g Release 1 (11.1.1) SSL Configuration Tool, and the equivalent commands or options in 11g Release 1 (11.1.1).

Table H-5 Equivalent Features for the SSL Configuration Tool

SSL Configuration Tool	SSL Configuration in 11g Release 1 (11.1.1)
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](#)
- [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:

- [Using a Different Version of Spring](#)
- [ClassNotFoundExceptions When Starting Managed Servers](#)

I.2.1 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.2.2 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 4.2.4](#) for information about resolving this problem.

I.3 Troubleshooting Fusion Middleware Control

The following sections describe problems and issues when using Fusion Middleware Control:

- [Troubleshooting the Display of Performance Metrics and Charts in Fusion Middleware Control](#)
- [Securing the Connection from Fusion Middleware Control to Oracle WebLogic Server Administration Console](#)

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:

- [What Are Agent-Monitored Targets?](#)
- [Setting Monitoring Credentials for All Agent-Monitored Targets in a Farm](#)
- [Changing the Monitoring Credentials for a Specific Agent-Monitored Target](#)
- [Verifying or Changing the Oracle Management Agent URL](#)

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 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/technology/documentation/oem.html>

1.3.1.2 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 1.3.1.3](#).

1.3.1.3 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 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.4.1](#).

See Also: *Oracle Fusion Middleware Release Notes*, available on the Oracle Technology Network:

