

Oracle® Fusion Middleware

Enterprise Deployment Guide for Oracle SOA Suite

12c (12.2.1)

E57888-04

February 2016

Describes how to install and configure Oracle Fusion Middleware components in an enterprise deployment.

Oracle Fusion Middleware Enterprise Deployment Guide for Oracle SOA Suite, 12c (12.2.1)

E57888-04

Copyright © 2014, 2016, Oracle and/or its affiliates. All rights reserved.

Primary Authors: Fermin Castro (MAA Engineer), Peter LaQuerre (Writer), HT Ma (Quality Assurance)

Contributors: Satheesh Amilineni, Sudhakar Betha, John Featherly, Mark Foster, Derek Kam, Rodger King, Stefan Koser, Ashwani Raj, Michael Rhys, Joe Paul, Maria Salzberger, Muthukumar Subramaniam, Mukesh Tailor, Chakravarthy Tenneti, Bonnie Vaughan, Shane Vermette

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

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

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

U.S. GOVERNMENT END USERS: Oracle programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, delivered to U.S. Government end users are "commercial computer software" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the programs, including any operating system, integrated software, any programs installed on the hardware, and/or documentation, shall be subject to license terms and license restrictions applicable to the programs. No other rights are granted to the U.S. Government.

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

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

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

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

Contents

Preface	xv
Audience	xv
Conventions.....	xv

Part I Understanding an Enterprise Deployment

1 Enterprise Deployment Overview

1.1 About the Enterprise Deployment Guide	1-1
1.2 When to Use the Enterprise Deployment Guide.....	1-2

2 Understanding a Typical Enterprise Deployment

2.1 Diagram of a Typical Enterprise Deployment.....	2-1
2.2 Understanding the Typical Enterprise Deployment Topology Diagram.....	2-2
2.2.1 Understanding the Firewalls and Zones of a Typical Enterprise Deployment.....	2-3
2.2.2 Understanding the Elements of a Typical Enterprise Deployment Topology	2-3
2.2.3 Receiving Requests Through Hardware Load Balancer.....	2-4
2.2.4 Understanding the Web Tier	2-6
2.2.5 Understanding the Application Tier	2-8
2.2.6 About the Data Tier.....	2-13

3 Understanding the Oracle SOA Suite Enterprise Deployment Topology

3.1 Understanding the Primary and Build-Your-Own Enterprise Deployment Topologies.....	3-2
3.2 Diagrams of the Primary Oracle SOA SuiteEnterprise Topologies.....	3-2
3.2.1 Diagram of the Oracle SOA Suite and Oracle Service Bus Topology.....	3-2
3.2.2 Diagram of the Oracle SOA Suite and Oracle Business Activity Monitoring Topology	3-3
3.3 Understanding the Primary Oracle SOA Suite Topology Diagrams	3-4
3.3.1 Understanding the Topology Options for Oracle Service Bus	3-5
3.3.2 Summary of Oracle SOA Suite Load Balancer Virtual Server Names.....	3-5
3.3.3 Understanding the Routing of SOA Composite Requests	3-6
3.3.4 Summary of the Managed Servers and Clusters on SOA Application Tier	3-9

3.4	Flow Charts and Roadmaps for Implementing the Primary Oracle SOA Suite Enterprise Topologies	3-9
3.4.1	Flow Chart of the Steps to Install and Configure the Primary Oracle SOA Suite Enterprise Topologies	3-9
3.4.2	Roadmap Table for Planning and Preparing for an Enterprise Deployment.....	3-11
3.4.3	Roadmap Table for Configuring the Oracle SOA Suite and Oracle Service Bus Enterprise Topology	3-11
3.4.4	Roadmap Table for Configuring the Oracle SOA Suite and Oracle Business Activity Monitoring Enterprise Topology	3-12
3.5	Building Your Own Oracle SOA Suite Enterprise Topology	3-13
3.5.1	Flow Chart of the "Build Your Own" Enterprise Topologies.....	3-13
3.5.2	Description of the Supported "Build Your Own" Topologies.....	3-14
3.6	About Installing and Configuring a Custom Enterprise Topology	3-16
3.7	About Using Automatic Service Migration for the Oracle SOA Suite Enterprise Topology	3-16

Part II Preparing for an Enterprise Deployment

4 Using the Enterprise Deployment Workbook

4.1	Introduction to the Enterprise Deployment Workbook.....	4-1
4.2	Typical Use Case for Using the Workbook	4-2
4.3	Using the Oracle SOA Suite Enterprise Deployment Workbook	4-2
4.3.1	Locating the Oracle SOA Suite Enterprise Deployment Workbook.....	4-2
4.3.2	Understanding the Contents of the Oracle SOA Suite Enterprise Deployment Workbook.....	4-2
4.4	Who Should Use the Enterprise Deployment Workbook?	4-5

5 Procuring Resources for an Enterprise Deployment

5.1	Hardware and Software Requirements for the Enterprise Deployment Topology	5-1
5.1.1	Hardware Load Balancer Requirements.....	5-2
5.1.2	Host Computer Hardware Requirements	5-3
5.1.3	Operating System Requirements for the Enterprise Deployment Topology	5-6
5.2	Reserving the Required IP Addresses for an Enterprise Deployment.....	5-6
5.2.1	What Is a Virtual IP (VIP) Address?	5-7
5.2.2	Why Use Virtual Host Names and Virtual IP Addresses?.....	5-7
5.2.3	Physical and Virtual IP Addresses Required by the Enterprise Topology	5-7
5.3	Identifying and Obtaining Software Downloads for an Enterprise Deployment.....	5-8

6 Preparing the Load Balancer and Firewalls for an Enterprise Deployment

6.1	Configuring Virtual Hosts on the Hardware Load Balancer	6-1
6.1.1	Overview of the Hardware Load Balancer Configuration.....	6-1
6.1.2	Typical Procedure for Configuring the Hardware Load Balancer.....	6-2
6.1.3	Summary of the Virtual Servers Required for an Enterprise Deployment.....	6-2

6.1.4	Additional Instructions for admin.example.com	6-3
6.1.5	Additional Instructions for soa.example.com	6-3
6.1.6	Additional Instructions for soainternal.example.com	6-3
6.1.7	Additional Instructions for osb.example.com	6-4
6.1.8	Additional Instructions for soahealthcare.example.com.....	6-4
6.2	Configuring the Firewalls and Ports for an Enterprise Deployment.....	6-4
7	Preparing the File System for an Enterprise Deployment	
7.1	Overview of Preparing the File System for an Enterprise Deployment.....	7-1
7.2	Shared Storage Recommendations When Installing and Configuring an Enterprise Deployment.....	7-2
7.3	Understanding the Recommended Directory Structure for an Enterprise Deployment	7-2
7.4	File System and Directory Variables Used in This Guide.....	7-5
7.5	About Creating and Mounting the Directories for an Enterprise Deployment	7-9
7.6	Summary of the Shared Storage Volumes in an Enterprise Deployment	7-10
8	Preparing the Host Computers for an Enterprise Deployment	
8.1	Verifying the Minimum Hardware Requirements for Each Host.....	8-1
8.2	Verifying Linux Operating System Requirements.....	8-2
8.2.1	Setting Linux Kernel Parameters	8-2
8.2.2	Setting the Open File Limit and Number of Processes Settings on UNIX Systems....	8-3
8.2.3	Verifying IP Addresses and Host Names in DNS or hosts File.....	8-4
8.3	Configuring Operating System Users and Groups.....	8-4
8.4	Enabling Unicode Support	8-4
8.5	Mounting the Required Shared File Systems on Each Host.....	8-5
8.6	Enabling the Required Virtual IP Addresses on Each Host	8-6
9	Preparing the Database for an Enterprise Deployment	
9.1	Overview of Preparing the Database for an Enterprise Deployment.....	9-2
9.2	About Database Requirements	9-2
9.2.1	Supported Database Versions	9-2
9.2.2	Additional Database Software Requirements.....	9-3
9.2.3	Setting the PROCESSES Database Initialization Parameter for an Enterprise Deployment	9-3
9.3	Creating Database Services	9-4
9.4	Using SecureFiles for Large Objects (LOBs) in an Oracle Database.....	9-5
9.5	About Database Backup Strategies	9-6
9.6	Implementing a Database Growth Management Strategy for Oracle SOA Suite	9-6
Part III Configuring the Enterprise Deployment		
10	Creating the Initial Infrastructure Domain for an Enterprise Deployment	
10.1	Variables Used When Creating the Infrastructure Domain	10-3

10.2	Understanding the Initial Infrastructure Domain.....	10-3
10.2.1	About the Infrastructure Distribution.....	10-3
10.2.2	Characteristics of the Initial Infrastructure Domain	10-4
10.3	Installing the Oracle Fusion Middleware Infrastructure in Preparation for an Enterprise Deployment.....	10-4
10.3.1	Installing a Supported JDK	10-4
10.3.2	Starting the Infrastructure Installer on SOAHOST1	10-6
10.3.3	Navigating the Infrastructure Installation Screens	10-6
10.3.4	Installing Oracle Fusion Middleware Infrastructure on the Other Host Computers	10-8
10.3.5	Checking the Directory Structure	10-8
10.4	Creating the Database Schemas.....	10-9
10.4.1	Installing and Configuring a Certified Database.....	10-9
10.4.2	Starting the Repository Creation Utility (RCU).....	10-10
10.4.3	Navigating the RCU Screens to Create the Schemas	10-10
10.5	Configuring the Infrastructure Domain	10-12
10.5.1	Starting the Configuration Wizard	10-13
10.5.2	Navigating the Configuration Wizard Screens to Configure the Infrastructure Domain	10-13
10.6	Configuring a Per Host Node Manager for an Enterprise Deployment.....	10-24
10.6.1	Creating a Per Host Node Manager Configuration	10-24
10.6.2	Creating the boot.properties File.....	10-26
10.6.3	Starting the Node Manager on SOAHOST1.....	10-26
10.6.4	Configuring the Node Manager Credentials and Type.....	10-27
10.7	Configuring the Domain Directories and Starting the Servers on SOAHOST1	10-29
10.7.1	Starting the Administration Server Using the Node Manager	10-29
10.7.2	Validating the Administration Server	10-30
10.7.3	Disabling the Derby Database	10-30
10.7.4	Creating a Separate Domain Directory for Managed Servers on SOAHOST1	10-31
10.7.5	Starting and Validating the WLS_WSM1 Managed Server on SOAHOST1.....	10-33
10.8	Propagating the Domain and Starting the Servers on SOAHOST2	10-34
10.8.1	Unpacking the Domain Configuration on SOAHOST2.....	10-34
10.8.2	Unpacking the Domain on SOAHOST2.....	10-35
10.8.3	Starting the Node Manager on SOAHOST2.....	10-36
10.8.4	Starting and Validating the WLS_WSM2 Managed Server on SOAHOST2.....	10-36
10.9	Modifying the Upload and Stage Directories to an Absolute Path	10-36
10.10	Creating a New LDAP Authenticator and Provisioning Enterprise Deployment Users and Group	10-37
10.10.1	About the Supported Authentication Providers.....	10-38
10.10.2	About the Enterprise Deployment Users and Groups.....	10-38
10.10.3	Prerequisites for Creating a New Authentication Provider and Provisioning Users and Groups	10-40
10.10.4	Provisioning a Domain Connector User in the LDAP Directory	10-40

10.10.5	Creating the New Authentication Provider	10-42
10.10.6	Provisioning an Enterprise Deployment Administration User and Group	10-46
10.10.7	Adding the New Administration User to the Administration Group	10-47
10.10.8	Updating the boot.properties File and Restarting the System.....	10-48
10.11	Adding the wsm-pm Role to the Administrators Group.....	10-49
11	Configuring the Web Tier for an Enterprise Deployment	
11.1	Variables Used When Configuring the Web Tier	11-2
11.2	About the Web Tier Domains	11-2
11.3	Installing Oracle HTTP Server on WEBHOST1.....	11-3
11.3.1	Starting the Installer on WEBHOST1	11-3
11.3.2	Navigating the Oracle HTTP Server Installation Screens	11-3
11.3.3	Verifying the Oracle HTTP Server Installation.....	11-4
11.4	Creating a Web Tier Domain on WEBHOST1	11-5
11.4.1	Starting the Configuration Wizard on WEBHOST1.....	11-5
11.4.2	Navigating the Configuration Wizard Screens for a Web Tier Domain	11-5
11.5	Installing and Configuring a Web Tier Domain on WEBHOST2.....	11-8
11.6	Starting the Node Manager and Oracle HTTP Server Instances on WEBHOST1 and WEBHOST2.....	11-8
11.6.1	Starting the Node Manager on WEBHOST1 and WEBHOST2.....	11-8
11.6.2	Starting the Oracle HTTP Server Instances	11-8
11.7	Configuring Oracle HTTP Server to Route Requests to the Application Tier.....	11-9
11.7.1	About the Oracle HTTP Server Configuration for an Enterprise Deployment.....	11-9
11.7.2	Modifying the httpd.conf File to Include Virtual Host Configuration Files.....	11-10
11.7.3	Creating the Virtual Host Configuration Files.....	11-11
11.7.4	Validating the Virtual Server Configuration on the Load Balancer	11-12
11.7.5	Configuring Routing to the Administration Server and Oracle Web Services Manager	11-12
11.7.6	Validating Access to the Management Consoles and Administration Server	11-14
12	Extending the Domain with Oracle SOA Suite	
12.1	Variables Used When Extending the Domain with Oracle SOA Suite.....	12-2
12.2	Synchronizing the System Clocks	12-3
12.3	Installing the Software for an Enterprise Deployment	12-3
12.3.1	Starting the Installer on SOAHOST1	12-3
12.3.2	Navigating the Installation Screens	12-3
12.3.3	Installing the Software on the Other Host Computers	12-4
12.3.4	Verifying the Installation.....	12-4
12.4	Creating the Oracle SOA Suite Database Schemas	12-5
12.4.1	Starting the Repository Creation Utility (RCU).....	12-6
12.4.2	Navigating the RCU Screens to Create the Schemas	12-6
12.4.3	Configuring SOA Schemas for Transactional Recovery.....	12-9
12.5	Extending the Enterprise Deployment Domain with Oracle SOA Suite.....	12-9

12.5.1	Starting the Configuration Wizard	12-9
12.5.2	Navigating the Configuration Wizard Screens to Extend the Domain with Oracle SOA Suite	12-10
12.6	Configuring a Default Persistence Store for Transaction Recovery	12-16
12.7	Propagating the Extended Domain to the Domain Directories and Machines	12-17
12.7.1	Packing Up the Extended Domain on SOAHOST1.....	12-18
12.7.2	Unpacking the Domain in the Managed Servers Domain Directory on SOAHOST1.....	12-19
12.7.3	Unpacking the Domain on SOAHOST2.....	12-20
12.8	Starting and Validating the WLS_SOA1 Managed Server	12-21
12.8.1	Starting the WLS_SOA1 Managed Server	12-21
12.8.2	Adding the SOAAdmin Role to the Administrators Group	12-22
12.8.3	Validating the Managed Server by Logging in to the SOA Infrastructure	12-22
12.9	Starting and Validating the WLS_SOA2 Managed Server	12-22
12.10	Validating the Location and Creation of the Transaction Logs	12-22
12.11	Configuring Oracle HTTP Server for the Extended Domain	12-23
12.11.1	Configuring Oracle HTTP Server for the WLS_SOA Managed Servers	12-23
12.11.2	Configuring the WebLogic Proxy Plug-In.....	12-26
12.11.3	Validating the Oracle SOA Suite URLs Through the Load Balancer.....	12-27
12.12	Post-Configuration Steps for Oracle SOA Suite	12-27
12.12.1	Configuring Oracle Adapters for Oracle SOA Suite	12-27
12.12.2	Enabling SSL Communication Between the SOA Servers and the Hardware Load Balancer	12-34
12.12.3	Considerations for sync-async interactions in a SOA cluster	12-34
12.13	Enabling Automatic Service Migration and JDBC Persistent Stores for Oracle SOA Suite.....	12-35

13 Adding Oracle Service Bus to the Topology

13.1	Variables Used when Adding Oracle Service Bus to the Enterprise Topology.....	13-2
13.2	About Configuring Oracle Service in Its Own Domain	13-3
13.3	Overview of Adding OSB to the Topology.....	13-3
13.4	Prerequisites for Extending the Domain to Include Oracle Service Bus	13-5
13.5	Installing Oracle Service Bus Software	13-5
13.5.1	Starting the Oracle Service Bus Installer	13-6
13.5.2	Navigating the OSB Installation Screens	13-6
13.5.3	Installing the Software on the Other Host Computers	13-7
13.5.4	Validating the OSB Installation	13-7
13.6	Extending the SOA or Infrastructure Domain to Include Oracle Service Bus.....	13-8
13.6.1	Starting the Configuration Wizard	13-8
13.6.2	Navigating the Configuration Wizard Screens for Oracle Service Bus.....	13-9
13.7	Configuring a Default Persistence Store for Transaction Recovery	13-15
13.8	Propagating the Extended Domain to the Domain Directories and Machines	13-16

13.8.1	Summary of the Tasks Required to Propagate the Changes to the Other Domain Directories and Machines	13-16
13.8.2	Starting and Validating the WLS_OSB1 Managed Server.....	13-17
13.8.3	Starting and Validating the WLS_OSB2 Managed Server.....	13-19
13.8.4	Validating the Location and Creation of the Transaction Logs.....	13-19
13.8.5	Verifying the Appropriate Targeting for OSB Singleton Services	13-19
13.9	Configuring Oracle HTTP Server for the Oracle Service Bus	13-20
13.10	Configuring the WebLogic Proxy Plug-In	13-23
13.11	Validating the Oracle Service Bus URLs Through the Load Balancer	13-23
13.12	Post-Configuration Tasks for Oracle Service Bus	13-24
13.12.1	Enabling High Availability for Oracle DB, File and FTP Adapters	13-24
13.12.2	Configuring Specific Oracle Service Bus Services for an Enterprise Deployment	13-24
13.12.3	Enabling SSL Communication Between the Oracle Service Bus Servers and the Hardware Load Balancer	13-25
13.12.4	Backing Up the Oracle Service Bus Configuration.....	13-25
13.13	Enabling Automatic Service Migration and JDBC Persistent Stores for Oracle Service Bus	13-25

14 Extending the Domain with Business Process Management

14.1	Variables Used in This Chapter	14-2
14.2	Prerequisites for Extending the SOA Domain to Include Oracle BPM.....	14-3
14.3	Installing Oracle Business Process Management for an Enterprise Deployment	14-3
14.3.1	Starting the Installation Program.....	14-3
14.3.2	Navigating the Oracle BPM Installation Screens.....	14-4
14.3.3	Verifying the Installation.....	14-5
14.4	Running the Configuration Wizard on SOAHOST1 to Extend a SOA Domain to Include BPM.....	14-6
14.4.1	Starting the Configuration Wizard	14-6
14.4.2	Navigating the Configuration Wizard Screens to Extend the Domain.....	14-6
14.5	Propagating the Extended Domain to the Domain Directories and Machines	14-9
14.6	Restarting the WLS_SOA Managed Servers with Business Process Management.....	14-10
14.7	Adding the Enterprise Deployment Administration User to the Oracle BPM Administrators Group	14-11
14.8	Configuring Oracle HTTP Server for Oracle Business Process Management	14-11
14.9	Enabling SSL Communication Between Business Process Management Servers and the Hardware Load Balancer	14-12
14.10	Validating Access to Business Process Management Through the Hardware Load Balancer	14-12
14.11	Configuring BPMJMSModule for the Oracle BPM Cluster	14-13
14.12	Backing Up the Oracle BPM Configuration.....	14-15
14.13	Enabling Automatic Service Migration and JDBC Persistent Stores for Oracle Business Process Management.....	14-16

15 Extending the Domain with Oracle Enterprise Scheduler

15.1	Variables Used in This Chapter	15-2
15.2	Overview of Adding Oracle Enterprise Scheduler to a SOA Domain	15-3
15.3	Creating the Database Schemas for ESS	15-4
15.3.1	Starting the Repository Creation Utility (RCU)	15-4
15.3.2	Navigating the RCU Screens to Create the Enterprise Scheduler Schemas.....	15-4
15.4	Extending the SOA Domain to Include Oracle Enterprise Scheduler	15-6
15.4.1	Starting the Configuration Wizard	15-6
15.4.2	Navigating the Configuration Wizard Screens to Extend the Domain with Oracle Enterprise Scheduler	15-7
15.5	Configuring a Default Persistence Store for Transaction Recovery	15-12
15.6	Propagating the Extended Domain to the Domain Directories and Machines	15-13
15.7	Adding the ESSAdmin Role to the SOA Administrators Group.....	15-14
15.8	Starting WLS_ESS1 Managed Server	15-14
15.9	Starting and Validating the WLS_ESS2 Managed Server	15-16
15.10	Validating the Location and Creation of the Transaction Logs	15-16
15.11	Configuring Oracle HTTP Server for the WLS_ESS Managed Servers	15-16
15.12	Configuring the WebLogic Proxy Plug-In	15-17
15.13	Validating Access to Oracle Enterprise Scheduler Through the Hardware Load Balancer	15-18
15.14	Backing Up the Oracle Enterprise Scheduler Configuration	15-18

16 Extending the Domain with Business Activity Monitoring

16.1	Variables Used in This Chapter	16-2
16.2	Prerequisites When Adding Oracle BAM to the Domain.....	16-3
16.2.1	Understanding the Installation Requirements Adding Oracle BAM to the Domain	16-3
16.2.2	Understanding the Database Schema Requirements for Oracle BAM.....	16-3
16.2.3	Backing Up the Existing Installation	16-4
16.3	Special Instructions When Configuring Oracle BAM on Separate Hosts	16-4
16.3.1	Procuring Additional Host Computers for Oracle BAM.....	16-4
16.3.2	Installation Requirements When Configuring Oracle BAM on Separate Hosts	16-4
16.3.3	Configuration Wizard Instructions When Configuring Oracle BAM on Separate Hosts	16-6
16.3.4	Propagating the Domain Configuration When Configuring Oracle BAM on Separate Hosts	16-6
16.4	Roadmap for Adding Oracle BAM to the Domain.....	16-6
16.5	Extending the SOA Domain to Include Oracle Business Activity Monitoring.....	16-8
16.5.1	Starting the Configuration Wizard	16-8
16.5.2	Navigating the Configuration Wizard Screens for Oracle BAM.....	16-8
16.6	Configuring a Default Persistence Store for Transaction Recovery	16-14
16.7	Propagating the Extended Domain to the Domain Directories and Machines	16-15

16.8	Adding the Enterprise Deployment Administration User to the Oracle BAM Administration Group.....	16-15
16.9	Starting WLS_BAM1 Managed Server	16-16
16.10	Starting and Validating the WLS_BAM2 Managed Server	16-17
16.11	Configuring Oracle HTTP Server for the WLS_BAM Managed Servers.....	16-18
16.12	Configuring the WebLogic Proxy Plug-In	16-18
16.13	Validating Access to Oracle BAM Through the Hardware Load Balancer.....	16-19
16.14	Enabling Automatic Service Migration and JDBC Persistent Stores for the Oracle BAM Servers.....	16-20
16.15	Backing Up the Oracle BAM Configuration	16-20
17	Extending the Domain with Oracle B2B	
17.1	Variables Used in This Chapter	17-2
17.2	Prerequisites for Extending the SOA Domain to Include Oracle B2B	17-3
17.3	Installing Oracle B2B for an Enterprise Deployment	17-3
17.3.1	Starting the Oracle B2B and Healthcare Installer on SOAHOST1	17-3
17.3.2	Navigating the Oracle B2B Installation Screens.....	17-4
17.3.3	Verifying the B2B or Healthcare Installation.....	17-5
17.4	Running the Configuration Wizard to Extend for Oracle B2B.....	17-6
17.4.1	Starting the Configuration Wizard	17-6
17.4.2	Navigating the Configuration Wizard Screens for Oracle B2B	17-6
17.5	Starting the B2B Suite Components	17-9
17.6	Updating the B2B Instance Identifier for Transports	17-9
17.7	Configuring Oracle HTTP Server for Oracle B2B	17-10
17.8	Adding the B2BAdmin Role to the SOA Administrators Group.....	17-11
17.9	Validating Access to Oracle B2B Through the Load Balancer	17-11
17.10	Backing Up the Configuration	17-12
17.11	Enabling Automatic Service Migration and JDBC Persistent Stores for Oracle B2B.....	17-13
18	Extending the Domain with Oracle SOA Suite for Healthcare Integration	
18.1	Variables Used When Extending the Domain for Oracle Healthcare.....	18-2
18.2	About Oracle SOA Suite for Healthcare Integration in an Enterprise Deployment.....	18-3
18.3	Prerequisites for Extending the Domain to Include Oracle Healthcare	18-3
18.4	Installing Oracle Healthcare for an Enterprise Deployment.....	18-4
18.4.1	Starting the Oracle B2B and Healthcare Installer on SOAHOST1	18-4
18.4.2	Navigating the Installation Screens for Oracle Healthcare Installation	18-5
18.4.3	Verifying the B2B or Healthcare Installation.....	18-6
18.5	Running the Configuration Wizard for Oracle Healthcare	18-7
18.5.1	Starting the Configuration Wizard	18-7
18.5.2	Navigating the Configuration Wizard Screens for Oracle Healthcare.....	18-7
18.6	Starting the Healthcare Components.....	18-10
18.7	Updating the B2B Instance Identifier and MLLP High Availability Mode.....	18-11
18.8	Disabling Connection Factory Affinity for Optimum Load Balancing.....	18-12

18.9	Configuring Oracle HTTP Server for Oracle Healthcare	18-12
18.10	Adding the B2BAdmin Role to the SOA Administrators Group.....	18-13
18.11	Validating Access to Oracle Healthcare Through the Load Balancer	18-13
18.12	Backing Up the Configuration	18-14
18.13	Enabling Automatic Service Migration for Oracle Healthcare	18-14

Part IV Common Configuration and Management Procedures for an Enterprise Deployment

19 Common Configuration and Management Tasks for an Enterprise Deployment

19.1	Configuration and Management Tasks for All Enterprise Deployments	19-1
19.1.1	Verifying Manual Failover of the Administration Server	19-2
19.1.2	Enabling SSL Communication Between the Middle Tier and the Hardware Load Balancer	19-6
19.1.3	Configuring Roles for Administration of an Enterprise Deployment.....	19-13
19.1.4	Using JDBC Persistent Stores for TLOGs and JMS in an Enterprise Deployment	19-16
19.1.5	Performing Backups and Recoveries for an Enterprise Deployment.....	19-26
19.2	Configuration and Management Tasks for an Oracle SOA Suite Enterprise Deployment	19-27
19.2.1	Deploying Oracle SOA Suite Composite Applications to an Enterprise Deployment	19-27
19.2.2	Using Shared Storage for Deployment Plans and SOA Infrastructure Applications Updates	19-28
19.2.3	Managing Database Growth in an Oracle SOA Suite Enterprise Deployment.....	19-28

20 Using Whole Server Migration and Service Migration in an Enterprise Deployment

20.1	About Whole Server Migration and Automatic Service Migration in an Enterprise Deployment.....	20-1
20.1.1	Understanding the Difference Between Whole Server and Service Migration.....	20-2
20.1.2	Implications of Using Whole Server Migration or Service Migration in an Enterprise Deployment.....	20-2
20.1.3	Understanding Which Products and Components Require Whole Server Migration and Service Migration	20-3
20.2	Creating a GridLink Data Source for Leasing	20-3
20.3	Configuring Whole Server Migration for an Enterprise Deployment	20-6
20.3.1	Editing the Node Manager's Properties File to Enable Whole Server Migration ...	20-6
20.3.2	Setting Environment and Superuser Privileges for the wlsifconfig.sh Script	20-7
20.3.3	Configuring Server Migration Targets.....	20-8
20.3.4	Testing Whole Server Migration	20-9
20.4	Configuring Automatic Service Migration in an Enterprise Deployment.....	20-11
20.4.1	Setting the Leasing Mechanism and Data Source for an Enterprise Deployment Cluster	20-11

20.4.2	Changing the Migration Settings for the Managed Servers in the Cluster	20-12
20.4.3	About Selecting a Service Migration Policy	20-12
20.4.4	Setting the Service Migration Policy for Each Managed Server in the Cluster	20-13
20.4.5	Restarting the Managed Servers and Validating Automatic Service Migration...	20-13
20.4.6	Failing Back Services After Automatic Service Migration	20-15

21 Configuring Single Sign-On for an Enterprise Deployment

21.1	About Oracle HTTP Server Webgate	21-2
21.2	General Prerequisites for Configuring Oracle HTTP Server 12c Webgate.....	21-2
21.3	Enterprise Deployment Prerequisites for Configuring OHS 12c Webgate	21-2
21.4	Configuring Oracle HTTP Server 12c WebGate for an Enterprise Deployment	21-3
21.5	Registering the Oracle HTTP Server 12c WebGate with Oracle Access Manager	21-4
21.5.1	Locating and Preparing the RREG Tool.....	21-4
21.5.2	About RREG In-Band and Out-of-Band Mode	21-5
21.5.3	Updating the Standard Properties in the OAM11gRequest.xml File	21-5
21.5.4	Updating the Protected, Public, and Excluded Resources for an Enterprise Deployment	21-7
21.5.5	Running the RREG Tool	21-8
21.5.6	Files and Artifacts Generated by RREG.....	21-10
21.5.7	Copying Generated Artifacts to the Oracle HTTP Server WebGate Instance Location	21-11
21.5.8	Restarting the Oracle HTTP Server Instance	21-12
21.6	Setting Up the WebLogic Server Authentication Providers.....	21-13
21.6.1	Backing Up Configuration Files	21-13
21.6.2	Setting Up the Oracle Access Manager Identity Assertion Provider.....	21-13
21.6.3	Setting the Order of Providers.....	21-14
21.7	Configuring Oracle ADF and OPSS Security with Oracle Access Manager	21-14

A Using Multi Data Sources with Oracle RAC

A.1	About Multi Data Sources and Oracle RAC.....	A-1
A.2	Typical Procedure for Configuring Multi Data Sources for an Enterprise Deployment.....	A-1

Preface

This guide explains how to install, configure, and manage a highly available Oracle Fusion Middleware enterprise deployment. For more information, see [About the Enterprise Deployment Guide](#).

See Also:

[Audience](#)

[Conventions](#)

Audience

In general, this document is intended for administrators of Oracle Fusion Middleware, who are assigned the task of installing and configuring Oracle Fusion Middleware software for production deployments.

Specific tasks can also be assigned to more specialized administrators, such as database administrators (DBAs) and network administrators, where applicable.

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.

Note:

This guide focuses on the implementation of the enterprise deployment reference topology on Oracle Linux systems.

The topology can be implemented on any certified, supported operating system, but the examples in this guide typically show the commands and configuration steps as they should be performed using the bash shell on Oracle Linux.

Part I

Understanding an Enterprise Deployment

This part of the Enterprise Deployment Guide contains the following topics.

See Also:

[Enterprise Deployment Overview](#)

[Understanding a Typical Enterprise Deployment](#)

[Understanding the Oracle SOA Suite Enterprise Deployment Topology](#)

Enterprise Deployment Overview

This chapter introduces the concept of an Oracle Fusion Middleware enterprise deployment.

It also provides information on when to use the Enterprise Deployment guide.

See Also:

[Understanding an Enterprise Deployment](#)

[About the Enterprise Deployment Guide](#)

An Enterprise Deployment Guide provides a comprehensive, scalable example for installing, configuring, and maintaining a secure, highly available, production-quality deployment of selected Oracle Fusion Middleware products. This resulting environment is known as an **enterprise deployment topology**.

[When to Use the Enterprise Deployment Guide](#)

This guide describes one of three primary installation and configuration options for Oracle Fusion Middleware. Use this guide to help you plan, prepare, install, and configure a multi-host, secure, highly available, production topology for selected Oracle Fusion Middleware products.

1.1 About the Enterprise Deployment Guide

An Enterprise Deployment Guide provides a comprehensive, scalable example for installing, configuring, and maintaining a secure, highly available, production-quality deployment of selected Oracle Fusion Middleware products. This resulting environment is known as an **enterprise deployment topology**.

By example, the enterprise deployment topology introduces key concepts and best practices that you can use to implement a similar Oracle Fusion Middleware environment for your organization.

Each Enterprise Deployment Guide provides detailed, validated instructions for implementing the reference topology. Along the way, the guide offers links to supporting documentation that explains concepts, reference material, and additional options for an Oracle Fusion Middleware enterprise deployment.

Note that the enterprise deployment topologies described in the enterprise deployment guides cannot meet the exact requirements of all Oracle customers. In some cases, you can consider alternatives to specific procedures in this guide, depending on whether the variations to the topology are documented and supported by Oracle.

Oracle recommends customers use the Enterprise Deployment Guides as a first option for deployment. If variations are required, then those variations should be verified by reviewing related Oracle documentation or by working with Oracle Support.

1.2 When to Use the Enterprise Deployment Guide

This guide describes one of three primary installation and configuration options for Oracle Fusion Middleware. Use this guide to help you plan, prepare, install, and configure a multi-host, secure, highly available, production topology for selected Oracle Fusion Middleware products.

Alternatively, you can:

- Use the instructions in *Installing SOA Suite and Business Process Management Quick Start for Developers* to install a **development environment**.

A development environment provides the software and tools that you can use to develop Java, Oracle Application Development Framework, and other applications that depend on Oracle technologies. Development environments are typically installed on a single host and do not require many of the features of a production environment.

- Use the instructions in one of the product-specific installation guides to install and configure a **standard installation topology** for a selected set of Oracle Fusion Middleware products.

A standard installation topology can be installed on a single host for evaluation purposes, but it can also serve as a starting point for scaling out to a more complex production environment.

For Oracle SOA Suite, see:

- *Installing and Configuring Oracle SOA Suite and Business Process Management*
- *Installing and Configuring Oracle Service Bus*
- *Installing and Configuring B2B and Healthcare*
- Review *Planning an Installation of Oracle Fusion Middleware*, which provides additional information to help you prepare for any Oracle Fusion Middleware installation.

Understanding a Typical Enterprise Deployment

This chapter describes the general characteristics of a typical Oracle Fusion Middleware enterprise deployment. You can apply the concepts here to any product-specific enterprise deployment.

This chapter provides information on Enterprise Deployment Topology diagram.

See Also:

[Understanding an Enterprise Deployment](#)

[Diagram of a Typical Enterprise Deployment](#)

This section illustrates a typical enterprise deployment, including the Web tier, application tier, and data tier.

[Understanding the Typical Enterprise Deployment Topology Diagram](#)

This section provides a detailed description of the typical enterprise topology diagram.

2.1 Diagram of a Typical Enterprise Deployment

This section illustrates a typical enterprise deployment, including the Web tier, application tier, and data tier.

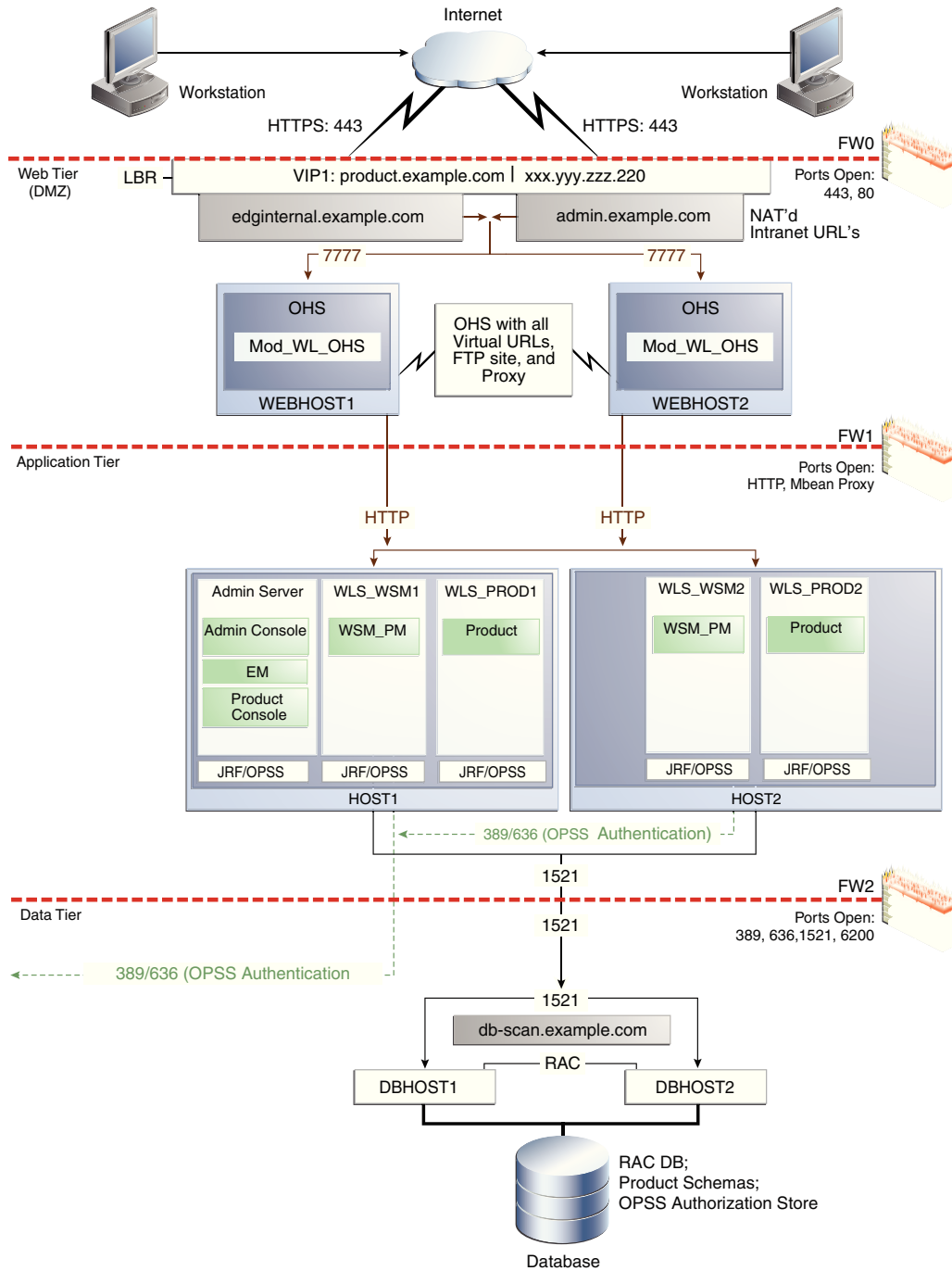
All Oracle Fusion Middleware enterprise deployments are designed to demonstrate the best practices for installing and configuring an Oracle Fusion Middleware production environment.

A best practices approach starts with the basic concept of a multi-tiered deployment and standard communications between the different software tiers.

[Figure 2-1](#) shows a typical enterprise deployment, including the Web tier, application tier and data tier. All enterprise deployments are based on these basic principles.

For a description of each tier and the standard protocols used for communications within a typical Oracle Fusion Middleware enterprise deployment, see [Understanding the Typical Enterprise Deployment Topology Diagram](#).

Figure 2-1 Typical Enterprise Deployment Topology Diagram



2.2 Understanding the Typical Enterprise Deployment Topology Diagram

This section provides a detailed description of the typical enterprise topology diagram.

See Also:

[Understanding the Firewalls and Zones of a Typical Enterprise Deployment](#)

[Understanding the Elements of a Typical Enterprise Deployment Topology](#)

[Receiving Requests Through Hardware Load Balancer](#)

[Understanding the Web Tier](#)

[Understanding the Application Tier](#)

[About the Data Tier](#)

2.2.1 Understanding the Firewalls and Zones of a Typical Enterprise Deployment

The topology is divided into several security zones, which are separated by firewalls:

- The Web tier (or DMZ), which is used for the hardware load balancer and Web servers (in this case, Oracle HTTP Server instances) that receive the initial requests from users. This zone is accessible only through a single virtual server name defined on the load balancer.
- The application tier, which is where the business and application logic resides.
- The data tier, which is not accessible from the Internet and reserved in this topology for the highly available database instances.

The firewalls are configured to allow data to be transferred only through specific communication ports. Those ports (or in some cases, the protocols that will need open ports in the firewall) are shown on each firewall line in the diagram.

For example:

- On the firewall protecting the Web tier, only the HTTP ports are open: 443 for HTTPS and 80 for HTTP.
- On the firewall protecting the Application tier, HTTP ports, and MBean proxy port are open.

Applications that require external HTTP access can use the Oracle HTTP Server instances as a proxy. Note that this port for outbound communications only and the proxy capabilities on the Oracle HTTP Server must be enabled.

- On the firewall protecting the data tier, the database listener port (typically, 1521) must be open.

The LDAP ports (typically, 389 and 636) are also required to be open for communication between the authorization provider and the LDAP-based identity store.

The ONS port (typically, 6200) is also required so the application tier can receive notifications about workload and events in the Oracle RAC Database. These events are used by the Oracle WebLogic Server connection pools to adjust quickly (creating or destroying connections), depending on the availability and workload on the Oracle RAC database instances.

For a complete list of the ports you must open for a specific Oracle Fusion Middleware enterprise deployment topology, see the chapter that describes the topology you want to implement, or refer to the *Enterprise Deployment Workbook* for the topology you are implement. For more information, see [Using the Enterprise Deployment Workbook](#) .

2.2.2 Understanding the Elements of a Typical Enterprise Deployment Topology

The enterprise deployment topology consists of the following high-level elements:

- A hardware load balancer that routes requests from the Internet to the Web servers in the Web tier. It also routes requests from internal clients or other components that are performing internal invocations within the corporate network.
- A Web tier, consisting of a hardware load balancer and two or more physical computers that host the Web server instances (for high availability).

The Web server instances are configured to authenticate users (via an external identity store and a single sign-on server) and then route the HTTP requests to the Oracle Fusion Middleware products and components running in the Application tier.

The Web server instances also host static Web content that does not require application logic to be delivered. Placing such content in the Web tier reduces the overhead on the application servers and eliminates unnecessary network activity.

- An Application tier, consisting of two or more physical computers that are hosting a cluster of Oracle WebLogic Server Managed Servers, and the Administration Server for the domain. The Managed Servers are configured to run the various Oracle Fusion Middleware products, such as Oracle SOA Suite, Oracle Service Bus, Oracle WebCenter Content, and Oracle WebCenter Portal, depending on your choice of products in the enterprise deployment.
- A data tier, consisting of two or more physical hosts that are hosting an Oracle RAC Database.

2.2.3 Receiving Requests Through Hardware Load Balancer

The following topics describe the hardware load balancer and its role in an enterprise deployment.

See Also:

[Purpose of the Hardware Load Balancer \(LBR\)](#)

[Summary of the Typical Load Balancer Virtual Server Names](#)

[HTTPS versus HTTP Requests to the External Virtual Server Name](#)

2.2.3.1 Purpose of the Hardware Load Balancer (LBR)

The following topics describe the types of requests handled by the hardware load balancer in an enterprise deployment.

See Also:

[HTTP Requests from the Internet to the Web server instances in the Web tier](#)

[MLLP Requests for Oracle SOA Suite for Healthcare Integration](#)

[Specific internal-only communications between the components of the Application tier](#)

2.2.3.1.1 HTTP Requests from the Internet to the Web server instances in the Web tier

The hardware load balancer balances the load on the Web tier by receiving requests to a single virtual host name and then routing each request to one of the Web server instances, based on a load balancing algorithm. In this way, the load balancer ensures that no one Web server is overloaded with HTTP requests.

For more information about the purpose of specific virtual host names on the hardware load balancer, see [Summary of the Typical Load Balancer Virtual Server Names](#).

Note that in the reference topology, only HTTP requests are routed from the hardware load balancer to the Web tier. Secure Socket Layer (SSL) requests are terminated at the load balancer and only HTTP requests are forwarded to the Oracle HTTP Server instances. This guide does not provide instructions for SSL configuration between the load balancer and the Oracle HTTP Server instances or between the Web tier and the Application tier.

The load balancer provides high availability by ensuring that if one Web server goes down, requests will be routed to the remaining Web servers that are up and running.

Further, in a typical highly available configuration, the hardware load balancers are configured such that a hot standby device is ready to resume service in case a failure occurs in the main load balancing appliance. This is important because for many types of services and systems, the hardware load balancer becomes the unique point of access to make invocations and, as a result, becomes a single point of failure (SPOF) for the whole system if it is not protected.

2.2.3.1.2 MLLP Requests for Oracle SOA Suite for Healthcare Integration

If you plan to configure Oracle SOA Suite for healthcare integration, then the hardware load balancer must also pass requests via the Minimum Lower Layer Protocol (MLLP) protocol. MLLP is required when you are using the Health Level 7 (HL7) standards to exchange healthcare documents.

For more information, see "Using the HL7 Document Protocol" in the *Healthcare Integration User's Guide for Oracle SOA Suite*.

2.2.3.1.3 Specific internal-only communications between the components of the Application tier

In addition, the hardware load balancer routes specific communications between the Oracle Fusion Middleware components and applications on the application tier. The internal-only requests are also routed through the load balancer, using a unique virtual host name.

2.2.3.2 Summary of the Typical Load Balancer Virtual Server Names

In order to balance the load on servers and to provide high availability, the hardware load balancer is configured to recognize a set of virtual server names. As shown in the diagram, the following virtual server names are recognized by the hardware load balancer in this topology:

- `product.example.com` - This virtual server name is used for all incoming traffic. Users enter this URL to access the Oracle Fusion Middleware product you have deployed and the custom applications available on this server. The load balancer then routes these requests (using a load balancing algorithm) to one of the servers in the Web tier. In this way, the single virtual server name can be used to route traffic to multiple servers for load balancing and high availability of the Web servers instances.
- `productinternal.example.com` - This virtual server name is for internal communications only.

The load balancer uses its **Network Address Translation (NAT)** capabilities to route any internal communication from the Application tier components that are directed to this URL. This URL is not exposed to external customers or users on the

Internet. Each product has specific uses for the internal URL, so in the deployment instructions, we prefix it with the product name.

- `admin.example.com` - This virtual server name is for administrators who need to access the Oracle Enterprise Manager Fusion Middleware Control and Oracle WebLogic Server Administration Console interfaces.

This URL is known only to internal administrators. It also uses the NAT capabilities of the load balancer to route administrators to the active Administration Server in the domain.

For the complete set of virtual server names you must define for your topology, see the the chapter that describes the product-specific topology.

2.2.3.3 HTTPS versus HTTP Requests to the External Virtual Server Name

Note that when you configure the hardware load balancer, a best practice is to assign the main external URL (for example, `http://myapplication.example.com`) to port 80 and port 443.

Any request on port 80 (non-SSL protocol) should be redirected to port 443 (SSL protocol). Exceptions to this rule include requests from public WSDLs. For more information, see [Configuring Virtual Hosts on the Hardware Load Balancer](#).

2.2.4 Understanding the Web Tier

The Web tier of the reference topology consists of the Web servers that receive requests from the load balancer. In the typical enterprise deployment, at least two Oracle HTTP Server instances are configured in the Web tier. The following topics provide more detail.

See Also:

[Benefits of Using Oracle HTTP Server Instances to Route Requests](#)

[Alternatives to Using Oracle HTTP Server in the Web Tier](#)

[Configuration of Oracle HTTP Server in the Web Tier](#)

[About Mod_WL_OHS](#)

2.2.4.1 Benefits of Using Oracle HTTP Server Instances to Route Requests

A Web tier with Oracle HTTP Server is not a requirement for many of the Oracle Fusion Middleware products. You can route traffic directly from the hardware load balancer to the WLS servers in the Application Tier. However, a Web tier does provide several advantages, which is why it is recommended as part of the reference topology:

- The Web tier provides DMZ public zone, which is a common requirement in security audits. If a load balancer routes directly to the WebLogic Server, requests move from the load balancer to the application tier in one single HTTP jump, which can cause security concerns.
- The Web tier allows the WebLogic Server cluster membership to be reconfigured (new servers added, others removed) without having to change the Web server configuration (as long as at least some of the servers in the configured list remain alive).
- Oracle HTTP Server delivers static content more efficiently and faster than WebLogic Server; it also provides FTP services, which are required for some

enterprise deployments, as well as the ability to create virtual hosts and proxies via the Oracle HTTP Server configuration files.

- Oracle HTTP Server provides HTTP redirection over and above what WebLogic Server provides. You can use Oracle HTTP Server as a front end against many different WebLogic Server clusters, and in some cases, control the routing via content based routing.
- Oracle HTTP Server provides the ability to integrate single sign-on capabilities into your enterprise deployment. For example, you can later implement single sign-on for the enterprise deployment, using Oracle Access Manager, which is part of the Oracle Identity and Access Management family of products.
- Oracle HTTP Server provides support for WebSocket connections deployed within WebLogic Server.