<http://www.oracle.com/technology/documentation/index.html>

I.4.1 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
```


A

- addCertificateRequest, 6-39
- addSelfSignedCertificate, 6-40
- Administration Server, 2-4, 4-2
 - recovery of, 18-4, 18-18, 18-19
 - recovery of host, 18-17
 - starting and stopping, 4-1
 - without credentials, 4-3
- administration users, 3-7, 3-21
- administrative changes, F-1
- ADR Command Interpreter (ADRCI), 13-5, 13-20
- adrci utility, 13-20
- agent-monitored targets, I-2
 - setting credentials for, I-3
- allotted port range, C-1
- applications
 - base documents, 14-3
 - customizations, 14-3
 - deploying, 10-1, 10-7
 - recovery of, 18-15
 - redeploying, 10-11
 - starting and stopping, 4-6
 - transferring to new repository, 14-18
 - undeploying, 10-10
- applyJRF command, 19-6
- audit policies
 - moving from test to production, 21-16
- authentication
 - SSL and, 6-2
- auto-login wallet, 8-20
- Automatic Diagnostic Repository (ADR), 13-3

B

- backing up files, 17-3
- backup and recovery
 - backup strategies, 16-3
 - creating record of environment, 17-6
 - overview, 16-1
 - restrictions, 16-33
- backups
 - Audit Framework and, 17-3
 - databases and, 17-6
 - domains, 17-4, 17-5
 - full, 16-4, 17-4

- Java components and, 17-4, 17-5
- LDAP data and, 17-2
- limitations, 17-2
- Managed Servers and, 17-5
- Middleware home and, 17-4
- Oracle instance homes, 17-4, 17-6
- OraInventory and, 17-5
- persistent stores and, 17-2
- recommendations, 17-1
- run-time artifacts, 16-5
- system components and, 17-4, 17-6
- types of, 16-4, 17-3
- WebLogic Server configuration files, 16-4

- BIPUBLISHER schema
 - datafile, D-3
 - description, D-2
 - tablespace, D-3
- BISCHEDULER schema
 - datafile, D-3
 - description, D-2
 - tablespace, D-3
- BISCORECARD schema
 - datafile, D-3
 - description, D-2
 - tablespace, D-3
- BISERVER schema
 - datafile, D-3
 - description, D-2
 - tablespace, D-3

- bulkdelete command, A-1
- bulkload command, A-1
- bulkmodify command, A-1
- Business Intelligence
 - schemas for, D-2

C

- catalog command, A-1
- certificate
 - converting to third-party, 8-33
 - deleting, 8-32
 - exporting, 8-30
 - importing, 8-31
 - lifecycle, 8-10, 8-28
 - managing with Fusion Middleware Control, 8-28
 - operations, 8-10

- replacing, 8-18
- requesting, 8-29
- certificate authority, 6-3
- certificate operations, 8-29
- Certificate Revocation, 6-35
- certificate revocation lists, H-9
 - deleting, H-13
 - listing, H-12
 - managing with orapki, H-8
 - renaming, H-10
 - uploading, H-11
 - uploading to LDAP directory, H-10
 - validation and, H-9
 - viewing, H-12
- Certificate Signing Request, 8-12
- changeKeyStorePassword command, 6-40
- changeWalletPassword command, 6-41
- changing IP address, 15-7
- character sets
 - changing for metadata repository, 14-27
- chgiphost command, 15-2, A-1
- ClassNotFound error
 - when starting Managed Servers, I-2
- CLASSPATH environment variable, 3-2, 3-3
- cloneMetadataPartition system MBean, 14-16
- cloning, 20-1
 - introduction, 20-1
 - Java options and, 20-4
 - limitations, 20-35
 - move plans and, 20-3
 - Oracle Internet Directory, 20-22
 - Oracle Virtual Directory, 20-23
 - process, 20-2
 - supported entities, 20-1
- cloning MDS Repository partition, 14-16
- cloning Middleware Home, 20-17
- cloning scripts
 - copyBinary, 20-6, 20-18
 - copyConfig, 20-9, 20-19, 20-21
 - extractMovePlan, 20-12
 - help, 20-4
 - pasteBinary, 20-7, 20-18
 - pasteConfig, 20-13, 20-15, 20-19, 20-22
- clusters, 2-5
 - creating, 19-6
 - monitoring, 11-5
 - recovery of, 18-14, 18-15
- command-line tools, 3-15, A-1
- components
 - recovery of, 18-7, 18-8
 - recovery of host, 18-24
 - starting, 4-5
 - starting and stopping, 4-4
 - stopping, 4-5
 - viewing status, 11-6
- config command
 - invoking the Configuration Wizard, A-1
- configureLogHandler command, 12-15, 12-17, 12-21
- configureSSL command, 6-41
- content pane
 - in Fusion Middleware Control, 3-8

- context pane
 - in Fusion Middleware Control, 3-9
- context root, 10-8
- copyBinary script, 20-6, 20-18
- copyConfig script, 20-9, 20-19, 20-21
- createIncident command, 13-19
- createKeyStore command, 6-42
- createMetadataLabel command, 14-24
- createMetadataPartition command, 14-18, 14-21
- createWallet command, 6-43
- CRL
 - configuring for validation, 6-35
 - creation, 6-35
 - renaming to hashed form, 6-34
- CRL integration, 6-33
- CRLAdmins directory administrative group, H-17
- cryptography
 - private key, 6-2
 - public key, 6-3

D

- dads.conf file, 5-14
- data sources
 - configuring, 10-3
 - creating, 10-4
 - managing, 10-5, 10-6
 - monitoring, 10-6
- database-based repository
 - creating, 14-2
 - starting, 4-7, 4-8
- databases
 - backing up, 17-6
 - recovery of, 18-17, 18-47
- DB2 databases
 - MDS and, 14-5
- default port numbers, C-1
- deleteKeyStore command, 6-43
- deleteMetadataLabel command, 14-27
- deleteMetadataPartiton command, 14-22
- deleteWallet command, 6-44
- deploy command, 10-10, 10-16
- deploying applications, 10-7
 - application security, 10-27
 - configure EJBs, 10-27
 - configure persistence, 10-28
 - configure Web modules, 10-27
 - overview, 10-1
- deployment plan
 - fetching, 10-26
- deployment plans
 - creating automatically, 10-8, 10-12, 10-13, 10-18, 10-22, 10-25
 - managing, 10-26
- deployment profile, 9-4
- deregisterMetadataDBRepository command, 14-11
- DER-encoded certificates, 8-3
- describeDump command, 13-18
- DHCP addresses

- changing, 15-7