For more information about Oracle HTTP Server, see Introduction to Oracle HTTP Server in *Administrator's Guide for Oracle HTTP Server*.

2.2.4.2 Alternatives to Using Oracle HTTP Server in the Web Tier

Although Oracle HTTP Server provides a variety of benefits in an enterprise topology, Oracle also supports routing requests directly from the hardware load balancer to the Managed Servers in the middle tier.

This approach provide the following advantages:

- Lower configuration and processing overhead than using a front-end Oracle HTTP Server Web tier front-end.
- Monitoring at the application level since the LBR can be configured to monitor specific URLs for each Managed Server (something that is not possible with OHS).

You can potentially use this load balancer feature to monitor SOA composite application URLs. Note that this enables routing to the Managed Servers only when all composites are deployed, and you must use the appropriate monitoring software.

2.2.4.3 Configuration of Oracle HTTP Server in the Web Tier

Starting with Oracle Fusion Middleware 12c, the Oracle HTTP Server software can be configured in one of two ways: as part of an existing Oracle WebLogic Server domain or in its own standalone domain. Each configuration offers specific benefits.

When you configure Oracle HTTP Server instances as part of an existing WebLogic Server domain, you can manage the Oracle HTTP Server instances, including the wiring of communications between the Web servers and the Oracle WebLogic Server Managed Servers using Oracle Enterprise Manager Fusion Middleware Control. When you configure Oracle HTTP Server in a standalone configuration, you can configure and manage the Oracle HTTP Server instances independently of the application tier domains.

For this enterprise deployment guide, the Oracle HTTP Server instances are configured as separate standalone domains, one on each Web tier host. You can choose to configure the Oracle HTTP Server instances as part of the application tier domain, but this enterprise deployment guide does not provide specific steps to configure the Oracle HTTP Server instances in that manner.

For more information, see "Understanding Oracle HTTP Server Installation Options" in *Installing and Configuring Oracle HTTP Server*.

2.2.4.4 About Mod_WL_OHS

As shown in the diagram, the Oracle HTTP Server instances use the WebLogic Proxy Plug-In (`mod_wl_ohs`) for proxying HTTP requests from Oracle HTTP Server to the Oracle WebLogic Server Managed Servers in the Application tier.

For more information, see "Overview of Web Server Proxy Plug-Ins 12.1.3" in *Using Oracle WebLogic Server Proxy Plug-Ins 12.1.3*.

2.2.5 Understanding the Application Tier

The application tier consists of two physical host computers, where Oracle WebLogic Server and the Oracle Fusion Middleware products are installed and configured. The application tier computers reside in the secured zone between firewall 1 and firewall 2.

The following topics provide more information.

See Also:

[Configuration of the Administration Server and Managed Servers Domain Directories](#)

[Using Oracle Web Services Manager in the Application Tier](#)

[Best Practices and Variations on the Configuration of the Clusters and Hosts on the Application Tier](#)

[About the Node Manager Configuration in a Typical Enterprise Deployment](#)

[About Using Unicast for Communications Within the Application Tier](#)

[Understanding OPSS and Requests to the Authentication and Authorization Stores](#)

[About Coherence Clusters In a Typical Enterprise Deployment](#)

2.2.5.1 Configuration of the Administration Server and Managed Servers Domain Directories

Unlike the Managed Servers in the domain, the Administration Server uses an active-passive high availability configuration. This is because only one Administration Server can be running within an Oracle WebLogic Server domain.

In the topology diagrams, the Administration Server on HOST1 is in the active state and the Administration Server on HOST2 is in the passive (inactive) state.

To support the manual fail over of the Administration Server in the event of a system failure, the typical enterprise deployment topology includes:

- A Virtual IP Address (VIP) for the routing of Administration Server requests
- The configuration of the Administration Server domain directory on a shared storage device.

In the event of a system failure (for example a failure of HOST1), you can manually reassign the Administration Server VIP address to another host in the domain, mount the Administration Server domain directory on the new host, and then start the Administration Server on the new host.

However, unlike the Administration Server, there is no benefit to storing the Managed Servers on shared storage. In fact, there is a potential performance impact when

Managed Server configuration data is not stored on the local disk of the host computer.

As a result, in the typical enterprise deployment, after you configure the Administration Server domain on shared storage, a copy of the domain configuration is placed on the local storage device of each host computer, and the Managed Servers are started from this copy of the domain configuration. You create this copy using the Oracle WebLogic Server pack and unpack utilities.

The resulting configuration consists of separate domain directories on each host: one for the Administration Server (on shared storage) and one for the Managed Servers (on local storage). Depending upon the action required, you must perform configuration tasks from one domain directory or the other.

For more information about structure of the Administration Server domain directory and the Managed Server domain directory, as well as the variables used to reference these directories, see [Understanding the Recommended Directory Structure for an Enterprise Deployment](#).

There is an additional benefit to the multiple domain directory model. It allows you to isolate the Administration Server from the Managed Servers. By default, the primary enterprise deployment topology assumes the Administration Server domain directory is on one of the Application Tier hosts, but if necessary, you could isolate the Administration Server further by running it from its own host, for example in cases where the Administration Server is consuming high CPU or RAM. Some administrators prefer to configure the Administration Server on a separate, dedicated host, and the multiple domain directory model makes that possible.

2.2.5.2 Using Oracle Web Services Manager in the Application Tier

Oracle Web Services Manager (Oracle WSM) provides a policy framework to manage and secure Web services in the Enterprise Deployment topology.

In most enterprise deployment topologies, the Oracle Web Services Manager Policy Manager runs on Managed Servers in a separate cluster, where it can be deployed in an active-active highly available configuration.

You can choose to target Oracle Web Services Manager and Fusion Middleware products or applications to the same cluster, as long as you are aware of the implications.

The main reasons for deploying Oracle Web Services Manager on its own managed servers is to improve performance and availability isolation. Oracle Web Services Manager often provides policies to custom Web services or to other products and components in the domain. In such a case, you do not want the additional Oracle Web Services Manager activity to affect the performance of any applications that are sharing the same managed server or cluster as Oracle Web Services Manager.

The eventual process of scaling out or scaling up is also better addressed when the components are isolated. You can scale out or scale up only the Fusion Middleware application Managed Servers where your products are deployed or only the Managed Servers where Oracle Web Services Manager is deployed, without affecting the other product.

2.2.5.3 Best Practices and Variations on the Configuration of the Clusters and Hosts on the Application Tier

In a typical enterprise deployment, you configure the Managed Servers in a cluster on two or more hosts in the application tier. For specific Oracle Fusion Middleware products, the enterprise deployment reference topologies demonstrate best practices

for the number of Managed Servers, the number of clusters, and what services are targeted to each cluster.

These best practices take into account typical performance, maintenance, and scale-out requirements for each product. The result is the grouping of Managed Servers into an appropriate set of clusters within the domain.

Variations of the enterprise deployment topology allow the targeting of specific products or components to additional clusters or hosts for improved performance and isolation.

For example, you can consider hosting the Administration Server on a separate and smaller host computer, which allows the FMW components and products to be isolated from the Administration Server.

For another example, in an Oracle SOA Suite deployment, you might deploy Oracle SOA Suite and Oracle Service Bus on different hosts. Similarly, you might target Oracle Business Activity Monitoring and Enterprise Scheduler to a separate cluster on separate host computers.

These variations in the topology are supported, but the enterprise deployment reference topology uses the minimum hardware resources while keeping high availability, scalability and security in mind. You should perform the appropriate resource planning and sizing, based on the system requirements for each type of server and the load that the system needs to sustain. Based on these decisions, you must adapt the steps to install and configure these variations accordingly from the instructions presented in this guide.

2.2.5.4 About the Node Manager Configuration in a Typical Enterprise Deployment

Starting with Oracle Fusion Middleware 12c, you can use either a per domain Node Manager or a per host Node Manager. The following sections of this topic provide more information on the impact of the Node Manager configuration on a typical enterprise deployment.

Note:

For general information about these two types of Node Managers, see Overview in *Administering Node Manager for Oracle WebLogic Server*.

About Using a Per Domain Node Manager Configuration

In a per domain Node Manager configuration—as opposed to a per host Node Manager configuration—you actually start two Node Manager instances on the Administration Server host: one from the Administration Server domain directory and one from the Managed Servers domain directory. In addition, a separate Node Manager instance runs on each of the other hosts in the topology.

The Node Manager controlling the Administration Server uses the listen address of the virtual host name created for the Administration Server. The Node Manager controlling the Managed Servers uses the listen address of the physical host. When the Administration Server fails over to another host, an additional instance of Node Manager is started to control the Administration Server on the failover host.

The key advantages of the per domain configuration are an easier and simpler initial setup of the Node Manager and the ability to set Node Manager properties that are unique to the Administration Server. This last feature was important in previous

releases because some features, such as Crash Recovery, applied only to the Administration Server and not to the Managed servers. In the current release, the Oracle SOA Suite products can be configured for Automated Service Migration, rather than Whole Server Migration. This means the Managed Servers, as well as the Administration Server, can take advantage of Crash Recovery, so there is no need to apply different properties to the Administration Server and Managed Server domain directories.

Another advantage is that the per domain Node Manager provides a default SSL configuration for Node Manager-to-Server communication, based on the Demo Identity store created for each domain.

About Using a Per Host Domain Manager Configuration

In a per-host Node Manager configuration, you start a single Node Manager instance to control the Administration Server and all Managed Servers on a host, even those that reside in different domains. This reduces the footprint and resource utilization on the Administration Server host, especially in those cases where multiple domains coexist on the same machine.

A per-host Node Manager configuration allows all Node Managers to use a listen address of ANY, so they listen on all addresses available on the host. This means that when the Administration Server fails over to a new host, no additional configuration is necessary. The per host configuration allows for simpler maintenance, because you can update and maintain a single Node Manager properties file on each host, rather than multiple node manager property files.

The per-host Node Manager configuration requires additional configuration steps. If you want SSL for Node Manager-to-Server communication, then you must configure an additional Identity and Trust store, and it also requires using Subject Alternate Names (SAN), because the Node Manager listens on multiple addresses. Note that SSL communications are typically not required for the application tier, because it is protected by two firewalls.

2.2.5.5 About Using Unicast for Communications Within the Application Tier

Oracle recommends the unicast communication protocol for communication between the Managed Servers and hosts within the Oracle WebLogic Server clusters in an enterprise deployment. Unlike multicast communication, unicast does not require cross-network configuration and it reduces potential network errors that can occur from multicast address conflicts as well.

When you consider using the multicast or unicast protocol for your own deployment, consider the type of network, the number of members in the cluster, and the reliability requirements for cluster membership. Also consider the following benefits of each protocol.

Benefits of unicast in an enterprise deployment:

- Uses a group leader that every server sends messages directly to. This leader is responsible for retransmitting the message to every other group member and other group leaders, if applicable.
- Works out of the box in most network topologies
- Requires no additional configuration, regardless of the network topology.
- Uses a single missed heartbeat to remove a server from the cluster membership list.

Benefits of multicast in an enterprise deployment:

- Multicast uses a more scalable peer-to-peer model where a server sends each message directly to the network once and the network makes sure that each cluster member receives the message directly from the network.
- Works out of the box in most modern environments where the cluster members are in a single subnet.
- Requires additional configuration in the router(s) and WebLogic Server (that is., Multicast TTL) if the cluster members span more than one subnet.
- Uses three consecutive missed heartbeats to remove a server from the cluster membership list.

Depending on the number of servers in your cluster and on whether the cluster membership is critical for the underlying application (for example in session-replication intensive applications or clusters with intensive RMI invocations across the cluster), each model may behave better.

Consider whether your topology is going to be part of an Active-Active disaster recovery system or if the cluster is going to traverse multiple subnets. In general, unicast will behave better in those cases.

For more information see the following resources:

- "Configuring Multicast Messaging for WebLogic Server Clusters" in the *High Availability Guide*
- "One-to-Many Communication Using Unicast" in *Administering Clusters for Oracle WebLogic Server*.

2.2.5.6 Understanding OPSS and Requests to the Authentication and Authorization Stores

Many of the Oracle Fusion Middleware products and components require an Oracle Platform Security Services (OPSS) security store for authentication providers (an identity store), policies, credentials, keystores, and for audit data. As a result, communications must be enabled so the Application tier can send requests to and from the security providers.

For authentication, this communication is to an LDAP directory, such as Oracle Internet Directory (OID) or Oracle Unified Directory (OUD), which typically communicates over port 389 or 636. When you configure an Oracle Fusion Middleware domain, the domain is configured by default to use the WebLogic Server Authentication provider. However, for an enterprise deployment, you must use a dedicated, centralized LDAP-compliant authentication provider.

For authorization (and the policy store), the location of the security store varies, depending upon the tier:

- For the application tier, the authorization store is database-based, so frequent connections from the Oracle WebLogic Server Managed Servers to the database are required for the purpose of retrieving the required OPSS data.
- For the Web tier, the authorization store is file-based, so connections to the database are not required.

For more information about OPSS security stores, see the following sections of *Securing Applications with Oracle Platform Security Services*:

- "Authentication Basics"
- "The OPSS Policy Model"

2.2.5.7 About Coherence Clusters In a Typical Enterprise Deployment

The standard Oracle Fusion Middleware enterprise deployment includes a Coherence cluster that contains storage-enabled Managed Coherence Servers.

This configuration is a good starting point for using Coherence, but depending upon your specific requirements, you can consider tuning and reconfiguring Coherence to improve performance in a production environment or to resolve possible port conflicts.

When reviewing port assignments, note that the Oracle Fusion Middleware products and components default to a Well Known Address (WKA) list that uses the port specified on the Coherence Clusters screen of the Configuration Wizard. The WKA list also uses the listen address of all servers that participate in the coherence cluster as the listen address for the WKA list. These settings can be customized using the WLS Administration Console.

For more information, refer to the following resources:

- For information about Coherence clusters, see "Configuring and Managing Coherence Clusters" in *Administering Clusters for Oracle WebLogic Server*.
- For information about tuning Coherence, see *Administering Oracle Coherence*.
- For information about storing HTTP session data in Coherence, see "Using Coherence*Web with WebLogic Server" in *Administering HTTP Session Management with Oracle Coherence*Web*.
- For more information about creating and deploying Coherence applications, see *Developing Oracle Coherence Applications for Oracle WebLogic Server*.

2.2.6 About the Data Tier

In the Data tier, an Oracle RAC database runs on the two hosts (DBHOST1 and DBHOST2). The database contains the schemas required by the Oracle SOA Suite components and the Oracle Platform Security Services (OPSS) policy store.

You can define multiple services for the different products and components in an enterprise deployment to isolate and prioritize throughput and performance accordingly. In this guide, one database service is used as an example. Furthermore, you can use other high availability database solutions to protect the database:

- Oracle Data Guard; for more information, see the *Oracle Data Guard Concepts and Administration*
- Oracle RAC One Node; for more information, see "Overview of Oracle RAC One Node" in the *Oracle Real Application Clusters Administration and Deployment Guide*

These solutions above provide protection for the database beyond the information provided in this guide, which focuses on using an Oracle RAC Database, given the scalability and availability requirements that typically apply to an enterprise deployment.

For more information about using Oracle Databases in a high availability environment, see "Database Considerations" in the *High Availability Guide*.

Understanding the Oracle SOA Suite Enterprise Deployment Topology

The following topics introduce and describe the Oracle SOA Suite enterprise deployment topologies. These topologies represent specific reference implementations of the concepts described in [Understanding a Typical Enterprise Deployment](#).

See Also:

[Understanding an Enterprise Deployment](#)

[Understanding the Primary and Build-Your-Own Enterprise Deployment Topologies](#)

This section provides instructions for installing and configuring the primary topologies and information about adding components required for your environment.

[Diagrams of the Primary Oracle SOA Suite Enterprise Topologies](#)

This section provides the diagrams of the two primary Oracle SOA Suite enterprise deployment topologies.

[Understanding the Primary Oracle SOA Suite Topology Diagrams](#)

This section provides information about the elements that are unique to the primary topology.

[Flow Charts and Roadmaps for Implementing the Primary Oracle SOA Suite Enterprise Topologies](#)

This section provides instructions in the form of flow charts and road maps to install and configure the enterprise deployment topology.

[Building Your Own Oracle SOA Suite Enterprise Topology](#)

This section describes the alternative topologies that you can implement, using some variations of the instructions in this guide.

[About Installing and Configuring a Custom Enterprise Topology](#)

If you choose to implement a topology that is not described in this guide, be sure to review the certification information, system requirements, and interoperability requirements for the products you want to include in the topology.

[About Using Automatic Service Migration for the Oracle SOA Suite Enterprise Topology](#)

To ensure high availability of the Oracle SOA Suite products and components, this guide recommends that you enable Oracle WebLogic

Server Automatic Service Migration for the clusters that you create as part of the reference topology.

3.1 Understanding the Primary and Build-Your-Own Enterprise Deployment Topologies

This section provides instructions for installing and configuring the primary topologies and information about adding components required for your environment.

This guide focuses on one or more primary reference topologies for the selected product. The exact topology you install and configure for your organization might vary, but for the primary topologies, this guide provides step-by-step instructions for installing and configuring those topologies.

In addition, this guide provides high-level information about how to design and build your own enterprise deployment topology. For the build-your-own topologies, the guide also provides information about how to add specific components or products required for your specific environment.

3.2 Diagrams of the Primary Oracle SOA Suite Enterprise Topologies

This section provides the diagrams of the two primary Oracle SOA Suite enterprise deployment topologies.

See Also:

[Diagram of the Oracle SOA Suite and Oracle Service Bus Topology](#)

[Diagram of the Oracle SOA Suite and Oracle Business Activity Monitoring Topology](#)

3.2.1 Diagram of the Oracle SOA Suite and Oracle Service Bus Topology

[Figure 3-1](#) shows a diagram of the Oracle SOA and Oracle Service Bus enterprise deployment topology.

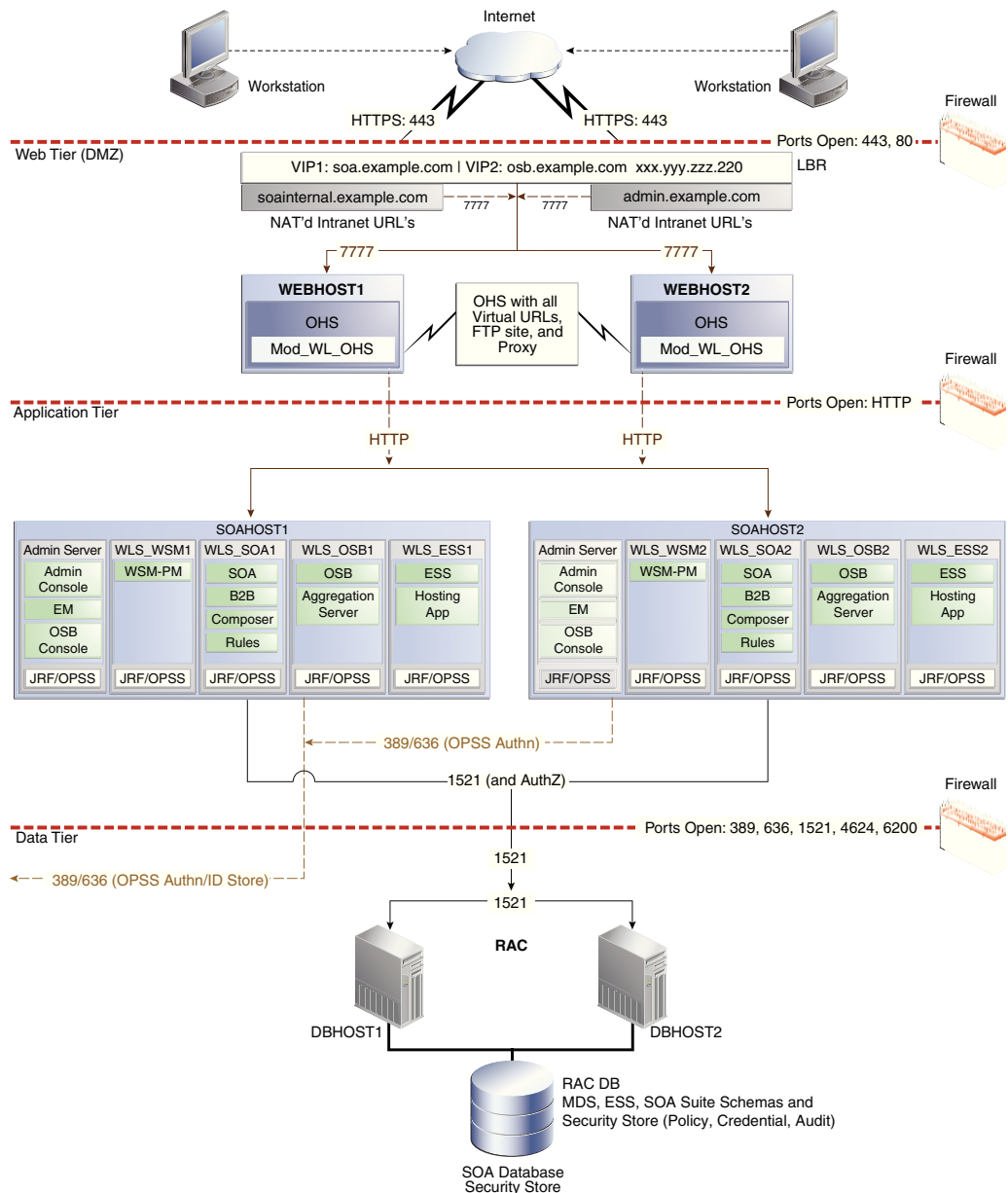
Note:

You can configure Oracle Service Bus in the same domain as Oracle SOA Suite or in its own domain. For more information, see [Understanding the Topology Options for Oracle Service Bus](#).

For a description of the standard elements shown in the diagram, see [Understanding the Typical Enterprise Deployment Topology Diagram](#).

For a description of the elements shown in the diagram, see [Understanding the Primary Oracle SOA Suite Topology Diagrams](#).

Figure 3-1 Oracle SOA Suite and Oracle Service Bus Enterprise Deployment Reference Topology Diagram



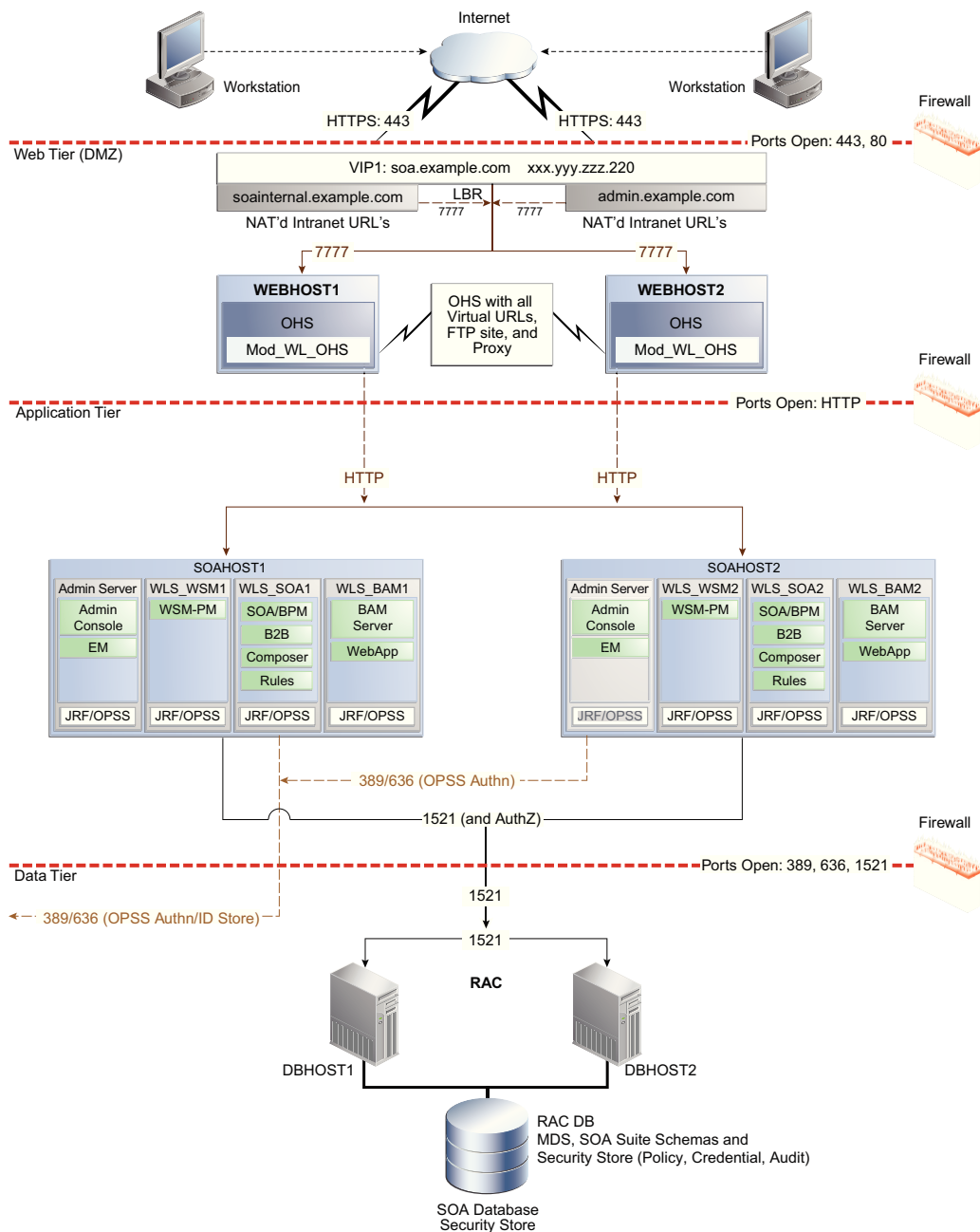
3.2.2 Diagram of the Oracle SOA Suite and Oracle Business Activity Monitoring Topology

Figure 3-2 shows a diagram of the Oracle SOA Suite and Oracle Business Activity Monitoring enterprise topology.

For a description of the standard elements shown in the diagram, see [Understanding the Typical Enterprise Deployment Topology Diagram](#).

For a description of the elements that are specific to the Oracle SOA Suite topologies, see [Understanding the Primary Oracle SOA Suite Topology Diagrams](#).

Figure 3-2 Oracle SOA Suite and Oracle Business Activity Monitoring Enterprise Topology Diagram



3.3 Understanding the Primary Oracle SOA Suite Topology Diagrams

This section provides information about the elements that are unique to the primary topology.

Most of the elements of Oracle SOA Suite topologies represent standard features of any enterprise topology that follows the Oracle-recommended best practices. These elements are described detail in [Understanding a Typical Enterprise Deployment](#).

Before you review the information here, it is assumed you have reviewed the information in [Understanding a Typical Enterprise Deployment](#) and that you are familiar with the general concepts of an enterprise deployment topology.

See the following sections for information about the elements that are unique to the topology described in this chapter:

See Also:

[Understanding the Topology Options for Oracle Service Bus](#)

[Summary of Oracle SOA Suite Load Balancer Virtual Server Names](#)

[Understanding the Routing of SOA Composite Requests](#)

[Summary of the Managed Servers and Clusters on SOA Application Tier](#)

3.3.1 Understanding the Topology Options for Oracle Service Bus

The Oracle SOA Suite and Oracle Service Bus topology diagram in this guide assumes a single domain that contains both SOA Suite and Oracle Service Bus. However, it is often advantageous to configure Oracle Service Bus in its own domain.

For example, consider separate domains when you are using Oracle Service Bus on an enterprise scale. In this scenario, you can then use Oracle Service Bus to route to multiple SOA domains and other services.

On the other hand, if you are using Oracle Service Bus primarily for mediating and providing routing for SOA Suite composite applications, configure Oracle Service Bus in the same domain, but in separate clusters for optimum performance and scalability.

When considering these options, take into account patching and other life cycle maintenance operations. For example, Oracle SOA Suite and Oracle SOA Suite sometimes have differing patching requirements. If the two products are in separate domains, it can be easier to patch one without affecting the other.

3.3.2 Summary of Oracle SOA Suite Load Balancer Virtual Server Names

In order to balance the load on servers and to provide high availability, the hardware load balancer is configured to recognize a set of virtual server names.

For information about the purpose of each of these server names, see [Summary of the Typical Load Balancer Virtual Server Names](#).

The following virtual server names are recognized by the hardware load balancer in Oracle SOA Suite topologies:

- `soa.example.com` - This virtual server name is used for all incoming traffic. It acts as the access point for all HTTP traffic to the runtime SOA components. The load balancer routes all requests to this virtual server name over SSL. As a result, clients access this service using the following secure address:
`soa.example.com:443`
- `osb.example.com` - This virtual server name that acts as the access point for all HTTP traffic to the runtime Oracle Service Bus resources and proxy services. The load balancer routes all requests to this virtual server name over SSL. As a result, clients access this service using the following secure address:
`osb.example.com:443`
- `soainternal.example.com` - This virtual server name is for internal communications between the application tier components only and is not exposed to the Internet.

Specifically, for the Oracle SOA Suite enterprise topology, this URL is used for both Oracle SOA Suite and Oracle Service Bus internal communications.

The traffic from clients to this URL is not SSL-enabled. Clients access this service using the following address and the requests are forwarded to port 7777 on WEBHOST1 and WEBHOST2:

```
soainternal.example.com:80
```

Note that this URL can also be set as the URL to be used for internal service invocations while modeling composites or at runtime with the appropriate Enterprise Manager MBeans. For more information, see [More About the soainternal Virtual Server Name](#).

- `admin.example.com` - This virtual server name is for administrators who need to access the Oracle Enterprise Manager Fusion Middleware Control and Oracle WebLogic Server Administration Console interfaces.

Instructions later in this guide explain how to:

- Configure the hardware load balancer to recognize and route requests to the virtual host names.
- Configure the Oracle HTTP Server instances on the Web Tier to recognize and properly route requests to the virtual host names and the correct host computers.

3.3.3 Understanding the Routing of SOA Composite Requests

The following topics provide additional information on configuring the enterprise deployment for Oracle SOA Suite composite applications.

See Also:

[More About the soainternal Virtual Server Name](#)

[About Web Services Optimizations for SOA Composite Applications](#)

[About Accessing SOA Composite Applications via Oracle HTTP Server](#)

[About Accessing Oracle SOA Suite Composite Applications Via the Load Balancer](#)

3.3.3.1 More About the soainternal Virtual Server Name

The `soainternal.example.com` virtual server name functions exactly the same as the `soa.example.com`, except that it is invoked by intranet clients and callbacks only. This topic provides additional details.

The `soainternal.example.com` virtual server name is not used explicitly during the installation and configuration of the enterprise deployment, but custom systems often expose services that should be consumed by internal-only clients. In those cases, for efficiency and security reasons, you should avoid using an external URL such as `soa.example.com`. Instead, you should use an address that cannot be invoked by Internet clients. SOA Composite applications, in particular, can use this internal URL in their end points, either directly or through deployment plans.

When you use the `soainternal.example.com` address, there are implications for the front end address specified for the system. Web Services optimizations (for example, direct RMI invocation instead of invocations that involve a full loopback to the load balancer endpoint) are triggered when the Front End address for the cluster

matches the invocation endpoint. For this reason, depending on the number and relevance of the expected internal invocations, consider setting the front end URL for the cluster and the `ServerURL` and `HTTPServerURL` properties to either the external or internal.

You can set the front end URL for a cluster when you are creating the cluster in the Configuration Wizard. You can also modify it later, using the Weblogic Server Administration Console. For more information, see “Configure HTTP Protocol” in the *Administration Console Online Help*.

For more information about setting the `ServerURL` and `HTTPServerURL` properties, see *Configuring SOA Infrastructure Properties*.

3.3.3.2 About Web Services Optimizations for SOA Composite Applications

When configuring internal callbacks so that SOA composite applications can communicate efficiently within the enterprise deployment, you should be aware of how the system checks for the proper end-point address for each request.

For webservice local optimization, the basic requirement is to make sure that the two SOA composites are colocated on the same Managed Server or process. To determine if the composites are colocated on the same server, Oracle SOA Suite compares the server on which the target service composite is deployed (host and port configuration) with those specified in the reference service endpoint URI.

- For target service host value, here is the sequence of checks in order precedence:
 - Checks the Server URL configuration property value on SOA Infrastructure Common Properties page.
 - If not specified, checks the `FrontendHost` and `FrontendHTTPPort` (or `FrontendHTTPSPort` if SSL is enabled) configuration property values from the cluster MBeans.
 - If not specified, checks the `FrontendHost` and `FrontendHTTPPort` (or `FrontendHTTPSPort` if SSL is enabled) configuration property values from the Oracle WebLogic Server MBeans.
 - If not specified, uses the DNS-resolved Inet address of localhost.
- For target service port value, here is the sequence of checks in order precedence:
 - Checks the port configured in `HttpServerURL` on SOA Infrastructure Common Properties page.
 - If not specified, checks the port configured in Server URL on SOA Infrastructure Common Properties page.
 - If not specified, checks the `FrontendHost` and `FrontendHTTPPort` (or `FrontendHTTPSPort` if SSL is enabled) configuration property values from the cluster MBeans.
 - If not specified, checks the `FrontendHost` and `FrontendHTTPPort` (or `FrontendHTTPSPort` if SSL is enabled) configuration property values from the Oracle WebLogic Server MBean.
 - If not specified, SOA Suite assumes 80 for HTTP and 443 for HTTPS URLs.

3.3.3.3 About Accessing SOA Composite Applications via Oracle HTTP Server

When routing requests from the Oracle HTTP Server instances on the Web tier to specific Oracle SOA Suite composite application URLs on the application, consider the following:

- In previous releases of Oracle Fusion Middleware, Oracle HTTP Server generated an HTTP 503 (Service Unavailable) message if a request to Oracle SOA Suite composite application was received by the Managed Server and the composite application was not yet loaded.
- In Oracle Fusion Middleware 12c, this behavior has changed. If requests for a composite arrives before the composite is active, then the HTTP requests are put on hold until the required artifacts are available and the composite reaches the active state.

This change in behavior allows you to route requests to composite applications that are not yet loaded during the startup of an Oracle SOA Suite Managed Server. However, the communication channel between the Oracle HTTP Server and Oracle WebLogic Server needs to account for the possibility of long delays in getting replies.

To address this issue, while configuring firewalls between Oracle HTTP Server and Oracle WebLogic Server, set the appropriate timeout to avoid shutting down of connections that are waiting for a composite to be loaded.

For more information, see [“Configuring the Firewalls and Ports for an Enterprise Deployment”](#).

Note that the Oracle HTTP Server instances route requests based on the availability of the Oracle WebLogic Server servers and not on the availability of any specific application. The instances continue to route the requests as long as the Oracle WebLogic Server is up and running.

3.3.3.4 About Accessing Oracle SOA Suite Composite Applications Via the Load Balancer

In the default configuration, the hardware load balancer routes all requests to the Web tier, which then routes the requests to the appropriate resource in the application tier.

However, you can configure the hardware load balancer to route directly to Managed Servers on the application tier. This configuration has some benefits, especially in an Oracle SOA Suite enterprise deployment:

- Configuration and processing overhead is lower than when using Oracle HTTP Server.
- It enables monitoring at the application level, because the load balancer can be configured to monitor specific URLs in each WLS server (something that is not possible with Oracle HTTP Server).

There is at least one disadvantage to this approach. If requests are routed directly from the load balancer to the Managed Servers, then each request will cross two firewalls without any proxy or interception. This might a security issue, depending on the network security policies in your organization.

If Oracle HTTP server directs an HTTP request for a composite to a Oracle SOA Suite Managed Server and the `soa-infra` application is not yet active, then the request will fail. Therefore, you should always verify that the `soa-infra` application is active after you start, restart, or migrate a server.

3.3.4 Summary of the Managed Servers and Clusters on SOA Application Tier

The Application tier hosts the Administration Server and Managed Servers in the Oracle WebLogic Server domain.

Depending upon the topology you select, the Oracle WebLogic Server domain for the Oracle SOA Suite domain consists of the clusters shown in [Table 3-1](#). These clusters function as active-active high availability configurations.

Table 3-1 Summary of the Clusters in the Oracle SOA Suite Enterprise Deployment Topology

Cluster	Managed Servers
Oracle SOA Suite, Oracle Business Process Management, and Oracle B2B Cluster	WLS_SOA1, WLS_SOA2
Oracle Web Services Manager Cluster	WLS_WSM1, WLS_WSM2
Oracle Service Bus Cluster	WLS_OSB1, WLS_OSB2
Oracle Oracle Enterprise Scheduler	WLS_ESS1, WLS_ESS2
Oracle Business Activity Monitoring Cluster	WLS_BAM1, WLS_BAM2

3.4 Flow Charts and Roadmaps for Implementing the Primary Oracle SOA Suite Enterprise Topologies

This section provides instructions in the form of flow charts and road maps to install and configure the enterprise deployment topology.

The following sections summarize the high-level steps you must perform to install and configure the enterprise topology described in this chapter.

See Also:

[Flow Chart of the Steps to Install and Configure the Primary Oracle SOA Suite Enterprise Topologies](#)

[Roadmap Table for Planning and Preparing for an Enterprise Deployment](#)

[Roadmap Table for Configuring the Oracle SOA Suite and Oracle Service Bus Enterprise Topology](#)

[Roadmap Table for Configuring the Oracle SOA Suite and Oracle Business Activity Monitoring Enterprise Topology](#)

3.4.1 Flow Chart of the Steps to Install and Configure the Primary Oracle SOA Suite Enterprise Topologies

[Figure 3-3](#) shows a flow chart of the steps required to install and configure the primary enterprise deployment topologies described in this chapter. The sections following the flow chart explain each step in the flow chart.

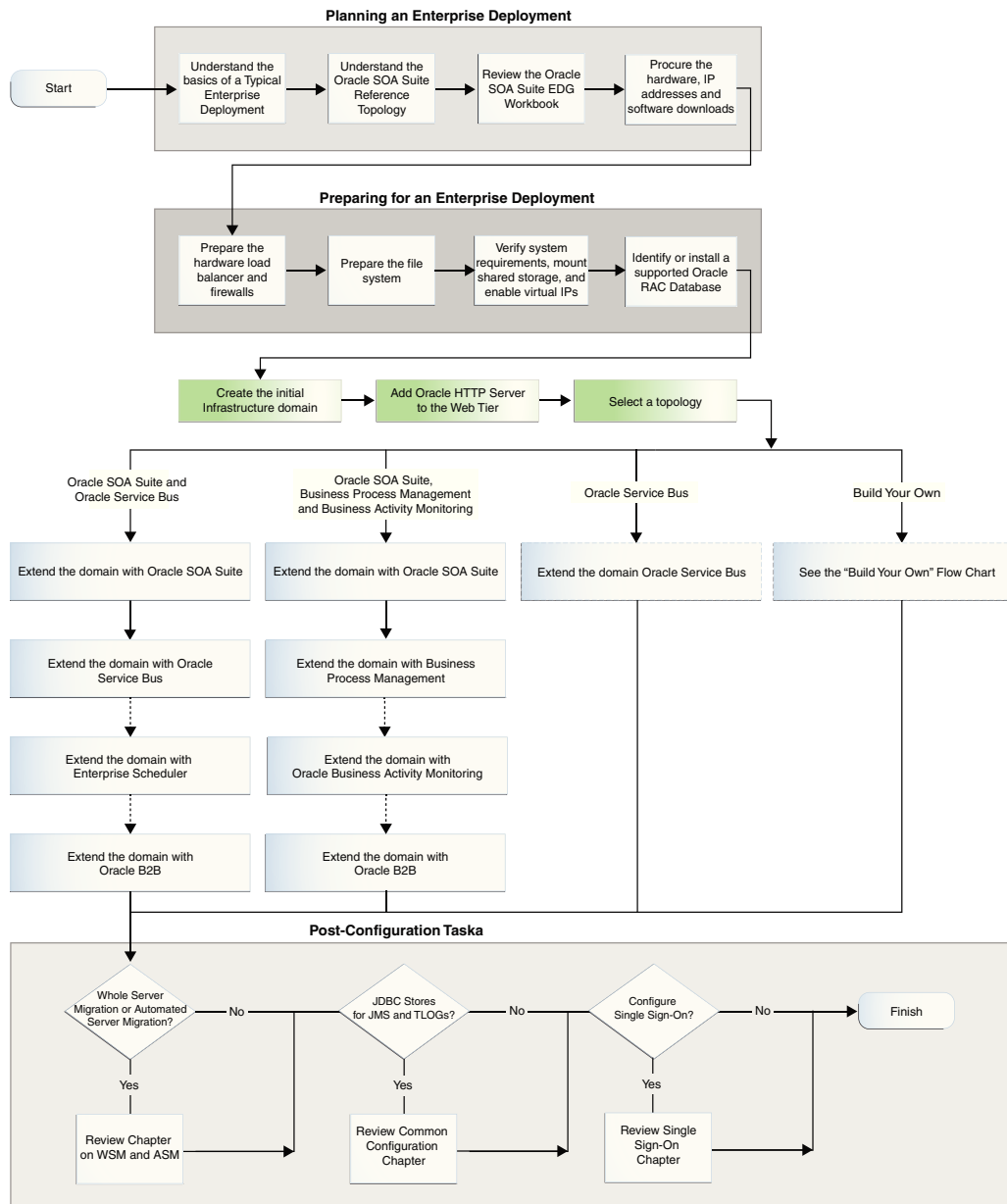
This guide is designed so you can start with a working Oracle SOA Suite domain and then later extend the domain to add additional capabilities.

This modular approach to building the topology allows you to make strategic decisions, based on your hardware and software resources, as well as the Oracle SOA Suite features that are most important to your organization.

It also allows you to validate and troubleshoot each individual product or component as they are configured.

This does not imply that configuring multiple products in one Configuration Wizard session is not supported; it is possible to group various extensions like the ones presented in this guide in one Configuration Wizard execution. However, the instructions in this guide focus primarily on the modular approach to building an enterprise deployment.

Figure 3-3 Flow Chart of the Enterprise Topology Configuration Steps



3.4.2 Roadmap Table for Planning and Preparing for an Enterprise Deployment

The following table describes each of the planning and preparing steps shown in the enterprise topology flow chart.

Flow Chart Step	More Information
Understand the basics of a Typical Enterprise Deployment	Understanding a Typical Enterprise Deployment
Understand the specific reference topology for the products you plan to deploy.	Review the product-specific topologies and the description of the topologies, including the virtual servers required and the summary of clusters and Managed Servers recommended for the product-specific deployment.
Review the Oracle SOA Suite EDG Workbook	Using the Enterprise Deployment Workbook
Procure the hardware, IP addresses and software downloads	Procuring Resources for an Enterprise Deployment
Prepare the hardware load balancer and firewalls	Preparing the Load Balancer and Firewalls for an Enterprise Deployment
Prepare the file system	Preparing the File System for an Enterprise Deployment
Verify system requirements, mount shared storage, and enable virtual IPs	Preparing the Host Computers for an Enterprise Deployment
Identify or install a supported Oracle RAC Database	Preparing the Database for an Enterprise Deployment

3.4.3 Roadmap Table for Configuring the Oracle SOA Suite and Oracle Service Bus Enterprise Topology

[Table 3-2](#) describes each of the configuration steps required when configuring the Oracle SOA Suite and Oracle Service Bus topology shown in [Figure 3-1](#).

These steps correspond to the Oracle SOA Suite and Oracle Service Bus Topology steps shown in the flow chart in [Figure 3-3](#).

Note:

You can configure Oracle Service Bus in the same domain as Oracle SOA Suite or in its own domain. For more information, see [Understanding the Topology Options for Oracle Service Bus](#).

Table 3-2 Roadmap Table for Configuring the Oracle SOA Suite and Oracle Service Bus Enterprise Topology

Flow Chart Step	More Information
Create the initial Infrastructure domain	Creating the Initial Infrastructure Domain for an Enterprise Deployment
Extend the domain to Include the Web Tier	Configuring the Web Tier for an Enterprise Deployment
Extend the domain with Oracle SOA Suite	Extending the Domain with Oracle SOA Suite
Extend the Domain with Oracle Service Bus	Extending the Domain with Oracle Service Bus
Extend the domain with Enterprise Scheduler	Extending the Domain with Oracle Enterprise Scheduler Note that extending the domain with Enterprise Scheduler is optional; perform the procedures in this chapter only if you want to configure Enterprise Scheduler.
Extend the domain with Oracle B2B	Extending the Domain with Oracle B2B Note that extending the domain with Oracle B2B is optional; perform the procedures in this chapter only if you want to configure Oracle B2B.

3.4.4 Roadmap Table for Configuring the Oracle SOA Suite and Oracle Business Activity Monitoring Enterprise Topology

[Table 3-3](#) describes each of the configuration steps required to configure the Oracle SOA Suite and Oracle Business Activity Monitoring topology shown in [Figure 3-2](#).

These steps correspond to the configuration steps shown for the Oracle SOA Suite Oracle Business Activity Monitoring topology in the flow chart in [Figure 3-3](#).

Table 3-3 Roadmap Table for Configuring the Oracle SOA Suite and Oracle Business Activity Monitoring Enterprise Topology

Flow Chart Step	More Information
Create the initial Infrastructure domain	Creating the Initial Infrastructure Domain for an Enterprise Deployment
Extend the domain to Include the Web Tier	Configuring the Web Tier for an Enterprise Deployment
Extend the domain with Oracle SOA Suite	Extending the Domain with Oracle SOA Suite

Table 3-3 (Cont.) Roadmap Table for Configuring the Oracle SOA Suite and Oracle Business Activity Monitoring Enterprise Topology

Flow Chart Step	More Information
Extend the domain with Business Process Management	Extending the Domain with Business Process Management
Extend the domain with Oracle Business Activity Monitoring	Extending the Domain with Business Activity Monitoring
Extend the domain with Oracle B2B	Extending the Domain with Oracle B2B

3.5 Building Your Own Oracle SOA Suite Enterprise Topology

This section describes the alternative topologies that you can implement, using some variations of the instructions in this guide.

This document provides step-by-step instructions for configuring the two primary enterprise topologies for Oracle SOA Suite, which are described in [Diagrams of the Primary Oracle SOA Suite Enterprise Topologies](#).

However, Oracle recognizes that the requirements of your organization may vary, depending on the specific set of Oracle Fusion Middleware products you purchase and the specific types of applications you deploy.

In many cases, you can install and configure an alternative topology--one that includes additional components, or one that does not include all the Oracle SOA Suite products shown in the primary topology diagrams.

See Also:

[Flow Chart of the "Build Your Own" Enterprise Topologies](#)

[Description of the Supported "Build Your Own" Topologies](#)

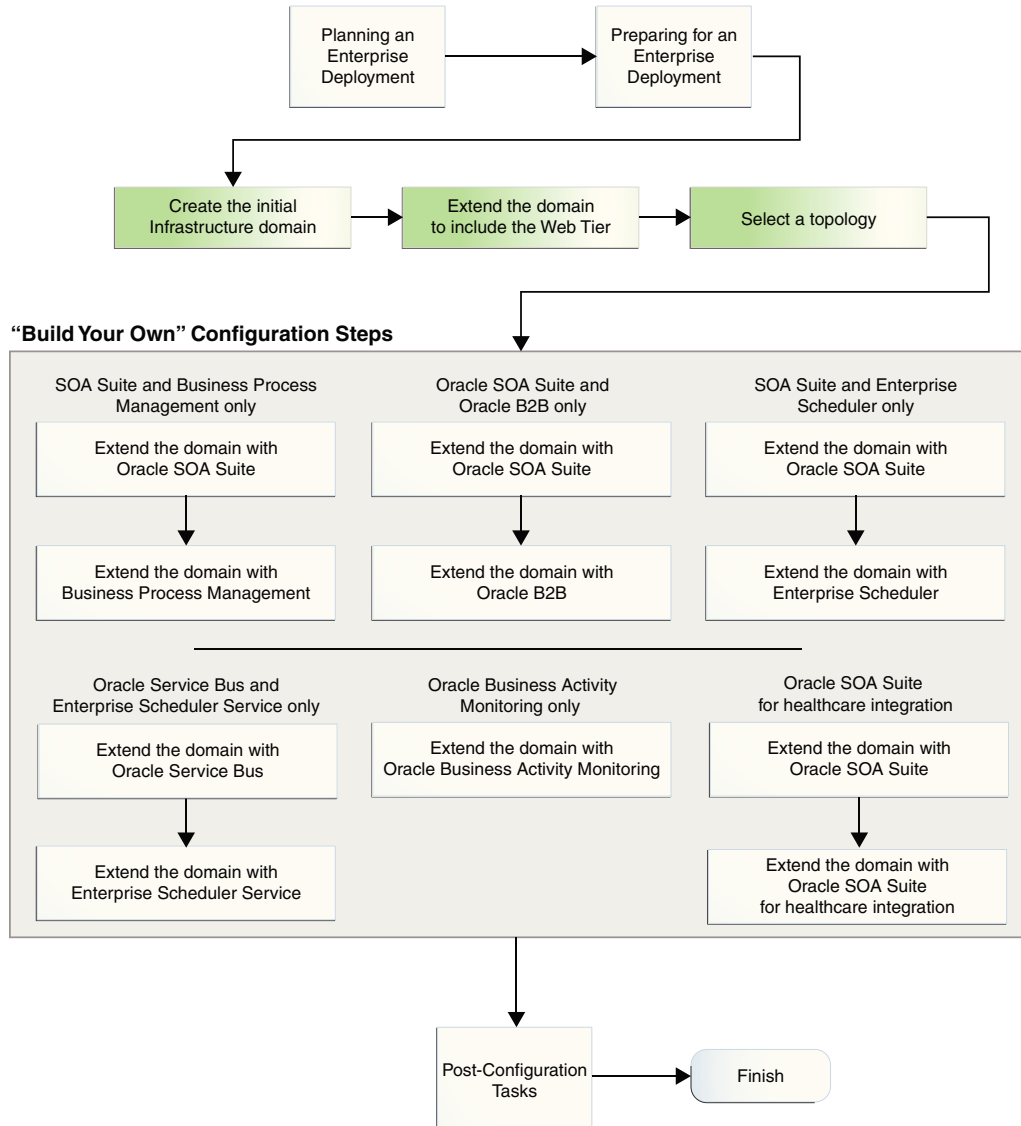
3.5.1 Flow Chart of the "Build Your Own" Enterprise Topologies

Building your own enterprise topology involves picking and choosing which Oracle Fusion Middleware products and which configuration steps you want to use to build your topology.