- moving to, 15-7
- diagnostic dumps, 13-5
- Diagnostic Framework, 13-1, 13-3, I-1
 - configuring, 13-9
 - diagnostic rules, 13-2
 - first-fault capture, 13-2
 - incident detection log filter, 13-2
 - incident flood control, 13-3
 - incidents, 13-3
 - invoking WLST, 3-17
 - MBean, 13-9
 - problem keys, 13-3
 - problems, 13-3
- Diagnostic Image, 13-2
- diagnostic messages
 - levels, 12-17
 - types, 12-17
- diagnostics
 - early detection, I-1
 - first-fault capture, 13-2, I-1
 - incident detection log filter, 13-2
 - messages, 12-22
 - troubleshooting, I-1
- disconnected mode
 - monitoring status, 11-1
- Discoverer
 - See* Oracle Business Intelligence Discoverer
- DISCOVERER schema
 - datafile, D-3
 - description, D-2
 - tablespaces, D-3
- DISPLAY environment variable, 3-1
- displayLogs command, 12-8, 12-11
- domain home, 2-3
- domain names
 - changing, 15-1
- domain templates
 - extending domains, 19-2
- domains
 - adding Managed Servers to, 19-4
 - extending, 19-2
 - recovery of, 18-2
 - WebLogic Server, 2-3
- dumps
 - diagnostic, 13-5
 - executing, 13-18
 - viewing list of, 13-17
- Dynamic Monitoring Service (DMS)
 - invoking WLST, 3-17
- dynamic target menu
 - in Fusion Middleware Control, 3-8

E

- ECID
 - See* Execution Context ID (ECID)
- encryption, 6-2
- environment variables
 - setting, 3-1

- ERROR message type, 12-17
- error messages
 - See* diagnostics
- eulbuilder.jar command-line tool, A-1
- executeDump command, 13-18
- Execution Context ID (ECID), 12-22
 - searching log files for, 12-11
- expand tree
 - in Fusion Middleware Control, 3-9
- exportKeyStore command, 6-44
- exportKeyStoreObject command, 6-45
- exportMetadata command, 14-18, 14-20, 14-21
- exportWallet command, 6-46
- exportWalletObject command, 6-47
- extractMovePlan script, 20-12

F

- farm menu
 - in Fusion Middleware Control, 3-8
- farms, 3-6
- first-fault component isolation, 12-22
- flood-control
 - enabling for incidents, 13-9
- flood-controlled incidents
 - Diagnostic Framework, 13-3
- frmcmp command
 - Oracle Forms Services, A-1
- Fusion Middleware Control
 - content pane, 3-8
 - context pane, 3-9
 - dynamic target menu, 3-8
 - expand tree, 3-9
 - farm menu, 3-8
 - recovery of, 18-43
 - refresh page, 3-9
 - right-click target menu, 3-8
 - securing, I-4
 - starting and stopping, 4-5
 - target information icon, 3-9
 - target name, 3-9
 - target navigation pane, 3-8
 - Topology Viewer, 3-9
 - troubleshooting, I-2
 - URL for, 3-6, B-1
 - using, 3-6

G

- generateKey command, 6-48
- getIncidentFile command, 13-15, 13-16
- getKeyStoreObject command, 6-49
- getLogLevel command, 12-20
- getMDSArchiveConfig command, 10-16
- getSSL command, 6-49
- getWalletObject command, 6-50

H

- high availability environments
 - starting and stopping, 4-8

- home pages, 3-6
- host names
 - changing, 15-1
- HTTP port
 - changing, 5-5
- HTTPS port
 - changing, 5-5
- Human Workflow
 - moving from test to production, 21-27

I

- iasua command, A-1
- IMMEDIATE option for Oracle Database shutdown, 4-9
- importKeyStore command, 6-51
- importKeyStoreObject command, 6-51
- importMetadata command, 14-18, 14-21
- importWallet command, 6-52
- importWalletObject command, 6-53
- incident flood control, 13-3
- INCIDENT_ERROR message type, 12-17
- incidents
 - Diagnostic Framework, 13-3
 - enabling creation, 13-9
 - listing, 13-16
 - managing, 13-18
 - packaging, 13-20, 13-21
 - purging, 13-22
 - viewing details, 13-16
- IP addresses
 - changing, 15-1, 15-7
 - metadata repository, 15-3
 - moving off-network, 15-7
 - moving to static address, 15-7
- IPC Listener
 - KEY value, 5-14
- IPM schema, D-2
- IPv4 protocol
 - support for, 15-8
- IPv6 protocol
 - Oracle Access Manager, 15-14
 - Oracle HTTP Server, 15-11
 - Oracle Single Sign-On, 15-12
 - Oracle Web Cache
 - disabling IPv6, 15-12
 - support for, 15-8
 - topologies supported, 15-10

J

- Java components, 2-1
 - recovery of, 18-25
- Java EE application, 10-1
- Java EE applications
 - deploying, 10-7
 - redeploying, 10-11
 - starting and stopping, 4-6
 - undeploying, 10-10
- Java keystore, 8-5

- Java Naming and Directory Interface (JNDI), 10-3
- Java Required Files (JRF)
 - configuring Managed Server for, 19-5
- JAVA_HOME environment variable, 3-2, 3-3
- JKS, 8-5
- JKS keystore, 8-1
 - component using, 6-7
 - lifecycle, 8-5

K

- keystore
 - changing password, 8-9
 - converting self-signed certificate, 8-16
 - creating, 8-6
 - deleting, 8-8
 - deleting certificate, 8-15
 - exporting, 8-7
 - exporting certificate, 8-14
 - generating new key, 8-11
 - importing, 8-9
 - importing certificate, 8-13
 - JKS and Oracle wallet, 6-3
 - location of, 8-18
 - types of, 6-7, 8-1
 - using Fusion Middleware Control, 8-5
- keystore and certificate maintenance, 8-18
- keystore management tools, 8-2

L

- labels
 - creating, 14-24
 - deleting, 14-27
 - listing, 14-24
 - metadata
 - managing, 14-23
 - promoting, 14-25
 - rolling back to, 14-25
- LD_LIBRARY_PATH environment variable, 3-2
- LD_LIBRARY_PATH_64 environment variable, 3-2
- ldapadd command, A-2
- ldapaddmt command, A-2
- ldapcompare command, A-2
- ldapdelete command, A-2
- ldapmoddn command, A-2
- ldapmodify command, A-2
- ldapmodifymt command, A-2
- ldap.ora file
 - directory SSL port for no authentication, H-12
- ldapsearch command, A-2
 - viewing context version, G-4
 - viewing schema version, G-3
- ldifmigrator command, A-2
- LIBPATH environment variable, 3-2
- listDumps command, 13-17
- listen ports
 - changing, 5-5
- listIncidents command, 13-16
- listKeyStoreObjects command, 6-54

- listKeyStores command, 6-55
- listLoggers command, 12-20
- listLogs command, 12-7
- listMetadataLabel command, 14-24
- listProblems command, 13-15
- listWalletObjects command, 6-55
- listWallets command, 6-56
- locking configuration
 - for WebLogic Server, 3-15
- log detection
 - enabling, 13-9
- log files
 - displaying count of messages, 12-12
 - downloading, 12-12
 - formats
 - setting, 12-20
 - levels, 12-17
 - retrieving, 12-20
 - setting, 12-20
 - listing, 12-5
 - locales
 - setting, 12-21
 - location, 12-14
 - naming, 12-14
 - overview, 12-1
 - retention period, 12-16
 - rotation, 12-15
 - size-based, 12-16
 - searching, 12-9, 12-11
 - by component type, 12-11
 - by ECID, 12-11
 - by time, 12-11, 12-12
 - by type of message, 12-11
 - specifying size of, 12-16
 - time-based rotation, 12-16
 - viewing, 12-5
- logging commands
 - invoking WLST, 3-17
- loss of host
 - recovery from, 18-17
 - limitations, 18-43

M

- managed beans
 - See* MBeans
- Managed Servers, 2-4, 4-2
 - adding to domain, 19-4
 - backing up, 17-5
 - recovery of, 18-5, 18-6
 - recovery of host, 18-20, 18-21, 18-22
 - starting and stopping, 4-1, 4-2
 - troubleshooting start problems, I-2
- MBeans
 - for Diagnostic Framework, 13-5
 - searching for, 3-20
 - viewing, 3-19
 - viewing for application, 3-20
- MDS Repository, 14-1, 14-3
 - benefits of database-based repository, 14-3

- changing configuration attributes, 10-29
- configuring application
 - to use different repository, 14-15
- creating database-based, 14-2
- creating labels, 14-24
- DB2 and, 14-5
- deleting labels, 14-27
- deregistering file-based, 14-12
- file-based
 - registering, 14-11
- listing labels, 14-24
- managing, 14-2, 14-6
- moving to database-based, 14-21
- promoting labels, 14-25
- purging labels, 14-25
- purging metadata versions, 14-22
- registering database-based, 14-8
- registering file-based, 14-12
- SQL Server and, 14-5
- supported databases, 14-4
- transferring metadata, 14-18
- versions, 14-3
- viewing, 14-14

- MDS schema
 - datafile, D-3
 - description, D-2
 - tablespace, D-3
- message correlation, 12-22
- message levels, 12-17
- message types, 12-17
- metadata
 - exporting from partition, 14-18
 - importing to partition, 14-18
 - transferring to new partition, 14-18
- Metadata Archive (MAR), 10-2, 14-2
- metadata labels
 - creating, 14-24
 - deleting, 14-27
 - listing, 14-24
 - managing, 14-23
 - promoting, 14-25
 - purging, 14-25
 - rolling back to, 14-25
- metadata repository, 2-6, 14-1
 - changing characters sets, 14-27
 - ports, changing, 5-10
 - release numbers, G-4
 - schemas
 - changing passwords, 14-27
 - schemas for components, D-1
 - starting, 4-6, 4-7, 4-8
 - stopping, 4-7
 - version numbers, G-4
- metrics
 - troubleshooting, I-2
- Middleware Home, 2-5
 - backing up, 17-4
 - cloning, 20-17
 - recovery of, 18-2
- migrating to a production environment, 21-1

- mod_osso
 - port numbers and, 5-6
- monitoring status, 11-1
- move plans
 - for cloning, 20-3
- moving from test to production environment, 21-1
 - audit policies, 21-16
 - Oracle Access Manager, 21-12
 - Oracle Adaptive Access Manager, 21-15
 - Oracle BI Discoverer, 21-65
 - Oracle Business Process Management, 21-32
 - Oracle Data Integrator, 21-94
 - Oracle Directory Integration Platform, 21-7
 - Oracle Enterprise Content Management Suite, 21-83
 - Oracle Forms Services, 21-61
 - Oracle HTTP Server, 21-53
 - Oracle Identity Federation, 21-7
 - Oracle Identity Management, 21-4
 - Oracle Identity Manager, 21-8
 - Oracle Identity Navigator, 21-11
 - Oracle Imaging and Process Management, 21-87, 21-93
 - Oracle Information Rights Management, 21-85, 21-91
 - Oracle Internet Directory, 21-7
 - Oracle Platform Security, 21-16
 - Oracle Portal, 21-60
 - Oracle Reports, 21-63
 - Oracle SOA Suite, 21-24
 - Oracle Universal Content Management, 21-85, 21-92
 - Oracle Universal Records Management, 21-90, 21-93
 - Oracle Virtual Directory, 21-7
 - Oracle Web Cache, 21-55
 - Oracle Web Center, 21-44
 - Oracle Web Services Manager, 21-16
- multiple installations on one host, 3-3
- MW_HOME environment variable, 3-2, 3-3

N

- navigation pane
 - in Fusion Middleware Control, 3-8
- Net Listener
 - starting, 4-7
- network configuration
 - changing, 15-1
 - Oracle HTTP Server, 15-2
 - Oracle Web Cache, 15-2
 - Oracle WebLogic Server, 15-1
- Node Manager, 2-5
 - configuring to enable scripts, 4-3
- NOTIFICATION message type, 12-17

O

- OAAM schema, D-1
- OAAM_OFFLINE schema, D-1

- OAAM_PARTN schema, D-1
- OCS schema, D-2
- ODI_REPO schema, D-2
- ODL
 - See* Oracle Diagnostic Logging (ODL)
- ODL Archives, 12-15
- ODL log, 12-15
- offline backup, 16-4
- off-network
 - moving on-network
 - DHCP address, 15-7
 - static IP address, 15-7
- OID schema
 - datafile, D-4
 - description, D-2
 - tablespace, D-4
- oidcmprec command, A-2
- oidctl command, A-2, A-3
- oiddiag command, A-3
- oidmon command, A-3
- oidprovtool command, A-3
- oidstats command, A-3
- OIM schema
 - datafile, D-4
 - tablespace, D-4
- online backup, 16-4
- on-network
 - moving off-network
 - IP address, 15-7
- ONS local port
 - changing, 5-7
- ONS remote port
 - changing, 5-7
- ONS request port
 - changing, 5-7
- OPMN
 - See* Oracle Process Manager and Notification Server (OPMN)
- opmnctl commands, 3-18, A-3
 - registerinstance, 18-4, 18-28, 18-29, 18-32, 18-34, 18-35, 18-37, 21-56
 - restartproc, 4-5
 - startall, 4-5
 - startproc, 4-5
 - status, 3-18, 11-2