[Figure 3-4](#) shows the high-level configuration steps required to build some typical alternative Oracle SOA Suite enterprise topologies. Each of the configuration steps corresponds to a chapter in this guide.

Note that modifications of the steps in this guide are necessary in order to implement the "Build Your Own" topologies. Refer to [Description of the Supported "Build Your Own" Topologies](#) for more information.

Figure 3-4 Flow Chart of the Oracle SOA Suite Build-Your-Own Topologies



3.5.2 Description of the Supported "Build Your Own" Topologies

Table 3-4 describes the configuration steps to follow if you want to use the instructions in this guide to build the enterprise topologies listed in Figure 3-4.

It also identifies some differences you will need to consider when you use the existing instructions in this guide to build each topology.

Table 3-4 Roadmap Table for Building Your Own Enterprise Topology

Table 3-4 (Cont.) Roadmap Table for Building Your Own Enterprise Topology

Topology	After configuring the Web Tier, refer to the following chapters...	Considerations and Dependencies
SOA Suite and Business Process Management only	<ol style="list-style-type: none"> 1. Extending the Domain with Oracle SOA Suite 2. Extending the Domain with Business Process Management 	<p>These instructions assume you will run the Oracle SOA Suite and Business Process Management installer twice--once to install Oracle SOA Suite and once to install Oracle Business Process Management.</p> <p>Alternatively, you can install both Oracle SOA Suite and Oracle Business Process Management at the same time by selecting the BPM install type during the installation.</p> <p>Similarly, you can configure this topology by running the Configuration Wizard only once by selecting both the SOA and Oracle Business Process Management templates during the Configuration Wizard session.</p>
Oracle SOA Suite and Oracle B2B only	<ol style="list-style-type: none"> 1. Extending the Domain with Oracle SOA Suite 2. Extending the Domain with Oracle B2B 	No special instructions required.
SOA Suite and Enterprise Scheduler only	<ol style="list-style-type: none"> 1. Extending the Domain with Oracle SOA Suite 2. Extending the Domain with Oracle Enterprise Scheduler 	No special instructions required.
Oracle Service Bus and Enterprise Scheduler only	<p>See</p> <ol style="list-style-type: none"> 1. Extending the Domain with Oracle Service Bus 2. Extending the Domain with Oracle Enterprise Scheduler 	<p>This topology does not require Oracle SOA Suite. However, the instructions in Extending the Domain with Oracle Service Bus assume you have already created a cluster of two SOA Managed Servers.</p> <p>As a result, when you create this topology, ignore any references to the SOA Managed Servers or the SOA Cluster.</p> <p>In addition, you must run the Repository Creation Utility (RCU) to create the SOAINFRA schema, which is also required by Oracle Service Bus.</p>

Table 3-4 (Cont.) Roadmap Table for Building Your Own Enterprise Topology

Topology	After configuring the Web Tier, refer to the following chapters...	Considerations and Dependencies
Oracle Business Activity Monitoring only	Extending the Domain with Business Activity Monitoring	<p>The instructions in Extending the Domain with Business Activity Monitoring assume you are extending an existing Oracle SOA Suite domain and that the Oracle SOA Suite software (which includes Oracle BAM) has already been installed in an Oracle home on shared storage.</p> <p>For this Oracle BAM-only topology, you will need to install Oracle SOA Suite into the Oracle Fusion Middleware Infrastructure Oracle home before you can configure the domain to include an Oracle BAM cluster.</p> <p>In addition, you must run the Repository Creation Utility (RCU) to create the required SOA schemas.</p>
Oracle SOA Suite for healthcare integration	<ol style="list-style-type: none"> 1. Extending the Domain with Oracle SOA Suite 2. Extending the Domain with Oracle SOA Suite for Healthcare Integration 	No special instructions required.

3.6 About Installing and Configuring a Custom Enterprise Topology

If you choose to implement a topology that is not described in this guide, be sure to review the certification information, system requirements, and interoperability requirements for the products you want to include in the topology.

After you verify that the topology is supported, then you can either use the instructions in this guide as a guide to installing and configuring the components you need, or you can install and configure a standard installation topology using the Oracle Fusion Middleware 12c installation guides and use the "Start Small and Scale Out" approach to configuring your environment.

For more information, see "Planning for a Production Environment" in *Planning an Installation of Oracle Fusion Middleware*.

3.7 About Using Automatic Service Migration for the Oracle SOA Suite Enterprise Topology

To ensure high availability of the Oracle SOA Suite products and components, this guide recommends that you enable Oracle WebLogic Server Automatic Service Migration for the clusters that you create as part of the reference topology.

For more information, see [Using Whole Server Migration and Service Migration in an Enterprise Deployment](#).

Part II

Preparing for an Enterprise Deployment

This part of the of the enterprise deployment guide contains the following topics.

See Also:

[Using the Enterprise Deployment Workbook](#)

[Procuring Resources for an Enterprise Deployment](#)

[Preparing the Load Balancer and Firewalls for an Enterprise Deployment](#)

[Preparing the File System for an Enterprise Deployment](#)

[Preparing the Host Computers for an Enterprise Deployment](#)

[Preparing the Database for an Enterprise Deployment](#)

Using the Enterprise Deployment Workbook

This chapter introduces the Enterprise Deployment Workbook; it describes how you can use the workbook to plan an enterprise deployment for your organization.

This chapter provides an introduction to the Enterprise Deployment workbook, use cases and information on who should use the Enterprise Deployment workbook.

See Also:

[Preparing for an Enterprise Deployment](#)

[Introduction to the Enterprise Deployment Workbook](#)

This section provides a brief introduction of the enterprise deployment workbook.

[Typical Use Case for Using the Workbook](#)

This section lists the roles and tasks involved in a typical use case of the enterprise deployment workbook.

[Using the Oracle SOA Suite Enterprise Deployment Workbook](#)

This section provides details for using the enterprise deployment workbook.

[Who Should Use the Enterprise Deployment Workbook?](#)

The information in the Enterprise Deployment Workbook is divided into categories. Depending on the structure of your organization and roles defined for your team, you can assign specific individuals in your organization to fill in the details of the workbook. Similarly the information in each category can be assigned to the individual or team responsible for planning, procuring, or setting up each category of resources.

4.1 Introduction to the Enterprise Deployment Workbook

This section provides a brief introduction of the enterprise deployment workbook.

The Oracle Fusion Middleware Enterprise Deployment Workbook is a companion document to this guide. It is a spreadsheet that can be used by architects, system engineers, database administrators, and others to plan and record all the details for an environment installation (such as server names, URLs, port numbers, installation paths, and other resources).

The Enterprise Deployment Workbook serves as a single document you can use to track input variables for the entire process, allowing for:

- Separation of tasks between architects, system engineers, database administrators, and other key organizational roles
- Comprehensive planning before the implementation

- Validation of planned decisions before actual implementation
- Consistency during implementation
- A record of the environment for future use

4.2 Typical Use Case for Using the Workbook

This section lists the roles and tasks involved in a typical use case of the enterprise deployment workbook.

A typical use case for the Enterprise Deployment Workbook involves the following roles and tasks, in preparation for an Oracle Fusion Middleware enterprise deployment:

- Architects read through the first five chapters of this guide, and fill in the corresponding sections of the Workbook.
- The Workbook is validated by other architects and system engineers.
- The architect uses the validated workbook to initiate network and system change requests with system engineering departments;
- The Administrators and System Integrators who are installing and configuring the software refer to the workbook and the subsequent chapters of this guide to perform the installation and configuration tasks.

4.3 Using the Oracle SOA Suite Enterprise Deployment Workbook

This section provides details for using the enterprise deployment workbook.

The following sections provide an introduction to the location and contents of the Oracle SOA Suite Enterprise Deployment Workbook:

See Also:

[Locating the Oracle SOA Suite Enterprise Deployment Workbook](#)

[Understanding the Contents of the Oracle SOA Suite Enterprise Deployment Workbook](#)

4.3.1 Locating the Oracle SOA Suite Enterprise Deployment Workbook

The Oracle SOA Suite Enterprise Deployment Workbook is available as a Microsoft Excel Spreadsheet in the Oracle Fusion Middleware documentation library. It is available as a link on the Install, Patch, and Upgrade page of the library.

4.3.2 Understanding the Contents of the Oracle SOA Suite Enterprise Deployment Workbook

The following sections describe the contents of the Oracle SOA Suite Enterprise Deployment Workbook. The workbook is divided into tabs, each containing a set of related variables and values you will need to install and configure the Enterprise Deployment topologies.

See Also:

[Using the Start Tab](#)

[Using the Hardware - Host Computers Tab](#)

[Using the Network - Virtual Hosts & Ports Tab](#)

[Using the Storage - Directory Variables Tab](#)

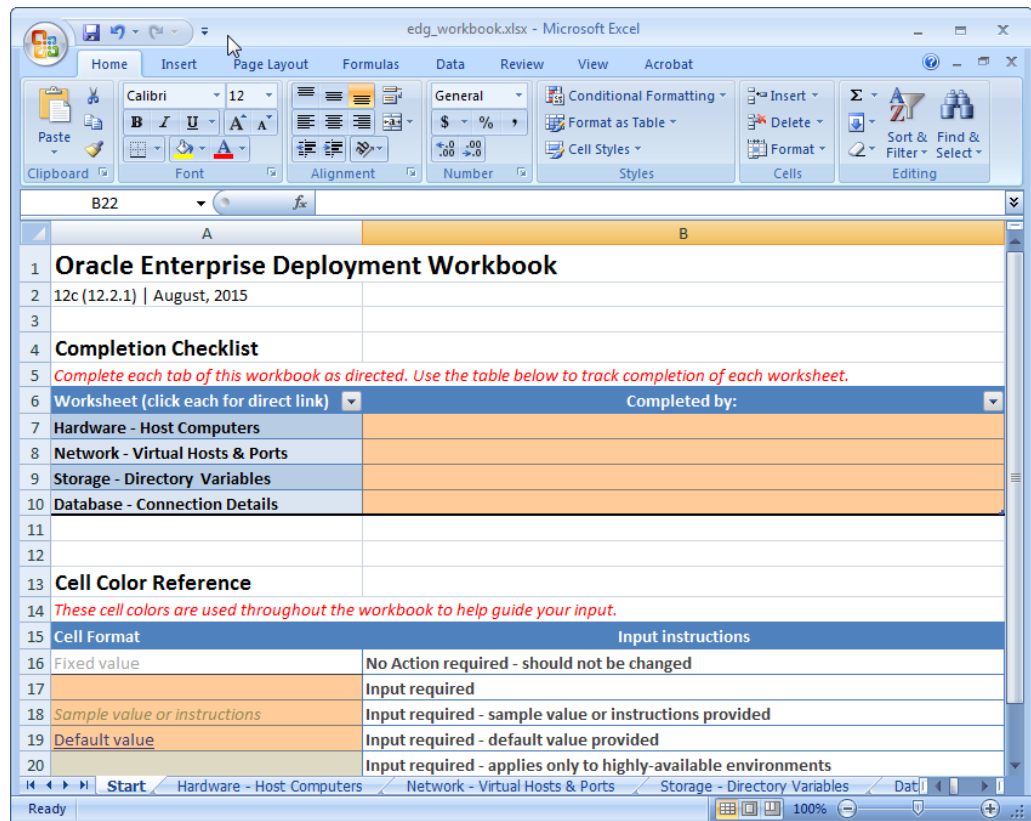
[Using the Database - Connection Details Tab](#)

4.3.2.1 Using the Start Tab

The Start tab of the Enterprise Deployment Workbook serves as a table of contents for the rest of the workbook. You can also use it to identify the people who will be completing the spreadsheet.

The Start tab also provides a key to identify the colors used to identify workbook fields that need values, as well as those that are provided for informational purposes.

The following image shows the Start tab of the spreadsheet.



4.3.2.2 Using the Hardware - Host Computers Tab

The Hardware - Host Computers tab lists the host computers required to install and configure the Oracle SOA Suite Enterprise Deployment Topology.

The reference topologies typically require a minimum of six host computers: two for the Web tier, two for the application tier, and two for the Oracle RAC database on the data tier. If you decide to expand the environment to include more systems, add a row for each additional host computer.

The **Abstract Host Name** is the name used throughout this guide to reference the host. For each row, procure a host computer, and enter the **Actual Host Name**. You can then use the actual host name when any of the abstract names is referenced in this guide.

For example, if a procedure in this guide references SOAHOST1, you can then replace the SOAHOST1 variable with the actual name provided on the Hardware - Host Computers tab of the workbook.

For easy reference, Oracle also recommends that you include the IP address, Operating System (including the version), number of CPUs, and the amount of RAM for each host. This information can be useful during the installation, configuration, and maintenance of the enterprise deployment.

For more information, see [Preparing the Host Computers for an Enterprise Deployment](#).

4.3.2.3 Using the Network - Virtual Hosts & Ports Tab

The Network - Virtual Hosts & Ports tab lists the virtual hosts that must be defined by your network administrator before you can install and configure the enterprise deployment topology.

The port numbers are important for several reasons. You must have quick reference to the port numbers so you can access the management consoles; the firewalls must also be configured to allow network traffic via specific ports.

Each virtual host, virtual IP address, and each network port serves a distinct purpose in the deployment. For more information, see [Preparing the Load Balancer and Firewalls for an Enterprise Deployment](#).

In the Network - Virtual Hosts table, review the items in the **Abstract Virtual Host or Virtual IP Name** column. These are the virtual host and virtual IP names used in the procedures in this guide. For each abstract name, enter the actual virtual host name defined by your network administrator. Whenever this guide references one of the abstract virtual host or virtual IP names, replace that value with the actual corresponding value in this table.

Similarly, in many cases, this guide assumes you are using default port numbers for the components or products you install and configure. However, in reality, you will likely have to use different port numbers. Use the Network - Port Numbers table to map the default port values to the actual values used in your specific installation.

4.3.2.4 Using the Storage - Directory Variables Tab

As part of preparing for an enterprise deployment, it is assumed you will be using a standard directory structure, which is recommended for Oracle enterprise deployments.

In addition, procedures in this book reference specific directory locations. Within the procedures, each directory is assigned a consistent variable, which you should replace with the actual location of the directory in your installation.

For each of the directory locations listed on this tab, provide the actual directory path in your installation.

In addition, for the application tier, it is recommended that many of these standard directories be created on a shared storage device. For those directories, the table also provides fields so you can enter the name of the shared storage location and the mount point used when you mounted the shared location.

For more information, see [Preparing the File System for an Enterprise Deployment](#).

4.3.2.5 Using the Database - Connection Details Tab

When you are installing and configuring the enterprise deployment topology, you will often have to make connections to a highly available Oracle Real Application Clusters

(RAC) database. In this guide, the procedures reference a set of variables that identify the information you will need to provide to connect to the database from tools, such as the Configuration Wizard and the Repository Creation Utility.

To be sure you have these values handy, use this tab to enter the actual values for these variables in your database installation.

For more information, see [Preparing the Database for an Enterprise Deployment](#).

4.4 Who Should Use the Enterprise Deployment Workbook?

The information in the Enterprise Deployment Workbook is divided into categories. Depending on the structure of your organization and roles defined for your team, you can assign specific individuals in your organization to fill in the details of the workbook. Similarly the information in each category can be assigned to the individual or team responsible for planning, procuring, or setting up each category of resources.

For example, the workbook can be filled in, reviewed, and used by people in your organization that fill the following roles:

- Information Technology (IT) Director
- Architect
- System Administrator
- Network Engineer
- Database Administrator

Procuring Resources for an Enterprise Deployment

Use the following topics to procure the required hardware, software, and network settings before you begin configuring the Oracle SOA Suite reference topology.

This chapter provides information on reserving the required IP addresses and identifying and obtaining software downloads for an enterprise deployment.

See Also:

[Preparing for an Enterprise Deployment](#)

[Hardware and Software Requirements for the Enterprise Deployment Topology](#)

This section specifies the hardware load balancer requirements, host computer hardware requirements, and operating system requirements for the enterprise deployment topology.

[Reserving the Required IP Addresses for an Enterprise Deployment](#)

This section lists the set of IP addresses that you must obtain and reserve before installation and configuration.

[Identifying and Obtaining Software Downloads for an Enterprise Deployment](#)

Before you begin installing and configuring the enterprise topology, you should locate and download the software distributions that you will need to implement the topology.

5.1 Hardware and Software Requirements for the Enterprise Deployment Topology

This section specifies the hardware load balancer requirements, host computer hardware requirements, and operating system requirements for the enterprise deployment topology.

This section includes the following sections.

See Also:

[Hardware Load Balancer Requirements](#)

This section lists the desired features of the external load balancer.

[Host Computer Hardware Requirements](#)

This section provides information to help you procure host computers that are configured to support the enterprise deployment topologies.

[Operating System Requirements for the Enterprise Deployment Topology](#)

This section provides details about the operating system requirements.

5.1.1 Hardware Load Balancer Requirements

This section lists the desired features of the external load balancer.

This enterprise topology uses an external load balancer. This external load balancer should have the following features:

- Ability to load-balance traffic to a pool of real servers through a virtual host name: Clients access services using the virtual host name (instead of using actual host names). The load balancer can then load balance requests to the servers in the pool.
- Port translation configuration should be possible so that incoming requests on the virtual host name and port are directed to a different port on the backend servers.
- Monitoring of ports on the servers in the pool to determine availability of a service.
- Virtual servers and port configuration: Ability to configure virtual server names and ports on your external load balancer, and the virtual server names and ports must meet the following requirements:
 - The load balancer should allow configuration of multiple virtual servers. For each virtual server, the load balancer should allow configuration of traffic management on more than one port. For example, for Oracle HTTP Server in the web tier, the load balancer needs to be configured with a virtual server and ports for HTTP and HTTPS traffic.
 - The virtual server names must be associated with IP addresses and be part of your DNS. Clients must be able to access the external load balancer through the virtual server names.
- Ability to detect node failures and immediately stop routing traffic to the failed node.
- Fault-tolerant mode: It is highly recommended that you configure the load balancer to be in fault-tolerant mode.
- It is highly recommended that you configure the load balancer virtual server to return immediately to the calling client when the backend services to which it forwards traffic are unavailable. This is preferred over the client disconnecting on its own after a timeout based on the TCP/IP settings on the client machine.
- Sticky routing capability: Ability to maintain sticky connections to components. Examples of this include cookie-based persistence, IP-based persistence, and so on.
- The load balancer should be able to terminate SSL requests at the load balancer and forward traffic to the backend real servers using the equivalent non-SSL protocol (for example, HTTPS to HTTP).

- SSL acceleration (this feature is recommended, but not required for the enterprise topology).
- The ability to route TCP/IP requests; this is a requirement for Oracle SOA Suite for healthcare integration, which uses the Minimum Lower Layer Protocol (MLLP) over TCP.

5.1.2 Host Computer Hardware Requirements

This section provides information to help you procure host computers that are configured to support the enterprise deployment topologies.

It includes the following topics.

See Also:

[General Considerations for Enterprise Deployment Host Computers](#)

This section specifies the general considerations required for the enterprise deployment host computers.

[Reviewing the Oracle Fusion Middleware System Requirements](#)

This section provides reference to the system requirements information to help you ensure that the environment meets the necessary minimum requirements.

[Typical Memory, File Descriptors, and Processes Required for an Enterprise Deployment](#)

This section specifies the typical memory, number of file descriptors, and operating system processes and tasks details required for an enterprise deployment.

[Typical Disk Space Requirements for an Enterprise Deployment](#)

This section specifies the disk space typically required for this enterprise deployment.

5.1.2.1 General Considerations for Enterprise Deployment Host Computers

This section specifies the general considerations required for the enterprise deployment host computers.

Before you start the process of configuring an Oracle Fusion Middleware enterprise deployment, you must perform the appropriate capacity planning to determine the number of nodes, CPUs, and memory requirements for each node depending on the specific system's load as well as the throughput and response requirements. These requirements will vary for each application or custom Oracle SOA Suite system being used.

The information in this chapter provides general guidelines and information that will help you determine the host computer requirements. It does not replace the need to perform capacity planning for your specific production environment.

Note:

As you obtain and reserve the host computers in this section, note the host names and system characteristics in the Enterprise Deployment Workbook. You will use these addresses later when you enable the IP addresses on each host computer.

For more information, see [Using the Enterprise Deployment Workbook](#)

5.1.2.2 Reviewing the Oracle Fusion Middleware System Requirements

This section provides reference to the system requirements information to help you ensure that the environment meets the necessary minimum requirements.

Review the [Oracle Fusion Middleware System Requirements and Specifications](#) to ensure that your environment meets the minimum installation requirements for the products you are installing.

The Requirements and Specifications document contains information about general Oracle Fusion Middleware hardware and software requirements, minimum disk space and memory requirements, database schema requirements, and required operating system libraries and packages.

It also provides some general guidelines for estimating the memory requirements for your Oracle Fusion Middleware deployment.

5.1.2.3 Typical Memory, File Descriptors, and Processes Required for an Enterprise Deployment

This section specifies the typical memory, number of file descriptors, and operating system processes and tasks details required for an enterprise deployment.

The following table summarizes the memory, file descriptors, and processes required for the Administration Server and each of the Managed Servers computers in a typical Oracle SOA Suite enterprise deployment. These values are provided as an example only, but they can be used to estimate the minimum amount of memory required for an initial enterprise deployment.

The example in this topic reflects the minimum requirements for configuring the Managed Servers and other services required on SOAHOST1, as depicted in the reference topologies.

When you are procuring machines, use the information in the **Approximate Top Memory** column as a guide when determining how much physical memory each host computer should have available.

After you procure the host computer hardware and verify the operating system requirements, review the software configuration to be sure the operating system settings are configured to accommodate the number of open files listed in the **File Descriptors** column and the number processes listed in the **Operating System Processes and Tasks** column.

For more information, see [Setting the Open File Limit and Number of Processes Settings on UNIX Systems](#).

Managed Server, Utility, or Service	Approximate Top Memory	Number of File Descriptors	Operating System Processes and Tasks
Administration Server	3.5 GB	3500	165

Managed Server, Utility, or Service	Approximate Top Memory	Number of File Descriptors	Operating System Processes and Tasks
WLS_WSM	3.0 GB	2000	130
WLS_SOA	4.0 GB	3100	240
WLS_OSB	4.0 GB	2200	180
WLS_ESS	3.5 GB	1300	35
WLS_BAM	3.5 GB	2300	210
WLST (connection to the Node Manager)	1.5 GB	910	20
Configuration Wizard	1.5 GB	700	20
Node Manager	1.0 GB	720	15
TOTAL	27.0 GB*	17000	1200

* Approximate total, with consideration for Operating System and other additional memory requirements.

5.1.2.4 Typical Disk Space Requirements for an Enterprise Deployment

This section specifies the disk space typically required for this enterprise deployment.

For the latest disk space requirements for the Oracle Fusion Middleware 12c (12.2.1) products, including the Oracle SOA Suite products, review the [Oracle Fusion Middleware System Requirements and Specifications](#).

In addition, the following table summarizes the disk space typically required for an Oracle SOA Suite enterprise deployment.

Use the this information and the information in [Preparing the File System for an Enterprise Deployment](#) to determine the disk space requirements required for your deployment.

Server	Disk
Database	nXm n = number of disks, at least 4 (striped as one disk) m = size of the disk (minimum of 30 GB)
WEBHOST _n	10 GB
SOAHOST _n (SOA only)	10 GB*
SOAHOST _n (SOA and OSB)	11 GB*

* For a shared storage Oracle home configuration, two installations suffice by making a total of 20 GB.

5.1.3 Operating System Requirements for the Enterprise Deployment Topology

This section provides details about the operating system requirements.

The Oracle Fusion Middleware software products and components described in this guide are certified on various operating systems and platforms, which are listed in the *Oracle Fusion Middleware System Requirements and Specifications*.

Note:

This guide focuses on the implementation of the enterprise deployment reference topology on Oracle Linux systems.

The topology can be implemented on any certified, supported operating system, but the examples in this guide typically show the commands and configuration steps as they should be performed using the bash shell on Oracle Linux.

5.2 Reserving the Required IP Addresses for an Enterprise Deployment

This section lists the set of IP addresses that you must obtain and reserve before installation and configuration.

Before you begin installing and configuring the enterprise topology, you must obtain and reserve a set of IP addresses:

- Physical IP (IP) addresses for each of the host computers you have procured for the topology
- A virtual IP (VIP) address for the Administration Server
- Additional VIP addresses for each Managed Server that is configured for Whole Server Migration

For Fusion Middleware 12c products, such as Oracle SOA Suite, that support Automatic Service Migration, VIPs for the Managed Servers are typically not necessary.

- A unique virtual host name to be mapped to each VIP.

You can then work with your network administrator to be sure these required VIPs are defined in your DNS server. (Alternatively, for non-production environments, you can use the `/etc/hosts` file to define these virtual hosts).

For more information, see the following topics.

See Also:[What Is a Virtual IP \(VIP\) Address?](#)

This section defines the virtual IP address and specifies its purpose.

[Why Use Virtual Host Names and Virtual IP Addresses?](#)

For an enterprise deployment, in particular, it is important that a set of VIPs--and the virtual host names to which they are mapped--are reserved and enabled on the corporate network.

[Physical and Virtual IP Addresses Required by the Enterprise Topology](#)

This section describes the physical IP (IP) and virtual IP (VIP) addresses required for the Administration Server and each of the Managed Servers in a typical Oracle SOA Suite enterprise deployment topology.

5.2.1 What Is a Virtual IP (VIP) Address?

This section defines the virtual IP address and specifies its purpose.

A virtual IP address is an unused IP Address that belongs to the same subnet as the host's primary IP address. It is assigned to a host manually. If a host computer fails, the virtual address can be assigned to a new host in the topology. For the purposes of this guide, we reference *virtual* IP addresses, which can be re-assigned from one host to another, and *physical* IP addresses, which are assigned permanently to hardware host computer.

5.2.2 Why Use Virtual Host Names and Virtual IP Addresses?

For an enterprise deployment, in particular, it is important that a set of VIPs--and the virtual host names to which they are mapped--are reserved and enabled on the corporate network.

Alternatively, host names can be resolved through appropriate `/etc/hosts` file propagated through the different nodes.

In the event of the failure of the host computer where the IP address is assigned, the IP address can be assigned to another host in the same subnet, so that the new host can take responsibility for running the Managed Servers assigned to it.

The reassignment of virtual IP address for the Administration Server must be performed manually, but the reassignment of virtual IP addresses for Managed Servers can be performed automatically using the Whole Server Migration feature of Oracle WebLogic Server.

Whether you should use Whole Server Migration or not depends upon the products you are deploying and whether they support Automatic Service Migration.

For example, starting with Oracle SOA Suite 12c, the SOA Suite products support automatic service migration. As a result, it is no longer necessary to reserve VIPs for each of the Managed Servers in the domain. Instead, a VIP is required for the Administration Server only.

5.2.3 Physical and Virtual IP Addresses Required by the Enterprise Topology

This section describes the physical IP (IP) and virtual IP (VIP) addresses required for the Administration Server and each of the Managed Servers in a typical Oracle SOA Suite enterprise deployment topology.

Before you begin to install and configure the enterprise deployment, reserve a set of host names and IP addresses that correspond to the VIPs in [Table 5-1](#).

You can assign any unique host name to the VIPs, but in this guide, we reference each VIP using the suggested host names in the table.

Note:

As you obtain and reserve the IP addresses and their corresponding virtual host names in this section, note the values of the IP addresses and host names in the Enterprise Deployment Workbook. You will use these addresses later when you enable the IP addresses on each host computer.

For more information, see [Using the Enterprise Deployment Workbook](#)

Table 5-1 Summary of the Virtual IP Addresses Required for the Enterprise Deployment

Virtual IP	VIP Maps to...	Description
VIP1	ADMINVHN	ADMINVHN is the virtual host name used as the listen address for the Administration Server and fails over with manual failover of the Administration Server. It is enabled on the node where the Administration Server process is running.

5.3 Identifying and Obtaining Software Downloads for an Enterprise Deployment

Before you begin installing and configuring the enterprise topology, you should locate and download the software distributions that you will need to implement the topology.

The following table lists the downloads you will need to obtain.

For general information about how to obtain Oracle Fusion Middleware software, see "Understanding and Obtaining Product Distributions" in *Planning an Installation of Oracle Fusion Middleware*.

For more specific information about locating and downloading specific Oracle Fusion Middleware products, see the *Oracle Fusion Middleware Download, Installation, and Configuration Readme Files* on OTN.

Distribution	Description
Oracle Fusion Middleware 12c (12.2.1.0.0) Infrastructure	Download this distribution to install the Oracle Fusion Middleware Infrastructure, which includes Oracle WebLogic Server and Java Required Files software required for Oracle Fusion Middleware products. This distribution also installs the Repository Creation Utility (RCU), which in previous Oracle Fusion Middleware releases was packaged in its own distribution.
Oracle HTTP Server 12c (12.2.1.0.0)	Download this distribution to install Oracle HTTP Server on the Web tier hosts.

Distribution	Description
Oracle Fusion Middleware 12c (12.2.1.0.0) SOA Suite and Business Process Management	Download this distribution to install the SOA Foundation and BPM software, which includes Oracle Business Activity Monitoring (BAM) and Oracle Enterprise Scheduler (ESS).
Oracle Fusion Middleware 12c (12.2.1.0.0) Service Bus	Download this distribution if you plan to install and configure Oracle Service Bus as part of the Oracle SOA Suite enterprise topology.
Oracle Fusion Middleware 12c (12.2.1.0.0) B2B and Healthcare	Download this distribution if you plan to install and configure Oracle B2B or Oracle B2B Healthcare as part of the Oracle SOA Suite enterprise topology.

Preparing the Load Balancer and Firewalls for an Enterprise Deployment

This chapter describes how to configure your network for an enterprise deployment.

See Also:

[Preparing for an Enterprise Deployment](#)

[Configuring Virtual Hosts on the Hardware Load Balancer](#)

This section explains how to configure the hardware load balancer for an enterprise deployment.

[Configuring the Firewalls and Ports for an Enterprise Deployment](#)

This section lists the ports that must be opened on the firewalls for an enterprise deployment.

6.1 Configuring Virtual Hosts on the Hardware Load Balancer

This section explains how to configure the hardware load balancer for an enterprise deployment.

The following topics explain how to configure the hardware load balancer, provide the summary of the virtual servers required, and provide additional instructions for these virtual servers.

See Also:

[Overview of the Hardware Load Balancer Configuration](#)

[Typical Procedure for Configuring the Hardware Load Balancer](#)

[Summary of the Virtual Servers Required for an Enterprise Deployment](#)

[Additional Instructions for admin.example.com](#)

[Additional Instructions for soa.example.com](#)

[Additional Instructions for soainternal.example.com](#)

[Additional Instructions for osb.example.com](#)

[Additional Instructions for soahealthcare.example.com](#)

6.1.1 Overview of the Hardware Load Balancer Configuration

As shown in the topology diagrams, you must configure the hardware load balancer to recognize and route requests to several virtual servers and associated ports for different types of network traffic and monitoring.

In the context of a load-balancing device, a virtual server is a construct that allows multiple physical servers to appear as one for load-balancing purposes. It is typically represented by an IP address and a service, and it is used to distribute incoming client requests to the servers in the server pool.

The virtual servers should be configured to direct traffic to the appropriate host computers and ports for the various services available in the enterprise deployment.

In addition, you should configure the load balancer to monitor the host computers and ports for availability so that the traffic to a particular server is stopped as soon as possible when a service is down. This ensures that incoming traffic on a given virtual host is not directed to an unavailable service in the other tiers.

Note that after you configure the load balancer, you can later configure the Web server instances in the Web tier to recognize a set of virtual hosts that use the same names as the virtual servers you defined for the load balancer. For each request coming from the hardware load balancer, the Web server can then route the request appropriately, based on the server name included in the header in the request. For more information, see [Configuring Oracle HTTP Server for Administration and Oracle Web Services Manager](#).

6.1.2 Typical Procedure for Configuring the Hardware Load Balancer

The following procedure outlines the typical steps for configuring a hardware load balancer for an enterprise deployment.

Note that the actual procedures for configuring a specific load balancer will differ, depending on the specific type of load balancer. Refer to the vendor-supplied documentation for actual steps.

1. Create a pool of servers. This pool contains a list of servers and the ports that are included in the load-balancing definition.

For example, for load balancing between the Web hosts, create a pool of servers that would direct requests to hosts WEBHOST1 and WEBHOST2 on port 7777.

2. Create rules to determine whether or not a given host and service is available and assign it to the pool of servers described in Step 1.
3. Create the required virtual servers on the load balancer for the addresses and ports that receive requests for the applications.

For a complete list of the virtual servers required for the enterprise deployment, see [Summary of the Virtual Servers Required for an Enterprise Deployment](#).

When you define each virtual server on the load balancer, consider the following:

- a. If your load balancer supports it, specify whether or not the virtual server is available internally, externally or both. Ensure that internal addresses are only resolvable from inside the network.
- b. Configure SSL Termination, if applicable, for the virtual server.
- c. Assign the pool of servers created in Step 1 to the virtual server.

6.1.3 Summary of the Virtual Servers Required for an Enterprise Deployment

This section provides the details of the virtual servers required for an enterprise deployment.

The following table provides a list of the virtual servers you must define on the hardware load balancer for the Oracle SOA Suite enterprise topology.

Virtual Host	Server Pool	Protocol	SSL Termination?	External?
admin.example.com:80	WEBHOST1.example.com:7777 WEBHOST2.example.com:7777	HTTP	No	No
soa.example.com:443	WEBHOST1.example.com:7777 WEBHOST2.example.com:7777	HTTPS	Yes	Yes
soainternal.example.com:80	WEBHOST1.example.com:7777 WEBHOST2.example.com:7777	HTTP	No	No
osb.example.com:443	WEBHOST1.example.com:7777 WEBHOST2.example.com:7777	HTTPS	No	Yes
soahealthcare.example.com:95nn	WEBHOST1.example.com:7777 WEBHOST2.example.com:7777	TCP	No	Yes

6.1.4 Additional Instructions for admin.example.com

This section provides the additional instructions required for the virtual server—admin.example.com.

When you configure this virtual server on the hardware load balancer:

- Enable address and port translation.
- Enable reset of connections when services or hosts are down.

6.1.5 Additional Instructions for soa.example.com

When you configure this virtual server on the hardware load balancer:

- Use port 80 and port 443. Any request that goes to port 80 (non-ssl protocol) should be redirected to port 443 (ssl protocol).
- Specify ANY as the protocol (non-HTTP protocols are required for B2B).
- Enable address and port translation.
- Enable reset of connections when services and/or nodes are down.
- Create rules to filter out access to /console and /em on this virtual server.

These context strings direct requests to the Oracle WebLogic Server Administration Console and to the Oracle Enterprise Manager Fusion Middleware Control and should be used only when accessing the system from admin.example.com.

6.1.6 Additional Instructions for soainternal.example.com

When you configure this virtual server on the hardware load balancer:

- Enable address and port translation.
- Enable reset of connections when services or nodes are down.

- As with the `soa.example.com`, create rules to filter out access to `/console` and `/em` on this virtual server.

6.1.7 Additional Instructions for `osb.example.com`

When you configure this virtual server on the hardware load balancer:

- Use port 80 and port 443. Any request that goes to port 80 (non-ssl protocol) should be redirected to port 443 (ssl protocol).
- Specify ANY as the protocol (non-HTTP protocols are required for B2B).
- Enable address and port translation.
- Enable reset of connections when services and/or nodes are down.
- Create rules to filter out access to `/console` and `/em` on this virtual server.

These context strings direct requests to the Oracle WebLogic Server Administration Console and to the Oracle Enterprise Manager Fusion Middleware Control and should be used only when accessing the system from `admin.example.com`.

6.1.8 Additional Instructions for `soahealthcare.example.com`

Each Healthcare Minimum Lower Layer Protocol (MLLP) endpoint requires a separate virtual server in the load balancer. The load balancer routes to the MLLP service that will run in one of the Healthcare Managed Servers on a specific port. The pool for this virtual server points to the hostname and port that was used in the Healthcare Console for creating the endpoint.

For example, if an endpoint is created on `SOAHOST1:9501` (with failover to `SOAHOST2:9501`), then you should create a virtual server, using 9501 as service port and with a pool containing `SOAHOST1:9501` and `SOAHOST2:9501`. The Healthcare load balancer virtual servers should use TCP as protocol and use Address and Port translation preserving the sort port of the connection.

6.2 Configuring the Firewalls and Ports for an Enterprise Deployment

This section lists the ports that must be opened on the firewalls for an enterprise deployment.

As an administrator, you should be familiar with the port numbers used by the various Oracle Fusion Middleware products and services. You can then ensure that the same port number is not used by two services on a host, and you can make sure that the proper ports are open on the firewalls in the enterprise topology.

The following tables lists the ports that you must open on the firewalls in the topology.

Note:

The TCP/IP port for B2B is a user-configured port and is not predefined. Similarly, the firewall ports depend on the definition of TCP/IP ports.

Firewall notation:

- FW0 refers to the outermost firewall.

- FW1 refers to the firewall between the web tier and the application tier.
- FW2 refers to the firewall between the application tier and the data tier.

Firewall Ports Common to All Fusion Middleware Enterprise Deployments

Type	Firewall	Port and Port Range	Protocol / Application	Inbound / Outbound	Other Considerations and Timeout Guidelines
Browser request	FW0	80	HTTP / Load Balancer	Inbound	Timeout depends on the size and type of HTML content.
Browser request	FW0	443	HTTPS / Load Balancer	Inbound	Timeout depends on the size and type of HTML content.
Browser request	FW1	80	HTTPS / Load Balancer	Outbound (for intranet clients)	Timeout depends on the size and type of HTML content.
Browser request	FW1	443	HTTPS / Load Balancer	Outbound (for intranet clients)	Timeout depends on the size and type of HTML content.
Callbacks and Outbound invocations	FW1	80	HTTPS / Load Balancer	Outbound	Timeout depends on the size and type of HTML content.
Callbacks and Outbound invocations	FW1	443	HTTPS / Load Balancer	Outbound	Timeout depends on the size and type of HTML content.
Load balancer to Oracle HTTP Server	n/a	7777	HTTP	n/a	n/a
OHS registration with Administration Server	FW1	7001	HTTP/t3	Inbound	Set the timeout to a short period (5-10 seconds).
OHS management by Administration Server	FW1	OHS Admin Port (7779)	TCP and HTTP, respectively	Outbound	Set the timeout to a short period (5-10 seconds).

Type	Firewall	Port and Port Range	Protocol / Application	Inbound / Outbound	Other Considerations and Timeout Guidelines
Session replication within a WebLogic Server cluster	n/a	n/a	n/a	n/a	By default, this communication uses the same port as the server's listen address.
Administration Console access	FW1	7001	HTTP / Administration Server and Enterprise Manager t3	Both	You should tune this timeout based on the type of access to the admin console (whether it is planned to use the Oracle WebLogic Server Administration Console from application tier clients or clients external to the application tier).
Database access	FW2	1521	SQL*Net	Both	Timeout depends on database content and on the type of process model used for SOA.
Coherence for deployment	n/a	8088 Range: 8000 - 8090		n/a	n/a
Oracle Unified Directory access	FW2	389 636 (SSL)	LDAP or LDAP/ssl	Inbound	You should tune the directory server's parameters based on load balancer, and not the other way around.

Type	Firewall	Port and Port Range	Protocol / Application	Inbound / Outbound	Other Considerations and Timeout Guidelines
Oracle Notification Server (ONS)	FW2	6200	ONS	Both	Required for Gridlink. An ONS server runs on each database server.

*External clients can access SOA servers directly on RMI or JMS (for example, for JDeveloper deployments and for JMX monitoring), in which case FW0 might need to be open or not depending on the security model that you implement.

Firewall Ports Specific to Oracle SOA Suite Enterprise Deployments

Type	Firewall	Port and Port Range	Protocol / Application	Inbound / Outbound	Other Considerations and Timeout Guidelines
WSM-PM access	FW1	7010 Range: 7010 - 7999	HTTP / WLS_WSM-PM n	Inbound	Set the timeout to 60 seconds.
SOA Server access	FW1*	8001 Range: 8000 - 8010	HTTP / WLS_SOA n	Inbound	Timeout varies based on the type of process model used for SOA.
Oracle Service Bus Access	FW1	8011 Range: 8011-8021	HTTP / WLS_OSBN	Inbound/ Outbound	Set the timeout to a short period (5-10 seconds).
BAM access	FW1	9001 Range: 9000 - 9080	HTTP / WLS_BAM n	Inbound	Connections to BAM WebApps are kept open until the report/browser is closed, so set the timeout as high as the longest expected user session.
MLLP Requests	FW0, FW1	9500 — 95 nm	Application: MLLP/HC	Inbound	Timeout depends on the expected MLLP transfer sizes.

Preparing the File System for an Enterprise Deployment

Involves understanding the requirements for local and shared storage, as well as the terminology used to reference important directories and file locations during the installation and configuration of the enterprise topology.

This chapter describes how to prepare the file system for an Oracle Fusion Middleware enterprise deployment.

See Also:

[Preparing for an Enterprise Deployment](#)

[Overview of Preparing the File System for an Enterprise Deployment](#)

This section provides an overview of the process of preparing the file system for an enterprise deployment.

[Shared Storage Recommendations When Installing and Configuring an Enterprise Deployment](#)

This section provides reference to the shared storage recommendations when installing and configuring an enterprise deployment.

[Understanding the Recommended Directory Structure for an Enterprise Deployment](#)

The diagrams in this section show the recommended directory structure for a typical Oracle Fusion Middleware enterprise deployment.

[File System and Directory Variables Used in This Guide](#)

This section lists and describes the file system and directory variables used throughout this guide.

[About Creating and Mounting the Directories for an Enterprise Deployment](#)

This section lists the best practices to be followed when creating or mounting the top-level directories in an enterprise deployment.

[Summary of the Shared Storage Volumes in an Enterprise Deployment](#)

This section provides a summary of the shared storage volumes required for an enterprise deployment.

7.1 Overview of Preparing the File System for an Enterprise Deployment

This section provides an overview of the process of preparing the file system for an enterprise deployment.

It is important to set up your storage in a way that makes the enterprise deployment easy to understand, configure, and manage. Oracle recommends setting up your storage according to information in this chapter. The terminology defined in this chapter is used in diagrams and procedures throughout the guide.

Use this chapter as a reference to help understand the directory variables used in the installation and configuration procedures.

Other directory layouts are possible and supported, but the model adopted in this guide was designed for maximum availability, providing both the best isolation of components and symmetry in the configuration and facilitating backup and disaster recovery. The rest of the document uses this directory structure and directory terminology.

7.2 Shared Storage Recommendations When Installing and Configuring an Enterprise Deployment

This section provides reference to the shared storage recommendations when installing and configuring an enterprise deployment.

Before you implement the detailed recommendations in this chapter, be sure to review the recommendations and general information about using shared storage in the *High Availability Guide*.

The recommendations in this chapter are based on the concepts and guidelines described in the *High Availability Guide*.

[Table 7-1](#) lists the key sections you should review and how those concepts apply to an enterprise deployment.

Table 7-1 Shared Storage Resources in the High Availability Guide

Section in <i>High Availability Guide</i>	Importance to an Enterprise Deployment
Shared Storage Prerequisites	Describes guidelines for disk format and the requirements for hardware devices that are optimized for shared storage.
Using Shared Storage for Binary (Oracle Home) Directories	Describes your options for storing the Oracle home on a shared storage device that is available to multiple hosts. For the purposes of the enterprise deployment, Oracle recommends using redundant Oracle homes on separate storage volumes. If a separate volume is not available, a separate partition on the shared disk should be used to provide redundant Oracle homes to application tier hosts.
Using Shared Storage for Domain Configuration Files	Describes the concept of creating separate domain homes for the Administration Server and the Managed Servers in the domain. For an enterprise deployment, the Administration Server domain home location is referenced by the <code>ASERVER_HOME</code> variable.
Shared Storage Requirements for JMS Stores and JTA Logs	Provides instructions for setting the location of the transaction logs and JMS stores for an enterprise deployment.

7.3 Understanding the Recommended Directory Structure for an Enterprise Deployment

The diagrams in this section show the recommended directory structure for a typical Oracle Fusion Middleware enterprise deployment.

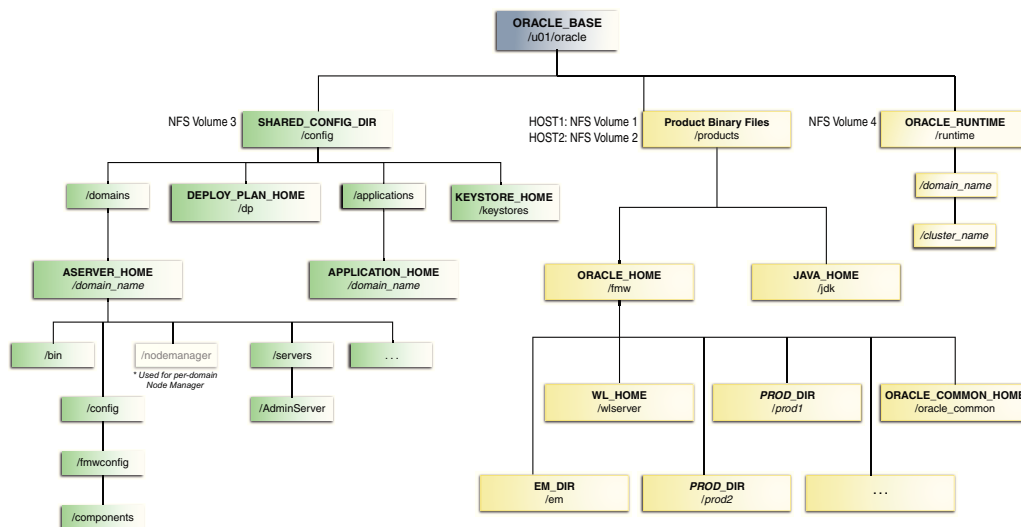
The directories shown in the diagrams contain binary files that are installed on disk by the Oracle Fusion Middleware installers, domain-specific files generated via the domain configuration process, as well as domain configuration files that are

propagated to the various host computers via the Oracle WebLogic Server pack and unpack commands:

- [Figure 7-1](#) shows the resulting directory structure on the shared storage device after you have installed and configured a typical Oracle Fusion Middleware enterprise deployment. The shared storage directories are accessible by the application tier host computers.
- [Figure 7-2](#) shows the resulting directory structure on the local storage device for a typical application tier host after you have installed and configured an Oracle Fusion Middleware enterprise deployment. The Managed Servers in particular are stored on the local storage device for the application tier host computers.
- [Figure 7-3](#) shows the resulting directory structure on the local storage device for a typical Web tier host after you have installed and configured an Oracle Fusion Middleware enterprise deployment. Note that the software binaries (in the Oracle home) are installed on the local storage device for each Web tier host.