 - stopall, 4-5
 - stopproc, 4-5
 - updatecomponentregistration, 5-13, 18-26, 18-35
 - updateinstanceregistration, 18-26, 18-27, 18-32, 18-35
- opmn.xml file
 - ports and, 5-7
- ORABAM schema
 - datafile, D-4
 - description, D-1
 - tablespaces, D-4
- Oracle Access Manager
 - backup and recovery recommendations, 16-13
 - IPV6 support, 15-14
 - moving from test to production, 21-12

- recovery of, 18-10, 18-30
- Oracle Adaptive Access Manager
 - backup and recovery recommendations, 16-14
 - moving from test to production, 21-15
 - recovery of, 18-10, 18-30
 - schemas for, D-1
- Oracle Application Development Framework
 - applications, 10-1
 - invoking WLST, 3-17
- Oracle B2B
 - backup and recovery recommendations, 16-17
 - moving from test to production, 21-25
 - schemas for, D-1
- Oracle BI Intelligence Enterprise Edition
 - See* Oracle Business Intelligence
- Oracle BPEL Process Manager
 - backup and recovery recommendations, 16-16
 - schemas for, D-1
- Oracle Business Activity Monitoring
 - backup and recovery recommendations, 16-16
 - moving from test to production, 21-30
 - schemas for, D-1
- Oracle Business Intelligence, 1-3
 - backup and recovery recommendations, 16-28
 - moving from test to production, 21-69
 - recovery of, 18-11, 18-38
- Oracle Business Intelligence Discoverer
 - backup and recovery recommendations, 16-27
 - command-line tool, A-1
 - moving from test to production, 21-58, 21-65
 - recovery of, 18-37
 - schemas for, D-2
- Oracle Business Intelligence Publisher
 - backup and recovery recommendations, 16-29
 - moving from test to production, 21-69
 - recovery of, 18-12, 18-41
- Oracle Business Process Management
 - backup and recovery recommendations, 16-19
 - moving from test to production, 21-32
 - recovery of, 18-10
 - schemas for, D-2
- Oracle Business Rules
 - backup and recovery recommendations, 16-18
 - schemas for, D-2
- Oracle Common home, 2-6
- Oracle Content Server
 - backup and recovery recommendations, 16-22
 - schemas for, D-2
- Oracle Data Integrator
 - backup and recovery recommendations, 16-30
 - moving from test to production, 21-94
 - recovery of, 18-13, 18-41
 - schema for, D-2
- Oracle Database
 - immediate shutdown, 4-9
 - recovery of, 18-47
- Oracle Diagnostic Logging (ODL), 12-1
 - message format, 12-2
 - message header fields, 12-2
- Oracle Directory Integration Platform
 - backup and recovery recommendations, 16-12
 - moving from test to production, 21-3, 21-7
 - recovery of, 18-28
 - schemas for, D-2
- Oracle Directory Services Manager
 - backup and recovery recommendations, 16-13
- Oracle Enterprise Content Management Suite, 1-3
 - backup and recovery recommendations, 16-31
 - moving from test to production, 21-83
- Oracle Enterprise Manager Fusion Middleware Control
 - See* Fusion Middleware Control
- Oracle Event Processing
 - schemas for, D-2
- Oracle Forms Services
 - backup and recovery recommendations, 16-25
 - moving from test to production, 21-58, 21-61
 - recovery of, 18-34
- Oracle Fusion Middleware
 - overview, 2-1
- Oracle Fusion Middleware Audit Framework, 17-3
 - invoking WLST, 3-17
- Oracle Fusion Middleware environment
 - starting, 4-6
 - stopping, 4-7
- Oracle Fusion Middleware Upgrade Assistant, A-1
- Oracle home, 2-6
 - recovery of, 18-3
- Oracle HTTP Server, 1-2
 - backup and recovery recommendations, 16-24
 - changing network configuration, 15-2
 - IPv6 support, 15-11
 - moving from test to production, 21-53
 - ports
 - changing listen, 5-4, 5-5
 - changing SSL listen, 5-5
 - less than 1024, 5-4
 - recovery of, 18-31
 - URL for, B-1
- Oracle Identity Federation, 1-3
 - backup and recovery recommendations, 16-13
 - moving from test to production, 21-7
 - recovery of, 18-28
 - schemas for, D-2
- Oracle Identity Management, 1-2
 - backup and recovery recommendations, 16-11
 - moving from test to production, 21-4
 - starting, 4-6
- Oracle Identity Manager
 - backup and recovery recommendations, 16-14
 - moving from test to production, 21-8
 - recovery of, 18-9, 18-29
 - schemas for, D-2
- Oracle Identity Navigator
 - backup and recovery recommendations, 16-15
 - moving from test to production, 21-11
 - recovery of, 18-10, 18-29
- Oracle Imaging and Process Management
 - backup and recovery recommendations, 16-32
 - moving from test to production, 21-87, 21-93

- recovery of, 18-13
- schemas for, D-2
- Oracle Information Rights Management
 - backup and recovery recommendations, 16-31
 - moving from test to production, 21-85, 21-91
 - recovery of, 18-13
 - schemas for, D-2
- Oracle instances, 2-5
 - environment variable, 3-2, 3-3
 - recovery of, 18-3
 - viewing log files, 12-7
 - viewing status, 3-18
- Oracle Internet Directory, 1-2
 - adding entries, A-2
 - administering provisioning entries, A-3
 - authenticating client, A-2
 - backup and recovery recommendations, 16-11
 - catalog entries, A-1
 - cloning, 20-22
 - comparing, A-2
 - comparing attribute values, A-2
 - creating entries in, A-1
 - deleting entries, A-2
 - deleting subtree in, A-1
 - diagnostic tool, A-3
 - Diffie-Hellman SSL port, H-12
 - estimating statistics, A-3
 - migrating data, A-2
 - modifying entries, A-1, A-2
 - monitoring, A-3
 - moving from test to production, 21-3, 21-7
 - ports
 - updating, 5-13
 - recovery of, 18-27
 - release numbers, G-3
 - replication tool, A-3
 - schemas for, D-2
 - searching entries, A-2
 - starting and stopping, A-2, A-3
 - version numbers, G-3
- Oracle Inventory
 - updating for recovery, 18-46