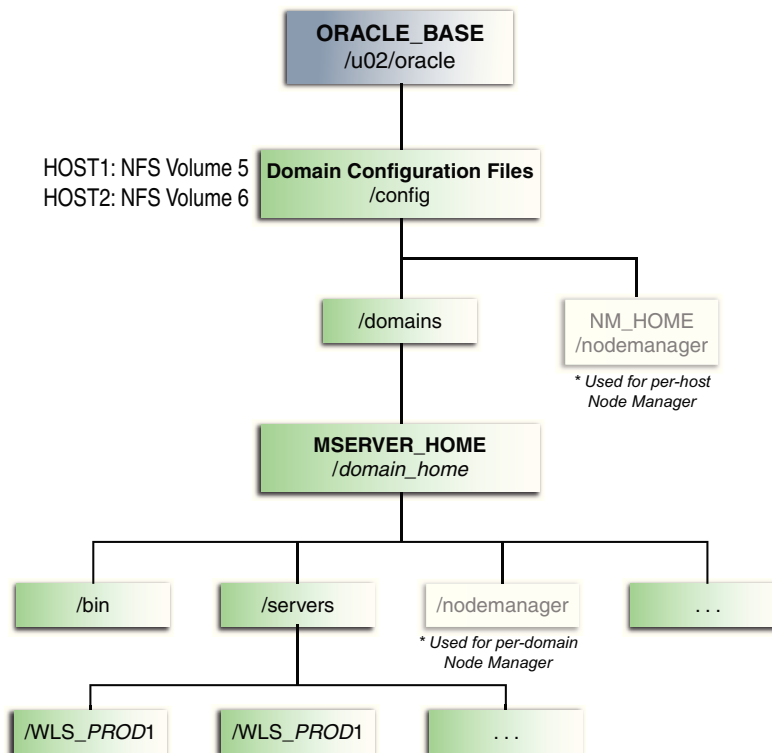
Where applicable, the diagrams also include the standard variables used to reference the directory locations in the installation and configuration procedures in this guide.

Figure 7-1 Recommended Shared Storage Directory Structure for an Enterprise Deployment



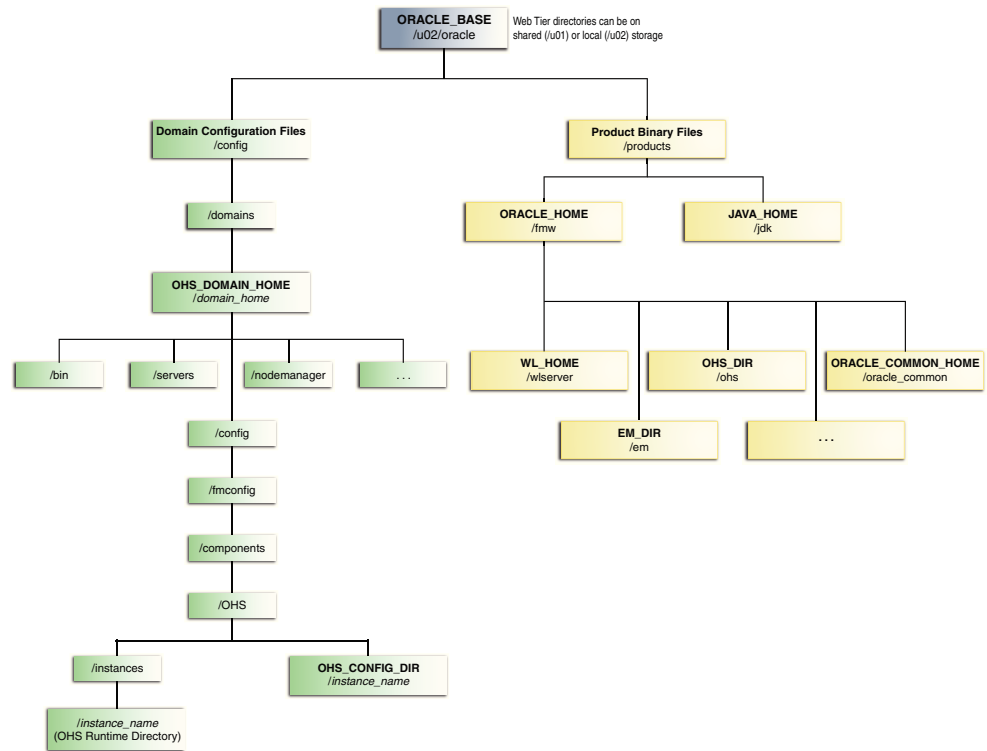
* For more information, see [About the Node Manager Configuration in a Typical Enterprise Deployment](#).

Figure 7-2 Recommended Local Storage Directory Structure for an Application Tier Host Computer in an Enterprise Deployment



* For more information, see [About the Node Manager Configuration in a Typical Enterprise Deployment](#).

Figure 7-3 Recommended Local Storage Directory Structure for a Web Tier Host Computer in an Enterprise Deployment



7.4 File System and Directory Variables Used in This Guide

This section lists and describes the file system and directory variables used throughout this guide.

[Table 7-2](#) lists the file system directories and the directory variables used to reference the directories on the Application tier. [Table 7-3](#) lists the file system directories and variables used to reference the directories on the Web tier.

For additional information about mounting these directories when you are using shared storage, see [About Creating and Mounting the Directories for an Enterprise Deployment](#).

Throughout this guide, the instructions for installing and configuring the topology refer to the directory locations using the variables shown here.

You can also define operating system variables for each of the directories listed in this section. If you define system variables for the particular UNIX shell you are using, you can then use the variables as they are used in this document, without having to map the variables to the actual values for your environment.

Note:

As you configure your storage devices to accommodate the recommended directory structure, note the actual directory paths in the Enterprise Deployment Workbook. You will use these addresses later when you enable the IP addresses on each host computer.

For more information, see [Using the Enterprise Deployment Workbook](#).

Table 7-2 Sample Values for Key Directory Variables on the Application Tier

Directory Variable	Description	Sample Value on the Application Tier
<i>ORACLE_BASE</i>	The base directory, under which Oracle products are installed.	<code>/u01/oracle</code>
<i>ORACLE_HOME</i>	The read-only location for the product binaries. For the application tier host computers, it is stored on shared disk. The Oracle home is created when you install the Oracle Fusion Middleware Infrastructure software. You can then install additional Oracle Fusion Middleware products into the same Oracle home.	<code>/u01/oracle/products/fmw</code>
<i>ORACLE_COMMON_HOME</i>	The directory within the Oracle Fusion Middleware Oracle home where common utilities, libraries, and other common Oracle Fusion Middleware products are stored.	<code>/u01/oracle/products/fmw/oracle_common</code>
<i>WL_HOME</i>	The directory within the Oracle home where the Oracle WebLogic Server software binaries are stored.	<code>/u01/oracle/products/fmw/wlserver</code>
<i>PROD_DIR</i>	Individual product directories for each Oracle Fusion Middleware product you install.	<code>/u01/oracle/products/fmw/prod_dir</code> The product can be <code>soa</code> , <code>wcc</code> , <code>bi</code> , or another value, depending on your enterprise deployment.
<i>EM_DIR</i>	The product directory used to store the Oracle Enterprise Manager Fusion Middleware Control software binaries.	<code>/u01/oracle/products/fmw/em</code>
<i>JAVA_HOME</i>	The location where you install the supported Java Development Kit (JDK).	<code>/u01/oracle/products/jdk</code>
<i>SHARED_CONFIG_DIR</i>	The shared parent directory for shared environment configuration files, including domain configuration, keystores, runtime artifacts, and application deployments	<code>/u01/oracle/config</code>
<i>ASERVER_HOME</i>	The Administration Server domain home, which is installed on shared disk.	<code>/u01/oracle/config/domains/domain_name</code> In this example, replace <code>domain_name</code> with the name of the WebLogic Server domain.
<i>MSERVER_HOME</i>	The Managed Server domain home, which is created via the <code>unpack</code> command on the local disk of each application tier host.	<code>/u02/oracle/config/domains/domain_name</code>

Table 7-2 (Cont.) Sample Values for Key Directory Variables on the Application Tier

Directory Variable	Description	Sample Value on the Application Tier
<i>APPLICATION_HOME</i>	The Application home directory, which is installed on shared disk, so the directory is accessible by all the application tier host computers.	<i>/u01/oracle/config/applications/domain_name</i>
<i>ORACLE_RUNTIME</i>	This directory contains the Oracle runtime artifacts, such as the JMS logs and TLogs. Typically, you mount this directory as a separate shared file system, which is accessible by all hosts in the domain. When you run the Configuration Wizard or perform post-configuration tasks, and you identify the location of JMS stores or tlogs persistent stores, then you can use this directory, qualified with the name of the domain, the name of the cluster, and the purpose of the directory. For example: <i>ORACLE_RUNTIME/cluster_name/jms</i>	<i>/u01/oracle/runtime/</i>
<i>NM_HOME</i>	The directory used by the Per Machine Node Manager start script and configuration files.	<i>/u02/oracle/config/node_manager</i>
	<hr/> Note: This directory is necessary only if you are using a Per Machine Node Manager configuration. <hr/>	
	For more information, see About the Node Manager Configuration in a Typical Enterprise Deployment .	
<i>DEPLOY_PLAN_HOME</i>	The deployment plan directory, which is used as the default location for application deployment plans.	<i>/u01/oracle/config/dp</i>
	<hr/> Note: This directory is required only when you are deploying custom applications to the application tier. <hr/>	

Table 7-2 (Cont.) Sample Values for Key Directory Variables on the Application Tier

Directory Variable	Description	Sample Value on the Application Tier
<code>KEYSTORE_HOME</code>	The shared location for custom certificates and keystores.	<code>/u01/oracle/config/keystores</code>

Table 7-3 Sample Values for Key Directory Variables on the Web Tier

Directory Variable	Description	Sample Value on the Web Tier
<code>OHS_ORACLE_HOME</code>	The read-only location for the Oracle HTTP Server product binaries. For the Web tier host computers, this directory is stored on local disk. The Oracle home is created when you install the Oracle HTTP Server software.	<code>/u02/oracle/products/fmw</code>
<code>ORACLE_COM_MON_HOME</code>	The directory within the Oracle HTTP Server Oracle home where common utilities, libraries, and other common Oracle Fusion Middleware products are stored.	<code>/u02/oracle/products/fmw/oracle_common</code>
<code>WL_HOME</code>	The directory within the Oracle home where the Oracle WebLogic Server software binaries are stored.	<code>/u02/oracle/products/fmw/wlserver</code>
<code>PROD_DIR</code>	Individual product directories for each Oracle Fusion Middleware product you install.	<code>/u02/oracle/products/fmw/ohs</code>
<code>JAVA_HOME</code>	The location where you install the supported Java Development Kit (JDK).	<code>/u02/oracle/products/jdk</code>
<code>OHS_DOMAIN_HOME</code>	The Domain home for the standalone Oracle HTTP Server domain, which is created when you install Oracle HTTP Server on the local disk of each Web tier host.	<code>/u02/oracle/config/domains/domain_name</code>
<code>OHS_CONFIG_DIR</code>	This is the location where you edit the Oracle HTTP Server configuration files (for example, <code>httpd.conf</code> and <code>moduleconf/*.conf</code>) on each Web host. Note this directory is also referred to as the OHS Staging Directory. Changes made here are later propagated to the OHS Runtime Directory. For more information, see “Staging and Run-time Configuration Directories” in the <i>Administrator’s Guide for Oracle HTTP Server</i> .	<code>/u02/oracle/config/domains/domain_name/config/fmwconfig/components/OHS/instance_name</code>

7.5 About Creating and Mounting the Directories for an Enterprise Deployment

This section lists the best practices to be followed when creating or mounting the top-level directories in an enterprise deployment.

When creating or mounting the top-level directories, note the following best practices:

- For the application tier, install the Oracle home (which contains the software binaries) on a second shared storage volume or second partition that is mounted to SOAHOST2. Be sure the directory path to the binaries on SOAHOST2 is identical to the directory path on SOAHOST1.

For example:

```
/u01/oracle/products/fmw/
```

For more information, see [Shared Storage Recommendations When Installing and Configuring an Enterprise Deployment](#).

- This enterprise deployment guide assumes that the Oracle Web tier software will be installed on a local disk.

The Web tier installation is typically performed on local storage to the WEBHOST nodes. When using shared storage, you can install the Oracle Web tier binaries (and create the Oracle HTTP Server instances) on shared disk. However, if you do so, then the shared disk *must* be separate from the shared disk used for the application tier, and you must consider the appropriate security restrictions for access to the storage device across tiers.

As with the application tier servers (SOAHOST1 and SOAHOST2), use the same directory path on both computers.

For example:

```
/u02/oracle/products/fmw/
```

- If you are configuring Oracle Service Bus (OSB) in its own domain, but on the same host as Oracle SOA Suite, then you must create an additional Oracle home (ORACLE_HOME) for the OSB binaries, and you should mount that Oracle home separately from the SOA Oracle home.

For example, the OSB_ORACLE_HOME might be mounted as follows:

```
/u03/oracle/products/fmw/osb
```

- Similarly, if you are configuring OSB in its own domain, but on the same host as Oracle SOA Suite, then you should mount the domain directories separately from the Oracle SOA Suite domain directories.

For example, the OSB Administration Server domain directory might be mounted as follows:

```
/u03/oracle/config/domains/osb_domain_name
```

And the OSB Managed Servers domain directory might be mounted as follows:

```
u04/oracle/config/domains/osb_domain_name
```

7.6 Summary of the Shared Storage Volumes in an Enterprise Deployment

This section provides a summary of the shared storage volumes required for an enterprise deployment.

The following table summarizes the shared volumes and their purpose in a typical Oracle Fusion Middleware enterprise deployment.

For more information, see [Shared Storage Recommendations When Installing and Configuring an Enterprise Deployment](#).

Volume in Shared Storage	Mounted to Host	Mount Directories	Description and Purpose
NFS Volume 1	SOAHOST1	/u01/oracle/products/	Local storage for the product binaries to be used by SOAHOST1; this is where the Oracle home directory and product directories are installed.
NFS Volume 2	SOAHOST2	/u01/oracle/products/	Local storage for the product binaries to be used by SOAHOST2; this is where the Oracle home directory and product directories are installed.
NFS Volume 3	SOAHOST1 SOAHOST2	/u01/oracle/config/	Administration Server domain configuration, mounted to all hosts; used initially by SOAHOST1, but can be failed over to any host.
NFS Volume 4	SOAHOST1 SOAHOST2	/u01/oracle/runtime/	The runtime artifacts directory, mounted to all hosts, contains runtime artifacts such as JMS logs, blogs, and any cluster-dependent shared files needed.
NFS Volume 5	SOAHOST1	/u02/oracle/config/	Local storage for the Managed Server domain directory to be used by SOAHOST1, if the private Managed Server domain directory resides on shared storage.

Volume in Shared Storage	Mounted to Host	Mount Directories	Description and Purpose
NFS Volume 6	SOAHOST2	/u02/oracle/config/	Local storage for the Managed Server domain directory to be used by SOAHOST2, if the private Managed Server domain directory resides on shared storage.
NFS Volume 7	WEBHOST1	/u02/oracle/	Local storage for the Oracle HTTP Server software binaries (Oracle home) and domain configuration files used by WEBHOST1, if the private Managed Server domain directory resides on shared storage.
NFS Volume 8	WEBHOST2	/u02/oracle/	Local storage for the Oracle HTTP Server software binaries (Oracle home) and domain configuration files used by WEBHOST2, if the private Managed Server domain directory resides on shared storage.

Preparing the Host Computers for an Enterprise Deployment

It explains how to mount the required shared storage systems to the host and how to enable the required virtual IP addresses on each host.

This chapter describes the tasks you must perform from each computer or server that will be hosting the enterprise deployment.

See Also:

[Preparing for an Enterprise Deployment](#)

[Verifying the Minimum Hardware Requirements for Each Host](#)

This section provides information about the minimum hardware requirements for each host.

[Verifying Linux Operating System Requirements](#)

Review this section for typical Linux operating system settings for an enterprise deployment.

[Configuring Operating System Users and Groups](#)

The lists in this section show the users and groups to define on each of the computers that will host the enterprise deployment.

[Enabling Unicode Support](#)

This section provides information about enabling Unicode support.

[Mounting the Required Shared File Systems on Each Host](#)

This section provides information about mounting the required shared file systems on each host.

[Enabling the Required Virtual IP Addresses on Each Host](#)

This section provides instruction to enable the required virtual IP addresses on each host.

8.1 Verifying the Minimum Hardware Requirements for Each Host

This section provides information about the minimum hardware requirements for each host.

After you have procured the required hardware for the enterprise deployment, log in to each host computer and verify the system requirements listed in [Hardware and Software Requirements for the Enterprise Deployment Topology](#).

If you are deploying to a virtual server environment, such as Oracle Exalogic, ensure that each of the virtual servers meets the minimum requirements.

Ensure that you have sufficient local disk storage and shared storage configured as described in [Preparing the File System for an Enterprise Deployment](#).

Allow sufficient swap and temporary space; specifically:

- **Swap Space**—The system must have at least 500 MB.
- **Temporary Space**—There must be a minimum of 500 MB of free space in `/tmp`.

8.2 Verifying Linux Operating System Requirements

Review this section for typical Linux operating system settings for an enterprise deployment.

To ensure the host computers meet the minimum operating system requirements, be sure you have installed a certified operating system and that you have applied all the necessary patches for the operating system.

In addition, review the following sections for typical Linux operating system settings for an enterprise deployment.

See Also:

[Setting Linux Kernel Parameters](#)

[Setting the Open File Limit and Number of Processes Settings on UNIX Systems](#)

[Verifying IP Addresses and Host Names in DNS or hosts File](#)

8.2.1 Setting Linux Kernel Parameters

The kernel-parameter and shell-limit values shown below are recommended values only. Oracle recommends that you tune these values to optimize the performance of the system. See your operating system documentation for more information about tuning kernel parameters.

Kernel parameters must be set to a minimum of those in Table on all nodes in the topology.

The values in the following table are the current Linux recommendations. For the latest recommendations for Linux and other operating systems, see *Oracle Fusion Middleware System Requirements and Specifications*.

If you are deploying a database onto the host, you might need to modify additional kernel parameters. Refer to the 12c (12.2.1) *Oracle Grid Infrastructure Installation Guide* for your platform.

Table 8-1 UNIX Kernel Parameters

Parameter	Value
kernel.sem	256 32000 100 142
kernel.shmmax	4294967295

To set these parameters:

1. Log in as `root` and add or amend the entries in the file `/etc/sysctl.conf`.
2. Save the file.
3. Activate the changes by issuing the command:

```
/sbin/sysctl -p
```

8.2.2 Setting the Open File Limit and Number of Processes Settings on UNIX Systems

On UNIX operating systems, the `Open File Limit` is an important system setting, which can affect the overall performance of the software running on the host computer.

For guidance on setting the `Open File Limit` for an Oracle Fusion Middleware enterprise deployment, see [Host Computer Hardware Requirements](#).

Note:

The following examples are for Linux operating systems. Consult your operating system documentation to determine the commands to be used on your system.

For more information, see the following sections.

See Also:

[Viewing the Number of Currently Open Files](#)

[Setting the Operating System Open File and Processes Limits](#)

8.2.2.1 Viewing the Number of Currently Open Files

You can see how many files are open with the following command:

```
/usr/sbin/lsof | wc -l
```

To check your open file limits, use the following commands.

C shell:

```
limit descriptors
```

Bash:

```
ulimit -n
```

8.2.2.2 Setting the Operating System Open File and Processes Limits

To change the `Open File Limit` values:

1. Log in as `root` and edit the following file:

```
/etc/security/limits.conf
```

2. Add the following lines to the `limits.conf` file. (The values shown here are for example only):

```
* soft nofile 4096
* hard nofile 65536
* soft nproc 2047
* hard nproc 16384
```

The `nfiles` values represent the open file limit; the `nproc` values represent the number of processes limit.

3. Save the changes, and close the `limits.conf` file.

4. Reboot the host computer.

8.2.3 Verifying IP Addresses and Host Names in DNS or hosts File

Before you begin the installation of the Oracle software, ensure that the IP address, fully qualified host name, and the short name of the host are all registered with your DNS server. Alternatively, you can use the local `hosts` file and add an entry similar to the following:

```
IP_Address Fully_Qualified_Name Short_Name
```

For example:

```
10.229.188.205 host1.example.com host1
```

8.3 Configuring Operating System Users and Groups

The lists in this section show the users and groups to define on each of the computers that will host the enterprise deployment.

Groups

You must create the following groups on each node.

- `oinstall`
- `dba`

Users

You must create the following user on each node.

- `nobody`—An unprivileged user.
- `oracle`—The owner of the Oracle software. You may use a different name. The primary group for this account must be `oinstall`. The account must also be in the `dba` group.

Note:

- The group `oinstall` must have write privileges to all the file systems on shared and local storage that are used by the Oracle software.
 - Each group must have the same Group ID on every node.
 - Each user must have the same User ID on every node.
-
-

8.4 Enabling Unicode Support

This section provides information about enabling Unicode support.

Your operating system configuration can influence the behavior of characters supported by Oracle Fusion Middleware products.

On UNIX operating systems, Oracle highly recommends that you enable Unicode support by setting the `LANG` and `LC_ALL` environment variables to a locale with the UTF-8 character set. This enables the operating system to process any character in Unicode. Oracle SOA Suite technologies, for example, are based on Unicode.

If the operating system is configured to use a non-UTF-8 encoding, Oracle SOA Suite components may function in an unexpected way. For example, a non-ASCII file name might make the file inaccessible and cause an error. Oracle does not support problems caused by operating system constraints.

8.5 Mounting the Required Shared File Systems on Each Host

This section provides information about mounting the required shared file systems on each host.

The shared storage configured, as described in [Shared Storage Recommendations When Installing and Configuring an Enterprise Deployment](#), must be available on the hosts that use it.

In an enterprise deployment, it is assumed that you have a hardware storage filer, which is available and connected to each of the host computers you have procured for the deployment.

You must mount the shared storage to all servers that require access.

Each host must have appropriate privileges set within the Network Attached Storage (NAS) or Storage Area Network (SAN) so that it can write to the shared storage.

Follow the best practices of your organization for mounting shared storage. This section provides an example of how to do this on Linux using NFS storage.

You must create and mount shared storage locations so that SOAHOST1 and SOAHOST2 can see the same location if it is a binary installation in two separate volumes.

For more information, see [Shared Storage Recommendations When Installing and Configuring an Enterprise Deployment](#).

You use the following command to mount shared storage from a NAS storage device to a Linux host. If you are using a different type of storage device or operating system, refer to your manufacturer documentation for information about how to do this.

Note:

The user account used to create a shared storage file system owns and has read, write, and execute privileges for those files. Other users in the operating system group can read and execute the files, but they do not have write privileges.

For more information about installation and configuration privileges, see "Selecting an Installation User" in the *Oracle Fusion Middleware Installation Planning Guide*.

In the following example, `nasfiler` represents the shared storage filer. Also note that these are examples only. Typically, the mounting of these shared storage locations should be done using the `/etc/fstabs` file on UNIX systems, so that the mounting of these devices survives a reboot. Refer to your operating system documentation for more information.

1. Create the mount directories on SOAHOST1, as described in [Summary of the Shared Storage Volumes in an Enterprise Deployment](#), and then mount the shared storage. For example:

```
mount -t nfs nasfiler:VOL1/oracle/products/ /u01/oracle/products/
```

2. Repeat the procedure on SOAHOST2 using VOL2.

Validating the Shared Storage Configuration

Ensure that you can read and write files to the newly mounted directories by creating a test file in the shared storage location you just configured.

For example:

```
$ cd newly mounted directory
$ touch testfile
```

Verify that the owner and permissions are correct:

```
$ ls -l testfile
```

Then remove the file:

```
$ rm testfile
```

Note:

The shared storage can be a NAS or SAN device. The following example illustrates creating storage for a NAS device from SOAHOST1. The options may differ depending on the specific storage device.

```
mount -t nfs -o
rw,bg,hard,nointr,tcp,vers=3,timeo=300,rsz=32768,wsz=32768 nasfiler:VOL1/
Oracle /u01/oracle
```

Contact your storage vendor and machine administrator for the correct options for your environment.

8.6 Enabling the Required Virtual IP Addresses on Each Host

This section provides instruction to enable the required virtual IP addresses on each host.

To prepare each host for the enterprise deployment, you must enable the virtual IP (VIP) addresses described in [Reserving the Required IP Addresses for an Enterprise Deployment](#).

It is assumed that you have already reserved the VIP addresses and host names and that they have been enabled by your network administrator. You can then enable the VIPs on the appropriate host.

Note that the virtual IP addresses used for the enterprise topology are not persisted because they are managed by Whole Server Migration (for selected Managed Servers and clusters) or by manual failover (for the Administration Server).

To enable the VIP addresses on each host, run the following commands as root:

```
/sbin/ifconfig interface:index IPAddress netmask netmask
/sbin/arping -q -U -c 3 -I interface IPAddress
```

where *interface* is eth0, or eth1, and *index* is 0, 1, or 2.

For example:

```
/sbin/ifconfig eth0:1 100.200.140.206 netmask 255.255.255.0
```

Enable your network to register the new location of the virtual IP address:

```
/sbin/arping -q -U -c 3 -I eth0 100.200.140.206
```

Validate that the address is available by using the ping command from another node, for example:

```
/bin/ping 100.200.140.206
```

Preparing the Database for an Enterprise Deployment

This chapter describes procedures for preparing your database for an Oracle SOA Suite enterprise deployment.

This chapter provides information about the database requirements, creating database services and about the database backup strategies.

See Also:

[Preparing for an Enterprise Deployment](#)

[Overview of Preparing the Database for an Enterprise Deployment](#)

This section provides information about how to configure a supported database as part of an Oracle Fusion Middleware enterprise deployment.

[About Database Requirements](#)

Check that the database meets the requirements described in these sections.

[Creating Database Services](#)

When multiple Oracle Fusion Middleware products are sharing the same database, each product should be configured to connect to a separate, dedicated database service.

[Using SecureFiles for Large Objects \(LOBs\) in an Oracle Database](#)

Beginning with Oracle Database 11g Release 1, Oracle introduced SecureFiles, a new LOB storage architecture. Oracle recommends using SecureFiles for the Oracle Fusion Middleware schemas, in particular for the Oracle SOA Suite schemas.

[About Database Backup Strategies](#)

This section provides brief information about the necessity of database backup strategies.

[Implementing a Database Growth Management Strategy for Oracle SOA Suite](#)

An Oracle SOA Suite enterprise deployment presents several challenges for database administrators, including managing the growth of the Oracle SOA Suite database. Underestimating the importance of

managing the database can lead to issues when the database is moved to a production environment.

9.1 Overview of Preparing the Database for an Enterprise Deployment

This section provides information about how to configure a supported database as part of an Oracle Fusion Middleware enterprise deployment.

Most Oracle Fusion Middleware products require a specific set of schemas that must be installed in a supported database. The schemas are installed using the Oracle Fusion Middleware Repository Creation Utility (RCU).

In an enterprise deployment, Oracle recommends a highly available Real Application Clusters (Oracle RAC) database for the Oracle Fusion Middleware product schemas.

9.2 About Database Requirements

Check that the database meets the requirements described in these sections.

See Also:

[Supported Database Versions](#)

[Additional Database Software Requirements](#)

[Setting the PROCESSES Database Initialization Parameter for an Enterprise Deployment](#)

9.2.1 Supported Database Versions

Use the following information to verify what databases are supported by each Oracle Fusion Middleware release and which version of the Oracle database you are currently running:

- For a list of all certified databases, refer to *Oracle Fusion Middleware Supported System Configurations*.
- To check the release of your database, query the `PRODUCT_COMPONENT_VERSION` view:

```
SQL> SELECT VERSION FROM SYS.PRODUCT_COMPONENT_VERSION WHERE  
        PRODUCT LIKE 'Oracle%';
```

Oracle Fusion Middleware requires that the database supports the AL32UTF8 character set. Check the database documentation for information on choosing a character set for the database.

For enterprise deployments, Oracle recommends using GridLink data sources to connect to Oracle RAC databases.

Note:

For more information about using GridLink data sources and SCAN, see "Using Active GridLink Data Sources" in *Administering JDBC Data Sources for Oracle WebLogic Server*.

9.2.2 Additional Database Software Requirements

In the enterprise topology, there are two database host computers in the data tier that host the two instances of the RAC database. We refer to these hosts as DBHOST1 and DBHOST2.

Before you install or configure the enterprise topology, you must be sure the following software is installed and available on DBHOST1 and DBHOST2:

- **Oracle Clusterware**

For more information, see the *Oracle Grid Infrastructure Installation Guide for Linux*.

- **Oracle Real Application Clusters**

For more information, see the *Oracle Real Application Clusters Installation Guide for Linux and UNIX*.

- **Time synchronization between Oracle RAC database instances**

The clocks of the database instances being used by servers in a Fusion Middleware cluster that is configured with server migration must be in sync.

- **Automatic Storage Management (optional)**

For more information, see the *Oracle Automatic Storage Management Administrator's Guide*.

9.2.3 Setting the PROCESSES Database Initialization Parameter for an Enterprise Deployment

Table 9-1 lists some of the typical Oracle SOA Suite enterprise topologies and the value you should use when setting the PROCESSES initialization parameter for each topology.

Use this information as a guide when configuring the Oracle RAC database for an enterprise deployment.

Table 9-1 Required Initialization Parameters

Configuration	Parameter	Required Value	Parameter Class
SOA	PROCESSES	300 or greater	Static
BAM	PROCESSES	100 or greater	Static
SOA and BAM	PROCESSES	400 or greater	Static
SOA and OSB	PROCESSES	800 or greater	Static

To check the value of the initialization parameter using SQL*Plus, you can use the SHOW PARAMETER command.

1. As the SYS user, issue the SHOW PARAMETER command as follows:

```
SQL> SHOW PARAMETER processes;
```

2. Set the initialization parameter using the following command:

```
SQL> ALTER SYSTEM SET processes=300 SCOPE=SPFILE;
```

3. Restart the database.

The method that you use to change a parameter's value depends on whether the parameter is static or dynamic, and on whether your database uses a parameter file or a server parameter file.

Note:

See the *Oracle Database Administrator's Guide* for details on parameter files, server parameter files, and how to change parameter values.

9.3 Creating Database Services

When multiple Oracle Fusion Middleware products are sharing the same database, each product should be configured to connect to a separate, dedicated database service.

Note:

The instructions in this section are for the Oracle Database 12c (12.1) release. If you are using another supported database, refer to the appropriate documentation library for more up-to-date and release-specific information.

For more information about connecting to Oracle databases using services, see “Overview of Using Dynamic Database Services to Connect to Oracle Databases” in the *Real Application Clusters Administration and Deployment Guide*.

In addition, the database service should be different from the default database service. For complete instructions on creating and managing database services for an Oracle Database 12c database, see “Overview of Automatic Workload Management with Dynamic Database Services” in the *Real Application Clusters Administration and Deployment Guide*.

Run-time connection load balancing requires configuring Oracle RAC Load Balancing Advisory with service-level goals for each service for which load balancing is enabled.

You can configure the Oracle RAC Load Balancing Advisory for `SERVICE_TIME` or `THROUGHPUT`. Set the connection load-balancing goal to **SHORT**.

You create and modify Oracle Database services using the `srvctl` utility.

To create and modify a database service:

1. Log in to SQL*Plus and create the service:

```
sqlplus "sys/password as sysdba"

SQL> EXECUTE DBMS_SERVICE.CREATE_SERVICE
(SERVICE_NAME => 'soaedg.example.com',
NETWORK_NAME => 'soaedg.example.com'
);
```

Note:

For the Service Name of the Oracle RAC database, use lowercase letters, followed by the domain name. For example:

```
soaedg.example.com
```

Note:

Enter the EXECUTE DBMS_SERVICE command shown on a single line.

For more information about the DBMS_SERVICE package, see *Oracle Database PL/SQL Packages and Types Reference*.

2. Add the service to the database and assign it to the instances using `srvctl`:

```
srvctl add service -d soadb -s soaedg.example.com -r soadb1,soadb2
```

3. Start the service:

```
srvctl start service -d soadb -s soaedg.example.com
```

Note:

For complete instructions on creating and managing database services with SRVCTL, see "Creating Services with SRVCTL" in the *Real Application Clusters Administration and Deployment Guide*.

4. Modify the service so it uses the Load Balancing Advisory and the appropriate service-level goals for run-time connection load balancing.

More specifically, use the following resources in the Oracle Database 12c *Real Application Clusters Administration and Deployment Guide* to set the SERVICE_TIME and THROUGHPUT service-level goals:

- "Overview of the Load Balancing Advisory"
- "Configuring Your Environment to Use the Load Balancing Advisory"

9.4 Using SecureFiles for Large Objects (LOBs) in an Oracle Database

Beginning with Oracle Database 11g Release 1, Oracle introduced SecureFiles, a new LOB storage architecture. Oracle recommends using SecureFiles for the Oracle Fusion Middleware schemas, in particular for the Oracle SOA Suite schemas.

For more information, see "Using Oracle SecureFiles LOBs" in the *Oracle Database SecureFiles and Large Objects Developer's Guide*.

In Oracle 12c Databases, the default setting for using SecureFiles is `PREFERRED`. This means that the database attempts to create a SecureFiles LOB unless a BasicFiles LOB is explicitly specified for the LOB or the parent LOB (if the LOB is in a partition or sub-partition). The Oracle Fusion Middleware schemas do not explicitly specify BasicFiles, which means that Oracle Fusion Middleware LOBs will default to SecureFiles when installed in an Oracle 12c database.

For Oracle 11g databases, the `db_securefile` system parameter controls the SecureFiles usage policy. This parameter can be modified dynamically. The following options can be used for using SecureFiles:

- **PERMITTED**: allows SecureFiles to be created (This is the default setting for `db_securefile`. The default storage method uses BasicFiles)
- **FORCE**: create all (new) LOBs as SecureFiles
- **ALWAYS**: try to create LOBs as SecureFiles, but fall back to BasicFiles if not possible (if ASSM is disabled)

Other values for the `db_securefile` parameter are:

- **IGNORE**: ignore attempts to create SecureFiles
- **NEVER**: disallow new SecureFiles creations

For Oracle 11g Databases, Oracle recommends that you set the `db_securefile` parameter to **FORCE** before creating the Oracle Fusion Middleware schemas with the Repository Creation Utility (RCU).

Note that the SecureFiles segments require tablespaces managed with automatic segment space management (ASSM). This means that LOB creation on SecureFiles will fail if ASSM is not enabled. However, the Oracle Fusion Middleware tablespaces are created by default with ASSM enabled. As a result, with the default configuration, nothing needs to be changed to enable SecureFiles for the Oracle Fusion Middleware schemas.

9.5 About Database Backup Strategies

This section provides brief information about the necessity of database backup strategies.

At key points in the installation and configuration of an enterprise deployment, this guide recommends that you back up your current environment. For example, after you install the product software and create the schemas for a particular Oracle Fusion Middleware product, you should perform a database backup. Performing a backup allows you to perform a quick recovery from any issue that might occur in the later configuration steps.

You can choose to use your own backup strategy for the database, or you can simply make a backup using operating system tools or RMAN for this purpose.

Oracle recommends using Oracle Recovery Manager for the database, particularly if the database was created using Oracle Automatic Storage Management. If possible, you can also perform a cold backup using operating system tools such as tar.

9.6 Implementing a Database Growth Management Strategy for Oracle SOA Suite

An Oracle SOA Suite enterprise deployment presents several challenges for database administrators, including managing the growth of the Oracle SOA Suite database. Underestimating the importance of managing the database can lead to issues when the database is moved to a production environment.

For information about determining an appropriate strategy and planning for capacity, testing, and monitoring, see "Introduction to Planning for Database Growth" in the *Administering Oracle SOA Suite and Oracle Business Process Management Suite*.

Part III

Configuring the Enterprise Deployment

This part of the Enterprise Deployment Guide contains the following topics:

See Also:

[Creating the Initial Infrastructure Domain for an Enterprise Deployment](#)

[Configuring the Web Tier for an Enterprise Deployment](#)

[Extending the Domain with Oracle SOA Suite](#)

[Adding Oracle Service Bus to the Topology](#)

The following topics describe the procedures for adding Oracle Service Bus (OSB) to the enterprise deployment topology.

[Extending the Domain with Business Process Management](#)

[Extending the Domain with Oracle Enterprise Scheduler](#)

[Extending the Domain with Business Activity Monitoring](#)

[Extending the Domain with Oracle B2B](#)

[Extending the Domain with Oracle SOA Suite for Healthcare Integration](#)

The following topics describe the procedures for extending the domain to include Oracle SOA Suite for healthcare integration (Oracle Healthcare).

Creating the Initial Infrastructure Domain for an Enterprise Deployment

The following topics describe how to install and configure an initial domain, which can be used as the starting point for an enterprise deployment. Later chapters in this guide describe how to extend this initial domain with the various products and components that comprise the enterprise topology you are deploying.

See Also:

[Configuring the Enterprise Deployment](#)

[Variables Used When Creating the Infrastructure Domain](#)

As you perform the tasks in this chapter, you will be referencing the directory variables listed in this section.

[Understanding the Initial Infrastructure Domain](#)

Before you create the initial Infrastructure domain, be sure to review the following key concepts.

[Installing the Oracle Fusion Middleware Infrastructure in Preparation for an Enterprise Deployment](#)

Use the following sections to install the Oracle Fusion Middleware Infrastructure software in preparation for configuring a new domain for an enterprise deployment.

[Creating the Database Schemas](#)

Before you can configure a Fusion Middleware Infrastructure domain, you must install the schemas listed in this section in a certified database for use with this release of Oracle Fusion Middleware.

[Configuring the Infrastructure Domain](#)

The following topics provide instructions for creating a WebLogic Server domain using the Fusion Middleware Configuration wizard.

[Configuring a Per Host Node Manager for an Enterprise Deployment](#)

For specific enterprise deployments, Oracle recommends that you configure a per-host Node Manager, as opposed to the default per-domain Node Manager.

[Configuring the Domain Directories and Starting the Servers on SOAHOST1](#)

After the domain is created and the node manager is configured, you can then configure the additional domain directories and start the Administration Server and the Managed Servers on SOAHOST1.

[Propagating the Domain and Starting the Servers on SOAHOST2](#)

After you start and validate the Administration Server and WLS_WSM1 Managed Server on SOAHOST1, you can then perform the following tasks on SOAHOST2.

[Modifying the Upload and Stage Directories to an Absolute Path](#)

After creating the domain and unpacking it to the Managed Server domain directory, verify and update the `upload` and `stage` directories for the Managed Servers.

[Creating a New LDAP Authenticator and Provisioning Enterprise Deployment Users and Group](#)

When you configure an Oracle Fusion Middleware domain, the domain is configured by default to use the WebLogic Server authentication provider (`DefaultAuthenticator`). However, for an enterprise deployment, Oracle recommends that you use a dedicated, centralized LDAP-compliant authentication provider.

[Adding the `wsm-pm` Role to the Administrators Group](#)

After you configure a new LDAP-based Authorization Provider and restart the Administration Server, add the enterprise deployment

administration LDAP group (SOA Administrators) as a member to the wsm-pm application stripe role: policy.Updater.

10.1 Variables Used When Creating the Infrastructure Domain

As you perform the tasks in this chapter, you will be referencing the directory variables listed in this section.

These directory variables are defined in [File System and Directory Variables Used in This Guide](#).

- ORACLE_HOME
- ASERVER_HOME
- MSERVER_HOME
- APPLICATION_HOME
- JAVA_HOME
- NM_HOME

In addition, you'll be referencing the following virtual IP (VIP) addresses and host names defined in [Physical and Virtual IP Addresses Required by the Enterprise Topology](#):

- ADMINVHN
- SOAHOST1
- SOAHOST2
- DBHOST1
- DBHOST2
- SCAN Address for the Oracle RAC Database (DB-SCAN.example.com)

10.2 Understanding the Initial Infrastructure Domain

Before you create the initial Infrastructure domain, be sure to review the following key concepts.

See Also:

[About the Infrastructure Distribution](#)

[Characteristics of the Initial Infrastructure Domain](#)

10.2.1 About the Infrastructure Distribution

You create the initial Infrastructure domain for an enterprise deployment using the Oracle Fusion Middleware Infrastructure distribution. This distribution contains both the Oracle WebLogic Server software and the Oracle JRF software.

The Oracle JRF software consists of Oracle Web Services Manager, Oracle Application Development Framework (Oracle ADF), Oracle Enterprise Manager Fusion Middleware Control (Fusion Middleware Control), the Repository Creation Utility (RCU), and other libraries and technologies required to support the Oracle Fusion Middleware products.

Later in this guide, you can then extend the domain to support the Oracle Fusion Middleware products required for your enterprise deployment.

For more information, see "Understanding Oracle Fusion Middleware Infrastructure" in *Understanding Oracle Fusion Middleware*.

10.2.2 Characteristics of the Initial Infrastructure Domain

[Table 10-1](#) lists some of the key characteristics of the initial Infrastructure domain. By reviewing and understanding these characteristics, you can better understand the purpose and context of the procedures used to configure the domain.

Many of these characteristics are described in more detail in [Understanding a Typical Enterprise Deployment](#).

Table 10-1 Characteristics of the Initial Infrastructure domain

Characteristic of the Domain	More Information
Uses a separate virtual IP (VIP) address for the Administration Server.	Configuration of the Administration Server and Managed Servers Domain Directories
Uses separate domain directories for the Administration Server and the Managed Servers in the domain.	Configuration of the Administration Server and Managed Servers Domain Directories
Includes a dedicated cluster for Oracle Web Services Manager	Using Oracle Web Services Manager in the Application Tier
Uses a per host Node Manager configuration.	About the Node Manager Configuration in a Typical Enterprise Deployment
Requires a separately installed LDAP-based authentication provider.	Understanding OPSS and Requests to the Authentication and Authorization Stores

10.3 Installing the Oracle Fusion Middleware Infrastructure in Preparation for an Enterprise Deployment

Use the following sections to install the Oracle Fusion Middleware Infrastructure software in preparation for configuring a new domain for an enterprise deployment.

See Also:

[Installing a Supported JDK](#)

[Starting the Infrastructure Installer on SOAHOST1](#)

[Navigating the Infrastructure Installation Screens](#)

[Installing Oracle Fusion Middleware Infrastructure on the Other Host Computers](#)

[Checking the Directory Structure](#)

10.3.1 Installing a Supported JDK

Oracle Fusion Middleware requires that a certified Java Development Kit (JDK) is installed on your system. See the following sections for more information:

See Also:

[Locating and Downloading the JDK Software](#)

[Installing the JDK Software](#)

10.3.1.1 Locating and Downloading the JDK Software

To find a certified JDK, see the certification document for your release on the Oracle Fusion Middleware Supported System Configurations page.

After you identify the Oracle JDK for the current Oracle Fusion Middleware release, you can download an Oracle JDK from the following location on Oracle Technology Network:

<http://www.oracle.com/technetwork/java/index.html>

Be sure to navigate to the download for the Java SE JDK.

10.3.1.2 Installing the JDK Software

Install the JDK in the following locations:

- On the shared storage device, where it will be accessible from each of the application tier host computers.
- On the local storage device for each of the Web tier host computers.
The Web tier host computers, which reside in the DMZ, do not necessarily have access to the shared storage on the application tier.

For more information about the recommended location for the JDK software, see the [Understanding the Recommended Directory Structure for an Enterprise Deployment](#).

The following example describes how to install a recent version of JDK 1.8:

1. Change directory to the location where you downloaded the JDK archive file.
2. Unpack the archive into the JDK home directory, and then run these commands:

```
cd download_dir
tar -xvzf jdk-8u51-linux-x64.tar.gz
```

Note that the JDK version listed here was accurate at the time this document was published. For the latest supported JDK, see the *Oracle Fusion Middleware System Requirements and Specifications* for the current Oracle Fusion Middleware release.

3. Move the JDK directory to the recommended location in the directory structure.

For example:

```
mv ./jdk1.8.0_51 /u01/oracle/products/jdk
```

For more information, see [File System and Directory Variables Used in This Guide](#).

4. Define the `JAVA_HOME` and `PATH` environment variables for running Java on the host computer.

For example:

```
export JAVA_HOME=/u01/oracle/products/jdk
export PATH=$JAVA_HOME/bin:$PATH
```

5. Run the following command to verify that the appropriate java executable is in the path and your environment variables are set correctly:

```
java -version
java version "1.8.0_51"
Java(TM) SE Runtime Environment (build 1.8.0_51-b16)
Java HotSpot(TM) 64-Bit Server VM (build 25.51-b03, mixed mode)
```

10.3.2 Starting the Infrastructure Installer on SOAHOST1

To start the installation program, perform the following steps.

1. Log in to SOAHOST1.
2. Go to the directory where you downloaded the installation program.
3. Launch the installation program by invoking the java executable from the JDK directory on your system, as shown in the example below.

```
JAVA_HOME/bin/java -d64 -jar distribution_file_name.jar
```

In this example:

- Replace *JAVA_HOME* with the environment variable or actual JDK location on your system.
- Replace *distribution_file_name* with the actual name of the distribution JAR file.

Note that if you download the distribution from the Oracle Technology Network (OTN), then the JAR file is typically packaged inside a downloadable ZIP file.

To install the software required for the initial Infrastructure domain, the distribution you want to install is **fmw_12.2.1.0.0_infrastructure.jar**.

For more information about the actual file names of each distribution, see [Identifying and Obtaining Software Downloads for an Enterprise Deployment](#).

When the installation program appears, you are ready to begin the installation. See [Navigating the Installation Screens](#) for a description of each installation program screen.

10.3.3 Navigating the Infrastructure Installation Screens

The installation program displays a series of screens, in the order listed in the following table.

If you need additional help with any of the installation screens, click the screen name.

Screen	Description
Installation Inventory Setup	<p>On UNIX operating systems, this screen will appear if this is the first time you are installing any Oracle product on this host. Specify the location where you want to create your central inventory. Make sure that the operating system group name selected on this screen has write permissions to the central inventory location.</p> <p>For more information about the central inventory, see "Understanding the Oracle Central Inventory" in <i>Installing Software with the Oracle Universal Installer</i>.</p>
Welcome	This screen introduces you to the product installer.
Auto Updates	Use this screen to automatically search My Oracle Support for available patches or automatically search a local directory for patches that you've already downloaded for your organization.
Installation Location	<p>Use this screen to specify the location of your Oracle home directory.</p> <p>For the purposes of an enterprise deployment, enter the value of the <code>ORACLE_HOME</code> variable listed in Table 7-2.</p>
Installation Type	<p>Use this screen to select the type of installation and consequently, the products and feature sets you want to install.</p> <p>For this topology, select Fusion Middleware Infrastructure.</p> <p>Note: The topology in this document does not include server examples. Oracle strongly recommends that you do not install the examples into a production environment.</p>
Prerequisite Checks	<p>This screen verifies that your system meets the minimum necessary requirements.</p> <p>If there are any warning or error messages, refer to the Oracle Fusion Middleware System Requirements and Specifications document on the Oracle Technology Network (OTN).</p>
Security Updates	<p>If you already have an Oracle Support account, use this screen to indicate how you would like to receive security updates.</p> <p>If you do not have one and are sure you want to skip this step, clear the check box and verify your selection in the follow-up dialog box.</p>
Auto Updates - Patch Selection	<p>This screen appears if both of the following statements are true:</p> <ul style="list-style-type: none"> You searched for available patches earlier in the installation session, using the Auto Updates screen. The Auto Updates feature located one or more application patches that must be applied to the Oracle home you are creating in this installation session. <p>This screen lists the patches that were found by the Auto Updates feature. Select one or more patches and click Next to apply the selected patches to the Oracle home.</p>

Screen	Description
Installation Summary	<p>Use this screen to verify the installation options you selected. If you want to save these options to a response file, click Save Response File and provide the location and name of the response file. Response files can be used later in a silent installation situation.</p> <p>For more information about silent or command-line installation, see "Using the Oracle Universal Installer in Silent Mode" in <i>Installing Software with the Oracle Universal Installer</i>.</p>
Installation Progress	This screen allows you to see the progress of the installation.
Installation Complete	This screen appears when the installation is complete. Review the information on this screen, then click Finish to dismiss the installer.

10.3.4 Installing Oracle Fusion Middleware Infrastructure on the Other Host Computers

If you have configured a separate shared storage volume or partition for SOAHOST2, then you must also install the Infrastructure on SOAHOST2.

For more information, see [Shared Storage Recommendations When Installing and Configuring an Enterprise Deployment](#).

To install the software on the other host computers in the topology, log in to each host, and use the instructions in [Starting the Infrastructure Installer on SOAHOST1](#) and [Navigating the Infrastructure Installation Screens](#) to create the Oracle home on the appropriate storage device.

Note:

In previous releases, the recommended enterprise topology included a colocated set of Oracle HTTP Server instances. In those releases, there was a requirement to install the Infrastructure on the Web Tier hosts (WEBHOST1 and WEBHOST2). However, for this release, the enterprise deployment topology assumes the Web servers are installed and configured in standalone mode, so they are not considered part of the application tier domain. For more information, see [Configuring the Web Tier for an Enterprise Deployment](#)

10.3.5 Checking the Directory Structure

After you install the Oracle Fusion Middleware Infrastructure and create the Oracle home, you should see the following directory and sub-directories. The contents of your installation vary based on the options you selected during the installation.

To check the directory structure:

1. Change directory to the `ORACLE_HOME` directory.
2. Enter the following command:

```
ls -l
```

The directory structure on your system should match the structure shown in the following example:


```
/u01/oracle/products/fmw/  
  
cfgtoollogs  
coherence  
em  
install  
inventory  
OPatch  
oracle_common  
oraInst.loc  
oui  
root.sh  
wlserver
```

For more information about the directory structure after the installation complete, see "What are the Key Oracle Fusion Middleware Directories?" in *Understanding Oracle Fusion Middleware*.

10.4 Creating the Database Schemas

Before you can configure a Fusion Middleware Infrastructure domain, you must install the schemas listed in this section in a certified database for use with this release of Oracle Fusion Middleware.

- Metadata Services (MDS)
- Audit Services (IAU)
- Audit Services Append (IAU_APPEND)
- Audit Services Viewer (IAU_VIEWER)
- Oracle Platform Security Services (OPSS)
- User Messaging Service (UMS)
- WebLogic Services (WLS)
- Common Infrastructure Services (STB)

You use the Repository Creation Utility (RCU) to create the schemas. This utility is installed in the Oracle home for each Oracle Fusion Middleware product. For more information about RCU and how the schemas are created and stored in the database, see "Preparing for Schema Creation" in *Creating Schemas with the Repository Creation Utility*.

Follow the instructions in this section to install the required schemas.

See Also:

[Installing and Configuring a Certified Database](#)

[Starting the Repository Creation Utility \(RCU\)](#)

[Navigating the RCU Screens to Create the Schemas](#)

10.4.1 Installing and Configuring a Certified Database

Make sure you have installed and configured a certified database, and that the database is up and running.

For more information, see the [Preparing the Database for an Enterprise Deployment](#).

10.4.2 Starting the Repository Creation Utility (RCU)

To start the Repository Creation Utility (RCU):

1. Navigate to the `ORACLE_HOME/oracle_common/bin` directory on your system.
2. Make sure the `JAVA_HOME` environment variable is set to the location of a certified JDK on your system. The location should be up to but not including the `bin` directory. For example, if your JDK is located in `/u01/oracle/products/jdk`:

On UNIX operating systems:

```
export JAVA_HOME=/u01/oracle/products/jdk
```

3. Start RCU:

On UNIX operating systems:

```
./rcu
```

10.4.3 Navigating the RCU Screens to Create the Schemas

Follow the instructions in this section to create the schemas for the Fusion Middleware Infrastructure domain:

- [Task 1, "Introducing RCU"](#)
- [Task 2, "Selecting a Method of Schema Creation"](#)
- [Task 3, "Providing Database Credentials"](#)
- [Task 4, "Specifying a Custom Prefix and Selecting Schemas"](#)
- [Task 5, "Specifying Schema Passwords"](#)
- [Task 6, "Completing Schema Creation"](#)

Task 1 Introducing RCU

Review the Welcome screen and verify the version number for RCU. Click **Next** to begin.

Task 2 Selecting a Method of Schema Creation

If you have the necessary permission and privileges to perform DBA activities on your database, select **System Load and Product Load Concurrently** on the Create Repository screen. The procedure in this document assumes that you have the necessary privileges.

If you do not have the necessary permission or privileges to perform DBA activities in the database, you must select **Prepare Scripts for System Load** on this screen. This option will generate a SQL script, which can be provided to your database administrator. See "Understanding System Load and Product Load" in *Creating Schemas with the Repository Creation Utility*.

Tip:

For more information about the options on this screen, see "Create repository" in *Creating Schemas with the Repository Creation Utility*.

Task 3 Providing Database Credentials

On the Database Connection Details screen, provide the database connection details for RCU to connect to your database.

In the **Host Name** field, enter the SCAN address of the Oracle RAC Database.

Click **Next** to proceed, then click **OK** in the dialog window confirming that connection to the database was successful.

Tip:

For more information about the options on this screen, see "Database Connection Details" in *Creating Schemas with the Repository Creation Utility*.

Task 4 Specifying a Custom Prefix and Selecting Schemas

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

The custom prefix is used to logically group these schemas together for use in this domain. For the purposes of this guide, use the prefix FMW1221.

Tip:

Make a note of the custom prefix you choose to enter here; you will need this later, during the domain creation process.

For more information about custom prefixes, see "Understanding Custom Prefixes" in *Creating Schemas with the Repository Creation Utility*.

2. Select **AS Common Schemas**.

When you select **AS Common Schemas**, all of the schemas in this section are automatically selected.

A schema called **Common Infrastructure Services** is also automatically created; this schema is grayed out and cannot be selected or deselected. This schema (the STB schema) enables you to retrieve information from RCU during domain configuration. For more information, see "Understanding the Service Table Schema" in *Creating Schemas with the Repository Creation Utility*.

Tip:

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

Specify a unique prefix for all schemas created in this session, so you can easily locate, reference, and manage the schemas later.

Select existing prefix:

Create new prefix:

Alpha numeric only. Cannot start with a number. No special characters.

Component	Schema Owner
<input checked="" type="checkbox"/> Oracle AS Repository Components	
<input checked="" type="checkbox"/> AS Common Schemas	
<input checked="" type="checkbox"/> User Messaging Service	FMW1221_UMS
<input checked="" type="checkbox"/> Metadata Services	FMW1221_MDS
<input checked="" type="checkbox"/> WebLogic Services	FMW1221_WLS
<input type="checkbox"/> Common Infrastructure Services	FMW1221_STB
<input checked="" type="checkbox"/> Oracle Platform Security Services	FMW1221_OPSS
<input checked="" type="checkbox"/> Audit Services	FMW1221_IAU
<input checked="" type="checkbox"/> Audit Services Append	FMW1221_IAU_APPEND
<input checked="" type="checkbox"/> Audit Services Viewer	FMW1221_IAU_VIEWER

Click **Next** to proceed, then click **OK** on the dialog window confirming that prerequisite checking for schema creation was successful.

Task 5 Specifying Schema Passwords

Specify how you want to set the schema passwords on your database, then specify and confirm your passwords.

Tip:

You must make a note of the passwords you set on this screen; you will need them later on during the domain creation process.

Task 6 Completing Schema Creation

Navigate through the remainder of the RCU screens to complete schema creation.

For the purposes of this guide, you can accept the default settings on the remaining screens, or you can customize how RCU creates and uses the required tablespaces for the Oracle Fusion Middleware schemas.

For more information about RCU and its features and concepts, see *Creating Schemas with the Repository Creation Utility*.

When you reach the Completion Summary screen, click **Close** to dismiss RCU.

10.5 Configuring the Infrastructure Domain

The following topics provide instructions for creating a WebLogic Server domain using the Fusion Middleware Configuration wizard.

For more information on other methods available for domain creation, see "Additional Tools for Creating, Extending, and Managing WebLogic Domains" in *Creating WebLogic Domains Using the Configuration Wizard*.

See Also:

[Starting the Configuration Wizard](#)

Navigating the Configuration Wizard Screens to Configure the Infrastructure Domain

10.5.1 Starting the Configuration Wizard

To begin domain configuration, run the following command in the Oracle Fusion Middleware Oracle home.

```
ORACLE_HOME/oracle_common/common/bin/config.sh
```

10.5.2 Navigating the Configuration Wizard Screens to Configure the Infrastructure Domain

Follow the instructions in this section to create and configure the domain for the topology.

Task 1 Selecting the Domain Type and Domain Home Location

On the Configuration Type screen, select **Create a New Domain**.

In the Domain Location field, specify the value of the *ASERVER_HOME* variable, as defined in [File System and Directory Variables Used in This Guide](#).

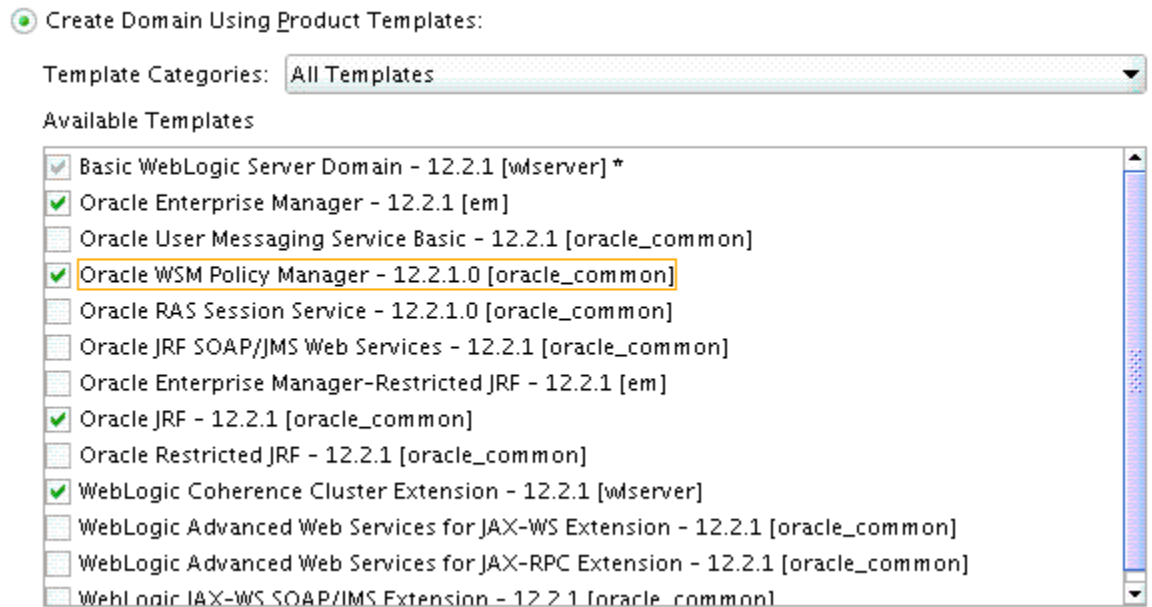
Tip:

More information about the other options on this screen of the Configuration Wizard, see "Configuration Type" in *Creating WebLogic Domains Using the Configuration Wizard*.

Task 2 Selecting the Configuration Templates

On the Templates screen, make sure **Create Domain Using Product Templates** is selected, then select the following templates:

- **Oracle Enterprise Manager - 12.2.1.0 [em]**
Selecting this template automatically selects the following dependencies:
 - Oracle JRF - 12.2.1.0 [oracle_common]
 - WebLogic Coherence Cluster Extension - 12.2.1.0 [wlserver]
- **Oracle WSM Policy Manager - 12.2.1.0 [oracle_common]**



Tip:

More information about the options on this screen can be found in Templates in *Creating WebLogic Domains Using the Configuration Wizard*.

Task 3 Selecting the Application Home Location

On the Application Location screen, specify the value of the `APPLICATION_HOME` variable, as defined in [File System and Directory Variables Used in This Guide](#).

Tip:

More information about the options on this screen can be found in Application Location in *Creating WebLogic Domains Using the Configuration Wizard*.

Task 4 Configuring the Administrator Account

On the Administrator Account screen, specify the user name and password for the default WebLogic Administrator account for the domain.

Make a note of the user name and password specified on this screen; you will need these credentials later to boot and connect to the domain's Administration Server.

Task 5 Specifying the Domain Mode and JDK

On the Domain Mode and JDK screen:

- Select **Production** in the Domain Mode field.
- Select the **Oracle Hotspot** JDK in the JDK field.

Selecting **Production Mode** on this screen gives your environment a higher degree of security, requiring a user name and password to deploy applications and to start the Administration Server.

Tip:

More information about the options on this screen, including the differences between development mode and production mode, can be found in Domain Mode and JDK in *Creating WebLogic Domains Using the Configuration Wizard*.

In production mode, a boot identity file can be created to bypass the need to provide a user name and password when starting the Administration Server. For more information, see [Creating the boot.properties File](#).

Task 6 Specifying the Database Configuration Type

Select **RCU Data** to activate the fields on this screen.

The **RCU Data** option instructs the Configuration Wizard to connect to the database and Service Table (STB) schema to automatically retrieve schema information for the schemas needed to configure the domain.

Note:

If you choose to select **Manual Configuration** on this screen, you will have to manually fill in the parameters for your schema on the JDBC Component Schema screen.

After selecting **RCU Data**, fill in the fields as shown in the following table. Refer to [Figure 10-1](#) for a partial screen shot of a sample Database Configuration Type screen.

Field	Description
DBMS/Service	<p>Enter the service name for the Oracle RAC database where you will install the product schemas. For example:</p> <p><code>orcl.example.com</code></p> <p>Be sure this is the common service name that is used to identify all the instances in the Oracle RAC database; do not use the host-specific service name.</p>
Host Name	<p>Enter the Single Client Access Name (SCAN) Address for the Oracle RAC database, which you entered in the <i>Enterprise Deployment Workbook</i>.</p>
Port	<p>Enter the port number on which the database listens. For example, 1521.</p>
Schema Owner Schema Password	<p>Enter the user name and password for connecting to the database's Service Table schema.</p> <p>This is the schema user name and password that was specified for the Service Table component on the "Schema Passwords" screen in RCU (see Creating the Database Schemas).</p> <p>The default user name is <code>prefix_STB</code>, where <code>prefix</code> is the custom prefix that you defined in RCU.</p>

Figure 10-1 Setting the Database Configuration Type for an Enterprise Deployment

Specify AutoConfiguration Options Using:

 RCU Data Manual Configuration

Enter the database connection details using the Repository Creation Utility service table (STB) schema credentials. The Wizard uses this connection to automatically configure the datasources required for components in this domain.

Vendor:	<input type="text" value="Oracle"/>	Driver:	<input type="text" value="*Oracle's Driver (Thin) for Service connections; Ver..."/>		
DBMS/Service:	<input type="text" value="orcl.example.com"/>	Host Name:	<input type="text" value="dbhost1.example.com"/>	Port:	<input type="text" value="1521"/>
Schema Owner:	<input type="text" value="FMW1221_STB"/>	Schema Password:	<input type="password" value="....."/>		

Click **Get RCU Configuration** when you are finished specifying the database connection information. The following output in the Connection Result Log indicates that the operating succeeded:

```
Connecting to the database server...OK
Retrieving schema data from database server...OK
Binding local schema components with retrieved data...OK
```

Successfully Done.

Click **Next** if the connection to the database is successful.

Tip:

More information about the **RCU Data** option can be found in "Understanding the Service Table Schema" in *Creating Schemas with the Repository Creation Utility*.

More information about the other options on this screen can be found in Datasource Defaults in *Creating WebLogic Domains Using the Configuration Wizard*

Task 7 Specifying JDBC Component Schema Information

Verify that the values on the JDBC Component Schema screen are correct for all schemas.

The schema table should be populated, because you selected **Get RCU Data** on the previous screen. As a result, the Configuration Wizard locates the database connection values for all the schemas required for this domain.

At this point, the values are configured to connect to a single-instance database. However, for an enterprise deployment, you should use a highly available Real Application Clusters (RAC) database, as described in [Preparing the Database for an Enterprise Deployment](#).

In addition, Oracle recommends that you use an Active GridLink datasource for each of the component schemas. For more information about the advantages of using GridLink data sources to connect to a RAC database, see "Database Considerations" in the *High Availability Guide*.

To convert the data sources to GridLink:

1. Select all the schemas by selecting the checkbox at in the first header row of the schema table.
2. Click **Convert to GridLink** and click **Next**.

Task 8 Providing the GridLink Oracle RAC Database Connection Details

On the GridLink Oracle RAC Component Schema screen, provide the information required to connect to the RAC database and component schemas, as shown in following table.

Element	Description and Recommended Value
SCAN, Host Name, and Port	Select the SCAN check box. In the Host Name field, enter the Single Client Access Name (SCAN) Address for the Oracle RAC database. In the Port field, enter the SCAN listening port for the database (for example, 1521)
ONS Host and Port	In the ONS Host field, enter the SCAN address for the Oracle RAC database. In the Port field, enter the ONS Remote port (typically, 6200).
Enable Fan	Verify that the Enable Fan check box is selected, so the database can receive and process FAN events.

Figure 10-2 Sample Values for the GridLink Oracle RAC Component Schema Scree

The screenshot shows the configuration interface for the GridLink Oracle RAC Component Schema. It includes several input fields and checkboxes:

- Driver:** *Oracle's Driver (Thin) for GridLink Connecti
- Service Name:** ORCL.US.ORACLE.COM
- Schema Owner:** Varies among component schemas
- Schema Password:** [Redacted]
- Enable FAN:** **Enable SSL:**
- Wallet File:** Enter a value
- Wallet Password:** Enter a value
- SCAN:** **Host Name:** db-scan.examp; **Port:** 1521

There are two tables displayed:

Service Listener	Port	Protocol

ONS Host	Port
db-scan.example.com	6200

<input checked="" type="checkbox"/>	RAC Component Schema	Service Name	Schema Owner	Schema Password
<input checked="" type="checkbox"/>	LocalSvcTbl Schema	ORCL.US.ORACLE.C	FMW1221_STB	[Redacted]
<input checked="" type="checkbox"/>	OWSM MDS Schema	ORCL.US.ORACLE.C	FMW1221_MDS	[Redacted]
<input checked="" type="checkbox"/>	OPSS Audit Schema	ORCL.US.ORACLE.C	FMW1221_JAU_APPI	[Redacted]
<input checked="" type="checkbox"/>	OPSS Audit Viewer Schema	ORCL.US.ORACLE.C	FMW1221_JAU_VIEW	[Redacted]
<input checked="" type="checkbox"/>	OPSS Schema	ORCL.US.ORACLE.C	FMW1221_OPSS	[Redacted]

For more information about specifying the information on this screen, as well as information about how to identify the correct SCAN address, see "Configuring Active GridLink Data Sources with Oracle RAC" in the *High Availability Guide*.

You can also click **Help** to display a brief description of each field on the screen.

Task 9 Testing the JDBC Connections

Use the JDBC Component Schema Test screen to test the data source connections you have just configured.

A green check mark in the Status column indicates a successful test. If you encounter any issues, see the error message in the Connection Result Log section of the screen, fix the problem, then try to test the connection again.

Tip:

More information about the other options on this screen can be found in Test Component Schema in *Creating WebLogic Domains Using the Configuration Wizard*

Task 10 Selecting Advanced Configuration

To complete domain configuration for the topology, select the following options on the Advanced Configuration screen:

- **Administration Server**
This is required to properly configure the listen address of the Administration Server.
- **Node Manager**
This is required to configure Node Manager.
- **Managed Servers, Clusters and Coherence**
This is required to configure the Managed Servers and cluster, and also for configuring the machine and targeting the Managed Servers to the machine.
- **JMS File Store**
This is required to configure the appropriate shared storage for JMS persistent stores.

Note:

When using the Advanced Configuration screen in the Configuration Wizard:

- If any of the above options are not available on the screen, then return to the Templates screen, and be sure you selected the required templates for this topology.
 - Do not select the **Domain Frontend Host Capture** advanced configuration option. You will later configure the frontend host property for specific clusters, rather than for the domain.
-
-

Task 11 Configuring the Administration Server Listen Address

On the Administration Server screen:

1. In the **Server Name** field, retain the default value - AdminServer.

2. In the **Listen Address** field, enter the virtual host name that corresponds to the VIP of the ADMINVHN that you procured in [Procuring Resources for an Enterprise Deployment](#) and enabled in [Preparing the Host Computers for an Enterprise Deployment](#).

For more information on the reasons for using the ADMINVHN virtual host, see [Reserving the Required IP Addresses for an Enterprise Deployment](#).

3. Leave the other fields at their default values.

In particular, be sure that no server groups are assigned to the Administration Server.

Task 12 Configuring Node Manager

Select **Manual Node Manager Setup** as the Node Manager type.

Tip:

For more information about the options on this screen, see "Node Manager" in *Creating WebLogic Domains Using the Configuration Wizard*.

For more information about per domain and per host Node Manager implementations, see [About the Node Manager Configuration in a Typical Enterprise Deployment](#).

For additional information, see "Configuring Node Manager on Multiple Machines" in *Administering Node Manager for Oracle WebLogic Server*.

Task 13 Configuring Managed Servers

Use the Managed Servers screen to create two new Managed Servers:

1. Click the **Add** button to create a new Managed Server.
2. Specify WLS_WSM1 in the **Server name** column.
3. In the **Listen Address** column, enter *SOAHOST1*.

Be sure to enter the host name that corresponds to SOAHOST1; do not use the IP address.

4. In the **Listen Port** column, enter 7010.
5. In the **Server Groups** drop-down list, select **JRF-MAN-SVR**, **WSM-CACHE-SVR**, and **WSMPM-MAN-SVR**. (See [Figure 10-3](#).)

These server groups ensure that the Oracle JRF and Oracle Web Services Manager (OWSM) services are targeted to the Managed Servers you are creating.

Server groups target Fusion Middleware applications and services to one or more servers by mapping defined groups of application services to each defined server group. Any application services that are mapped to a given server group are automatically targeted to all servers that are assigned to that group. For more information, see "Application Service Groups, Server Groups, and Application Service Mappings" in *Domain Template Reference*.

Note:

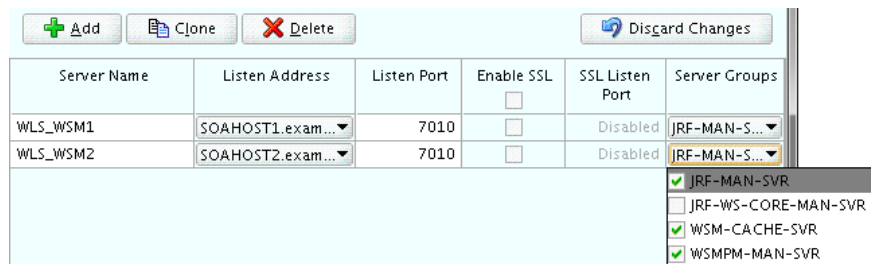
Once caching for Oracle Web Services is configured automatically by the WSM-CACHE-SVR server group and is suitable for most applications. Nonce is a unique number that can be used only once in a SOAP request and is used to prevent replay attacks. Nonce caching will naturally scale with the number of added Managed Servers running Web service applications.

For advanced caching configurations, see "Caching the Nonce with Oracle Coherence" in *Securing Web Services and Managing Policies with Oracle Web Services Manager*, which provides additional guidance for the use of nonce caching and the WSM-CACHE-SVR server-group.

- Repeat this process to create a second Managed Server named WLS_WSM2. For the **Listen Address**, enter SOAHOST2. For the **Listen Port**, enter 7010. Apply the same server groups you applied to the first managed server to the WLS_WSM2.

The Managed Server names suggested in this procedure (WLS_WSM1 and WLS_WSM2) will be referenced throughout this document; if you choose different names then be sure to replace them as needed.

Figure 10-3 Using the Configuration Wizard to Define Managed Servers for an Enterprise Deployment



Tip: More information about the options on this screen can be found in Managed Servers in *Creating WebLogic Domains Using the Configuration Wizard*.

Task 14 Configuring a Cluster

Use the Clusters screen to create a new cluster:

- Click the **Add** button.
- Specify WSM-PM_Cluster in the **Cluster Name** field.
- Leave the other fields empty.



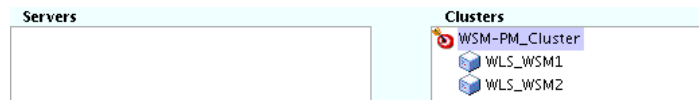
Tips:

For more information about the options on this screen, see "Clusters" in *Creating WebLogic Domains Using the Configuration Wizard*.

Task 15 Assigning Managed Servers to the Cluster

Use the Assign Servers to Clusters screen to assign WLS_WSM1 and WLS_WSM2 to the new cluster WSM-PM_Cluster:

1. In the **Clusters** pane, select the cluster to which you want to assign the servers; in this case, WSM-PM_Cluster.
2. In the **Servers** pane, assign WLS_WSM1 to WSM-PM_Cluster by doing one of the following:
 - Click once on WLS_WSM1 to select it, then click on the right arrow to move it beneath the selected cluster (WSM-PM_Cluster) in the Clusters pane.
 - OR
 - Double-click on WLS_WSM1 to move it beneath the selected cluster (WSM-PM_Cluster) in the clusters pane.
3. Repeat these steps to assign the WLS_WSM2 Managed Server to the WSM-PM_Cluster.

**Tip:**

More information about the options on this screen can be found in Assign Servers to Clusters in *Creating WebLogic Domains Using the Configuration Wizard*.

Task 16 Configuring Coherence Clusters

Use the Coherence Clusters screen to configure the Coherence cluster that is automatically added to the domain.

In the **Cluster Listen Port**, enter 9991.

Note:

For Coherence licensing information, refer to "Oracle Coherence" in *Oracle Fusion Middleware Licensing Information*.

Task 17 Creating Machines

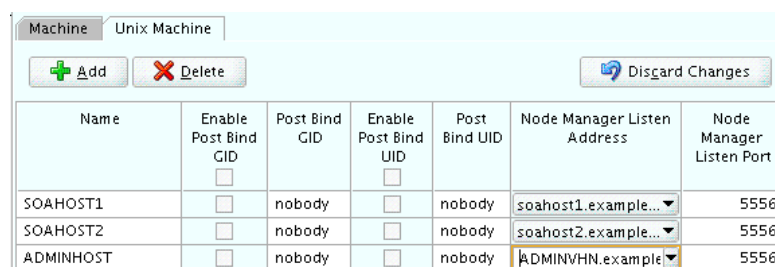
Use the Machines screen to create three new machines in the domain. A machine is required in order for the Node Manager to be able to start and stop the servers.

1. Select the **Unix Machine** tab.

2. Click the **Add** button to create three new Unix machines.
Use the values in [Table 10-2](#) to define the Name and Node Manager Listen Address of each machine. [Figure 10-4](#) shows a portion of the Machines screen, with example values for each machine.
3. Verify the port in the Node Manager Listen Port field.
The port number 5556, shown in this example, may be referenced by other examples in the documentation. Replace this port number with your own port number as needed.

Name	Node Manager Listen Address	Node Manager Listen Port
SOAHOST1	The value of the SOAHOST1 host name variable. For example, SOAHOST1.example.com.	5556
SOAHOST2	The value of the SOAHOST2 host name variable. For example, SOAHOST2.example.com.	5556
ADMINHOST	Enter the value of the ADMINVHN variable.	5556

Figure 10-4 Example Values for the Configuration Wizard Unix Machines Screen



Tip:
More information about the options on this screen can be found in Machines in *Creating WebLogic Domains Using the Configuration Wizard*.

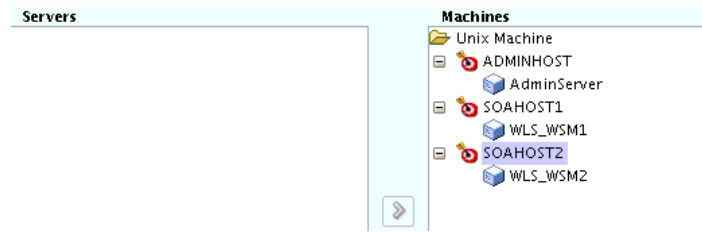
Task 18 Assigning Servers to Machines

Use the Assign Servers to Machines screen to assign the Administration Server and the two Managed Servers to the appropriate machine.

The Assign Servers to Machines screen is similar to the Assign Managed Servers to Clusters screen. Select the target machine in the Machines column, select the Managed Server in the left column, and click the right arrow to assign the server to the appropriate machine.

Assign the servers as follows:

- Assign the AdminServer to the ADMINHOST machine.
- Assign the WLS-WSM1 Managed Server to the SOAHOST1 machine.
- Assign the WLS-WSM2 Managed Server to the SOAHOST2 machine.

**Tip:**

More information about the options on this screen can be found in Assign Servers to Machines in *Creating WebLogic Domains Using the Configuration Wizard*.

Task 19 Configuring the JMS File Store

When you configure a domain using the Oracle WSM Policy Manager configuration template, you should select the proper location of the Metadata Services (MDS) JMS File Store, especially when you are configuring an enterprise deployment.

Enter the following location in the Directory column of the JMS File Store screen:

```
ORACLE_RUNTIME/domain_name/WSM-PM_Cluster
```

Replace *ORACLE_RUNTIME* with the actual value of the variable, as defined in [File System and Directory Variables Used in This Guide](#).

Replace *domain_name* with the name of the domain you are creating.

Task 20 Reviewing Your Configuration Specifications and Configuring the Domain

The Configuration Summary screen contains the detailed configuration information for the domain you are about to create. Review the details of each item on the screen and verify that the information is correct.

You can go back to any previous screen if you need to make any changes, either by using the **Back** button or by selecting the screen in the navigation pane.

Domain creation will not begin until you click **Create**.

Tip:

More information about the options on this screen can be found in Configuration Summary in *Creating WebLogic Domains Using the Configuration Wizard*.

Task 21 Writing Down Your Domain Home and Administration Server URL

The Configuration Success screen will show the following items about the domain you just configured:

- Domain Location
- Administration Server URL

You must make a note of both items as you will need them later; the domain location is needed to access the scripts used to start the Administration Server.

Click **Finish** to dismiss the configuration wizard.

10.6 Configuring a Per Host Node Manager for an Enterprise Deployment

For specific enterprise deployments, Oracle recommends that you configure a per-host Node Manager, as opposed to the default per-domain Node Manager.

For more information about the advantages of a per host Node Manager, see [About the Node Manager Configuration in a Typical Enterprise Deployment](#)

See Also:

[Creating a Per Host Node Manager Configuration](#)

The step in configuring a per-host Node Manager is to create a configuration directory and two new node manager configuration files. You must also edit the default `startNodeManager.sh` file.

[Creating the boot.properties File](#)

You must create a `boot.properties` if you want start the Node Manager without being prompted for the Node Manager credentials. This step is required in an enterprise deployment. The credentials you enter in this file are encrypted when you start the Administration Server.

[Starting the Node Manager on SOAHOST1](#)

After you manually set up the Node Manager to use a per-host Node Manager configuration, you can start the Node Manager on SOAHOST1, using the `startNodeManager.sh` script.

[Configuring the Node Manager Credentials and Type](#)

By default, a per-host Node Manager configuration does not use Secure Socket Layer (SSL) for for Node Manager-to-server communications. As a result, you must configure each machine in the domain to use a communication type of "plain," rather than SSL. In addition, you should set the Node Manager credentials so you can connect to the Administration Server and Managed Servers in the domain.

10.6.1 Creating a Per Host Node Manager Configuration

The step in configuring a per-host Node Manager is to create a configuration directory and two new node manager configuration files. You must also edit the default `startNodeManager.sh` file.

To create a per-host Node Manager configuration, perform the following tasks, first on SOAHOST1, and then on SOAHOST2:

1. Log in to SOAHOST1 and create a directory for the Node Manager configuration files :

For example:

```
mkdir -p /u02/oracle/config/nodemanager
```

Note that this directory should be on a local disk, because it is specific to the host. This directory location is known as the Node Manager home, and it is identified by the `NM_HOME` directory variable in examples in this guide.

2. Change directory to the Node Manager home directory:

```
cd NM_HOME
```


3. Create a new text file called `nodemanager.properties` and add the values shown in [Example 10-1](#) to this new file.

For more information about the properties you can add to the `nodemanager.properties` file, see Node Manager Properties in *Administering Node Manager for Oracle WebLogic Server*.

As part of this configuration in the `nodemanager.properties` file, you will enable Crash Recovery for the Node Manager. For more information, see “Node Manager and System Crash Recovery” in *Administering Node Manager for Oracle WebLogic Server*.

4. Locate the `startNodeManager.sh` file in the following directory:

```
WL_HOME/server/bin
```

5. Copy the `startNodeManager.sh` file to the Node Manager home directory.
6. Edit the new `startNodeManager.sh` file and add the `NODEMGR_HOME` property as follows:

```
NODEMGR_HOME="NM_HOME"
```

In this example, replace `NM_HOME` with the actual path to the Node Manager home.

7. Create another new file in the Node Manager home directory, called `nodemanager.domains`.

The `nodemanager.domains` file provides additional security by restricting Node Manager client access to the domains listed in this file.

8. Perform steps 1 through 7 on SOAHOST2.
9. Add the following entries to the new `nodemanager.domains` files:

On SOAHOST1, add values for both the Administration Server domain home and the Managed Servers domain home:

```
soaedg_domain=MSERVER_HOME;ASERVER_HOME
```

On SOAHOST2, add the value for the Managed Servers domain home only:

```
soaedg_domain=MSERVER_HOME
```

In these examples, replace `ASERVER_HOME` and `MSERVER_HOME` with the values of the respective variables, as described in [File System and Directory Variables Used in This Guide](#).

Example 10-1 Contents of the `nodemanager.properties` File

```
DomainsFile=/u02/oracle/config/nodemanager/nodemanager.domains
LogLimit=0
PropertiesVersion=12.2.1
AuthenticationEnabled=true
NodeManagerHome=/u02/oracle/config/nodemanager
#Include the specific JDK home
JavaHome=/u01/oracle/products/jdk
LogLevel=INFO
DomainsFileEnabled=true
StartScriptName=startWebLogic.sh
```

```
#Leave blank for listening on ANY
ListenAddress=
NativeVersionEnabled=true
ListenPort=5556
LogToStderr=true
SecureListener=false
LogCount=1
StopScriptEnabled=false
QuitEnabled=false
LogAppend=true
StateCheckInterval=500
CrashRecoveryEnabled=true
StartScriptEnabled=true
LogFile=/u02/oracle/config/nodemanager/nodemanager.log
LogFormatter=weblogic.nodemanager.server.LogFormatter
ListenBacklog=50
```

10.6.2 Creating the boot.properties File

You must create a `boot.properties` if you want start the Node Manager without being prompted for the Node Manager credentials. This step is required in an enterprise deployment. The credentials you enter in this file are encrypted when you start the Administration Server.

To create a `boot.properties` file for the Administration Server:

1. Create the following directory structure:

```
mkdir -p ASERVER_HOME/servers/AdminServer/security
```

2. In a text editor, create a file called `boot.properties` in the `security` directory created in the previous step, and enter the Administration Server credentials that you defined when you ran the Configuration Wizard to create the domain:

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

Note:

When you start the Administration Server, the `username` and `password` entries in the file get encrypted.

For security reasons, minimize the amount of time the entries in the file are left unencrypted; after you edit the file, you should start the server as soon as possible so that the entries get encrypted.

3. Save the file and close the editor.

10.6.3 Starting the Node Manager on SOAHOST1

After you manually set up the Node Manager to use a per-host Node Manager configuration, you can start the Node Manager on `SOAHOST1`, using the `startNodeManager.sh` script.

To start the Node Manager on `SOAHOST1`:

1. Change directory to the Node Manager home directory:

```
cd NM_HOME
```

2. Run the following command to start the Node Manager and send the output of the command to an output file, rather than to the current terminal shell:

```
nohup ./startNodeManager.sh > ./nodemanager.out 2>&1 &
```

3. Monitor the the nodemanager.out file; make sure the NodeManager starts successfully. The output should eventually contain a string similar to the following:

```
<INFO><Plain socket listener started on port 5556>
```

10.6.4 Configuring the Node Manager Credentials and Type

By default, a per-host Node Manager configuration does not use Secure Socket Layer (SSL) for for Node Manager-to-server communications. As a result, you must configure each machine in the domain to use a communication type of “plain,” rather than SSL. In addition, you should set the Node Manager credentials so you can connect to the Administration Server and Managed Servers in the domain.

The following procedure temporarily starts the Administration Server with the default start script, so you can perform these tasks. After you perform these tasks, you can stop this temporary session and use the Node Manager to start the Administration Server.

1. Start the Administration Server, using the default start script:

- a. Change directory to the following directory:

```
cd ASERVER_HOME/bin
```

- b. Run the start script:

```
./startWebLogic.sh
```

Watch the output to the terminal, until you see the following:

```
<Server state changed to RUNNING>
```

2. Log in to the WebLogic Server Administration Console, using the WebLogic administrator user and password.
3. Configure the Node Manager type:
 - a. Click **Lock & Edit**.
 - b. In the **Domain Structure** navigation tree, expand **Domain**, and then **Environment**.
 - c. Click **Machines**.
 - d. Click the link for the **ADMINHOST** machine.
 - e. Click the **Node Manager** tab.
 - f. Change the **Type** property from SSL to **Plain**.
 - g. Click **Save**.
 - h. Repeat this task for each machine in the domain.
 - i. Click **Activate Changes**.

4. Set the Node Manager credentials:
 - a. Click **Lock & Edit**.
 - b. In the **Domain Structure** navigation pane, click the name of the domain.
 - c. Select the **Security** tab.
The **Security > General** tab should be selected.
 - d. Scroll down and expand the **Advanced** security options.
 - e. Make a note of the user name in the **NodeManager Username** field.
Optionally, you can edit the value to create a new Node Manager user name.
 - f. Enter a new password in the **NodeManager Password** and **Confirm NodeManager Password** fields
 - g. Click **Save** and then **Activate Changes**.
5. In a new terminal window, use the following steps to refresh the SystemSerialized.dat file. Without this step, you won't be able to connect to the Node Manager and use it to start the servers in domain:
 - a. Change directory to the


```
cd ORACLE_COMMON_HOME/common/bin
```
 - b. Start the WebLogic Server Scripting Tool (WLST):


```
./wlst.sh
```
 - c. Connect to the Administration Server, using the following WLST command:


```
connect('admin_user','admin_password','admin_url')
```

 For example:


```
connect('weblogic','mypassword','t3://ADMINVHN:7001')
```
 - d. Use the nmEnroll command to enables the Node Manager to manage servers in a specified WebLogic domain.


```
nmEnroll('ASERVER_HOME')
```

 For example:


```
nmEnroll('/u01/oracle/config/domains/soaedg_domain')
```
6. Optionally, if you want to customize any startup properties for the Administration Server, you can use the following WLST command to create a startup.properties file for the Administration Server:


```
nmGenBootStartupProps('AdminServer')
```

 The startup.properties file is created in the following directory:


```
ASERVER_HOME/servers/AdminServer/data/nodemanager/
```
7. Return to the terminal window where you started the Administration Server with the start script.

8. Press **Ctrl/C** to stop the Administration Server process.

Wait for the Administration Server process to end and for the terminal command prompt to appear.

10.7 Configuring the Domain Directories and Starting the Servers on SOAHOST1

After the domain is created and the node manager is configured, you can then configure the additional domain directories and start the Administration Server and the Managed Servers on SOAHOST1.

See Also:

[Starting the Administration Server Using the Node Manager](#)

After you have configured the domain and configured the Node Manager, you can start the Administration Server, using the Node Manager. In an enterprise Deployment, the Node Manager is used to start and stop the Administration Server and all the Managed Servers in the domain.

[Validating the Administration Server](#)

Before proceeding with the configuration steps, validate that the Administration Server has started successfully by making sure you have access to the Oracle WebLogic Server Administration Console and Oracle Enterprise Manager Fusion Middleware Control, which both are installed and configured on the Administration Servers.

[Disabling the Derby Database](#)

[Creating a Separate Domain Directory for Managed Servers on SOAHOST1](#)

When you initially create the domain for enterprise deployment, the domain directory resides on a shared disk. This default domain directory will be used to run the Administration Server. You can now create a copy of the domain on the local storage for both SOAHOST1 and SOAHOST2. The domain directory on the local (or private) storage will be used to run the Managed Servers.

[Starting and Validating the WLS_WSM1 Managed Server on SOAHOST1](#)

After you have configured Node Manager and created the Managed Server domain directory, you can use Oracle Enterprise Manager Fusion Middleware Control to start the WLS_WSM1 Managed Server on SOAHOST1.

10.7.1 Starting the Administration Server Using the Node Manager

After you have configured the domain and configured the Node Manager, you can start the Administration Server, using the Node Manager. In an enterprise Deployment, the Node Manager is used to start and stop the Administration Server and all the Managed Servers in the domain.

To start the Administration Server using the Node Manager:

1. Start the WebLogic Scripting Tool (WLST):

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

2. Connect to Node Manager using the Node Manager credentials you defined in the WebLogic Server Administration Console:

```
wls:/offline>nmConnect('nodemanager_username','nodemanager_password',  
                      'ADMINVHN','5556','domain_name',  
                      'ASERVER_HOME','PLAIN')
```

Note:

This user name and password are used only to authenticate connections between Node Manager and clients. They are independent of the server administrator ID and password and are stored in the `nm_password.properties` file located in the following directory:

```
ASERVER_HOME/config/nodemanager
```

3. Start the Administration Server:

```
nmStart('AdminServer')
```

Note:

When you start the Administration Server, it attempts to connect to Oracle Web Services Manager for WebServices policies. It is expected that, since the WSM-PM Managed Servers are not yet started, the following message will appear in the Administration Server log:

```
<Warning><oracle.wsm.resources.policymanager>  
<WSM-02141><Unable to connect to the policy access service due to Oracle WSM  
policy manager host server being down.>
```

4. Exit WLST:

```
exit()
```

10.7.2 Validating the Administration Server

Before proceeding with the configuration steps, validate that the Administration Server has started successfully by making sure you have access to the Oracle WebLogic Server Administration Console and Oracle Enterprise Manager Fusion Middleware Control, which both are installed and configured on the Administration Servers.

To navigate to Fusion Middleware Control, enter the following URL, and log in with the Oracle WebLogic Server administrator credentials:

```
ADMINVHN:7001/em
```

To navigate to the Oracle WebLogic Server Administration Console, enter the following URL, and log in with the same administration credentials:

```
ADMINVHN:7001/console
```

10.7.3 Disabling the Derby Database

Before you create the Managed Server directory and start the Managed Servers, disable the embedded Derby database, which is a file-based database, packaged with

Oracle WebLogic Server. The Derby database is used primarily for development environments. As a result, you must disable it when you are configuring a production-ready enterprise deployment environment; otherwise, the Derby database process will start automatically when you start the Managed Servers.

To disable the Derby database:

1. Navigate to the following directory in the Oracle home.

```
WL_HOME/common/derby/lib
```

2. Rename the Derby library jar file:

```
mv derby.jar disable_derby.jar
```

10.7.4 Creating a Separate Domain Directory for Managed Servers on SOAHOST1

When you initially create the domain for enterprise deployment, the domain directory resides on a shared disk. This default domain directory will be used to run the Administration Server. You can now create a copy of the domain on the local storage for both SOAHOST1 and SOAHOST2. The domain directory on the local (or private) storage will be used to run the Managed Servers.

Placing the MSERVER_HOME on local storage is recommended to eliminate the potential contention and overhead cause by servers writing logs to shared storage. It is also faster to load classes and jars need from the domain directory, so any tmp or cache data that Managed Servers use from the domain directory is processed quicker.

As described in [Preparing the File System for an Enterprise Deployment](#), the path to the Administration Server domain home is represented by the ASERVER_HOME variable, and the path to the Managed Server domain home is represented by the MSERVER_HOME variable.

To create the Managed Server domain directory:

1. Log in to SOAHOST1 and run the pack command to create a template as follows:

```
cd ORACLE_COMMON_HOME/common/bin

./pack.sh -managed=true
         -domain=ASERVER_HOME
         -template=complete_path/soadomaintemplate.jar
         -template_name=soa_domain_template
```

In this example:

- Replace *ASERVER_HOME* with the actual path to the domain directory you created on the shared storage device.
 - Replace *complete_path* with the complete path to the location where you want to create the domain template jar file. You will need to reference this location when you copy or unpack the domain template jar file.
 - *soadomaintemplate.jar* is a sample name for the jar file you are creating, which will contain the domain configuration files.
 - *soa_domain_template* is the name assigned to the domain template file.
2. Make a note of the location of the *soadomaintemplate.jar* file you just created with the pack command.

You must specify a full path for the template jar file as part of the `-template` argument to the pack command:

```
ORACLE_COMMON_HOME/common/bin/
```

Tip:

For more information about the pack and unpack commands, see "Overview of the Pack and Unpack Commands" in *Creating Templates and Domains Using the Pack and Unpack Commands*.

3. If you haven't already, create the recommended directory structure for the Managed Server domain on the SOAHOST1 local storage device.

Use the examples in [File System and Directory Variables Used in This Guide](#) as a guide.

4. Run the `unpack` command to unpack the template in the domain directory onto the local storage, as follows:

```
cd ORACLE_COMMON_HOME/common/bin

./unpack.sh -domain=MSERVER_HOME \
            -overwrite_domain=true \
            -template=complete_path/soadomaintemplate.jar \
            -log_priority=DEBUG \
            -log=/tmp/unpack.log \
            -app_dir=APPLICATION_HOME \
```

Note:

The `-overwrite_domain` option in the `unpack` command allows unpacking a managed server template into an existing domain and existing applications directories. For any file that is overwritten, a backup copy of the original is created. If any modifications had been applied to the start scripts and ear files in the managed server domain directory, they must be restored after this `unpack` operation.

Additionally, to customize server startup parameters that apply to all servers in a domain, you can create a file called `setUserOverrides.sh` and configure it to, for example, add custom libraries to the WebLogic Server classpath, specify additional java command line options for running the servers, or specify additional environment variables. Any customizations you add to this file are preserved during domain upgrade operations, and are carried over to remote servers when using the pack and unpack commands.

In this example:

- Replace `MSERVER_HOME` with the complete path to the domain home to be created on the local storage disk. This is the location where the copy of the domain will be unpacked.
- Replace `complete_path` with the complete path to the location where you created or copied the template jar file.
- `soadomaintemplate.jar` is the name of the template jar file you created when you ran the pack command to pack up the domain on the shared storage device.

Tip:

For more information about the pack and unpack commands, see "Overview of the Pack and Unpack Commands" in *Creating Templates and Domains Using the Pack and Unpack Commands*.

5. Change directory to the newly created Managed Server directory and verify that the domain configuration files were copied to the correct location on the SOAHOST1 local storage device.

10.7.5 Starting and Validating the WLS_WSM1 Managed Server on SOAHOST1

After you have configured Node Manager and created the Managed Server domain directory, you can use Oracle Enterprise Manager Fusion Middleware Control to start the WLS_WSM1 Managed Server on SOAHOST1.

1. Enter the following URL into a browser to display the Fusion Middleware Control login screen:

`http://ADMINVHN:7001/em`

In this example:

- Replace *ADMINVHN* with the host name assigned to the ADMINVHN Virtual IP address in [Identifying and Obtaining Software Downloads for an Enterprise Deployment](#).
- Port 7001 is the typical port used for the Administration Server console and Fusion Middleware Control. However, you should use the actual URL that was displayed at the end of the Configuration Wizard session when you created the domain.

Tip:

For more information about managing Oracle Fusion Middleware using Oracle Enterprise Manager Fusion Middleware, see "Getting Started Using Oracle Enterprise Manager Fusion Middleware Control" in *Administering Oracle Fusion Middleware*.

2. Log in to Fusion Middleware Control using the Administration Server credentials.
3. Select the **Servers** pane to view the Managed Servers in the domain.

The screenshot shows the Oracle Enterprise Manager Fusion Middleware Control interface. On the left, there are three summary cards: 'Servers' (2 Down, 1 Up), 'Clusters' (1 Down), and 'Deployments' (1 Active, 2 New). The main area displays the 'Administration Server' details (Name: AdminServer, Host: host42.example.com, Listen Port: 7001) and a 'Servers' table. The table has columns for Name, Status, Cluster, Machine, and State. The WLS_WSM1 server is selected and highlighted in blue, showing a red status icon and 'Shutdown' state. The WLS_WSM2 server also shows a red status icon and 'Shutdown' state. The Administration Server (AdminServer(admin)) shows a green status icon and 'Running' state.

Name	Status	Cluster	Machine	State
AdminServer(admin)	↑		ADMINHOST	Running
WLS_WSM1	↓	WSM-PM_Cluster	SOAHOST1	Shutdown
WLS_WSM2	↓	WSM-PM_Cluster	SOAHOST2	Shutdown

4. Select only the **WLS_WSM1** Managed Server, and then click **Control** > **Start** on the tool bar.
5. To verify that the Managed Server is working correctly, open your browser and enter the following URL:

`SOAHOST1:7010/wsm-pm/`

Enter the domain admin user name and password when prompted.