- Oracle JRF, 19-5
 - applying, 19-5
 - backup and recovery recommendations, 16-22
 - invoking JRF, 3-17
- Oracle Management Agent
 - changing URL, I-4
 - recovery of, 18-44
- Oracle Mediator
 - schemas for, D-2
- Oracle Metadata Services
 - invoking WLST, 3-17
 - schemas for, D-2
- Oracle Platform Security Services, 1-3
 - backup and recovery recommendations, 16-23
 - invoking WLST, 3-17
 - moving from test to production, 21-16
- Oracle Portal, 1-3
 - backup and recovery recommendations, 16-25
 - moving from test to production, 21-58, 21-60
 - ports
 - changing, 5-8
 - recovery of, 18-32
 - schemas for, D-2
- Oracle Process Manager and Notification Server (OPMN), 3-18, A-3
 - ports
 - changing, 5-7
- Oracle Real-Time Decisions
 - backup and recovery recommendations, 16-30
 - moving from test to production, 21-79
 - recovery of, 18-12, 18-41
 - schema for, D-2
- Oracle Reports
 - backup and recovery recommendations, 16-26
 - moving from test to production, 21-58, 21-63
 - recovery of, 18-35
- Oracle Service Bus
 - backup and recovery recommendations, 16-17
- Oracle Single Sign-On
 - changing Oracle Internet Directory, A-4
 - IPv6 support, 15-12
 - ports, updating, 5-13
 - schema for, D-2
 - updating URL, A-4
- Oracle SOA Suite, 1-1
 - backup and recovery recommendations, 16-15
 - composite application, 10-2
 - moving from test to production, 21-24
 - recovery of, 18-24, 18-30
 - schemas for, D-2
- Oracle Universal Content Management
 - backup and recovery recommendations, 16-32
 - moving from test to production, 21-85, 21-92
 - recovery of, 18-13, 18-42
 - schema for, D-2
- Oracle Universal Records Management
 - backup and recovery recommendations, 16-33
 - moving from test to production, 21-90, 21-93
 - recovery of, 18-14, 18-43
- Oracle User Messaging Service
 - schema for, D-2
- Oracle Virtual Directory, 1-2
 - backup and recovery recommendations, 16-12
 - cloning, 20-23
 - moving from test to production, 21-3, 21-7
 - recovery of, 18-27
- Oracle wallet, 8-2
 - and JKS keystore, H-3
 - auto-login, 8-20
 - changing to third-party, 8-26, 8-37
 - components using, 6-3
 - creating, 8-23
 - deleting, 8-26, 8-27
 - exporting, 8-26
 - importing, 8-27
 - lifecycle, 8-22
 - maintenance, 8-35
 - managing in Fusion Middleware Control, 8-22

- naming conventions, 8-21
- operations, 8-23
- types, 8-20
- Oracle Wallet Manager, H-1
 - equivalent features for, H-20
- Oracle Web Cache, 1-2
 - backup and recovery recommendations, 16-24
 - changing network configuration, 15-2
 - disabling IPV6, 15-12
 - moving from test to production, 21-55
 - ports
 - changing, 5-7
 - recovery of, 18-32
- Oracle Web Services
 - invoking WLST, 3-17
- Oracle Web Services Manager, 1-3
 - backup and recovery recommendations, 16-23
 - invoking WLST, 3-17
 - moving from test to production, 21-16
 - schemas for, D-3
- Oracle WebCenter, 1-1
 - application, 10-2
 - backup and recovery recommendations, 16-19
 - deploying applications, 10-22
 - moving from test to production, 21-44
 - schema for, D-2, D-3
 - schemas for, D-3
- Oracle WebCenter Activities Graph
 - recovery, 18-10
- Oracle WebCenter Analytics
 - backup and recovery recommendations, 16-22
- Oracle WebCenter Discussions
 - schemas for, D-3
- Oracle WebCenter Discussions Server
 - backup and recovery recommendations, 16-20
- Oracle WebCenter Portlets
 - backup and recovery recommendations, 16-20
- Oracle WebCenter Wiki and Blog Server
 - backup and recovery recommendations, 16-21
 - schemas for, D-3
- Oracle WebLogic Scripting Tool (WLST)
 - commands for system components, 3-17
 - custom commands, 3-16
 - See also* WLST commands
- Oracle WebLogic Server, 1-1
 - backing up, 16-4
 - backup and recovery recommendations, 16-8
 - changing network configuration, 15-1
 - changing port numbers, 5-3
 - JMS
 - backup and recovery recommendations, 16-9
- Oracle WebLogic Server Administration
 - Console, 3-14
- ORACLE_HOME environment variable, 3-2, 3-3
- ORACLE_INSTANCE environment variable, 3-2, 3-3
- ORAIRM schema, D-2
 - datafile, D-4
 - tablespace, D-4
- orapki utility, H-1, H-10
 - adding certificate requests, H-7, H-18
 - adding certificates, H-18
 - adding root certificates, H-8
 - adding trusted certificates, H-8
 - adding user certificates, H-8
 - certificate creation, H-14
 - changing wallet password with, H-19
 - commands, H-13
 - creating auto-login wallets with, H-7
 - creating signed certificates, H-6, H-14
 - creating wallets with, H-7, H-19
 - deleting certificate revocation lists, H-14, H-15
 - displaying certificate revocation lists, H-15
 - displaying certificates, H-14
 - displaying help, H-6
 - equivalent features for, H-22
 - exporting certificate requests, H-8
 - exporting certificates, H-8, H-20
 - exporting trust chain, H-20
 - generating CRL hash value, H-16
 - listing certificate revocation lists, H-16, H-17, H-18
 - managing certificate revocation lists, H-8
 - managing wallets with, H-7
 - new features, H-2
 - obtaining certificate status, H-17
 - overview, H-5
 - syntax, H-5
 - uploading certificate revocation lists, H-17
 - usage, H-2
 - verifying CRL signature, H-18
 - viewing certificates, H-6, H-19
 - viewing wallets with, H-7
- ORASDPM schema, D-2
 - datafile, D-4
 - tablespace, D-4
- ORASSO schema
 - datafile, D-4
 - description, D-2
 - tablespaces, D-4

P

- partitions
 - about, 14-4
 - cloning, 14-16
 - creating, 14-18, 14-21
 - deleting, 14-22
 - exporting metadata from, 14-18, 14-20, 14-21
 - importing metadata to, 14-18, 14-21
 - transferring metadata to, 14-18
- password-protected wallet, 8-20
- passwords
 - changing for administrative user, 3-21
- pasteBinary script, 20-7, 20-18
- pasteConfig script, 20-13, 20-15, 20-19, 20-22
- PATH environment variable, 3-2, 3-3
- PKI, 6-2
- port numbers
 - changing, 5-2

- managing, 5-1
- viewing, 5-1
 - with command line, 5-1
 - with Fusion Middleware Control, 5-2
- PORTAL schema
 - datafile, D-4
 - description, D-2
 - tablespace, D-4
- PORTLET schema
 - datafile, D-4
 - description, D-3
 - tablespaces, D-4
- ports
 - changing, 5-2
 - metadata repository, 5-10
 - OPMN, 5-7
 - Oracle HTTP Server, 5-4, 5-5
 - Oracle Portal, 5-8
 - Oracle Web Cache, 5-7
 - WebLogic Server, 5-3
 - managing, 5-1
 - updating
 - Oracle Internet Directory, 5-13
 - Oracle Single Sign-On, 5-13

See also port numbers
- private key cryptography, 6-2
- problem keys
 - Diagnostic Framework, 13-3
- problems
 - Diagnostic Framework, 13-3
- promoteMetadataLabel command, 14-25
- public key cryptography, 6-3
- purgeMetadata command, 14-22, 14-23
- purgeMetadataLabels command, 14-26
- purging data, 14-28
 - MDS Repository, 14-29
 - Oracle Application Development Framework, 14-28
 - Oracle Business Intelligence Publisher, 14-29
 - Oracle SOA Suite, 14-28
 - Oracle Universal Content Management, 14-29
 - Oracle Web Center, 14-30
 - Oracle Web Services
 - JRF data, 14-30
 - Oracle Web Services Manager, 14-29
 - Oracle WebCenter, 14-29
 - Oracle WebCenter Activity Stream, 14-30
 - Oracle WebCenter Analytics, 14-30, 14-32
 - Oracle WebLogic Server, 14-29
 - Oracle WebLogic Services, 14-28, 14-30
 - Web Services, 14-28, 14-30
- purging metadata version history
 - from MDS, 14-22

R

- recovery, 18-1
 - Administration Server, 18-4
 - Administration Server host and, 18-17
 - applications and, 18-15
 - clusters, 18-14
 - components, 18-7
 - components host and, 18-24
 - database, 18-47
 - databases, 18-17, 18-47
 - domain, 18-2
 - Fusion Middleware Control, 18-43
 - Java components, 18-25
 - loss of host, 18-17
 - limitations, 18-43
 - Managed Server, 18-5
 - Managed Server host and, 18-20
 - Middleware Home and, 18-2
 - Oracle Access Manager, 18-10, 18-30
 - Oracle Adaptive Access Manager, 18-10, 18-30
 - Oracle BI Enterprise Edition, 18-11, 18-38
 - Oracle Business Intelligence Discoverer, 18-37
 - Oracle Business Intelligence Publisher, 18-12, 18-41
 - Oracle Business Process Management, 18-10
 - Oracle Data Integrator, 18-13, 18-41
 - Oracle Directory Integration Platform, 18-28
 - Oracle Forms Services, 18-34
 - Oracle home, 18-3
 - Oracle HTTP Server, 18-31
 - Oracle Identity Federation, 18-28
 - Oracle Identity Manager, 18-9, 18-29
 - Oracle Identity Navigator, 18-10, 18-29
 - Oracle Imaging and Process Management, 18-13
 - Oracle Information Rights Management, 18-13
 - Oracle instance home, 18-3
 - Oracle Internet Directory, 18-27
 - Oracle Management Agent, 18-44
 - Oracle Portal, 18-32
 - Oracle Real-Time Decisions, 18-12, 18-41
 - Oracle Reports, 18-35
 - Oracle SOA Suite, 18-30
 - Oracle Universal Content Management, 18-13, 18-42
 - Oracle Universal Records Management, 18-14, 18-43
 - Oracle Virtual Directory, 18-27
 - Oracle Web Cache, 18-32
 - Oracle WebCenter Activities Graph, 18-10
 - recommendations, 18-1
 - strategies, 16-7
 - system components, 18-26
 - Windows Registry, 18-46
- redeploy command, 10-12
- redeploying applications, 10-11
- refresh pages
 - in Fusion Middleware Control, 3-9
- register components
 - updating, 5-13, 18-27, 18-35
- register instance
 - updating, 18-26, 18-32
- registerinstance command, 18-4, 18-28, 18-29, 18-32, 18-34, 18-35, 18-37, 21-56
- registerMetadataDBRepository command, 14-10
- Relationship ID (RID), 12-22

- release numbers
 - application server, G-2
 - component, G-2
 - format, G-1
 - metadata repository, G-4
 - Oracle Internet Directory, G-3
 - viewing, G-2
- Remote Diagnostic Agent (RDA), 13-22, I-5
- removeKeyStoreObject command, 6-56
- removeWalletObject command, 6-57
- remtool command, A-3
- Repository Creation Utility (RCU)
 - using, 14-2
- right-click target menu
 - in Fusion Middleware Control, 3-8
- roles, 3-10
 - MDS Repository and, 14-7
- RTD schema
 - datafile, D-4
 - description, D-2
 - tablespaces, D-4

S

- scalability, 19-1
- schemas
 - database-based repository
 - managing, 14-27
 - for components, D-1
- Secure Sockets Layer
 - See* SSL
- security, 6-1
- self-signed certificate, 8-16
- setAppMetadataRepository command, 10-16
- setLogLevel command, 12-20
- setNMProps script, 4-4
- SHLIB_PATH environment variable, 3-2
- showIncident command, 13-16
- SHUTDOWN IMMEDIATE, 4-9
- SOAINFRA schema
 - datafile, D-4
 - description, D-1, D-2
 - tablespaces, D-4
- software inventory
 - viewing, G-2
- Spring
 - using different version, I-1
- SQL Server databases
 - MDS and, 14-5
- SSL, 6-1
 - authentication modes, 6-8
 - best practices, 6-37
 - certificate lifecycle, 8-10
 - client-side, 6-23
 - concepts, 6-2
 - configuring, 6-1
 - with script, 7-1
 - CRL integration, 6-33
 - data sources on Oracle WebLogic Server, 6-31
 - data tier, 6-23

- for component using PKCS#11 wallet, 6-32
- for configuration tools, 6-9
- for Web tier, 6-9
- HSM device, 6-32
- in middle tier, 6-17
- in Oracle Fusion Middleware, 6-1, 6-6
- invoking WLST, 3-17
- LDAP authenticator
 - outbound, 6-19
- OPSS
 - outbound, 6-18
- Oracle Database, 6-29
- Oracle Directory Integration Platform, 6-21
- Oracle Directory Services Manager, 6-21
- Oracle Discoverer, 6-23