10.8 Propagating the Domain and Starting the Servers on SOAHOST2

After you start and validate the Administration Server and WLS_WSM1 Managed Server on SOAHOST1, you can then perform the following tasks on SOAHOST2.

See Also:

[Unpacking the Domain Configuration on SOAHOST2](#)

[Unpacking the Domain on SOAHOST2](#)

[Starting the Node Manager on SOAHOST2](#)

[Starting and Validating the WLS_WSM2 Managed Server on SOAHOST2](#)

10.8.1 Unpacking the Domain Configuration on SOAHOST2

Now that you have the Administration Server and the first WLS_WSM1 Managed Server running on SOAHOST1, you can configure the domain on SOAHOST2.

1. Log in to SOAHOST2.
2. If you haven't already, create the recommended directory structure for the Managed Server domain on the SOAHOST2 storage device.

Use the examples in [File System and Directory Variables Used in This Guide](#) as a guide.
3. Make sure the `soadomaintemplate.jar` accessible to SOAHOST2.

For example, if you are using a separate shared storage volume or partition for SOAHOST2, then copy the template to the volume or partition mounted to SOAHOST2.

4. Run the `unpack` command to unpack the template in the domain directory onto the local storage, as follows:

```
cd ORACLE_COMMON_HOME/common/bin

./unpack.sh -domain=MSERVER_HOME
            -overwrite_domain=true
            -template=complete_path/soadomaintemplate.jar
            -log_priority=DEBUG
            -log=/tmp/unpack.log
            -app_dir=APPLICATION_HOME
```

In this example:

- Replace *MSERVER_HOME* with the complete path to the domain home to be created on the local storage disk. This is the location where the copy of the domain will be unpacked.
- Replace *complete_path/soadomaintemplate.jar* with the complete path and file name of the domain template jar file that you created when you ran the pack command to pack up the domain on the shared storage device.
- Replace *APPLICATION_HOME* with the complete path to the Application directory for the domain on shared storage. For more information, see [File System and Directory Variables Used in This Guide](#).

Tip:

For more information about the pack and unpack commands, see "Overview of the Pack and Unpack Commands" in *Creating Templates and Domains Using the Pack and Unpack Commands*.

5. Change directory to the newly created *MSERVER_HOME* directory and verify that the domain configuration files were copied to the correct location on the SOAHOST2 local storage device.

10.8.2 Unpacking the Domain on SOAHOST2

This procedure assumes you have copied the file that you created earlier in a location that is accessible from both SOAHOST1 and SOAHOST2; such as the *ASERVER_HOME* directory, which is located on the shared storage filer:

1. Log in to SOAHOST2.
2. If you haven't already, create the recommended directory structure for the Managed Server domain on the SOAHOST2 storage device.

Use the examples in [File System and Directory Variables Used in This Guide](#) as a guide.

3. Make sure the *soadomaintemplate.jar* accessible to SOAHOST2.

For example, if you are using a separate shared storage volume or partition for SOAHOST2, then copy the template to the volume or partition mounted to SOAHOST2.

4. Run the `unpack` command to unpack the template in the domain directory onto the local storage, as follows:

```
cd ORACLE_COMMON_HOME/common/bin

./unpack.sh -domain=MSERVER_HOME
            -overwrite_domain=true
            -template=complete_path/soadomaintemplate.jar
            -log_priority=DEBUG
            -log=/tmp/unpack.log
            -app_dir=APPLICATION_HOME
```

In this example:

- Replace `MSERVER_HOME` with the complete path to the domain home to be created on the local storage disk. This is the location where the copy of the domain will be unpacked.
- Replace `complete_path/soadomaintemplate.jar` with the complete path and file name of the domain template jar file that you created when you ran the pack command to pack up the domain on the shared storage device.
- Replace `APPLICATION_HOME` with the complete path to the Application directory for the domain on shared storage. For more information, see [File System and Directory Variables Used in This Guide](#).

Tip:

For more information about the pack and unpack commands, see "Overview of the Pack and Unpack Commands" in *Creating Templates and Domains Using the Pack and Unpack Commands*.

5. Change directory to the newly created `MSERVER_HOME` directory and verify that the domain configuration files were copied to the correct location on the `SOAHOST2` local storage device.

10.8.3 Starting the Node Manager on SOAHOST2

After you manually set up the Node Manager to use a per host Node Manager configuration, you can start the Node Manager using the following commands on `SOAHOST2`:

1. Change directory to the Node Manager home directory:

```
cd NM_HOME
```

2. Run the following command to start the Node Manager and send the output of the command to an output file, rather than to the current terminal shell:

```
nohup ./startNodeManager.sh > nodemanager.out 2>&1 &
```

10.8.4 Starting and Validating the WLS_WSM2 Managed Server on SOAHOST2

Use the procedure in [Starting and Validating the WLS_WSM1 Managed Server on SOAHOST1](#) to start and validate the `WLS_WSM2` Managed Server on `SOAHOST2`.

10.9 Modifying the Upload and Stage Directories to an Absolute Path

After creating the domain and unpacking it to the Managed Server domain directory, verify and update the `upload` and `stage` directories for the Managed Servers.

This step is necessary to avoid potential issues when performing remote deployments and for deployments that require the stage mode.

To update these directory paths for all the Managed Servers in the Managed Server domain home directory:

1. Log in to the Oracle WebLogic Server Administration Console.
2. In the left navigation tree, expand **Domain**, and then **Environment**.
3. Click **Lock & Edit**.

4. Click **Servers**.
5. For each Managed Server in the Managed Server domain home directory:
 - a. Click the name of the Managed Server.
 - b. Click the **Configuration** tab, and then click the **Deployment** tab.
 - c. Verify that the **Staging Directory Name** is set to the following:
`MSERVER_HOME/servers/server_name/stage`

 Replace `MSERVER_HOME` with the directory path for the `MSERVER_HOME` directory; replace `server_name` with the name of the Server you are editing.

 Note that if you plan to configure the staging directory for the Administration Server, then you must set the property is set to:
`ASERVER_HOME/servers/AdminServer/stage`
 - d. Update the **Upload Directory Name** to the following value:
`ASERVER_HOME/servers/AdminServer/upload`

 Replace `ASERVER_HOME` with the directory path for the `ASERVER_HOME` directory.
 - e. Click **Save**.
 - f. Return to the Summary of Servers screen.
6. When you have modified these values for each Managed Server, click **Activate Changes**.
7. Restart all Managed Servers.

10.10 Creating a New LDAP Authenticator and Provisioning Enterprise Deployment Users and Group

When you configure an Oracle Fusion Middleware domain, the domain is configured by default to use the WebLogic Server authentication provider (`DefaultAuthenticator`). However, for an enterprise deployment, Oracle recommends that you use a dedicated, centralized LDAP-compliant authentication provider.

The following topics describe how to use the Oracle WebLogic Server Administration Console to create a new authentication provider for the enterprise deployment domain. This procedure assumes you have already installed and configured a supported LDAP directory, such as Oracle Unified Directory or Oracle Internet Directory.

See Also:

[About the Supported Authentication Providers](#)

[About the Enterprise Deployment Users and Groups](#)

[Prerequisites for Creating a New Authentication Provider and Provisioning Users and Groups](#)

[Provisioning a Domain Connector User in the LDAP Directory](#)

[Creating the New Authentication Provider](#)

[Provisioning an Enterprise Deployment Administration User and Group](#)

[Adding the New Administration User to the Administration Group](#)

[Updating the boot.properties File and Restarting the System](#)

10.10.1 About the Supported Authentication Providers

Oracle Fusion Middleware supports a variety of LDAP authentication providers. For more information, see "Identity Store Types and WebLogic Authenticators" in *Securing Applications with Oracle Platform Security Services*.

The instructions in this guide assume you will be using one of the following providers:

- Oracle Unified Directory
- Oracle Internet Directory
- Oracle Virtual Directory

Note:

By default, the instructions here describe how to configure the identity service instance to support querying against a single LDAP identity store.

However, you can configure the service to support a virtualized identity store, which queries multiple LDAP identity stores, using LibOVD.

For more information about configuring a Multi-LDAP lookup, refer to "Configuring the Identity Store Service" in *Securing Applications with Oracle Platform Security Services*.

10.10.2 About the Enterprise Deployment Users and Groups

The following topics provide important information on the purpose and characteristics of the enterprise deployment administration users and groups.

See Also:

[About Using Unique Administration Users for Each Domain](#)

[About the Domain Connector User](#)

[About Adding Users to the Central LDAP Directory](#)

[About Product-Specific Roles and Groups for Oracle SOA Suite](#)

[Example Users and Roles Used in This Guide](#)

10.10.2.1 About Using Unique Administration Users for Each Domain

When you use a central LDAP user store, you can provision users and groups for use with multiple Oracle WebLogic Server domains. As a result, there is a possibility that one WebLogic administration user can have access to all the domains within an enterprise.

Such an approach is not recommended. Instead, it is a best practice to assign a unique distinguished name (DN) within the directory tree for the users and groups you provision for the administration of your Oracle Fusion Middleware domains.

For example, if you plan to install and configure an Oracle SOA Suite enterprise deployment domain, then create a user called **weblogic_soa** and an administration group called **SOA Administrators**.

10.10.2.2 About the Domain Connector User

Oracle recommends that you create a separate domain connector user (for example, `soaLDAP`) in your LDAP directory. This user allows the domain to connect to the LDAP directory for the purposes of user authentication. It is recommended that this user be a non-administrative user.

In a typical Oracle Identity and Access Management deployment, you create this user in the `systemids` container. This container is used for system users that are not normally visible to users. Placing the user into the `systemids` container ensures that customers who have Oracle Identity Manager do not reconcile this user.

10.10.2.3 About Adding Users to the Central LDAP Directory

After you configure a central LDAP directory to be the authenticator for the enterprise domain, then you should add all new users to the new authenticator and not to the default WebLogic Server authenticator.

To add new users to the central LDAP directory, you cannot use the WebLogic Administration Console. Instead, you must use the appropriate LDAP modification tools, such as `ldapbrowser` or `JXplorer`.

When you are using multiple authenticators (a requirement for an enterprise deployment), login and authentication will work, but role retrieval will not. The role is retrieved from the first authenticator only. If you want to retrieve roles using any other authenticator, then you must enable virtualization for the domain.

Enabling virtualization involves the following steps:

1. Locate and open the following configuration file with a text editor:

```
DOMAIN_HOME/config/fmwconfig/jps-config.xml
```

2. Add or update the following property, as follows:

```
<property name="virtualize" value="true"/>
```

For more information about the `virtualize` property, see “OPSS System and Configuration Properties” in *Securing Applications with Oracle Platform Security Services*.

10.10.2.4 About Product-Specific Roles and Groups for Oracle SOA Suite

Each Oracle Fusion Middleware product implements its own predefined roles and groups for administration and monitoring.

As a result, as you extend the domain to add additional products, you can add these product-specific roles to the `SOA Administrators` group. After they are added to the `SOA Administrators` group, each product administrator user can administer the domain with the same set of privileges for performing administration tasks.

Instructions for adding additional roles to the `SOA Administrators` group are provided in [Common Configuration and Management Tasks for an Enterprise Deployment](#).

10.10.2.5 Example Users and Roles Used in This Guide

In this guide, the examples assume that you provision the following administration user and group with the DNs shown below:

- Admin User DN:

```
cn=weblogic_soa,cn=users,dc=example,dc=com
```

- Admin Group DN:

```
cn=SOA Administrators,cn=groups,dc=example,dc=com
```

- Product-specific LDAP Connector User:

```
cn=soaLDAP,cn=systemids,dc=example,dc=com
```

This is the user you will use to connect WebLogic Managed Servers to the LDAP authentication provider. This user must have permissions to read and write to the Directory Trees:

```
cn=users,dc=example,dc=com
cn=groups,dc=example,dc=com
```

Note:

When using Oracle Unified Directory, this user will need to be granted membership in the following groups to provide read and write access:

```
cn=orclFAUserReadPrivilegeGroup,cn=groups,dc=example,dc=com
cn=orclFAUserWritePrivilegeGroup,cn=groups,dc=example,dc=com
cn=orclFAGroupReadPrivilegeGroup,cn=groups,dc=example,dc=com
cn=orclFAGroupWritePrivilegeGroup,cn=groups,dc=example,dc=com
```

10.10.3 Prerequisites for Creating a New Authentication Provider and Provisioning Users and Groups

Before you create a new LDAP authentication provider, back up the relevant configuration files:

```
ASERVER_HOME/config/config.xml
ASERVER_HOME/config/fmwconfig/jps-config.xml
ASERVER_HOME/config/fmwconfig/system-jazn-data.xml
```

In addition, back up the `boot.properties` file for the Administration Server in the following directory:

```
DOMAIN_HOME/servers/AdminServer/security
```

10.10.4 Provisioning a Domain Connector User in the LDAP Directory

This example shows how to create a user called `soaLDAP` in the central LDAP directory.

To provision the user in the LDAP provider:

1. Create an ldif file named `domain_user.ldif` with the contents shown below and then save the file:


```
dn: cn=soaLDAP,cn=systemids,dc=example,dc=com
changetype: add
orclsamaccountname: soaLDAP
userpassword: password
objectclass: top
objectclass: person
objectclass: organizationalPerson
objectclass: inetorgperson
objectclass: orcluser
objectclass: orcluserV2
mail: soaLDAP@example.com
givenname: soaLDAP
sn: soaLDAP
cn: soaLDAP
uid: soaLDAP
```

Note:

If you are using Oracle Unified Directory, then add the following four group memberships to the end of the LDIF file to grant the appropriate read/write privileges:

```
dn:
cn=orclFAUserReadPrivilegeGroup,cn=groups,dc=us,dc=oracle,dc=com
changetype: modify
add: uniquemember
uniquemember: cn=soaLDAP,cn=systemids,dc=us,dc=oracle,dc=com

dn: cn=orclFAGroupReadPrivilegeGroup,cn=groups,dc=us,dc=oracle,dc=com
changetype: modify
add: uniquemember
uniquemember: cn=soaLDAP,cn=systemids,dc=us,dc=oracle,dc=com

dn: cn=orclFAUserWritePrivilegeGroup,cn=groups,dc=example,dc=com
changetype: modify
add: uniquemember
uniquemember: cn=soaLDAP,cn=systemids,dc=example,dc=com

dn: cn=orclFAGroupWritePrivilegeGroup,cn=groups,dc=example,dc=com
changetype: modify
add: uniquemember
uniquemember: cn=soaLDAP,cn=systemids,dc=example,dc=com
```

2. Provision the user in the LDAP directory.

For example, for an Oracle Unified Directory LDAP provider:

```
OID_INSTANCE_HOME/bin/ldapmodify -a \
    -h oudhost.example.com
    -D "cn=oudadmin" \
    -w password \
    -p 1389 \
    -f domain_user.ldif
```

For Oracle Internet Directory:

```
OID_ORACLE_HOME/bin/ldapadd -h oidhost.example.com \
    -p 3060 \
    -D cn="orcladmin" \
```

```
-w password \  
-c \  
-v \  
-f domain_user.ldif
```

10.10.5 Creating the New Authentication Provider

To configure a new LDAP-based authentication provider:

1. Log in to the WebLogic Server Administration Console.
2. Click **Security Realms** in the left navigational bar.
3. Click the **myrealm** default realm entry.
4. Click the **Providers** tab.

Note that there is a `DefaultAuthenticator` provider configured for the realm. This is the default WebLogic Server authentication provider.

<input type="checkbox"/>	Name	Description	Version
<input type="checkbox"/>	Trust Service Identity Asserter	Trust Service Identity Assertion Provider	1.0
<input type="checkbox"/>	DefaultAuthenticator	WebLogic Authentication Provider	1.0
<input type="checkbox"/>	DefaultIdentityAsserter	WebLogic Identity Assertion provider	1.0

5. Click **Lock & Edit** in the Change Center.
6. Click the **New** button below the **Authentication Providers** table.
7. Enter a name for the provider.

Use one of the following names, based on the LDAP directory service you are planning to use as your credential store:

- `OUDAuthenticator` for Oracle Unified Directory
- `OIDAuthenticator` for Oracle Internet Directory
- `OVDAuthenticator` for Oracle Virtual Directory

8. Select the authenticator type from the **Type** drop-down list.

Select one of the following types, based on the LDAP directory service you are planning to use as your credential store:

- `OracleUnifiedDirectoryAuthenticator` for Oracle Unified Directory
- `OracleInternetDirectoryAuthenticator` for Oracle Internet Directory
- `OracleVirtualDirectoryAuthenticator` for Oracle Virtual Directory

9. Click **OK** to return to the Providers screen.
10. On the Providers screen, click the newly created authenticator in the table.
11. Select **SUFFICIENT** from the **Control Flag** drop-down menu.



Setting the control flag to **SUFFICIENT** indicates that if the authenticator can successfully authenticate a user, then the authenticator should accept that authentication and should not continue to invoke any additional authenticators.

If the authentication fails, it will fall through to the next authenticator in the chain. Make sure all subsequent authenticators also have their control flags set to **SUFFICIENT**; in particular, check the **DefaultAuthenticator** and make sure that its control flag is set to **SUFFICIENT**.

12. Click **Save** to save the control flag settings.
13. Click the **Provider Specific** tab and enter the details specific to your LDAP server, as shown in the following table.

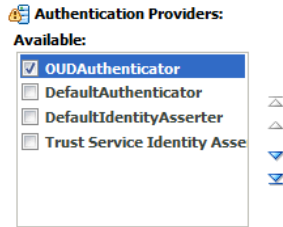
Note that only the required fields are discussed in this procedure. For information about all the fields on this page, consider the following resources:

- To display a description of each field, click **Help** on the **Provider Specific** tab.
- For more information on setting the **User Base DN**, **User From Name Filter**, and **User Attribute** fields, see "Configuring Users and Groups in the Oracle Internet Directory and Oracle Virtual Directory Authentication Providers" in *Administering Security for Oracle WebLogic Server*.

Parameter	Sample Value	Value Description
Host	For example: <code>oud.example.com</code>	The LDAP server's server ID.
Port	For example: <code>1689</code>	The LDAP server's port number.
Principal	For example: <code>cn=soaLDAP, cn=systemids, dc=example, dc=com</code>	The LDAP user DN used to connect to the LDAP server.
Credential	Enter LDAP password.	The password used to connect to the LDAP server.
SSL Enabled	Unchecked (clear)	Specifies whether SSL protocol is used when connecting to the LDAP server.
User Base DN	For example: <code>cn=users, dc=example, dc=com</code>	Specify the DN under which your users start.

Parameter	Sample Value	Value Description
All Users Filter	<code>(&(uid=*)(objectclass=person))</code>	<p>Instead of a default search criteria for All Users Filter, search all users based on the <code>uid</code> value.</p> <p>If the User Name Attribute for the user object class in the LDAP directory structure is a type other than <code>uid</code>, then change that type in the User From Name Filter field.</p> <p>For example, if the User Name Attribute type is <code>cn</code>, then this field should be set to:</p> <pre>(&(cn=*)(objectclass=person))</pre>
User From Name Filter	<p>For example:</p> <pre>(&(uid=%u)(objectclass=person))</pre>	<p>If the User Name Attribute for the user object class in the LDAP directory structure is a type other than <code>uid</code>, then change that type in the settings for the User From Name Filter.</p> <p>For example, if the User Name Attribute type is <code>cn</code>, then this field should be set to:</p> <pre>(&(cn=%u)(objectclass=person)).</pre>
User Name Attribute	For example: <code>uid</code>	The attribute of an LDAP user object that specifies the name of the user.
Group Base DN	<p>For example:</p> <pre>cn=groups,dc=example,dc=com</pre>	Specify the DN that points to your Groups node.
Use Retrieved User Name as Principal	Checked	Must be turned on.
GUID Attribute	<code>entryuuid</code>	<p>This value is prepopulated with <code>entryuuid</code> when <code>OracleUnifiedDirectoryAuthenticator</code> is used for OUD. Check this value if you are using Oracle Unified Directory as your authentication provider.</p>

14. Click **Save** to save the changes.
15. Return to the Providers page by clicking **Security Realms** in the right navigation pane, clicking the default realm name (**myrealm**), and then **Providers**.
16. Click **Reorder**, and then use the resulting page to make the Provider you just created first in the list of authentication providers.



17. Click **OK**.

18. In the Change Center, click **Activate Changes**.

19. Restart the Administration Server and all managed servers.

To stop the Managed Servers, log in to Fusion Middleware Control, select the Managed Servers in the Target Navigator and click **Shut Down** in the toolbar.

To stop and start the Administration Server using the Node Manager:

a. Start WLST:

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

b. Connect to Node Manager using the Node Manager credentials you defined in when you created the domain in the Configuration Wizard:

```
wls:/offline>nmConnect('nodemanager_username','nodemanager_password',
    'ADMINVHN','5556','domain_name',
    'ASERVER_HOME','PLAIN')
```

c. Stop the Administration Server:

```
nmKill('AdminServer')
```

d. Start the Administration Server:

```
nmStart('AdminServer')
```

e. Exit WLST:

```
exit()
```

To start the Managed Servers, log in to Fusion Middleware Control, select the Managed Servers, and click **Start Up** in the toolbar.

20. After the restart, review the contents of the following log file:

```
ASERVER_HOME/servers/AdminServer/logs/AdminServer.log
```

Verify that no LDAP connection errors occurred. For example, look for errors such as the following:

```
The LDAP authentication provider named "OUDAuthenticator" failed to make
connection to ldap server at ...
```

If you see such errors in the log file, then check the authorization provider connection details to verify they are correct and try saving and restarting the Administration Server again.

21. After you restart and verify that no LDAP connection errors are in the log file, try browsing the users and groups that exist in the LDAP provider:

In the Administration Console, navigate to the **Security Realms > myrealm > Users and Groups** page. You should be able to see all users and groups that exist in the LDAP provider structure.

10.10.6 Provisioning an Enterprise Deployment Administration User and Group

This example shows how to create a user called **weblogic_soa** and a group called **SOA Administrators**.

To provision the administration user and group in LDAP provider:

1. Create an ldif file named `admin_user.ldif` with the contents shown below and then save the file:

```
dn: cn=weblogic_soa,cn=users,dc=example,dc=com
changetype: add
orclsamaccountname: weblogic_soa
userpassword: password
objectclass: top
objectclass: person
objectclass: organizationalPerson
objectclass: inetorgperson
objectclass: orcluser
objectclass: orcluserV2
mail: weblogic_soa@example.com
givenname: weblogic_soa
sn: weblogic_soa
cn: weblogic_soa
uid: weblogic_soa
```

2. Provision the user in the LDAP directory.

For example, for an Oracle Unified Directory LDAP provider:

```
OID_INSTANCE_HOME/bin/ldapmodify -a \
    -h oudhost.example.com
    -D "cn=oudadmin" \
    -w password \
    -p 1389 \
    -f admin_user.ldif
```

For Oracle Internet Directory:

```
OID_ORACLE_HOME/bin/ldapadd -h oidhost.example.com \
    -p 3060 \
    -D cn="orcladmin" \
    -w password \
    -c \
    -v \
    -f admin_user.ldif
```

3. Create an ldif file named `admin_group.ldif` with the contents shown below and then save the file:

```
dn: cn=SOA Administrators,cn=Groups,dc=example,dc=com
displayname: SOA Administrators
objectclass: top
objectclass: SOA Administrators
objectclass: orclGroup
uniquemember: cn=weblogic_soa,cn=users,dc=example,dc=com
```

```
cn:SOA Administrators
description: Administrators Group for the Oracle SOA Suite Domain
```

4. Provision the group in the LDAP Directory.

For Oracle Unified Directory:

```
OID_INSTANCE_HOME/bin/ldapmodify -a \
-D "cn=oudadmin" \
-h oudhost.example.com \
-w password \
-p 3060 \
-f admin_group.ldif
```

For Oracle Internet Directory:

```
OID_ORACLE_HOME/bin/ldapadd -h oid.example.com \
-p 3060 \
-D cn="orcladmin" \
-w password \
-c \
-v \
-f admin_group.ldif
```

5. Verify that the changes were made successfully:
 - a. Log in to the Oracle WebLogic Server Administration Console.
 - b. In the left pane of the console, click **Security Realms**.
 - c. Click the default security realm (**myrealm**).
 - d. Click the **Users and Groups** tab.
 - e. Verify that the administrator user and group you provisioned are listed on the page.

10.10.7 Adding the New Administration User to the Administration Group

After adding the users and groups to Oracle Internet Directory, the group must be assigned the Administration role within the WebLogic domain security realm. This enables all users that belong to the group to be administrators for the domain.

To assign the Administration role to the new enterprise deployment administration group:

1. Log in to the WebLogic Administration Server Console using the administration credentials that you provided in the Configuration Wizard.

Do not use the credentials for the administration user you created and provided for the new authentication provider.

2. In the left pane of the Administration Console, click **Security Realms**.
3. Click the default security realm (**myrealm**).
4. Click the **Roles and Policies** tab.
5. Expand the **Global Roles** entry in the table and click **Roles**.

[-] Domain
[-] Global Roles
[-] Roles
[-] JCOM

6. Click the **Admin** role.

[-] Role Name 	Provider Name
[-] Admin	XACMLRoleMapper

7. Click **Add Conditions** button.
8. Select **Group** from the **Predicate List** drop-down menu, and then click **Next**.
9. Enter `SOA Administrators` in the **Group Argument Name** field, and then click **Add**.

`SOA Administrators` is added to the list box of arguments.
10. Click **Finish** to return to the Edit Global Role page.

The `SOA Administrators` group is now listed.
11. Click **Save** to finish adding the **Admin** Role to the `SOA Administrators` group.
12. Validate that the changes were made by logging in to the WebLogic Administration Server Console using the new `weblogic_soa` user credentials.

If you can log in to the Oracle WebLogic Server Administration Console and Fusion Middleware Control with the credentials of the new administration user you just provisioned in the new authentication provider, then you have configured the provider successfully.

10.10.8 Updating the `boot.properties` File and Restarting the System

After you create the new administration user and group, you must update the Administration Server `boot.properties` file with the administration user credentials that you created in the LDAP directory:

1. On `SOAHOST1`, go the following directory:

`ASERVER_HOME/servers/AdminServer/security`
2. Rename the existing `boot.properties` file:

`mv boot.properties boot.properties.backup`
3. Use a text editor to create a file called `boot.properties` under the security directory.
4. Enter the following lines in the file:

`username=weblogic_soa`
`password=password`

5. Save the file.
6. Restart the Administration Server.

10.11 Adding the wsm-pm Role to the Administrators Group

After you configure a new LDAP-based Authorization Provider and restart the Administration Server, add the enterprise deployment administration LDAP group (SOA Administrators) as a member to the wsm-pm application stripe role: `policy.Updater`.

To perform this task, refer to [Configuring Roles for Administration of Oracle SOA Suite Products](#).

Configuring the Web Tier for an Enterprise Deployment

This chapter describes how to install and configure a standalone Oracle HTTP Server domain that contains two Oracle HTTP Server instances: one on WEBHOST1 and one on WEBHOST2.

This chapter provides information on variables used when configuring the web tier and installing and configuring a web tier domain.

See Also:

[Configuring the Enterprise Deployment](#)

[Variables Used When Configuring the Web Tier](#)

As you perform the tasks in this chapter, you will be referencing the directory variables listed in this section.

[About the Web Tier Domains](#)

In an enterprise deployment, each Oracle HTTP Server instance is configured on a separate host and in its own standalone domain. This

allows for a simple configuration that requires a minimum amount of configuration and a minimum amount of resources to run and maintain.

[Installing Oracle HTTP Server on WEBHOST1](#)

The following sections describe how to install the Oracle HTTP Server software on the web tier.

[Creating a Web Tier Domain on WEBHOST1](#)

The following sections describe how to create a new Oracle HTTP Server standalone domain on the first Web tier host.

[Installing and Configuring a Web Tier Domain on WEBHOST2](#)

After you install Oracle HTTP Server and configure a Web Tier domain on WEBHOST1, then you must also perform the same tasks on WEBHOST2.

[Starting the Node Manager and Oracle HTTP Server Instances on WEBHOST1 and WEBHOST2](#)

The following sections describe how to start the Oracle HTTP Server instances on WEBHOST1 and WEBHOST2.

[Configuring Oracle HTTP Server to Route Requests to the Application Tier](#)

The following sections describe how to update the Oracle HTTP Server configuration files so the web server instances route requests to the servers in the domain.

11.1 Variables Used When Configuring the Web Tier

As you perform the tasks in this chapter, you will be referencing the directory variables listed in this section.

The values for several directory variables are defined in [File System and Directory Variables Used in This Guide](#).

- *OHS_ORACLE_HOME*
- *OHS_DOMAIN_HOME*

In addition, you'll be referencing the following virtual IP (VIP) address and host names:

- ADMINVHN
- WEBHOST1
- WEBHOST2

11.2 About the Web Tier Domains

In an enterprise deployment, each Oracle HTTP Server instance is configured on a separate host and in its own standalone domain. This allows for a simple configuration that requires a minimum amount of configuration and a minimum amount of resources to run and maintain.

For more information about the role and configuration of the Oracle HTTP Server instances in the web tier, see [Understanding the Web Tier](#).

11.3 Installing Oracle HTTP Server on WEBHOST1

The following sections describe how to install the Oracle HTTP Server software on the web tier.

See Also:

[Starting the Installer on WEBHOST1](#)

[Navigating the Oracle HTTP Server Installation Screens](#)

[Verifying the Oracle HTTP Server Installation](#)

11.3.1 Starting the Installer on WEBHOST1

To start the installation program, perform the following steps.

1. Log in to WEBHOST1.
2. Go to the directory in which you downloaded the installation program.
3. Launch the installation program by entering the following command:

```
./fmw_12.2.1.0.0_ohs_linux64.bin
```

When the installation program appears, you are ready to begin the installation.

11.3.2 Navigating the Oracle HTTP Server Installation Screens

The following table lists the screens in the order that the installation program displays them.

If you need additional help with any of the installation screens, click the screen name.

Screen	Description
Welcome	This screen introduces you to the product installer.
Auto Updates	Use this screen to automatically search My Oracle Support for available patches or automatically search a local directory for patches that you've already downloaded for your organization.
Installation Location	Use this screen to specify the location of your Oracle home directory. For the purposes of an enterprise deployment, enter the value of the OHS_ORACLE_HOME variable listed in Table 7-3 .
Installation Type	Select Standalone HTTP Server (Managed independently of WebLogic server) . This installation type allows you to configure the Oracle HTTP Server instances independently from any other existing Oracle WebLogic Server domains.

Screen	Description
Prerequisite Checks	<p>This screen verifies that your system meets the minimum necessary requirements.</p> <p>If there are any warning or error messages, verify that your host computers and the required software meet the system requirements and certification information described in Host Computer Hardware Requirements and Operating System Requirements for the Enterprise Deployment Topology.</p>
Security Updates	<p>If you already have an Oracle Support account, use this screen to indicate how you would like to receive security updates.</p> <p>If you do not have an account, or if you are sure you want to skip this step, then clear the check box and verify your selection in the follow-up dialog box.</p>
Installation Summary	<p>Use this screen to verify the installation options you selected. If you want to save these options to a response file, click Save Response File and provide the location and name of the response file. Response files can be used later in a silent installation situation.</p> <p>For more information about silent or command line installation, see "Using the Oracle Universal Installer in Silent Mode" in <i>Installing Software with the Oracle Universal Installer</i>.</p>
Installation Progress	<p>This screen allows you to see the progress of the installation.</p>
Installation Complete	<p>This screen appears when the installation is complete. Review the information on this screen, then click Finish to dismiss the installer.</p>

11.3.3 Verifying the Oracle HTTP Server Installation

To verify that your Oracle HTTP Server installation completed successfully, list files that were installed in the new Oracle home directory. You should see the following directories in the Oracle HTTP Server Oracle home:

```
bin
cfgtoollogs
crs
css
has
install
inventory
jlib
ldap
lib
network
nls
ohs
OPatch
```

```
oracle_common
oracore
oraInst.loc
oui
perl
plsq
plugins
precomp
rdbms
root.sh
slax
sqlplus
srv
webgate
wls
xdk
```

11.4 Creating a Web Tier Domain on WEBHOST1

The following sections describe how to create a new Oracle HTTP Server standalone domain on the first Web tier host.

See Also:

[Starting the Configuration Wizard on WEBHOST1](#)

[Navigating the Configuration Wizard Screens for a Web Tier Domain](#)

11.4.1 Starting the Configuration Wizard on WEBHOST1

To start the Configuration Wizard, navigate to the following directory and start the WebLogic Server Configuration Wizard, as follows:

```
cd OHS_ORACLE_HOME/oracle_common/common/bin
./config.sh
```

11.4.2 Navigating the Configuration Wizard Screens for a Web Tier Domain

Oracle recommends that you create a standalone domain for the Oracle HTTP Server instances on each Web tier host.

The following topics describe how to create a new standalone Oracle HTTP Server:

- [Task 1, "Selecting the Domain Type and Domain Home Location"](#)
- [Task 2, "Selecting the Configuration Templates"](#)
- [Task 3, "Selecting the JDK for the Web Tier Domain."](#)
- [Task 4, "Adding System Components"](#)
- [Task 5, "OHS Server Screen"](#)
- [Task 6, "Reviewing Your Configuration Specifications and Configuring the Domain"](#)
- [Task 7, "Writing Down Your Domain Home"](#)

Task 1 Selecting the Domain Type and Domain Home Location

On the Configuration Type screen, select **Create a new domain**.

In the **Domain Location** field, enter the value assigned to the OHS_DOMAIN_HOME variable.

Note the following:

- The Configuration Wizard will create the new directory that you specify here.
- Create the directory on local storage, so the web servers do not have any dependencies on on storage devices outside the DMZ.

[other]:

- More information about the Domain home directory can be found in "Choosing a Domain Home" in *Planning an Installation of Oracle Fusion Middleware*.
 - More information about the other options on this screen can be found in "Configuration Type" in *Creating WebLogic Domains Using the Configuration Wizard*.
 - For more information about the Web tier and the DMZ, see [Understanding the Firewalls and Zones of a Typical Enterprise Deployment](#).
 - For more information about the OHS_DOMAIN_HOME directory variable, see [File System and Directory Variables Used in This Guide](#).
-
-

Task 2 Selecting the Configuration Templates

On the Templates screen, select **Oracle HTTP Server (Standalone) - 12.2.1.0 [ohs]**.

Tip:

More information about the options on this screen can be found in "Templates" in *Creating WebLogic Domains Using the Configuration Wizard*.

Task 3 Selecting the JDK for the Web Tier Domain.

Select the Oracle Hotspot JDK, which was installed in the Web tier Oracle home when you installed the Oracle HTTP Server software.

Task 4 Adding System Components

On the System Components screen, create one Oracle HTTP Server instance. The screen should by default have a single instance defined.

1. Note that the default instance name is ohs1 in the **System Component** field. You use this default name.
2. Make sure OHS is selected in the **Component Type** field.
3. Use the **Restart Interval Seconds** field to specify the number of seconds to wait before attempting a restart if an application is not responding.

4. Use the **Restart Delay Seconds** field to specify the number of seconds to wait between restart attempts.

Task 5 OHS Server Screen

Use the OHS Server screen to configure the OHS servers in your domain:

1. Select **ohs1** from the **System Component** drop-down menu.
2. In the **Listen Address** field, enter *WEBHOST1*.

All of the remaining fields are pre-populated, but you can change the values as required for your organization. For more information about the fields on this screen, see "OHS Server" in *Creating WebLogic Domains Using the Configuration Wizard*.

3. In the **Server Name** field, verify the value of the listen address and listen port. It should appear as follows:

```
WEBHOST1:7777
```

Task 6 Configuring Node Manager

Select **Per Domain Default Location** as the Node Manager type, and specify the Node Manager credentials.

Note:

More information about the options on this screen can be found in "[Node Manager](#)" in *Creating Domains Using the Configuration Wizard*.

More information about the Node Manager types can be found in "[Node Manager Overview](#)" in *Administering Node Manager for Oracle WebLogic Server*.

Task 7 Reviewing Your Configuration Specifications and Configuring the Domain

The Configuration Summary screen contains the detailed configuration information for the domain you are about to extend. Review the details of each item on the screen and verify that the information is correct.

You can go back to any previous screen if you need to make any changes, either by using the **Back** button or by selecting the screen in the navigation pane.

Domain creation will not begin until you click **Create**.

Tip:

More information about the options on this screen can be found in "Configuration Summary" in *Creating WebLogic Domains Using the Configuration Wizard*.

Task 8 Writing Down Your Domain Home

The Configuration Success screen will show the updated domain home location.

Make a note of the information provided here, as you will need it to start the servers and access the Administration Server.

Click **Finish** to dismiss the configuration wizard.

11.5 Installing and Configuring a Web Tier Domain on WEBHOST2

After you install Oracle HTTP Server and configure a Web Tier domain on WEBHOST1, then you must also perform the same tasks on WEBHOST2.

1. Log in to WEBHOST2 and install Oracle HTTP Server, using the instructions in [Installing Oracle HTTP Server on WEBHOST1](#).
2. Configure a new standalone domain on WEBHOST2, using the instructions in [Creating a Web Tier Domain on WEBHOST1](#).

Use the name `ohs2` for the instance on WEBHOST2, and be sure to replace all occurrences of WEBHOST1 with WEBHOST2 and all occurrences of `ohs1` with `ohs2` in each of the examples.

11.6 Starting the Node Manager and Oracle HTTP Server Instances on WEBHOST1 and WEBHOST2

The following sections describe how to start the Oracle HTTP Server instances on WEBHOST1 and WEBHOST2.

See Also:

[Starting the Node Manager on WEBHOST1 and WEBHOST2](#)

[Starting the Oracle HTTP Server Instances](#)

11.6.1 Starting the Node Manager on WEBHOST1 and WEBHOST2

Before you can start the Oracle HTTP Server instances, you must start the Node Manager on WEBHOST1 and WEBHOST2:

1. Log in to WEBHOST1 and navigate to the following directory:

```
OHS_DOMAIN_HOME/bin
```
2. Start the Node Manager as shown below, using `nohup` and `nodemanager.out` as an example output file:

```
nohup OHS_DOMAIN_HOME/bin/startNodeManager.sh > OHS_DOMAIN_HOME/nodemanager/nodemanager.out 2>&1 &
```
3. Log in to WEBHOST2 and perform steps 1 and 2.

For more information about additional Node Manager configuration options, see *Administering Node Manager for Oracle WebLogic Server*.

11.6.2 Starting the Oracle HTTP Server Instances

To start the Oracle HTTP Server instances:

1. Navigate to the following directory on WEBHOST1:

```
OHS_DOMAIN_HOME/bin
```

For more information about the location of the `OHS_DOMAIN_HOME` directory, see [File System and Directory Variables Used in This Guide](#).

2. Enter the following command:

```
./startComponent.sh ohs1
```

3. When prompted, enter the Node Manager password.
4. Repeat steps 1 through 3 to start the ohs2 instance on WEBHOST2.

For more information, see "Starting Oracle HTTP Server Instances" in *Administering Oracle HTTP Server*.

11.7 Configuring Oracle HTTP Server to Route Requests to the Application Tier

The following sections describe how to update the Oracle HTTP Server configuration files so the web server instances route requests to the servers in the domain.

See Also:

[About the Oracle HTTP Server Configuration for an Enterprise Deployment](#)

[Modifying the httpd.conf File to Include Virtual Host Configuration Files](#)

[Creating the Virtual Host Configuration Files](#)

[Validating the Virtual Server Configuration on the Load Balancer](#)

[Configuring Routing to the Administration Server and Oracle Web Services Manager](#)

[Validating Access to the Management Consoles and Administration Server](#)

11.7.1 About the Oracle HTTP Server Configuration for an Enterprise Deployment

The following topics provide overview information about the changes required to the Oracle HTTP Server configuration in an enterprise deployment.

See Also:

[Purpose of the Oracle HTTP Server Virtual Hosts](#)

[About the WebLogicCluster Parameter of the <VirtualHost> Directive](#)

[Recommended Structure of the Oracle HTTP Server Configuration Files](#)

11.7.1.1 Purpose of the Oracle HTTP Server Virtual Hosts

The reference topologies in this guide require that you define a set of virtual servers on the hardware load balancer. You can then configure Oracle HTTP Server to recognize requests to specific virtual hosts (that map to the load balancer virtual servers) by adding <VirtualHost> directives to the Oracle HTTP Server instance configuration files.

For each Oracle HTTP Server virtual host, you define a set of specific URLs (or context strings) that route requests from the load balancer through the Oracle HTTP Server instances to the appropriate Administration Server or Managed Server in the Oracle WebLogic Server domain.

11.7.1.2 About the WebLogicCluster Parameter of the <VirtualHost> Directive

A key parameter of the Oracle HTTP Server <VirtualHost> directive is the `WebLogicCluster` parameter, which is part of the WebLogic Proxy Plug-In for Oracle HTTP Server. When configuring Oracle HTTP Server for an enterprise deployment, consider the following information when adding this parameter to the Oracle HTTP Server configuration files.

The servers specified in the `WebLogicCluster` parameter are important only at startup time for the plug-in. The list needs to provide at least one running cluster member for the plug-in to discover other members of the cluster. The listed cluster member must be running when Oracle HTTP Server is started. Oracle WebLogic Server and the plug-in work together to update the server list automatically with new, failed, and recovered cluster members.

Some example scenarios:

- Example 1: If you have a two-node cluster and then add a third member, you do not need to update the configuration to add the third member. The third member will be discovered on the fly at runtime.
- Example 2: You have a three-node cluster but only two nodes are listed in the configuration. However, if both listed nodes are down when you start Oracle HTTP Server, then the plug-in would fail to route to the cluster. You must ensure that at least one of the listed nodes is running when you start Oracle HTTP Server.

If you list all members of the cluster, then you guarantee you can route to the cluster, assuming at least one member is running when Oracle HTTP Server is started.

11.7.1.3 Recommended Structure of the Oracle HTTP Server Configuration Files

Rather than adding multiple virtual host definitions to the `httpd.conf` file, Oracle recommends that you create separate, smaller, and more specific configuration files for each of the virtual servers required for the products you are deploying. This avoids populating an already large `httpd.conf` file with additional content, and it can make troubleshooting configuration problems easier.

For example, in a typical Oracle Fusion Middleware Infrastructure domain, you can add a specific configuration file called `admin_vh.conf` that contains the virtual host definition for the Administration Server virtual host (ADMINVHN).

11.7.2 Modifying the httpd.conf File to Include Virtual Host Configuration Files

Perform the following tasks to prepare the `httpd.conf` file for the additional virtual hosts required for an enterprise topology:

1. Log in to WEBHOST1.
2. Locate the `httpd.conf` file for the first Oracle HTTP Server instance (`ohs1`) in the domain directory:

```
cd OHS_DOMAIN_HOME/config/fmwconfig/components/OHS/ohs1/
```
3. Open the `httpd.conf` file in a text editor and make the following changes:
 - a. Create a `NameVirtualHost` entry as follows immediately below the comment: "# Use name-based virtual hosting" and before the "# VirtualHost example" comment:

```
NameVirtualHost WEBHOST1:7777
```

In this example, replace WEBHOST1 with the value of the WEBHOST1 variable. For more information, see [File System and Directory Variables Used in This Guide](#).

- b. Verify that there is an INCLUDE statement in the `httpd.conf` that includes all `*.conf` files in the `moduleconf` subdirectory:

```
IncludeOptional "moduleconf/*.conf"
```

This statement makes it possible to create the separate virtual host files for each component, making it easier to update, maintain, and scale out the virtual host definitions.

4. Save the `httpd.conf` file.
5. Log in to WEBHOST2 and perform steps 2 through 4 to update the `httpd.conf` file for the `ohs2` instance.

On WEBHOST2, replace all instances of WEBHOST1 with WEBHOST2 and all instances of `ohs1` with `ohs2`.

11.7.3 Creating the Virtual Host Configuration Files

To create the virtual host configuration files:

Note: Before you create the virtual host configuration files, be sure you have configured the virtual servers on the load balancer, as described in [Purpose of the Oracle HTTP Server Virtual Hosts](#).

1. Log in to WEBHOST1 and change directory to the configuration directory for the first Oracle HTTP Server instance (`ohs1`):

```
cd OHS_DOMAIN_HOME/config/fmwconfig/components/OHS/ohs1/moduleconf
```

2. Create the `admin_vh.conf` file and add the following directive:

```
<VirtualHost WEBHOST1:7777>
  ServerName admin.example.com:80
  ServerAdmin you@your.address
  RewriteEngine On
  RewriteOptions inherit
</VirtualHost>
```

3. Create the `soainternal_vh.conf` file and add the following directive:

```
<VirtualHost WEBHOST1:7777>
  ServerName soainternal.example.com:80
  ServerAdmin you@your.address
  RewriteEngine On
  RewriteOptions inherit
</VirtualHost>
```

4. Restart the `ohs1` instance:
 - a. Change directory to the following location:

```
cd OHS_DOMAIN_HOME/bin
```



```

        WLSRequest ON
        WebLogicHost ADMINVHN
        WeblogicPort 7001
    </Location>

    <Location /em>
        WLSRequest ON
        WebLogicHost ADMINVHN
        WeblogicPort 7001
    </Location>

```

The `admin_vh.conf` file should appear as it does in [Task 1, "admin_vh.conf file"](#).

3. Add the following directives to the `soainternal_vh.conf` file within the `<VirtualHost>` tag:

```

# WSM-PM
<Location /wsm-pm>
    WLSRequest ON
    WebLogicCluster SOAHOST1:7010,SOAHOST2:7010
    WLProxySSL ON
    WLProxySSLPassThrough ON
</Location>

```

The `soainternal_vh.conf` file should appear as it does in [Task 1, "soainternal_vh.conf file"](#).

For more information about the `WebLogicCluster` parameter in this example, see [About the WebLogicCluster Parameter of the <VirtualHost> Directive](#).

4. Restart the `ohs1` instance:

- a. Change directory to the following location:

```
cd OHS_DOMAIN_HOME/bin
```

- b. Enter the following commands to stop and start the instance:

```
./stopComponent.sh ohs1
./startComponent.sh ohs1
```

5. Copy the `admin_vh.conf` file and the `soainternal_vh.conf` file to the configuration directory for the second Oracle HTTP Server instance (`ohs2`) on `WEBHOST2`:

```
OHS_DOMAIN_HOME/config/fmwconfig/components/OHS/ohs2/moduleconf/
```

6. Edit the `admin_vh.conf` and `soainternal_vh.conf` files and change any references to `WEBHOST1` to `WEBHOST2` in the `<VirtualHost>` directives.

7. Restart the `ohs2` instance:

- a. Change directory to the following location:

```
cd OHS_DOMAIN_HOME/bin
```

- b. Enter the following commands to stop and start the instance:

```
./stopComponent.sh ohs2
./startComponent.sh ohs2
```

admin_vh.conf file

```
<VirtualHost WEBHOST1:7777>
  ServerName admin.example.com:7777
  ServerAdmin you@your.address
  RewriteEngine On
  RewriteOptions inherit

# Admin Server and EM
<Location /console>
  WLSRequest ON
  WebLogicHost ADMINVHN
  WeblogicPort 7001
</Location>

<Location /consolehelp>
  WLSRequest ON
  WebLogicHost ADMINVHN
  WeblogicPort 7001
</Location>

<Location /em>
  WLSRequest ON
  WebLogicHost ADMINVHN
  WeblogicPort 7001
</Location>
</VirtualHost>
```

soainternal_vh.conf file

Contents of this file:

```
<VirtualHost WEBHOST1:7777>
  ServerName soainternal.example.com:80
  ServerAdmin you@your.address
  RewriteEngine On
  RewriteOptions inherit

# WSM-PM
<Location /wsm-pm>
  WLSRequest ON
  WebLogicCluster SOAHOST1:7010,SOAHOST2:7010
</Location>
</VirtualHost>
```

11.7.6 Validating Access to the Management Consoles and Administration Server

To verify the changes you have made in this chapter:

1. Use the following URL to the hardware load balancer to display the Oracle WebLogic Server Administration Console, and log in using the Oracle WebLogic Server administrator credentials:

```
http://admin.example.com/console
```

This validates that the `admin.example.com` virtual host on the load balancer is able to route requests to the Oracle HTTP Server instances on the web tier, which in turn can route requests for the Oracle WebLogic Server Administration Console to the Administration Server in the application tier.