- Oracle Forms, 6-22
- Oracle HTTP Server, 6-14
- Oracle Identity and Access Management, 6-20
- Oracle Identity Federation, 6-21
- Oracle Internet Directory, 6-24
- Oracle Portal, 6-23
- Oracle Reports, 6-22
- Oracle SOA Suite, 6-20
- Oracle Virtual Directory, 6-26
- Oracle Web Cache, 6-10
- Oracle WebCenter, 6-20
- Oracle WebLogic Server, 6-18
 - outbound, 6-18
- Oracle WebLogic Server to Oracle database, 6-19
- overview, 6-2
- properties files, 6-58
- tools, 6-7, 6-8, 8-2
 - keystore management, 8-1
 - keytool, 6-21
 - Oracle Wallet Manager, H-1
 - orapki, H-1
 - SSL Configuration Tool, H-23
 - WLST, 6-37, 8-2
 - WLST commands, 6-37
- SSL Automation Tool, 7-1
- SSL Configuration Tool
 - equivalent features for, H-23
- SSL Listen port
 - changing, 5-5
- SSL protocol, 6-4
- ssocfg command, A-4
- ssooconf.sql command, A-4
- startApplication command, 4-5, 4-6
- starting
 - Administration Server, 4-2
 - without credentials, 4-3
 - applications, 4-5
 - components, 4-4, 4-5
 - Managed Servers, 4-2
 - without credentials, 4-3
 - metadata repository, 4-6
 - Net Listener, 4-7
 - Oracle Identity Management, 4-6
 - subprocesses, 4-5
- starting and stopping, 4-1 to 4-9

- state command, 11-1
- static IP address
 - moving off-network, 15-7
 - moving to, 15-7
- status
 - viewing, 11-1
 - for components, 11-6
- status command, 11-2
- stopApplication command, 4-5, 4-6
- stopping, 4-2, 4-3
 - applications, 4-5
 - components, 4-4, 4-5
 - Managed Server, 4-2
 - without credentials, 4-3
 - subprocesses, 4-5
- stopping and starting, 4-1 to 4-9
- system components, 2-1, 3-18
 - recovery of, 18-26
- System MBean Browser, 3-19
 - cloning MDS partition, 14-16
- system MBeans
 - cloneMetadataPartition, 14-16

T

- T2P_JAVA_OPTIONS environment variable, 20-4
- target information icon
 - in Fusion Middleware Control, 3-9
- target menu
 - in Fusion Middleware Control, 3-8
- target name
 - in Fusion Middleware Control, 3-9
- target navigation pane
 - in Fusion Middleware Control, 3-8
- TEMP environment variable, 3-3
- test to production, 21-1
 - moving audit policies, 21-16
 - moving Human Workflow, 21-27
 - moving Oracle Access Manager, 21-12
 - moving Oracle Adaptive Access Manager, 21-15
 - moving Oracle BI Discoverer, 21-65
 - moving Oracle Business Intelligence Discoverer, 21-58
 - moving Oracle Business Process Management, 21-32
 - moving Oracle Data Integrator, 21-94
 - moving Oracle Directory Integration Platform, 21-3, 21-7
 - moving Oracle Directory Services Manager, 21-3
 - moving Oracle Enterprise Content Management Suite, 21-83
 - moving Oracle Forms Services, 21-58, 21-61
 - moving Oracle HTTP Server, 21-53
 - moving Oracle Identity Federation, 21-7
 - moving Oracle Identity Management, 21-4
 - moving Oracle Identity Manager, 21-8
 - moving Oracle Identity Navigator, 21-11
 - moving Oracle Identity and Process Management, 21-87, 21-93
 - moving Oracle Information Rights

- Management, 21-85, 21-91
- moving Oracle Internet Directory, 21-3, 21-7
- moving Oracle Platform Security, 21-16
- moving Oracle Portal, 21-58, 21-60
- moving Oracle Reports, 21-58, 21-63
- moving Oracle Single Sign-On Server, 21-3
- moving Oracle SOA Suite, 21-24
- moving Oracle Universal Content Management, 21-85, 21-92
- moving Oracle Universal Records Management, 21-90, 21-93
- moving Oracle Virtual Directory, 21-3, 21-7
- moving Oracle Web Cache, 21-55
- moving Oracle Web Services Manager, 21-16
- moving Oracle WebCenter, 21-44
- Oracle BI Publisher, 21-69
- Oracle Business Intelligence, 21-69
- Oracle Real-Time Decisions, 21-79
- overview, 21-1
- TMP environment variable, 3-3
- Topology Viewer, 11-13
 - in Fusion Middleware Control, 3-9
- TRACE message type, 12-17
- troubleshooting, I-1 to I-5
 - Fusion Middleware Control, I-2

U

- UCM schema, D-2
- undeploy command, 10-11
- undeploying applications, 10-10
- updatecomponentregistration command, 5-13, 18-26, 18-27, 18-35
- updateinstanceregistration command, 18-26, 18-32
- user names
 - administrator, 3-7
- users, 3-10

V

- version numbers
 - application server, G-2
 - component, G-2
 - format, G-1
 - metadata repository, G-4
 - Oracle Internet Directory, G-3
 - viewing, G-2
- versions
 - in MDS Repository, 14-3

W

- wallets
 - managing with orapki, H-7
- WARNING message type, 12-17
- WEBCENTER schema
 - datafile, D-4
 - description, D-3
 - tablespaces, D-4
- WebLogic Diagnostics Framework (WLDF), 13-2
- WebLogic Server home, 2-6

- WIKI schema
 - datafile, D-4
 - tablespace, D-4
- Windows Registry
 - recovery of, 18-46
- wlst command, A-4
- WLST commands
 - applyJRF, 19-6
 - configureLogHandler, 12-15, 12-17, 12-21
 - createIncident, 13-19
 - createMetadataLabel, 14-24
 - createMetadataPartition, 14-18, 14-21
 - deleteMetadataLabel, 14-27
 - deleteMetadataPartition, 14-22
 - deploy, 10-10, 10-16
 - deregisterMetadataDBRepository, 14-11
 - describeDump, 13-18
 - displayLogs, 12-8, 12-11
 - executeDump, 13-18
 - exportMetadata, 14-18, 14-20, 14-21
 - for SSL, 6-37
 - getIncidentFile, 13-15, 13-16
 - getMDSArchiveConfig, 10-16
 - importMetadata, 14-18, 14-21
 - listDumps, 13-17
 - listIncidents, 13-16
 - listLogs, 12-7
 - listMetadataLabel, 14-24
 - listProblems, 13-15
 - promoteMetadataLabel, 14-25
 - purgeMetadata, 14-22, 14-23
 - purgeMetadataLabels, 14-26
 - redploy, 10-12
 - registerMetadataDBRepository, 14-10
 - setAppMetadataRepository, 10-16
 - showIncident, 13-16
 - startApplication, 4-5, 4-6
 - state, 11-1
 - stopApplication, 4-5, 4-6
 - undeploy, 10-11