2. Similarly, you should be able to access the Fusion Middleware Control using a similar URL:

`http://admin.example.com/em`

Extending the Domain with Oracle SOA Suite

The following topics describe how to extend the enterprise deployment domain with the Oracle SOA Suite software.

See Also:

[Configuring the Enterprise Deployment](#)

[Variables Used When Extending the Domain with Oracle SOA Suite](#)

As you perform the tasks in this chapter, you will be referencing the directory variables listed in this section.

[Synchronizing the System Clocks](#)

Before you extend the domain to include Oracle SOA Suite, verify that the system clocks on each host computer are synchronized. You can do this by running the `date` command as simultaneously as possible on the hosts in each cluster.

[Installing the Software for an Enterprise Deployment](#)

The following sections describe how to install the software for an enterprise deployment.

[Creating the Oracle SOA Suite Database Schemas](#)

Before you can configure an Oracle SOA Suite Foundation domain, you must install the required schemas in a certified database for use with this release of Oracle Fusion Middleware.

[Extending the Enterprise Deployment Domain with Oracle SOA Suite](#)

This section provides instructions for extending the existing enterprise deployment domain with the Oracle SOA Suite software.

[Configuring a Default Persistence Store for Transaction Recovery](#)

Each Managed Server uses a transaction log that stores information about committed transactions that are coordinated by the server and that may not have been completed. Oracle WebLogic Server uses this transaction log for recovery from system crashes or network failures. To leverage the migration capability of the Transaction Recovery Service for the Managed Servers within a cluster, store the transaction log in a location accessible to each Managed Server and its backup server.

[Propagating the Extended Domain to the Domain Directories and Machines](#)

After you have extended the domain with the Oracle SOA Suite instances, and you have restarted the Administration Server on

SOAHOST1, you must then propagate the domain changes to the domain directories and machines.

[Starting and Validating the WLS_SOA1 Managed Server](#)

Now that you have extended the domain, restarted the Administration Server, and propagated the domain to the other hosts, you can start the newly configured Oracle SOA Suite Managed Servers.

[Starting and Validating the WLS_SOA2 Managed Server](#)

After you have validated the successful configuration and startup of the WLS_SOA1 Managed Server, you can then start and validate the WLS_SOA2 Managed Server.

[Validating the Location and Creation of the Transaction Logs](#)

After WLS_SOA1 and WLS_SOA2 are up and running, verify that the transaction log directory and transaction logs were created as expected.

[Configuring Oracle HTTP Server for the Extended Domain](#)

The following sections describe how to configure the Oracle HTTP Server instances so they route requests for both public and internal URLs to the proper Webcenter Portal clusters in the enterprise topology.

[Post-Configuration Steps for Oracle SOA Suite](#)

After you install and configure Oracle SOA Suite and Oracle Business Process Management, consider the following post-configuration tasks.

[Enabling Automatic Service Migration and JDBC Persistent Stores for Oracle SOA Suite](#)

To ensure that Oracle SOA Suite is configured for high availability, configure the Oracle SOA Suite Managed Servers for automatic service migration.

12.1 Variables Used When Extending the Domain with Oracle SOA Suite

As you perform the tasks in this chapter, you will be referencing the directory variables listed in this section.

The values for several directory variables are defined in [File System and Directory Variables Used in This Guide](#).

- ORACLE_HOME
- ASERVER_HOME
- MSERVER_HOME
- APPLICATION_HOME
- DEPLOY_PLAN_HOME
- OHS_DOMAIN_HOME
- JAVA_HOME
- ORACLE_RUNTIME

In addition, you'll be referencing the following virtual IP (VIP) address defined in [Reserving the Required IP Addresses for an Enterprise Deployment](#):

- ADMINVHN

Actions in this chapter will be performed on the following host computers:

- SOAHOST1
- SOAHOST2
- WEBHOST1
- WEBHOST2

12.2 Synchronizing the System Clocks

Before you extend the domain to include Oracle SOA Suite, verify that the system clocks on each host computer are synchronized. You can do this by running the `date` command as simultaneously as possible on the hosts in each cluster.

Alternatively, there are third-party and open-source utilities you can use for this purpose.

12.3 Installing the Software for an Enterprise Deployment

The following sections describe how to install the software for an enterprise deployment.

See Also:

[Starting the Installer on SOAHOST1](#)

[Navigating the Installation Screens](#)

[Installing the Software on the Other Host Computers](#)

[Verifying the Installation](#)

12.3.1 Starting the Installer on SOAHOST1

To start the installation program:

1. Log in to SOAHOST1.
2. Go to the directory where you downloaded the installation program.
3. Launch the installation program by invoking the `java` executable from the JDK directory on your system, as shown in the example below.

```
JAVA_HOME/bin/java -d64 -jar fmw_12.2.1.0.0_product.jar
```

Be sure to replace the JDK location in these examples with the actual JDK location on your system.

For information about downloading the software and locating the actual installer file name for your product, see [Identifying and Obtaining Software Downloads for an Enterprise Deployment](#).

When the installation program appears, you are ready to begin the installation.

12.3.2 Navigating the Installation Screens

The installation program displays a series of screens, in the order listed in the following table.

If you need additional help with any of the installation screens, click the screen name.

Screen	Description
Welcome	This screen introduces you to the product installer.
Auto Updates	Use this screen to automatically search My Oracle Support for available patches or automatically search a local directory for patches that you've already downloaded for your organization.
Installation Location	Use this screen to specify the location of your Oracle home directory. For more information about Oracle Fusion Middleware directory structure, see "Selecting Directories for Installation and Configuration" in <i>Planning an Installation of Oracle Fusion Middleware</i> .
Installation Type	Use this screen to select the type of installation and consequently, the products and feature sets you want to install. <ul style="list-style-type: none"> • Select SOA Suite <p>NOTE: The topology in this document does not include the examples. Oracle strongly recommends that you do not install the examples into a production environment.</p>
Prerequisite Checks	This screen verifies that your system meets the minimum necessary requirements. If there are any warning or error messages, you can refer to one of the documents in the Roadmap for Verifying Your System Environment section in <i>Planning Your Oracle Fusion Middleware Infrastructure Installation</i> .
Installation Summary	Use this screen to verify the installation options you selected. Click Install to begin the installation.
Installation Progress	This screen allows you to see the progress of the installation. Click Next when the progress bar reaches 100% complete.
Installation Complete	Review the information on this screen, then click Finish to dismiss the installer.

12.3.3 Installing the Software on the Other Host Computers

If you have configured a separate shared storage volume or partition for SOAHOST2, then you must also install the software on SOAHOST2. For more information, see [Shared Storage Recommendations When Installing and Configuring an Enterprise Deployment](#).

Note that the location where you install the Oracle home (which contains the software binaries) will vary, depending upon the host. To identify the proper location for you Oracle home directories, refer to the guidelines in [File System and Directory Variables Used in This Guide](#).

12.3.4 Verifying the Installation

After you complete the installation, you can verify it by successfully completing the following tasks.

See Also:

[Reviewing the Installation Log Files](#)

[Checking the Directory Structure](#)

[Viewing the Contents of Your Oracle Home](#)

12.3.4.1 Reviewing the Installation Log Files

Review the contents of the installation log files to make sure that no problems were encountered. For a description of the log files and where to find them, see "Understanding Installation Log Files" in *Installing Software with the Oracle Universal Installer*.

12.3.4.2 Checking the Directory Structure

The contents of your installation vary based on the options you selected during the installation.

The addition of Oracle SOA Suite adds the following directory and sub-directories:

```
/u01/oracle/products/fmw/soa
```

```
aiafp  
bam  
bin  
bpm  
common  
integration  
jlib  
plugins  
readme.txt  
reports  
soa
```

For more information about the directory structure you should see after installation, see "What are the Key Oracle Fusion Middleware Directories?" in *Understanding Oracle Fusion Middleware*.

12.3.4.3 Viewing the Contents of Your Oracle Home

You can also view the contents of your Oracle home using the `viewInventory` script. For more information, see "Viewing the contents of an Oracle home" in *Installing Software with the Oracle Universal Installer*.

12.4 Creating the Oracle SOA Suite Database Schemas

Before you can configure an Oracle SOA Suite Foundation domain, you must install the required schemas in a certified database for use with this release of Oracle Fusion Middleware.

Follow the instructions in this section to install the schemas.

See Also:

[Starting the Repository Creation Utility \(RCU\)](#)

[Navigating the RCU Screens to Create the Schemas](#)

[Configuring SOA Schemas for Transactional Recovery](#)

12.4.1 Starting the Repository Creation Utility (RCU)

To start the Repository Creation Utility (RCU):

1. Navigate to the `ORACLE_HOME/oracle_common/bin` directory on your system.
2. Make sure the `JAVA_HOME` environment variable is set to the location of a certified JDK on your system. The location should be up to but not including the `bin` directory. For example, if your JDK is located in `/u01/oracle/products/jdk`:

On UNIX operating systems:

```
export JAVA_HOME=/u01/oracle/products/jdk
```

3. Start RCU:

On UNIX operating systems:

```
./rcu
```

12.4.2 Navigating the RCU Screens to Create the Schemas

Schema creation involves the following tasks:

- [Task 1, "Introducing RCU"](#)
- [Task 2, "Selecting a Method of Schema Creation"](#)
- [Task 3, "Providing Database Connection Details"](#)
- [Task 4, "Specifying a Custom Prefix and Selecting Schemas"](#)
- [Task 5, "Specifying Schema Passwords"](#)
- [Task 6, "Specifying Custom Variables"](#)
- [Task 7, "Verifying the Tablespaces for the Required Schemas"](#)
- [Task 8, "Completing Schema Creation"](#)
- [Task 9, "Verifying the Schema Creation"](#)

Task 1 Introducing RCU

Click **Next**.

Task 2 Selecting a Method of Schema Creation

If you have the necessary permission and privileges to perform DBA activities on your database, select **System Load and Product Load Concurrently**. This procedure assumes that you have the necessary privileges.

If you do not have the necessary permission or privileges to perform DBA activities in the database, you must select **Prepare Scripts for System Load** on this screen. This option will generate a SQL script, which can be provided to your database administrator. See "Understanding System Load and Product Load" in *Creating Schemas with the Repository Creation Utility*.

Task 3 Providing Database Connection Details

Provide the database connection details for RCU to connect to your database.

In the **Host Name** field, enter the SCAN address of the Oracle RAC Database.

Enter the **DBMS/Service** details.

Enter the **Schema Owner** and **Schema Password** details.

Click **Next** to proceed, then click **OK** on the dialog window confirming that connection to the database was successful.

Task 4 Specifying a Custom Prefix and Selecting Schemas

Choose **Select existing prefix**, and then select the prefix you used when you created the initial domain.

From the list of schemas, select the **SOA Suite** schema. This will automatically select **SOA Infrastructure**. In addition, the following dependent schemas have already been installed with the Infrastructure and are grayed out:

- **Metadata Services**
- **Audit Services**
- **Audit Services Append**
- **Audit Services Viewer**
- **Oracle Platform Security Services**
- **User Messaging Service**

The custom prefix is used to logically group these schemas together for use in this domain only; you must create a unique set of schemas for each domain as schema sharing across domains is not supported.

Tip:

For more information about custom prefixes, see "Understanding Custom Prefixes" in *Creating Schemas with the Repository Creation Utility*.

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

Tip:

You must make a note of the custom prefix you choose to enter here; you will need this later on during the domain creation process.

Click **Next** to proceed, then click **OK** on the dialog window confirming that prerequisite checking for schema creation was successful.

Task 5 Specifying Schema Passwords

Specify how you want to set the schema passwords on your database, then specify and confirm your passwords.

Tip:

You must make a note of the passwords you set on this screen; you will need them later on during the domain creation process.

Task 6 Specifying Custom Variables

Specify the custom variables for the SOA Infrastructure schema.

For the enterprise deployment topology, enter `LARGE` for the **Database Profile** custom variable; if you are planning on using Oracle Healthcare, then enter `YES` for the **Healthcare Integration** variable.

For more information, see "About the Custom Variables Required for the SOA Suite Schemas" in *Installing and Configuring Oracle SOA Suite and Business Process Management*.

Component	Custom Variable	Value
SOA Infrastructure	Database Profile (SMALL/MED/LARGE)	LARGE
	Healthcare Integration(YES/NO)	NO

Task 7 Verifying the Tablespaces for the Required Schemas

On the Map Tablespaces screen, review the information, and then click **Next** to accept the default values.

Click **OK** in the confirmation dialog box.

Task 8 Completing Schema Creation

Navigate through the remainder of the RCU screens to complete schema creation. When you reach the Completion Summary screen, click **Close** to dismiss RCU.

Task 9 Verifying the Schema Creation

To verify that the schemas were created successfully, and to verify the database connection details, use `SQL*Plus` or another utility to connect to the database, using the `SOAINFRA` schema name and the password you provided.

For example:

```
./sqlplus
SQL*Plus: Release 11.2.0.4.0 Production on Fri Nov 1 08:44:18 2013
Copyright (c) 1982, 2013, Oracle. All rights reserved.
Enter user-name: FMW1221_SOAINFRA
Enter password: soainfra_password

Connected to:
Oracle Database 11g Enterprise Edition Release 11.2.0.4.0 - 64bit Production
With the Partitioning, OLAP, Data Mining and Real Application Testing options

SQL>
```

12.4.3 Configuring SOA Schemas for Transactional Recovery

After you have installed the Oracle SOA Suite schemas successfully, use the procedure in this section to configure the schemas for transactional recovery.

This procedure sets the appropriate database privileges so that the Oracle WebLogic Server transaction manager can query the schemas for transaction state information and issue the appropriate commands, such as commit and rollback, during recovery of in-flight transactions after a WebLogic Server is unexpectedly unavailable.

These privileges should be granted to the owner of the SOAINFRA schema, which you defined when you created the schemas with the Repository Creation Utility.

To configure the SOA schemas for transactional recovery privileges:

1. Log on to SQL*Plus as a user with sysdba privileges. For example:

```
sqlplus "/ as sysdba"
```

2. Enter the following commands:

```
SQL> Grant select on sys.dba_pending_transactions to soa_schema_prefix_soainfra;
```

```
Grant succeeded.
```

```
SQL> Grant force any transaction to soa_schema_prefix_soainfra;
```

```
Grant succeeded.
```

```
SQL>
```

12.5 Extending the Enterprise Deployment Domain with Oracle SOA Suite

This section provides instructions for extending the existing enterprise deployment domain with the Oracle SOA Suite software.

Extending the domain involves the following tasks.

See Also:

[Starting the Configuration Wizard](#)

[Navigating the Configuration Wizard Screens to Extend the Domain with Oracle SOA Suite](#)

12.5.1 Starting the Configuration Wizard

Note:

If you added any customizations directly to the start scripts in the domain, those will be overwritten by the Configuration Wizard. To customize server startup parameters that apply to all servers in a domain, you can create a file called `setUserOverrides.sh` and configure it to, for example, add custom libraries to the WebLogic Server classpath, specify additional Java command-line options for running the servers, or specify additional environment variables. Any customizations you add to this file are preserved during domain upgrade operations, and are carried over to remote servers when using the `pack` and `unpack` commands.

To start the Configuration Wizard:

1. Shut down the domain completely before extending the domain. From the WebLogic Server Console, stop all managed servers and verify, and then stop the Administration Server.

For more information, see the instructions for shutting down the Administration Server with Node Manager in [Starting the Configuration Wizard on SOAHOST1](#).

2. Navigate to the following directory and start the WebLogic Server Configuration Wizard.

```
cd ORACLE_HOME/oracle_common/common/bin
./config.sh
```

12.5.2 Navigating the Configuration Wizard Screens to Extend the Domain with Oracle SOA Suite

Follow the instructions in this section to create and configure the domain for the topology.

Note:

You can use the same procedure described in this section to extend an existing domain. If your needs do not match the instructions given in the procedure, be sure to make your selections accordingly, or refer to the supporting documentation for additional details.

Domain creation and configuration includes the following tasks.

- [Task 1, "Selecting the Domain Type and Domain Home Location"](#)
- [Task 2, "Selecting the Configuration Template"](#)
- [Task 3, "Specifying the Database Configuration Type"](#)
- [Task 4, "Specifying JDBC Component Schema Information"](#)
- [Task 5, "Providing the GridLink Oracle RAC Database Connection Details"](#)
- [Task 6, "Testing the JDBC Connections"](#)
- [Task 7, "Selecting Advanced Configuration"](#)
- [Task 8, "Configuring Managed Servers"](#)
- [Task 9, "Configuring a Cluster"](#)
- [Task 10, "Assigning Managed Servers to the Cluster"](#)
- [Task 11, "Configuring Coherence Clusters"](#)
- [Task 12, "Verifying the Existing Machines"](#)
- [Task 13, "Assigning Servers to Machines"](#)
- [Task 14, "Configuring the JMS File Store"](#)

- [Task 15, "Reviewing Your Configuration Specifications and Configuring the Domain"](#)
- [Task 16, "Writing Down Your Domain Home and Administration Server URL"](#)
- [Task 17, "Start the Administration Server"](#)

Task 1 Selecting the Domain Type and Domain Home Location

On the Configuration Type screen, select **Update an existing domain**.

In the **Domain Location** field, select the value of the *ASERVER_HOME* variable, which represents the complete path to the Administration Server domain home you created when you created the initial domain.

Do not enter the value of the *MSERVER_HOME* variable, which represents the location of the Managed Servers domain directory.

For more information about the directory location variables, see [File System and Directory Variables Used in This Guide](#)

Tip:

More information about the other options on this screen can be found in Configuration Type in *Creating WebLogic Domains Using the Configuration Wizard*.

Task 2 Selecting the Configuration Template

On the Templates screen, make sure **Update Domain Using Product Templates** is selected, then select the following templates:

- **Oracle SOA Suite - 12.2.1.0 [soa]**

The following additional templates should already be selected, because they were used to create the initial domain:

- Oracle Enterprise Manager - 12.2.1.0 [em]
- Oracle WSM Policy Manager - 12.2.1.0 [oracle_common]
- Oracle JRF - 12.2.1.0 [oracle_common]
- WebLogic Coherence Cluster Extension - 12.2.1.0 [wlserver]

Tip:

More information about the options on this screen can be found in Templates in *Creating WebLogic Domains Using the Configuration Wizard*.

Task 3 Specifying the Database Configuration Type

On the Database Configuration Type screen, select **RCU Data**.

All fields are pre-populated, because you already configured the domain to reference the Fusion Middleware schemas that are required for the Infrastructure domain.

Verify and ensure that credentials in all the fields are the same that you have provided while configuring Oracle Fusion Middleware Infrastructure.

Click **Get RCU Configuration** after you finish verifying the database connection information. The following output in the Connection Result Log indicates that the operating succeeded:

```
Connecting to the database server...OK
Retrieving schema data from database server...OK
Binding local schema components with retrieved data...OK
```

Successfully Done.

Tip:

For more information about the **RCU Data** option, see "Understanding the Service Table Schema" in *Creating Schemas with the Repository Creation Utility*.

For more information about the other options on this screen, see "Datasource Defaults" in *Creating WebLogic Domains Using the Configuration Wizard*.

Task 4 Specifying JDBC Component Schema Information

On the JDBC Component Schema screen, select all the SOA schemas in the table.

When you select the schemas, the fields on the page are activated and the database connection fields are populated automatically.

Click **Convert to GridLink** and click **Next**.

Task 5 Providing the GridLink Oracle RAC Database Connection Details

On the GridLink Oracle RAC Component Schema screen, provide the information required to connect to the RAC database and component schemas, as shown in the following table.

Element	Description and Recommended Value
SCAN, Host Name, and Port	Select the SCAN check box. In the Host Name field, enter the Single Client Access Name (SCAN) Address for the Oracle RAC database. In the Port field, enter the SCAN listening port for the database (for example, 1521)
ONS Host and Port	In the ONS Host field, enter the SCAN address for the Oracle RAC database. In the Port field, enter the ONS Remote port (typically, 6200).
Enable Fan	Verify that the Enable Fan check box is selected, so the database can receive and process FAN events.

Task 6 Testing the JDBC Connections

Use the JDBC Component Schema Test screen to test the data source connections you have just configured.

A green check mark in the **Status** column indicates a successful test. If you encounter any issues, see the error message in the Connection Result Log section of the screen, fix the problem, then try to test the connection again.

Tip:

For more information about the other options on this screen, see "Test Component Schema" in *Creating WebLogic Domains Using the Configuration Wizard*.

Task 7 Keystore

Use this screen to specify details about the keystore to be used in the new domain.

For a typical enterprise deployment, you can leave the default values.

For more information, see Keystore in *Creating WebLogic Domains Using the Configuration Wizard*.

Task 8 Selecting Advanced Configuration

To complete domain configuration for the topology, select the following options on the Advanced Configuration screen:

- **Managed Server, Clusters and Coherence**
- **JMS File Store**

Task 9 Configuring Managed Servers

On the Managed Servers screen, a new Managed Server for Oracle SOA Suite appears in the list of servers. This server was created automatically by the Oracle SOA Suite configuration template you selected in [Task 1, "Selecting the Configuration Template"](#).

Perform the following tasks to modify the default Oracle SOA Suite Managed Server and create a second Oracle SOA Suite Managed Server:

1. Rename the default Oracle SOA Suite Managed Server to WLS_SOA1.
2. Click **Add** to create a new Oracle SOA Suite Managed Server, and name it WLS_SOA2.

Tip:

The server names recommended here will be used throughout this document; if you choose different names, be sure to replace them as needed.

3. Use the information in the following table to fill in the rest of the columns for each Oracle SOA Suite Managed Server.

Tip:

More information about the options on the Managed Server screen can be found in Managed Servers in *Creating WebLogic Domains Using the Configuration Wizard*.

Server Name	Listen Address	Listen Port	Enable SSL	SSL Listen Port	Server Groups
WLS_WSM1	SOAHOST1	7010	No	Disabled	WSMPM-MAN-SVR, JRF-MAN-SVR, and WSM-CACHE-SVR
WLS_WSM2	SOAHOST2	7010	No	Disabled	WSMPM-MAN-SVR, JRF-MAN-SVR, and WSM-CACHE-SVR

Server Name	Listen Address	Listen Port	Enable SSL	SSL Listen Port	Server Groups
WLS_SOA1	SOAHOST1	8001	No	Disabled	SOA-MGD-SVRS-ONLY
WLS_SOA2	SOAHOST2	8001	No	Disabled	SOA-MGD-SVRS-ONLY

Task 10 Configuring a Cluster

In this task, you create a cluster of Managed Servers to which you can target the Oracle SOA Suite software.

You will also set the **Frontend Host** property for the cluster, which ensures that, when necessary, WebLogic Server will redirect Web services callbacks and other redirects to `soa.example.com` on the load balancer rather than the address in the HOST header of each request.

For more information about the `soa.example.com` virtual server address, see [Configuring Virtual Hosts on the Hardware Load Balancer](#).

Use the Clusters screen to create a new cluster:

1. Click the **Add** button.
2. Specify `SOA_Cluster` in the **Cluster Name** field.
3. Specify `soa.example.com` in the **Frontend Host** field.
4. Specify 80 as the **Frontend HTTP Port** and 443 as the **Frontend HTTPS** port.

Note:

By default, server instances in a cluster communicate with one another using unicast. If you want to change your cluster communications to use multicast, refer to "Considerations for Choosing Unicast or Multicast" in *Administering Clusters for Oracle WebLogic Server*.

Tip:

More information about the options on this screen can be found in Clusters in *Creating WebLogic Domains Using the Configuration Wizard*.

Task 11 Assigning Managed Servers to the Cluster

Use the Assign Servers to Clusters screen to assign `WLS_SOA1` and `WLS_SOA2` to the new cluster `SOA_Cluster`:

1. In the Clusters pane, select the cluster to which you want to assign the servers; in this case, `SOA_Cluster`.
2. In the Servers pane, assign `WLS_SOA1` to `SOA_Cluster` by doing one of the following:
 - Click `WLS_SOA1` Managed Server once to select it, and then click on the right arrow to move it beneath the selected cluster in the Clusters pane.
 - Double-click `WLS_SOA1` to move it beneath the selected cluster in the clusters pane.

3. Repeat to assign WLS_SOA2 to SOA_Cluster.

Tip:

More information about the options on this screen can be found in Assign Servers to Clusters in *Creating WebLogic Domains Using the Configuration Wizard*.

Task 12 Configuring Coherence Clusters

Use the Coherence Clusters screen to configure the Coherence cluster that is automatically added to the domain. Leave the port number value at 9991, as it was defined during the initial Infrastructure domain creation.

Note:

For Coherence licensing information, refer to "Oracle Coherence" in *Oracle Fusion Middleware Licensing Information*.

Task 13 Verifying the Existing Machines

Under the **Unix Machine** tab, verify the names of the machines you created when creating the initial Infrastructure domain.

Click **Next** to proceed.

Task 14 Assigning Servers to Machines

Use the Assign Servers to Machines screen to assign the Oracle SOA Suite Managed Servers you just created to the corresponding machines in the domain.

Assign WLS_SOA1 to SOAHOST1, and assign WLS_SOA2 to SOAHOST2.

Tip:

More information about the options on this screen can be found in Assign Servers to Machines in *Creating WebLogic Domains Using the Configuration Wizard*.

Task 15 Configuring the JMS File Store

In the JMS File Stores screen, assign the following directory for each of the SOA Persistence stores, including UMS and BPM file stores:

```
ORACLE_RUNTIME/domain_name/soa_cluster/jms
```

In this example, replace *ORACLE_RUNTIME* with the value of the variable for your environment. Replace *soa_cluster* with the name you assigned to the cluster.

Task 16 Reviewing Your Configuration Specifications and Configuring the Domain

The Configuration Summary screen contains the detailed configuration information for the domain you are about to create. Review the details of each item on the screen and verify that the information is correct.

You can go back to any previous screen if you need to make any changes, either by using the **Back** button or by selecting the screen in the navigation pane.

Domain creation will not begin until you click **Update**.

Tip:

More information about the options on this screen can be found in Configuration Summary in *Creating WebLogic Domains Using the Configuration Wizard*.

Task 17 Writing Down Your Domain Home and Administration Server URL

The Configuration Success screen will show the following items about the domain you just configured, including:

- Domain Location
- Administration Server URL

Make a note of both these items, because you will need them later; you will need the domain location to access the scripts used to start the Administration Server, and you will need the Administration Server URL to access the WebLogic Server Administration Console and Oracle Enterprise Manager Fusion Middleware Control. Click **Finish** to dismiss the configuration wizard.

Task 18 Start the Administration Server

Start the Administration Server to ensure the changes you have made to the domain have been applied.

12.6 Configuring a Default Persistence Store for Transaction Recovery

Each Managed Server uses a transaction log that stores information about committed transactions that are coordinated by the server and that may not have been completed. Oracle WebLogic Server uses this transaction log for recovery from system crashes or network failures. To leverage the migration capability of the Transaction Recovery Service for the Managed Servers within a cluster, store the transaction log in a location accessible to each Managed Server and its backup server.

Note:

To enable migration of the Transaction Recovery Service, specify a location on a persistent storage solution that is available to other servers in the cluster. All Managed Servers in the cluster must be able to access this directory. This directory must also exist before you restart the server.

The recommended location is a dual-ported SCSI disk or on a Storage Area Network (SAN). Note that it is important to set the appropriate replication and backup mechanisms at the storage level to guarantee protection in cases of a storage failure.

This information applies for file-based transaction logs. You can also configure a database-based persistent store for translation logs. For more information, see [Using JDBC Persistent Stores for TLOGs and JMS in an Enterprise Deployment](#).

To set the location for the default persistence stores:

1. Log into the Oracle WebLogic Server Administration Console:

```
ADMINVHN:7001/console
```

2. In the Change Center section, click **Lock & Edit**.
3. For each of the Managed Servers in the cluster:
 - a. In the Domain Structure window, expand the **Environment** node, and then click the **Servers** node.
The Summary of Servers page appears.
 - b. Click the name of the server (represented as a hyperlink) in **Name** column of the table.
The settings page for the selected server appears and defaults to the Configuration tab.
 - c. On the Configuration tab, click the **Services** tab.
 - d. In the Default Store section of the page, enter the path to the folder where the default persistent stores will store its data files.
For the enterprise deployment, use the `ORACLE_RUNTIME` directory location. This subdirectory serves as the central, shared location for transaction logs for the cluster. For more information, see [File System and Directory Variables Used in This Guide](#).
For example:

```
ORACLE_RUNTIME/domain_name/cluster_name/tlogs
```


Replace `cluster_name` with the name of the cluster you just created.
 - e. Click **Save**.
4. Click **Save** and **Activate Changes**.

Note:

You will validate the location and the creation of the transaction logs later in the configuration procedure.

12.7 Propagating the Extended Domain to the Domain Directories and Machines

After you have extended the domain with the Oracle SOA Suite instances, and you have restarted the Administration Server on SOAHOST1, you must then propagate the domain changes to the domain directories and machines.

[Table 12-1](#) summarizes the steps required to propagate the changes to all the domain directories and machines.

Note that there is no need to propagate the updated domain to the WEBHOST1 and WEBHOST2 machines because there are no changes to the Oracle HTTP Server instances on those host computers.

Table 12-1 Summary of Tasks Required to Propagate the Domain Changes to Domain Directories and Machines

Table 12-1 (Cont.) Summary of Tasks Required to Propagate the Domain Changes to Domain Directories and Machines

Task	Description	More Information
Pack up the Extended Domain on SOAHOST1	Use the pack command to create a new template JAR file that contains the new Oracle SOA Suite Managed Servers configuration. When you pack up the domain, create a template JAR file called <code>soadomaintemplateExtSOA.jar</code> .	Packing Up the Extended Domain on SOAHOST1
Unpack the Domain in the Managed Servers directory on SOAHOST1	Unpack the template JAR file in the Managed Servers directory on SOAHOST1 local storage.	Unpacking the Domain in the Managed Servers Domain Directory on SOAHOST1
Unpack the Domain on SOAHOST2	Unpack the template JAR file in the Managed Servers directory on the SOAHOST2 local storage.	Unpacking the Domain on SOAHOST2

See Also:

[Packing Up the Extended Domain on SOAHOST1](#)

[Unpacking the Domain in the Managed Servers Domain Directory on SOAHOST1](#)

[Unpacking the Domain on SOAHOST2](#)

12.7.1 Packing Up the Extended Domain on SOAHOST1

Use the following steps to create a template JAR file that contains the domain configuration information:

1. Log in to SOAHOST1 and run the pack command to create a template JAR file as follows:

```
cd ORACLE_COMMON_HOME/common/bin

./pack.sh -managed=true
         -domain=ASERVER_HOME
         -template=full_path/soadomaintemplate.jar
         -template_name=soa_domain_template
```

In this example:

- Replace `ASERVER_HOME` with the actual path to the domain directory you created on the shared storage device.
- Replace `full_path` with the complete path to the directory where you want the template jar file saved.
- `soadomaintemplate.jar` is a sample name for the JAR file you are creating, which will contain the domain configuration files, including the configuration files for the Oracle HTTP Server instances.

- `soa_domain_template` is the name assigned to the domain template file.
2. Make a note of the location of the template JAR file you just created with the `pack` command.

Tip:

For more information about the `pack` and `unpack` commands, see "Overview of the Pack and Unpack Commands" in *Creating Templates and Domains Using the Pack and Unpack Commands*.

12.7.2 Unpacking the Domain in the Managed Servers Domain Directory on SOAHOST1

To copy the updated domain configuration information from the Administration Server domain directory to the Managed Servers domain directory:

1. Log in to SOAHOST1 if you haven't already.
2. If you haven't already, create the recommended directory structure for the Managed Server domain on the SOAHOST1 local storage device.

Use the examples in [File System and Directory Variables Used in This Guide](#) as a guide.

3. Run the `unpack` command to unpack the template in the domain directory onto the local storage, as follows:

```
cd ORACLE_COMMON_HOME/common/bin

./unpack.sh -domain=MSERVER_HOME \
  -template=full_path/soadomaintemplateExtSOA.jar \
  -overwrite_domain=true \
  -app_dir=APPLICATION_HOME
```

In this example:

- Replace `MSERVER_HOME` with the complete path to the domain home to be created on the local storage disk. This is the location where the copy of the domain will be unpacked.
- `soadomaintemplateExtSOA.jar` is the directory path and name of the template you created when you ran the `pack` command to pack up the domain on the shared storage device.
- The `-overwrite_domain=true` argument is necessary when you are unpacking a Managed Server template into an existing domain and existing applications directories.

For any file that is overwritten, a backup copy of the original is created. If any modifications have been applied to the start scripts and EAR files in the Managed Server domain directory, they must be restored after this `unpack` operation.

- Replace `APPLICATION_HOME` with the complete path to the applications directory for the domain on local storage.

Tip:

For more information about the `pack` and `unpack` commands, see "Overview of the Pack and Unpack Commands" in *Creating Templates and Domains Using the Pack and Unpack Commands*.

4. Change directory to the newly created `MSERVER_HOME` directory and verify that the domain configuration files were copied to the correct location on the `SOAHOST1` local storage device.

12.7.3 Unpacking the Domain on SOAHOST2

This procedure assumes you have copied the file that you created earlier in a location that is accessible from both `SOAHOST1` and `SOAHOST2`; such as the `ASERVER_HOME` directory, which is located on the shared storage filer:

1. Log in to `SOAHOST2`.
2. If you haven't already, create the recommended directory structure for the Managed Server domain on the `SOAHOST2` storage device.

Use the examples in [File System and Directory Variables Used in This Guide](#) as a guide.

3. Make sure the `soadomaintemplate.jar` accessible to `SOAHOST2`.

For example, if you are using a separate shared storage volume or partition for `SOAHOST2`, then copy the template to the volume or partition mounted to `SOAHOST2`.

4. Run the `unpack` command to unpack the template in the domain directory onto the local storage, as follows:

```
cd ORACLE_COMMON_HOME/common/bin

./unpack.sh -domain=MSERVER_HOME
            -overwrite_domain=true
            -template=complete_path/soadomaintemplate.jar
            -log_priority=DEBUG
            -log=/tmp/unpack.log
            -app_dir=APPLICATION_HOME
```

In this example:

- Replace `MSERVER_HOME` with the complete path to the domain home to be created on the local storage disk. This is the location where the copy of the domain will be unpacked.
- Replace `complete_path/soadomaintemplate.jar` with the complete path and file name of the domain template jar file that you created when you ran the `pack` command to pack up the domain on the shared storage device.
- Replace `APPLICATION_HOME` with the complete path to the Application directory for the domain on shared storage. For more information, see [File System and Directory Variables Used in This Guide](#).

Tip:

For more information about the pack and unpack commands, see "Overview of the Pack and Unpack Commands" in *Creating Templates and Domains Using the Pack and Unpack Commands*.

5. Change directory to the newly created MSERVER_HOME directory and verify that the domain configuration files were copied to the correct location on the SOAHOST2 local storage device.

12.8 Starting and Validating the WLS_SOA1 Managed Server

Now that you have extended the domain, restarted the Administration Server, and propagated the domain to the other hosts, you can start the newly configured Oracle SOA Suite Managed Servers.

This process involves three tasks as described in the following sections.

See Also:

[Starting the WLS_SOA1 Managed Server](#)

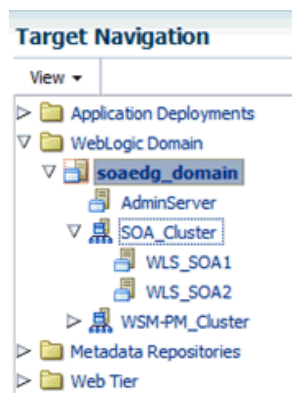
[Adding the SOAAdmin Role to the Administrators Group](#)

[Validating the Managed Server by Logging in to the SOA Infrastructure](#)

12.8.1 Starting the WLS_SOA1 Managed Server

To start the WLS_SOA1 Managed Server:

1. Enter the following URL into a browser to display the Fusion Middleware Control login screen:
`http://ADMINVHN:7001/em`
2. Log in to Fusion Middleware Control using the Administration Server credentials.
3. In the **Target Navigation** pane, expand the domain to view the Managed Servers in the domain.



4. Select only the **WLS_SOA1** Managed Server and click **Start Up** on the Oracle WebLogic Server toolbar.

Note:

SOA Servers depend on the policy access service to be functional. This implies that the WSM-PM Managed Servers in the domain need to be up and running and reachable before the SOA servers are started.

5. When the startup operation is complete, navigate to the Domain home page and verify that the WLS_SOA1 Managed Server is up and running.

12.8.2 Adding the SOAdmin Role to the Administrators Group

Before you validate the Oracle SOA Suite configuration on the WLS_SOA1 Managed Server, add the SOAdmin administration role to the enterprise deployment administration group (SOA Administrators).

To perform this task, refer to [Configuring Roles for Administration of Oracle SOA Suite Products](#).

12.8.3 Validating the Managed Server by Logging in to the SOA Infrastructure

After you add the SOAdmin role to the SOA Administrators group, you can then validate the configuration of the Oracle SOA Suite software on the WLS_SOA1 Managed Server as follows:

1. Use your Web browser to navigate to the following URL:

```
http://SOAHOST1:8001/soa-infra/
```

2. Log in using the enterprise deployment administrator user credentials (weblogic_soa).

You should see a web page with the following title:

```
"Welcome to the Oracle SOA Platform on WebLogic"
```

12.9 Starting and Validating the WLS_SOA2 Managed Server

After you have validated the successful configuration and startup of the WLS_SOA1 Managed Server, you can then start and validate the WLS_SOA2 Managed Server.

To start and validate the WLS_SOA2 Managed Server, use the procedure in [Starting and Validating the WLS_SOA1 Managed Server](#) for WLS_SOA2 Managed Server.

For the validation URL, enter the following URL in your web browser and log in using the enterprise deployment administrator user (weblogic_soa):

```
http://SOAHOST2:8001/soa-infra/
```

12.10 Validating the Location and Creation of the Transaction Logs

After WLS_SOA1 and WLS_SOA2 are up and running, verify that the transaction log directory and transaction logs were created as expected.

Run the following command to verify, based on the steps you performed in [Configuring a Default Persistence Store for Transaction Recovery](#):

```
ORACLE_RUNTIME/SOA_Cluster/tlogs
```

- `_WLS_WLS_SOA1000000.DAT`

- `_WLS_WLS_SOA2000000.DAT`

12.11 Configuring Oracle HTTP Server for the Extended Domain

The following sections describe how to configure the Oracle HTTP Server instances so they route requests for both public and internal URLs to the proper Webcenter Portal clusters in the enterprise topology.

See Also:

[Configuring Oracle HTTP Server for the WLS_SOA Managed Servers](#)

[Configuring the WebLogic Proxy Plug-In](#)

[Validating the Oracle SOA Suite URLs Through the Load Balancer](#)

12.11.1 Configuring Oracle HTTP Server for the WLS_SOA Managed Servers

To configure the Oracle HTTP Server instances in the Web tier so they route requests correctly to the Oracle SOA Suite cluster, use the following procedure to create an additional Oracle HTTP Server configuration file that creates and defines the parameters of the `soa.example.com` virtual server.

This procedure assumes you performed the Oracle HTTP Server configuration tasks described in [Configuring Oracle HTTP Server to Route Requests to the Application Tier](#).

To create the virtual host configuration file so requests are routed properly to the Oracle SOA Suite clusters:

1. Log in to WEBHOST1 and change directory to the configuration directory for the first Oracle HTTP Server instance (OHS_1):

```
cd OHS_DOMAIN_HOME/config/fmwconfig/components/OHS/ohs1/moduleconf/
```

2. Create the `soa_vh.conf` file and add the following directive:

```
<VirtualHost WEBHOST1:7777>
  ServerName https://soa.example.com:443
  ServerAdmin you@your.address
  RewriteEngine On
  RewriteOptions inherit
</VirtualHost>
```

3. Add the following directives inside the `<VirtualHost>` tags:

Note:

The URL entry for `/workflow` is optional. It is for workflow tasks associated with Oracle ADF task forms. The `/workflow` URL itself can be a different value, depending on the form.

```
<Location /soa-infra>
  WLSRequest ON
  WebLogicCluster SOAHOST1:8001,SOAHOST2:8001
  WLProxySSL ON
  WLProxySSLPassThrough ON
</Location>
```

```

# SOA inspection.wsil
<Location /inspection.wsil>
  WLSRequest ON
  WebLogicCluster SOAHOST1:8001,SOAHOST2:8001
  WLProxySSL ON
  WLProxySSLPassThrough ON
</Location>

# Worklist
<Location /integration>
  WLSRequest ON
  WebLogicCluster SOAHOST1:8001,SOAHOST2:8001
  WLProxySSL ON
  WLProxySSLPassThrough ON
</Location>

# UMS prefs
<Location /sdpmessaging/userprefs-ui>
  WLSRequest ON
  WebLogicCluster SOAHOST1:8001,SOAHOST2:8001
  WLProxySSL ON
  WLProxySSLPassThrough ON
</Location>

# Default to-do taskflow
<Location /DefaultToDoTaskFlow>
  WLSRequest ON
  WebLogicCluster SOAHOST1:8001,SOAHOST2:8001
  WLProxySSL ON
  WLProxySSLPassThrough ON
</Location>

# Workflow
<Location /workflow>
  WLSRequest ON
  WebLogicCluster SOAHOST1:8001,SOAHOST2:8001
  WLProxySSL ON
  WLProxySSLPassThrough ON
</Location>

#Required if attachments are added for workflow tasks
<Location /ADFAttachmentHelper>
  WLSRequest ON
  WebLogicCluster SOAHOST1:8001,SOAHOST2:8001
  WLProxySSL ON
  WLProxySSLPassThrough ON
</Location>

# SOA composer application
<Location /soa/composer>
  WLSRequest ON
  WebLogicCluster SOAHOST1:8001,SOAHOST2:8001
  WLProxySSL ON
  WLProxySSLPassThrough ON
</Location>

<Location /frevvo>
  WLSRequest ON
  WebLogicCluster SOAHOST1:8001,SOAHOST2:8001
  WLProxySSL ON
  WLProxySSLPassThrough ON

```

```
</Location>
</VirtualHost>
```

The `soa_vh.conf` file will appear as it does in [Example 12-1](#).

4. Copy the `soa_vh.conf` file to the configuration directory for the second Oracle HTTP Server instance (`ohs2`):

```
OHS_DOMAIN_HOME/config/fmwconfig/components/OHS/ohs2/moduleconf/
```

5. Edit the `soa_vh.conf` and change any references to `WEBHOST1` to `WEBHOST2` in the `<VirtualHost>` directives.
6. Restart the Oracle HTTP servers on `WEBHOST1` and `WEBHOST2`.

Example 12-1 *soa_vh.conf* file

```
<VirtualHost WEBHOST1:7777>
  ServerName https://soa.example.com:443
  ServerAdmin you@your.address
  RewriteEngine On
  RewriteOptions inherit

<Location /soa-infra>
  WLSRequest ON
  WebLogicCluster SOAHOST1:8001,SOAHOST2:8001
  WLProxySSL ON
  WLProxySSLPassThrough ON
</Location>

# SOA inspection.wsil
<Location /inspection.wsil>
  WLSRequest ON
  WebLogicCluster SOAHOST1:8001,SOAHOST2:8001
  WLProxySSL ON
  WLProxySSLPassThrough ON
</Location>

# Worklist
<Location /integration>
  WLSRequest ON
  WebLogicCluster SOAHOST1:8001,SOAHOST2:8001
  WLProxySSL ON
  WLProxySSLPassThrough ON
</Location>

# UMS prefs
<Location /sdpmessaging/userprefs-ui>
  WLSRequest ON
  WebLogicCluster SOAHOST1:8001,SOAHOST2:8001
  WLProxySSL ON
  WLProxySSLPassThrough ON
</Location>

# Default to-do taskflow
<Location /DefaultToDoTaskFlow>
  WLSRequest ON
  WebLogicCluster SSOAHOST1:8001,SOAHOST2:8001
  WLProxySSL ON
  WLProxySSLPassThrough ON
```

```

</Location>

# Workflow
<Location /workflow>
  WLSRequest ON
  WebLogicCluster SOAHOST1:8001,SOAHOST2:8001
  WLProxySSL ON
  WLProxySSLPassThrough ON
</Location>

#Required if attachments are added for workflow tasks
<Location /ADFAttachmentHelper>
  WLSRequest ON
  WebLogicCluster SOAHOST1:8001,SOAHOST2:8001
  WLProxySSL ON
  WLProxySSLPassThrough ON
</Location>

# SOA composer application
<Location /soa/composer>
  WLSRequest ON
  WebLogicCluster SOAHOST1:8001,SOAHOST2:8001
  WLProxySSL ON
  WLProxySSLPassThrough ON
</Location>

<Location /frevvo>
  WLSRequest ON
  WebLogicCluster SOAHOST1:8001,SOAHOST2:8001
  WLProxySSL ON
  WLProxySSLPassThrough ON
</Location>
</VirtualHost>

```

Note:

If internal invocations are going to be used in the system, add the appropriate locations to the soainternal virtual host.

12.11.2 Configuring the WebLogic Proxy Plug-In

Before you can validate that requests are routed correctly through the Oracle HTTP Server instances, you must set the `WebLogic Plug-In Enabled` parameter for the clusters you just configured.

1. Log in to the Oracle WebLogic Server Administration Console.
2. In the **Domain Structure** pane, expand the **Environment** node.
3. Click **Clusters**.
4. Select the cluster to which you want to proxy requests from Oracle HTTP Server.

The **Configuration: General** tab is displayed.

5. Scroll down to the **Advanced** section and expand it.

6. Click **Lock & Edit** in the Change Center.
7. Set **WebLogic Plug-In Enabled** to **yes**.
8. Click **Save** and click **Activate Changes**.
9. Click **Activate Changes** in the Change Center.
10. Restart all Managed Servers in all of the clusters that you modified in this section.

12.11.3 Validating the Oracle SOA Suite URLs Through the Load Balancer

To validate the configuration of the Oracle HTTP Server virtual hosts and to verify that the hardware load balancer can route requests through the Oracle HTTP Server instances to the application tier:

1. Verify that the server status is reported as **Running** in the Administration Console.

If the server is shown as **Starting** or **Resuming**, wait for the server status to change to **Started**. If another status is reported (such as **Admin** or **Failed**), check the server output log files for errors.

2. Verify that you can access these URLs:

- `https://soa.example.com:443/soa-infra`
- `https://soa.example.com:443/integration/worklistapp`
- `https://soa.example.com:443/sdpMessaging/userprefs-ui`
- `https://soa.example.com:443/soa/composer`

3. Verify that Identity Service can be invoked successfully on the application tier by accessing the following load balancer URL:

`https://soa.example.com:443/integration/services/IdentityService/identity?WSDL`

12.12 Post-Configuration Steps for Oracle SOA Suite

After you install and configure Oracle SOA Suite and Oracle Business Process Management, consider the following post-configuration tasks.

See Also:

[Configuring Oracle Adapters for Oracle SOA Suite](#)

[Enabling SSL Communication Between the SOA Servers and the Hardware Load Balancer](#)

[Considerations for sync-async interactions in a SOA cluster](#)

12.12.1 Configuring Oracle Adapters for Oracle SOA Suite

If the Oracle SOA Suite applications you are developing take advantage of any of the Oracle adapters for Oracle SOA Suite, then you should make sure the adapters are configured to work efficiently and securely in the enterprise topology.

See the following topics for more information.

See Also:

[Enabling High Availability for Oracle File and FTP Adapters](#)

[Enabling High Availability for Oracle JMS Adapters](#)

[Enabling High Availability for the Oracle Database Adapter](#)

12.12.1.1 Enabling High Availability for Oracle File and FTP Adapters

If the Oracle SOA Suite applications you are developing or deploying require the Oracle File and FTP Adapters, you must configure the adapters for high availability in the enterprise deployment topology.

Use the following sections to complete this task.

See Also:

[Understanding the Oracle File and FTP Adapter Configuration](#)

[Configuring the Oracle File Adapter in the Administration Console](#)

[Editing the JCA File Within the Composite Application](#)

[Configuring the Oracle FTP Adapter](#)

12.12.1.1.1 Understanding the Oracle File and FTP Adapter Configuration

The Oracle File and FTP adapters enable a BPEL process or an Oracle Mediator to read and write files on private file systems and on remote file systems through the FTP (File Transfer Protocol).

When configured properly, these adapters support high availability for an active-active topology with Oracle BPEL Process Manager and Oracle Mediator service engines for both inbound and outbound operations.

For general information about this task, see "Configuring Oracle File and FTP Adapters" in *Understanding Technology Adapters*. The instructions provided here are specific to the Oracle SOA Suite enterprise deployment.

Note:

The File Adapter picks up a file from the inbound directory, processes it, and then outputs a file to the output directory. Because the File Adapter is non-transactional, files can be processed twice. As a result, it is possible to get duplicate files when there is failover in the RAC backend or in the SOA managed servers.

12.12.1.1.2 Configuring the Oracle File Adapter in the Administration Console

To make the Oracle File Adapter highly available, first modify the Oracle File Adapter deployment descriptor for the connection-instance corresponding to `eis/HAFfileAdapter`.

You can perform this task from the Oracle WebLogic Server console:

1. Navigate to and log into the Oracle WebLogic Server Administration Console.

For example:

`http://ADMINVHN:7001/console`

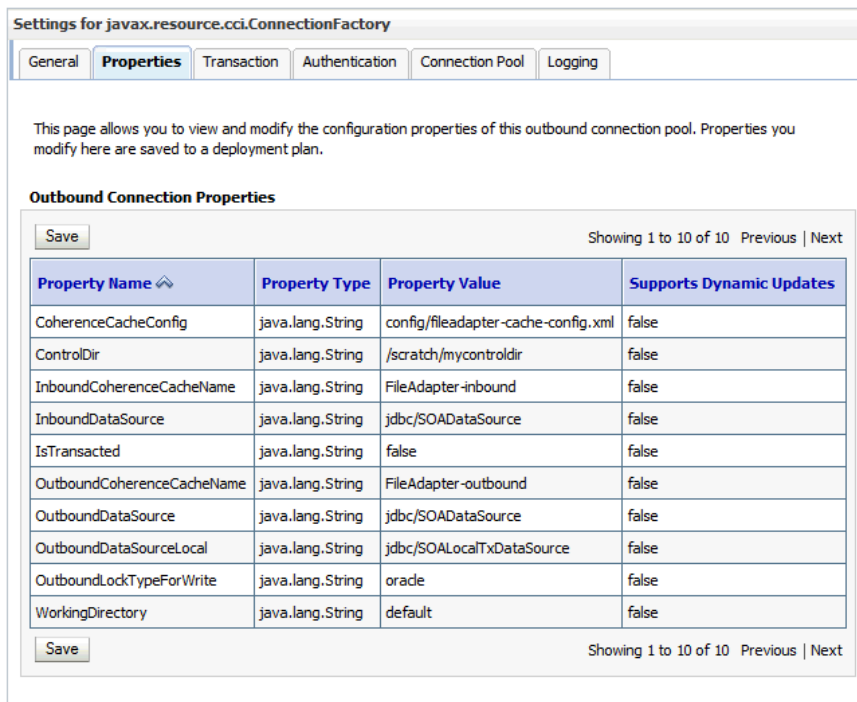
2. In the left pane of the console, click **Deployments**.
3. Locate the **FileAdapter** resource adapter in the Summary of Deployments table.

<input type="checkbox"/>	Name	State	Health	Type
<input type="checkbox"/>	emcoresdkimpl_jar(11.2.0.1.0,12.1.0.0.0)	Active		Library
<input type="checkbox"/>	emcoresdk_jar(11.2.0.1.0,12.1.0.0.0)	Active		Library
<input type="checkbox"/>	emcore_jar	Active		Library
<input type="checkbox"/>	em_core_ppc_pojo_jar	Active		Library
<input type="checkbox"/>	em_sdkcore_ppc_public_pojo_jar	Active		Library
<input type="checkbox"/>	ess.em	Active		Library
<input type="checkbox"/>	ESSAPP	Active		Enterprise Application
<input type="checkbox"/>	EssNativeHostingApp (V1.0)	Active		Enterprise Application
<input type="checkbox"/>	FileAdapter	Active		Resource Adapter
<input type="checkbox"/>	frevvo	Installed		Enterprise Application

4. Click **FileAdapter** to display the Settings for FileAdapter page.
5. Click **Configuration**.
6. Click **Outbound Connection Pools**.
7. Expand **javax.resource.cci.ConnectionFactory** to see the configured connection factories.
8. Click **eis/HAFileAdapter**.

The Outbound Connection Properties for the connection factory appears ([Figure 12-1](#)).

Figure 12-1 Oracle WebLogic Server Console - Settings for `javax.resource.cci.ConnectionFactory`



9. Click **Lock & Edit**.

The property value column becomes editable (you can click on any of the rows in the Property Value column and modify the value).

10. Enter the values as shown in [Table 12-2](#).

Note:

Update controlDir and check other values against the default values as mentioned in [Table 12-2](#).

Table 12-2 Values to Provide for the `javax.resource.cci.ConnectionFactory`

Parameter	Description
controlDir	Enter the directory where you want the control files to be stored. You must set it to a shared location if multiple WebLogic Server instances run in a cluster. Structure the directory for shared storage as follows: <code>ORACLE_RUNTIME/domain_name/cluster_name/fadapter</code>
inboundDataSource	Set the value to <code>jdbc/SOADataSource</code> .
outboundDataSource	Set the value to <code>jdbc/SOADataSource</code> .
outboundDataSourceLocal	Set the value to <code>jdbc/SOALocalTxDataSource</code> . This is the data source where the schemas corresponding to high availability are pre-created.

Table 12-2 (Cont.) Values to Provide for the `javax.resource.cci.ConnectionFactory`

Parameter	Description
<code>outboundLockTypeForWrite</code>	<p>Set the value to <code>oracle</code> if you are using Oracle Database. By default the Oracle File and FTP Adapters use an in-memory mutex to lock outbound write operations. You must choose from the following values for synchronizing write operations:</p> <ul style="list-style-type: none"> <code>memory</code>: The Oracle File and FTP Adapters use an in-memory mutex to synchronize access to the file system. <code>oracle</code>: The adapter uses Oracle Database sequence. <code>db</code>: The adapter uses a pre-created database table (<code>FILEADAPTER_MUTEX</code>) as the locking mechanism. You must use this option only if you are using a schema other than the Oracle Database schema. <code>user-defined</code>: The adapter uses a user-defined mutex. To configure the user-defined mutex, you must implement the mutex interface: <code>"oracle.tip.adapter.file.Mutex"</code> and then configure a new binding-property with the name <code>"oracle.tip.adapter.file.mutex"</code> and value as the fully qualified class name for the mutex for the outbound reference.
<code>workingDirectory</code>	Leave this value as "default".

11. Click **Save** after you update the properties. The Save Deployment Plan page appears.

12. Create `DEPLOY_PLAN_HOME` directory.

```
mkdir -p DEPLOY_PLAN_HOME/soaedg_domain
```

In this example, replace `DEPLOY_PLAN_HOME` with the actual path to the deployment plan directory defined in [File System and Directory Variables Used in This Guide](#).

13. Enter a shared storage location for the deployment plan. The directory structure is as follows:

```
DEPLOY_PLAN_HOME/soaedg_domain/FileAdapterPlan.xml
```

14. Click **Save and Activate** to save and apply your changes.

15. Update the deployment in the console:

- a. Click **Deployments**.
- b. Click **Lock & Edit**.
- c. Select the File Adapter.
- d. Select **Update this application in place with new deployment plan changes (A deployment plan must be specified for this option.)** and select the deployment plan saved in a shared storage location; all servers in the cluster must be able to access the plan.
- e. Click **Finish**.
- f. Activate the changes.

16. Verify that the FileAdapter deployment is activated and running:

- a. In the Administration Console, click **Deployments** in the left pane.
- b. Locate the FileAdapter deployment in the Deployments table.
- c. If it is not in the active state, then select **FileAdapter** under **Summary of Deployments**, Select **Start**, and then **Servicing All Requests**.

12.12.1.1.3 Editing the JCA File Within the Composite Application

After you have configured the FileAdapter deployment in the Administration Console, you can edit the .jca file that is included in the composite applications to be deployed so that they can use the connection factory configured in the previous steps, as shown in [Example 12-2](#).

Note:

The location attribute is set to `eis/HFileAdapter` for the connection factory.

Example 12-2 Example of the File Adapter .JCA File Modifications for an Enterprise Deployment

```
<adapter-config name="FlatStructureOut"
  adapter="File Adapter"
  xmlns="http://platform.integration.oracle/blocks/adapter/fw/metadata">
  <connection-factory location="eis/HFileAdapter" adapterRef="" />
  <endpoint-interaction portType="Write_ptt"
    operation="Write">
    <interaction-spec className="oracle.tip.adapter.file.outbound.FileInteractionSpec">
      <property ../>
      <property ../>
    </interaction-spec>
  </endpoint-interaction>
</adapter-config>
```

12.12.1.1.4 Configuring the Oracle FTP Adapter

If your application requires an FTP Adapter, then repeat the procedures [Configuring the Oracle File Adapter in the Administration Console](#) and [Editing the JCA File Within the Composite Application](#), with the following differences:

- Select the **FtpAdapter** deployment in the list of deployments in the Administration Console.
- Modify the `eis/Ftp/HAFtpAdapter` connection factory instance for the FTPAdapter deployment.
- Enter a shared storage location for the deployment plan. The directory structure is as follows:

```
DEPLOY_PLAN_HOME/soaedg_domain/FtpAdapterPlan.xml
```

- Update the ControlDir property so it points to the following location:

```
ORACLE_RUNTIME/domain_name/cluster_name/ftadapter
```

12.12.1.2 Enabling High Availability for Oracle JMS Adapters

When the Oracle JMS adapter communicates with multiple servers in a cluster, the adapter's connection factory property `FactoryProperties` must list available

servers. If it does not list servers, the connection establishes to only one random server. If that particular server goes down, no further messages are processed.

To verify the adapter's JCA connection factory:

1. Log into your Oracle WebLogic Server Administration Console using the following URL:

```
http://ADMINVHN:7001/console
```

2. Click **Deployments** in the left pane for Domain Structure.
3. Click **JmsAdapter** under **Summary of Deployments** on the right pane.
4. Click the **Configuration** tab.
5. Click the **Outbound Connection Pools** tab and expand `oracle.tip.adapter.jms.IJmsConnectionFactory` to see the configured connection factories.
6. Click the specific instance you are using (for example, `eis/wls/Queue`). The Outbound Connection Properties for the connection factory opens.
7. Click **Lock & Edit**.
8. In the **FactoryProperties** field (click on the corresponding cell under Property value), enter the following, all on one line, separated by semicolons:

```
java.naming.factory.initial=weblogic.jndi.WLInitialContextFactory;
java.naming.provider.url=t3://SOAHOST1:8001,SOAHOST2:8001;
java.naming.security.principal=weblogic;
java.naming.security.credentials=mypassword
```

9. Click **Save** after you update the properties. The Save Deployment Plan page appears.
10. Enter a shared storage location for the deployment plan. The directory structure is as follows:

```
DEPLOY_PLAN_HOME/soaedg_domain/JMSAdapterPlan.xml
```

11. Click **Save and Activate**.

Update the deployment in the console:

1. Click **Deployments**.
2. Click **Lock & Edit**
3. Select the JMS Adapter.
4. Click **Update**.
5. Select **Update this application in place with new deployment plan changes (A deployment plan must be specified for this option.)** and select the deployment plan saved in a shared storage location; all servers in the cluster must be able to access the plan.
6. Click **Finish**.

7. Activate the changes.

12.12.1.3 Enabling High Availability for the Oracle Database Adapter

To ensure High Availability while leveraging the Oracle Database Adapter, the Logical Delete Polling Strategy is used normally as it performs better than a physical delete. However, when you have a clustered environment where multiple nodes are polling for the same data, a single record might get processed more than once. To avoid this problem, Oracle Database Adapter uses a distributed polling technique that uses an Oracle Database feature called skip locking.

If you were using the Logical Delete Polling Strategy approach previously, you can simply remove (in `db.jca`) or clear (Logical Delete Page of wizard) the `MarkReservedValue`, and you automatically get skip locking.

The benefits of using skip locking over a reserved value include:

- Skip locking scales better in a cluster and under load.
- All work is in one transaction (as opposed to update/reserve, then commit, then select in a new transaction), so the risk of facing a non-recoverable situation in a high availability environment is minimized.
- No unique `MarkReservedValue` must be specified. Previously, for this to work you would have to configure a complex variable, such as `R${weblogic.Name-2}-${IP-2}-${instance}`.

If you are using Logical Delete polling, and you set `MarkReservedValue`, skip locking is not used.

For more information, see "Scalability" and "Polling Strategies" in the Oracle Fusion Middleware User's Guide for Technology Adapters.

12.12.2 Enabling SSL Communication Between the SOA Servers and the Hardware Load Balancer

After you extend the domain with Oracle SOA Suite, you should also ensure that the Administration Server and Managed Servers can access the front-end, SSL URL of the hardware load balancer.

This will allow SOA Composite applications and web services to invoke callbacks and other communications with the front-end, secure URL.

For more information, see [Enabling SSL Communication Between the Middle Tier and the Hardware Load Balancer](#).

12.12.3 Considerations for sync-async interactions in a SOA cluster

In a SOA cluster, the following scenarios are not supported:

- Synchronous BPEL process with mid-process receive.
- Synchronous BPEL process calling asynchronous services.
- Callback from synchronous processes.

12.13 Enabling Automatic Service Migration and JDBC Persistent Stores for Oracle SOA Suite

To ensure that Oracle SOA Suite is configured for high availability, configure the Oracle SOA Suite Managed Servers for automatic service migration.

For more information on enabling automatic service migration, see [Configuring Automatic Service Migration in an Enterprise Deployment](#).

For additional high availability, you can also configure your transaction logs store and JMS store in a database. For more information, see [Using JDBC Persistent Stores for TLOGs and JMS in an Enterprise Deployment](#).

Adding Oracle Service Bus to the Topology

The following topics describe the procedures for adding Oracle Service Bus (OSB) to the enterprise deployment topology.

See Also:

[Configuring the Enterprise Deployment](#)

[Variables Used when Adding Oracle Service Bus to the Enterprise Topology](#)

As you perform the tasks in this chapter, you will be referencing the directory variables listed in this section.

[About Configuring Oracle Service in Its Own Domain](#)

When adding Oracle Service Bus to your enterprise topology, you can add it to the existing SOA domain, or you can create a new domain for Oracle Service Bus, separate from the Oracle SOA Suite domain.

[Overview of Adding OSB to the Topology](#)

This section provides an overview of the steps you need to perform to add Oracle Service Bus to your domain.

[Prerequisites for Extending the Domain to Include Oracle Service Bus](#)

Before extending the current domain, ensure that your existing deployment meets the following prerequisites.

[Installing Oracle Service Bus Software](#)

The following topics describe how to install Oracle Service Bus in an enterprise deployment.

[Extending the SOA or Infrastructure Domain to Include Oracle Service Bus](#)

This section provides instructions for extending the existing enterprise deployment SOA domain with the Oracle Service Bus.

[Configuring a Default Persistence Store for Transaction Recovery](#)

Each Managed Server uses a transaction log that stores information about committed transactions that are coordinated by the server and that may not have been completed. Oracle WebLogic Server uses this transaction log for recovery from system crashes or network failures. To leverage the migration capability of the Transaction Recovery Service for the Managed Servers within a cluster, store the transaction log in a location accessible to each Managed Server and its backup server.

[Propagating the Extended Domain to the Domain Directories and Machines](#)

After you have extended the domain with the OSB instances, and you have restarted the Administration Server on SOAHOST1, you must then propagate the domain changes to the domain directories and machines.

[Configuring Oracle HTTP Server for the Oracle Service Bus](#)

To configure the Oracle HTTP Server instances in the Web tier so they route requests correctly to the Oracle Service Bus cluster, use the following procedure to create an additional Oracle HTTP Server configuration file that creates and defines the parameters of the soa.example.com virtual server.

[Configuring the WebLogic Proxy Plug-In](#)

Set the WebLogic Plug-In Enabled parameter for the OSB cluster.

[Validating the Oracle Service Bus URLs Through the Load Balancer](#)

Verify the Oracle Service Bus URLs to ensure that appropriate routing and failover is working from the hardware load balancer to the HTTP Server instances to the Oracle Service Bus components.

[Post-Configuration Tasks for Oracle Service Bus](#)

After you install and configure Oracle Service Bus in the domain, consider the following post-configuration tasks.

[Enabling Automatic Service Migration and JDBC Persistent Stores for Oracle Service Bus](#)

To ensure that Oracle Service Bus is configured for high availability, configure the Oracle Service Bus Managed Servers for automatic service migration for failover and zero data loss.

13.1 Variables Used when Adding Oracle Service Bus to the Enterprise Topology

As you perform the tasks in this chapter, you will be referencing the directory variables listed in this section.

The values for several directory variables are defined in [File System and Directory Variables Used in This Guide](#).

- ORACLE_HOME
- ASERVER_HOME
- MSERVER_HOME
- JAVA_HOME
- OHS_DOMAIN_HOME

In addition, you'll be referencing the following virtual IP (VIP) addresses defined in [Physical and Virtual IP Addresses Required by the Enterprise Topology](#):

- ADMINVHN

Actions in these topics will be performed on the following host computers:

- SOAHOST1
- SOAHOST2

- WEBHOST1
- WEBHOST2

13.2 About Configuring Oracle Service in Its Own Domain

When adding Oracle Service Bus to your enterprise topology, you can add it to the existing SOA domain, or you can create a new domain for Oracle Service Bus, separate from the Oracle SOA Suite domain.

For more information, see [Understanding the Topology Options for Oracle Service Bus](#).

If you decide to configure Oracle Service Bus in a separate domain, then keep in mind the following when you are using the instructions for adding Oracle Service to your topology:

- Ignore any references to the SOA Managed Servers or the SOA Cluster. These elements of the domain will only exist if you are extending a domain that has already been extended with Oracle SOA Suite.
- You must run the Repository Creation Utility (RCU) to create the SOAINFRA schema for the Oracle Service Bus domain. This schema is required by Oracle Service Bus. You must use a unique SOAINFRA schema and schema prefix for the Oracle Service Bus domain.

13.3 Overview of Adding OSB to the Topology

This section provides an overview of the steps you need to perform to add Oracle Service Bus to your domain.

[Table 13-1](#) lists and describes the high-level steps for extending an existing SOA domain or an existing Infrastructure domain for Oracle Service Bus. These steps assume you have already performed the steps required to create an initial Infrastructure domain and then you have extended the domain to include Oracle SOA Suite.

Table 13-1 Steps for Extending a SOA Domain to Include Oracle Service Bus

Step	Description	More Information
Install Oracle Service Bus software	Install OSB software on the target system.	Installing Oracle Service Bus Software
Optionally, install the SOAINFRA schema in a supported database.	OSB requires the SOAINFRA schema for the <code>wlsbjmsrpDataSource</code> data source. If you are planning to run OSB in its own domain, then you must be sure that you have installed a separate SOAINFRA schema for OSB in a supported database. Be sure to use a unique schema for the SOAINFRA schema that will be used by the OSB domain.	Creating the Oracle SOA Suite Database Schemas

Table 13-1 (Cont.) Steps for Extending a SOA Domain to Include Oracle Service Bus

Step	Description	More Information
Optionally, create a new Infrastructure domain.	If you are planning to run OSB in its own domain, then you must first create an Infrastructure domain, so you can extend that domain with OSB.	Creating the Initial Infrastructure Domain for an Enterprise Deployment
Run the Configuration Wizard to Extend the Domain	Extend the SOA or Infrastructure domain to contain Oracle Service Bus components.	Extending the SOA or Infrastructure Domain to Include Oracle Service Bus
Configure a Default Persistence Store for Transaction Recovery	To leverage the migration capability of the Transaction Recovery Service for the servers within a cluster, store the transaction log in a location accessible to a server and its backup servers.	Configuring a Default Persistence Store for Transaction Recovery
Propagate the Domain Configuration to the Managed Server Directory in SOAHOST1 and to SOAHOST2	Oracle Service Bus requires some updates to the WebLogic Server start scripts. Propagate these changes using the pack and unpack commands.	Propagating the Extended Domain to the Domain Directories and Machines
Start the Oracle Service Bus Servers	Oracle Service Bus servers extend an already existing domain. As a result, the Administration Server and respective Node Managers are already running in SOAHOST1 and SOAHOST2.	Starting and Validating the WLS_OSB1 Managed Server
Validate the WLS_OSB Managed Servers	Verify that the server status is reported as Running in the Admin Console and access URLs to verify status of servers.	Starting and Validating the WLS_OSB2 Managed Server
Configuring Oracle HTTP Server for the WLS_OSBn Managed Servers	To enable Oracle HTTP Server to route to Oracle Service Bus console and Oracle Service Bus service, set the WebLogicCluster parameter to the list of nodes in the cluster.	Configuring Oracle HTTP Server for the Oracle Service Bus
Validating Access Through Oracle HTTP Server	Verify that the server status is reported as Running.	Validating the Oracle Service Bus URLs Through the Load Balancer
Enable High Availability for Oracle File and FTP Adapters	Make Oracle File and FTP Adapters highly available for outbound operations using the database mutex locking operation.	Enabling High Availability for Oracle DB_ File and FTP Adapters

Table 13-1 (Cont.) Steps for Extending a SOA Domain to Include Oracle Service Bus

Step	Description	More Information
Backing up the Oracle Service Bus Configuration	To back up the domain configuration for immediate restoration in case of failures in future procedures.	Backing Up the Oracle Service Bus Configuration

13.4 Prerequisites for Extending the Domain to Include Oracle Service Bus

Before extending the current domain, ensure that your existing deployment meets the following prerequisites.

- Back up the installation - If you have not yet backed up the existing Fusion Middleware Home and domain, Oracle recommends backing it up now.
To back up the existing Fusion Middleware Home and domain, see [Performing Backups and Recoveries in the SOA Enterprise Deployments](#).
- Verify that you have installed the Infrastructure and SOA software binaries in an Oracle home on shared storage and they are available from SOAHOST1 and SOAHOST2.
- If Oracle Service Bus is being configured in the same domain as SOA, then the appropriate SOAINFRA schema (used by the wlsbjmsrpDataSource) will be already available. If OSB is being configured in its own domain, then you must run RCU to install the SOAINFRA schema in a supported database, using a different schema prefix than the SOAINFRA schema used by the SOA domain.
- You have already configured Node Manager, Administration Server, (optionally SOA Servers) and WSM Servers as described in previous chapters to run a SOA system. Optionally, you may have already configured Server migration, transaction logs, coherence, and all other configuration steps for the SOA System.
- If you haven't done so already, verify that the system clocks on each host computer are synchronized. You can do this by running the date command as simultaneously as possible on the hosts in each cluster.

Alternatively, there are third-party and open-source utilities you can use for this purpose.

13.5 Installing Oracle Service Bus Software

The following topics describe how to install Oracle Service Bus in an enterprise deployment.

See Also:

[Starting the Oracle Service Bus Installer](#)

[Navigating the OSB Installation Screens](#)

[Installing the Software on the Other Host Computers](#)

[Validating the OSB Installation](#)

13.5.1 Starting the Oracle Service Bus Installer

To start the installation program, perform the following steps.

1. Log in to the target system, SOAHOST1.
2. Go to the directory in which you downloaded the installation program.
3. Set the `path` for the java executable:

```
export JAVA_HOME=JAVA_HOME
export PATH=$JAVA_HOME/bin:$PATH
```

In this example, replace `JAVA_HOME` with the value this variable listed in [File System and Directory Variables Used in This Guide](#) and entered in the *Enterprise Deployment Workbook*.

4. Launch the installation program by entering the following command:

```
java -d64 -jar fmw_12.2.1.0.0_osb.jar
```

When the installation program appears, you are ready to begin the installation.

13.5.2 Navigating the OSB Installation Screens

[Table 13-2](#) provides description of each installation program screen.

Table 13-2 SOA Installation Screens

Screen	Description
Welcome	This screen introduces you to the product installer.
Auto Updates	Use this screen to automatically search My Oracle Support for available patches or automatically search a local directory for patches that you've already downloaded for your organization.
Installation Location	Use this screen to specify the location of your Oracle home directory. If you plan to extend the existing SOA domain, then install the OSB software into the existing Oracle home, where the SOA software has already been installed. If you plan to configure OSB in a separate domain, then install the OSB software in the Infrastructure Oracle home.
Installation Type	Use this screen to select the type of installation and consequently, the products and feature sets you want to install. For this topology, select Service Bus .

Table 13-2 (Cont.) SOA Installation Screens

Screen	Description
Prerequisite Checks	This screen verifies that your system meets the minimum necessary requirements. If there are any warning or error messages, you can refer to one of the following documents in Section 1.4 .
Installation Progress	This screen allows you to see the progress of the installation.
Installation Complete	This screen appears when the installation is complete. Review the information on this screen, then click Finish to dismiss the installer.

13.5.3 Installing the Software on the Other Host Computers

If you have configured a separate shared storage volume or partition for SOAHOST2, then you must also install the software on SOAHOST2. For more information, see [Shared Storage Recommendations When Installing and Configuring an Enterprise Deployment](#).

Note that the location where you install the Oracle home (which contains the software binaries) will vary, depending upon the host. To identify the proper location for you Oracle home directories, refer to the guidelines in [File System and Directory Variables Used in This Guide](#).

13.5.4 Validating the OSB Installation

After you complete the installation, you can verify it by successfully completing the following tasks.

See Also:

[Reviewing the Installation Log Files](#)

[Checking the Directory Structure](#)

[Viewing the Contents of Your Oracle Home](#)

13.5.4.1 Reviewing the Installation Log Files

Review the contents of the installation log files to make sure that no problems were encountered. For a description of the log files and where to find them, see "Understanding Installation Log Files" in *Installing Software with the Oracle Universal Installer*.

13.5.4.2 Checking the Directory Structure

The contents of your installation vary based on the options you selected during the installation.

The addition of Oracle Service Bus adds the following directory and sub-directories:

`ORACLE_HOME/osb/`

`bin`

common
 config
 doc
 financial
 L10N
 lib
 osb
 plugins
 tools

For more information about the directory structure you should see after installation, see "What are the Key Oracle Fusion Middleware Directories?" in *Understanding Oracle Fusion Middleware*.

13.5.4.3 Viewing the Contents of Your Oracle Home

You can also view the contents of your Oracle home using the `viewInventory` script. For more information, see "Viewing the contents of an Oracle home" in *Installing Software with the Oracle Universal Installer*.

13.6 Extending the SOA or Infrastructure Domain to Include Oracle Service Bus

This section provides instructions for extending the existing enterprise deployment SOA domain with the Oracle Service Bus.

Extending the domain involves the following tasks as described in the following sections.

See Also:

[Starting the Configuration Wizard](#)

[Navigating the Configuration Wizard Screens for Oracle Service Bus](#)

13.6.1 Starting the Configuration Wizard

Note:

If you added any customizations directly to the start scripts in the domain, those will be overwritten by the configuration wizard. To customize server startup parameters that apply to all servers in a domain, you can create a file called `setUserOverrides.sh` and configure it, for example, add custom libraries to the WebLogic Server classpath, specify additional java command line options for running the servers, or specify additional environment variables. Any customizations you add to this file are preserved during domain upgrade operations, and are carried over to remote servers when using the **pack** and **unpack** commands.

To begin domain configuration:

1. Shut down the Administration Server to prevent any configuration locks, saves, or activations from occurring during the configuration of the domain.

For more information, see the instructions for shutting down the Administration Server with Node Manager in [Starting the Configuration Wizard on SOAHOST1](#).

2. Navigate to the following directory and start the WebLogic Server Configuration Wizard.

```
ORACLE_HOME/oracle_common/common/bin
./config.sh
```

13.6.2 Navigating the Configuration Wizard Screens for Oracle Service Bus

In this step, you extend the domain created in [Extending the Domain with Oracle SOA Suite](#), and add the Oracle Service Bus software components and Managed Servers.

The steps reflected in this section would be very similar if Oracle Service Bus was extending a domain containing only an Administration Server and a WSM-PM Cluster, but some of the options, libraries and components shown in the screens could vary.

Domain creation and configuration includes the following tasks:

- [Task 1, "Selecting the Domain Type and Domain Home Location"](#)
- [Task 2, "Selecting the Configuration Template"](#)
- [Task 3, "Specifying the Datasource Configuration Type"](#)
- [Task 4, "Specifying JDBC Component Schema Information"](#)
- [Task 5, "Providing the GridLink Oracle RAC Database Connection Details"](#)
- [Task 6, "Testing the JDBC Connections"](#)
- [Task 7, "Selecting Advanced Configuration"](#)
- [Task 8, "Configuring Managed Servers"](#)
- [Task 9, "Configuring a Cluster"](#)
- [Task 10, "Assigning Managed Servers to the Cluster"](#)
- [Task 11, "Configuring Coherence Clusters"](#)
- [Task 12, "Verifying the Existing Machines"](#)
- [Task 13, "Assigning Servers to Machines"](#)
- [Task 14, "Configuring the JMS File Store"](#)
- [Task 15, "Reviewing Your Configuration Specifications and Configuring the Domain"](#)
- [Task 16, "Start the Administration Server"](#)

Task 1 Selecting the Domain Type and Domain Home Location

On the Configuration Type screen, select **Update an existing domain**.

In the **Domain Location** field, select the value of the ASERVER_HOME variable, which represents the complete path to the Administration Server domain home you created in [Creating the Initial Infrastructure Domain for an Enterprise Deployment](#).

For more information about the directory location variables, see [File System and Directory Variables Used in This Guide](#)

Tip:

More information about the other options on this screen can be found in Configuration Type in *Creating WebLogic Domains Using the Configuration Wizard*.

Task 2 Selecting the Configuration Template

On the Templates screen, make sure **Update Domain Using Product Templates** is selected, then select the following template:

Oracle Service Bus - 12.2.1.0 [osb]

Selecting this template automatically selects the following as dependencies:

- WebLogic Advanced Web Services for JAX-RPC Extension - 12.2.1 [oracle_common]
- ODSI XQuery 2004 Components - 12.1.3.0 [oracle_common]

Note:

There is no 12.2.1 template for ODSI. The 12.1.3 template will work for your 12.2.1 configuration.

In addition, the following additional templates should already be selected, because they were used to create the initial domain:

- Oracle Enterprise Manager - 12.2.1.0 [em]
- Oracle WSM Policy Manager - 12.2.1.0 [oracle_common]
- Oracle JRF - 12.2.1.0 [oracle_common]
- WebLogic Coherence Cluster Extension - 12.2.1.0 [wlserver]

Task 3 Specifying the Datasource Configuration Type

All fields are pre-populated, because you already configured the domain to reference the Fusion Middleware schemas that are required for the Infrastructure domain. Verify and ensure that credentials in all the fields are the same that you have provided while configuring Oracle Fusion Middleware Infrastructure.

Click **Get RCU Configuration** after you finish verifying the database connection information. The following output in the Connection Result Log indicates that the operation succeeded:

```
Connecting to the database server...OK
Retrieving schema data from database server...OK
Binding local schema components with retrieved data...OK
Successfully Done.
```


Tip:

More information about the RCU Data option can be found in "Understanding the Service Table Schema" in Creating Schemas with the Repository Creation Utility.

More information about the other options on this screen can be found in "Datasource Defaults" in Creating WebLogic Domains Using the Configuration Wizard.

Task 4 Specifying JDBC Component Schema Information

Select the **OSB JMS Reporting Provider** component schema, click the **Convert to Gridlink** option and click **Next**.

Click **Next**.

Task 5 Providing the GridLink Oracle RAC Database Connection Details

On the GridLink Oracle RAC Component Schema screen, provide the information required to connect to the RAC database and component schemas, as shown in the following table.

Element	Description and Recommended Value
SCAN, Host Name, and Port	Select the SCAN check box. In the Host Name field, enter the Single Client Access Name (SCAN) Address for the Oracle RAC database. In the Port field, enter the SCAN listening port for the database (for example, 1521)
ONS Host and Port	In the ONS Host field, enter the SCAN address for the Oracle RAC database. In the Port field, enter the ONS Remote port (typically, 6200).
Enable Fan	Verify that the Enable Fan check box is selected, so the database can receive and process FAN events.

Task 6 Testing the JDBC Connections

On the Test JDBC Data Sources screen, confirm that all connections were successful.

The connections are tested automatically. The Status column displays the results. If all connections are not successful, click Previous to return to the previous screen and correct your entries.

Click **Next** when all the connections are successful.

Task 7 Selecting Advanced Configuration

On the Select Advanced Configuration screen, select the following:

- Managed Servers, Clusters, and Coherence
- JMS File Store

Click **Next**.

Task 8 Configuring Managed Servers

On the Managed Servers screen, add the required managed servers for Oracle Service Bus.

- Select the automatically created server and click **Rename** to change the name to WLS_OSB1.
- Click **Add** to add another new server and enter WLS_OSB2 as the server name.
- Give servers WLS_OSB1 and WLS_OSB2 the attributes listed in [Table 13-3](#).
- Select **OSB-MGD-SVRS-ONLY** as the server group for the OSB Servers. Deselect **OSB-MGD-SVRS-COMBINED** that is selected by default.

In the end, the configuration for the added servers should match [Table 13-3](#).

Click **Next**.

Name	Listen Address	Listen Port	SSL Listen Port	SSL Enabled	Server Groups
WLS_SOA1	SOAHOST1	8001	n/a	No	SOA-MGD-SVRS-ONLY
WLS_SOA2	SOAHOST2	8001	n/a	No	SOA-MGD-SVRS-ONLY
WLS_WSM 1	SOAHOST1	7010	n/a	No	JRF-MAN-SVR WSMPM-MAN-SVR WSM-CACHE-SVR
WLS_WSM 2	SOAHOST2	7010	n/a	No	JRF-MAN-SVR WSMPM-MAN-SVR WSM-CACHE-SVR
WLS_OSB1	SOAHOST1	8011	n/a	No	OSB-MGD-SVRS-ONLY
WLS_OSB2	SOAHOST2	8011	n/a	No	OSB-MGD-SVRS-ONLY

The WLS_SOA Managed Servers will appear if you are extending an existing Oracle SOA Suite domain with Oracle Service Bus.

Task 9 Configuring a Cluster

In this task, you create a cluster of Managed Servers to which you can target the Oracle SOA Suite software.

You will also set the **Frontend Host** property for the cluster, which ensures that, when necessary, WebLogic Server will redirect Web services callbacks and other redirects to `osb.example.com` on the load balancer rather than the address in the HOST header of each request.

For more information about the `osb.example.com` virtual server address, see [Configuring Virtual Hosts on the Hardware Load Balancer](#).

Use the Clusters screen to create a new cluster:

1. Click the **Add** button.
2. Specify `OSB_Cluster` in the **Cluster Name** field.
3. Specify `osb.example.com` in the **Frontend Host** field.

4. Specify 80 as the **Frontend HTTP Port** and 443 as the **Frontend HTTPS** port.

Cluster Name	Cluster Address	Frontend Host	Frontend HTTP Port	Frontend HTTPS
WSM-PM_Cluster			0	0
SOA_Cluster		soa.example.com	80	443
OSB_Cluster		osb.example.com	80	443

Note:

By default, server instances in a cluster communicate with one another using unicast. If you want to change your cluster communications to use multicast, refer to "Considerations for Choosing Unicast or Multicast" in *Administering Clusters for Oracle WebLogic Server*.

Tip:

More information about the options on this screen can be found in Clusters in *Creating WebLogic Domains Using the Configuration Wizard*.

Task 10 Assigning Managed Servers to the Cluster

On the Assign Servers to Clusters screen, assign servers to clusters as follows:

Note that the WLS_SOA Managed Servers will appear only if you are extending an existing Oracle SOA Suite domain with Oracle Service Bus.

- SOA_Cluster - If you are extending a SOA domain.
 - WLS_SOA1
 - WLS_SOA2
- WSM-PM_Cluster:
 - WLS_WSM1
 - WLS_WSM2
- OSB_Cluster:
 - WLS_OSB1
 - WLS_OSB2

Click **Next**.

Task 11 Configuring Coherence Clusters

Use the Coherence Clusters screen to configure the Coherence cluster that is automatically added to the domain. Leave the port number value at 9991, as it was defined during the initial Infrastructure domain creation.

Task 12 Verifying the Existing Machines

Confirm that the following entries appear:

Name	Node Manager Listen Address
SOAHOST1	SOAHOST1
SOAHOST2	SOAHOST2
ADMINHOST	ADMINVHN

Leave all other fields to their default values.

Click **Next**.

Task 13 Assigning Servers to Machines

On the Assign Servers to Machines screen, assign servers to machines as follows:

- ADMINHOST:
 - AdminServer
- SOAHOST1
 - WLS_SOA1 (if extending a SOA domain)
 - WLS_WSM1
 - WLS_OSB1
- SOAHOST2:
 - WLS_SOA2 (if extending a SOA domain)
 - WLS_WSM2
 - WLS_OSB2

Click **Next**.

Task 14 Configuring the JMS File Store

Configure the JMS File Stores that were just created during the configuration session (Wsee FileStore, UMS JMS FileStore, and FileStore).

On the JMS File Stores screen, assign the following directory for each of the OSB Persistence stores:

`ORACLE_RUNTIME/cluster_name/OSB_Cluster/jms`

In this example, replace `ORACLE_RUNTIME` with the value of the variable for your environment. Replace `OSB_Cluster` with the name you assigned to the OSB cluster.

Ignore the JMS configuration warning that appears.

Task 15 Reviewing Your Configuration Specifications and Configuring the Domain

The Configuration Summary screen contains the detailed configuration information for the domain you are about to create. Review the details of each item on the screen and verify that the information is correct.

Click **Update**.

In the Extending Domain screen, click **Done**.

Task 16 Start the Administration Server

Start the Administration Server to ensure the changes you have made to the domain have been applied.

13.7 Configuring a Default Persistence Store for Transaction Recovery

Each Managed Server uses a transaction log that stores information about committed transactions that are coordinated by the server and that may not have been completed. Oracle WebLogic Server uses this transaction log for recovery from system crashes or network failures. To leverage the migration capability of the Transaction Recovery Service for the Managed Servers within a cluster, store the transaction log in a location accessible to each Managed Server and its backup server.

Note:

To enable migration of the Transaction Recovery Service, specify a location on a persistent storage solution that is available to other servers in the cluster. All Managed Servers in the cluster must be able to access this directory. This directory must also exist before you restart the server.

The recommended location is a dual-ported SCSI disk or on a Storage Area Network (SAN). Note that it is important to set the appropriate replication and backup mechanisms at the storage level to guarantee protection in cases of a storage failure.

This information applies for file-based transaction logs. You can also configure a database-based persistent store for translation logs. For more information, see [Using JDBC Persistent Stores for TLOGs and JMS in an Enterprise Deployment](#).

To set the location for the default persistence stores:

1. Log into the Oracle WebLogic Server Administration Console:

```
ADMINVHN:7001/console
```

2. In the Change Center section, click **Lock & Edit**.

3. For each of the Managed Servers in the cluster:

- a. In the Domain Structure window, expand the **Environment** node, and then click the **Servers** node.

The Summary of Servers page appears.

- b. Click the name of the server (represented as a hyperlink) in **Name** column of the table.

The settings page for the selected server appears and defaults to the Configuration tab.

- c. On the Configuration tab, click the **Services** tab.

- d. In the Default Store section of the page, enter the path to the folder where the default persistent stores will store its data files.

For the enterprise deployment, use the `ORACLE_RUNTIME` directory location. This subdirectory serves as the central, shared location for transaction logs for the cluster. For more information, see [File System and Directory Variables Used in This Guide](#).

For example:

```
ORACLE_RUNTIME/domain_name/cluster_name/tlogs
```

Replace `cluster_name` with the name of the cluster you just created.

e. Click **Save**.

4. Click **Save** and **Activate Changes**.

Note:

You will validate the location and the creation of the transaction logs later in the configuration procedure.

13.8 Propagating the Extended Domain to the Domain Directories and Machines

After you have extended the domain with the OSB instances, and you have restarted the Administration Server on SOAHOST1, you must then propagate the domain changes to the domain directories and machines.

Note that there is no need to propagate the updated domain to the WEBHOST1 and WEBHOST2 machines, because there are no changes to the Oracle HTTP Server instances on those host computers.

Refer to the following sections for more information.

See Also:

[Summary of the Tasks Required to Propagate the Changes to the Other Domain Directories and Machines](#)

[Starting and Validating the WLS_OSB1 Managed Server](#)

[Starting and Validating the WLS_OSB2 Managed Server](#)

[Validating the Location and Creation of the Transaction Logs](#)

[Verifying the Appropriate Targeting for OSB Singleton Services](#)

13.8.1 Summary of the Tasks Required to Propagate the Changes to the Other Domain Directories and Machines

[Table 13-4](#) summarizes the steps required to propagate the changes to all the domain directories and machines.

Table 13-4 Summary of Tasks Required to Propagate the Domain Changes to Domain Directories and Machines

Table 13-4 (Cont.) Summary of Tasks Required to Propagate the Domain Changes to Domain Directories and Machines

Task	Description	More Information
Pack up the Extended Domain on SOAHOST1	Use the Pack command to create a new template jar file that contains the new OSB Servers configuration. When you pack up the domain, create a template jar file called soadomaintemplateExtOSB.jar.	Packing Up the Extended Domain on SOAHOST1
Unpack the Domain in the Managed Servers Directory on SOAHOST1	Unpack the template jar file in the Managed Servers directory on SOAHOST1 local storage.	Unpacking the Domain in the Managed Servers Domain Directory on SOAHOST1
Unpack the Domain on SOAHOST2	Unpack the template jar file in the Managed Servers directory on the SOAHOST2 local storage.	Unpacking the Domain on SOAHOST2

13.8.2 Starting and Validating the WLS_OSB1 Managed Server

After extending the domain, restarting the Administration Server, and propagating the domain to the other hosts, use the following procedure to start the WLS_OSB1 server and validate that it's configured successfully:

See Also:

[Starting the WLS_OSB1 Managed Server](#)

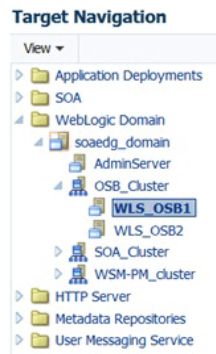
[Adding the MiddlewareAdministrators Role to the Enterprise Deployment Administration Group](#)

[Validating the Managed Server by Logging in to the SOA Infrastructure](#)

13.8.2.1 Starting the WLS_OSB1 Managed Server

1. Enter the following URL into a browser to display the Fusion Middleware Control login screen:

`http://ADMINVHN:7001/em`
2. Log in to Fusion Middleware Control using the Administration Server credentials.
3. In the **Target Navigation** pane, expand the domain to view the Managed Servers in the domain.



4. Select only the **WLS_OSB1** Managed Server and click **Start Up** on the Oracle WebLogic Server toolbar.

Note:

OSB Servers depend on the policy access service to be functional. This implies that the WSM-PM Managed Servers in the domain need to be up and running and reachable before the OSB servers are started.

5. When the startup operation is complete, navigate to the Domain home page and verify that the WLS_OSB1 Managed Server is up and running.

13.8.2.2 Adding the MiddlewareAdministrators Role to the Enterprise Deployment Administration Group

Before you validate the Oracle Service Bus configuration on the WLS_OSB1 Managed Server, add the Oracle Service Bus `MiddlewareAdministrators` administration role to the enterprise deployment administration group (`SOA Administrators`) and add the `IntegrationAdministrators` group in the external LDAP directory.

To perform this task, refer to [Configuring Roles for Administration of Oracle SOA Suite Products](#).

13.8.2.3 Validating the Managed Server by Logging in to the SOA Infrastructure

After you add the `MiddlewareAdministrator` role to the `SOA Administrators` group, you can then validate the configuration of the Oracle Service Bus software on the WLS_OSB1 Managed Server as follows.

1. Use your Web browser to navigate to the following URL:

```
http://SOAHOST1:8011/sbinspection.wsil
```

Replace `SOAHOST1` with the value of this variable in the *Enterprise Deployment Workbook*. For more information, see [Physical and Virtual IP Addresses Required by the Enterprise Topology](#).

2. Log in using the enterprise deployment administration user (`SOA Administrators`).

With the default installation, this should result in the following HTTP response to the Web services call:

```
<ins:inspection xmlns:ins="http://schemas.xmlsoap.org/ws/2001/10/inspection/" />
```


13.8.3 Starting and Validating the WLS_OSB2 Managed Server

Follow similar steps as in the previous section for WLS_OSB2:

1. Log in to Fusion Middleware Control using the Administration Server credentials.
2. In the Target Navigation pane, expand the domain to view the Managed Servers in the domain.
3. Select only the WLS_OSB2 Managed Server and click **Start Up** on the Oracle WebLogic Server tool bar.
4. When the startup operation is complete, navigate to the Domain home page and verify that the WLS_OSB2 Managed Server is up and running. Access the equivalent URLs for the WLS_OSB2:

```
http://SOAHOST2:8011/sbinspection.wsil
```

5. Verify the correct deployment of the Oracle Service Bus console to the Administration Server by accessing the following URL:

```
http://ADMINVHN:7001/servicebus/
```

13.8.4 Validating the Location and Creation of the Transaction Logs

After the WLS_OSB1 and WLS_OSB2 Managed Servers are up and running, verify that the transaction log directory and transaction logs were created as expected, based on the steps you performed in [Configuring a Default Persistence Store for Transaction Recovery](#):

```
ORACLE_RUNTIME/domain_name/OSB_Cluster/tlogs
```

- `_WLS_WLS_OSB1000000.DAT`
- `_WLS_WLS_OSB2000000.DAT`

13.8.5 Verifying the Appropriate Targeting for OSB Singleton Services

Oracle Service Bus uses some services that are singletons and that should run only in one of the WLS servers in the OSB_Cluster.

Verify that the appropriate targeting exists and that the following application is targeted only to WLS_OSB1:

- Service Bus Domain Singleton Marker Application

To do this follow these steps:

1. In a browser, go to the following URL:

```
http://ADMINVHN:7001/console
```

2. Log in as the administrator.
3. In the Domain Structure tree on the left, click **Deployments**.

4. Find the **Service Bus Domain Singleton Marker Application**. Verify that in the **Targets** column in the table only WLS_OSB1 appears as target.

13.9 Configuring Oracle HTTP Server for the Oracle Service Bus

To configure the Oracle HTTP Server instances in the Web tier so they route requests correctly to the Oracle Service Bus cluster, use the following procedure to create an additional Oracle HTTP Server configuration file that creates and defines the parameters of the soa.example.com virtual server.

This procedure assumes you performed the Oracle HTTP Server configuration tasks described in [Configuring Oracle HTTP Server for Administration and Oracle Web Services Manager](#).

Note:

If you configure any HTTP proxy services, you can start the context paths for the proxy services with a common name to facilitate the routing for all the proxy services. For example:

```
/osb/project-name/folder-name/proxy-service-name
```

To set the parameter:

1. Log in to WEBHOST1 and change directory to the configuration directory for the first Oracle HTTP Server instance (ohs1):

```
cd OHS_DOMAIN_HOME/config/fmwconfig/components/OHS/ohs1/moduleconf
```

2. Create a new configuration file, called `osb_vh.conf` file, and add the following `<VirtualHost>` directive to the file:

```
<VirtualHost WEBHOST1:7777>
  ServerName https://osb.example.com:443
  ServerAdmin you@your.address
  RewriteEngine On
  RewriteOptions inherit
</VirtualHost>
```

3. Add the following directives inside the `<VirtualHost>` tags:

```
<Location /sbinspection.wsil>
  WLSRequest ON
  WebLogicCluster SOAHOST1:8011,SOAHOST2:8011
  WLProxySSL ON
  WLProxySSLPassThrough ON
</Location>

<Location /sbresource>
  WLSRequest ON
  WebLogicCluster SOAHOST1:8011,SOAHOST2:8011
  WLProxySSL ON
  WLProxySSLPassThrough ON
</Location>

<Location /osb>
  WLSRequest ON
  WebLogicCluster SOAHOST1:8011,SOAHOST2:8011
```

```

    WLProxySSL ON
    WLProxySSLPassThrough ON
  </Location>

  <Location /alsb>
    WLSRequest ON
    WebLogicCluster SOAHOST1:8011,SOAHOST2:8011
    WLProxySSL ON
    WLProxySSLPassThrough ON
  </Location>

```

The `osb_vh.conf` file will appear as it does in [Example 13-1](#).

4. Add the following entry to the `admin_vh.conf` file within the `<VirtualHost>` tags:

```

<Location /sbconsole >
  WLSRequest ON
  WebLogicHost ADMINVHN
  WeblogicPort 7001
</Location>

<Location /servicebus>
  WLSRequest ON
  WebLogicHost ADMINVHN
  WeblogicPort 7001
</Location>

<Location /lwpfconsole >
  WLSRequest ON
  WebLogicHost ADMINVHN
  WeblogicPort 7001
</Location>

```

The `admin_vh.conf` file will appear as it does in [Example 13-2](#).

5. Copy the `soa_vh.conf` file and the `admin_vh.conf` file to the configuration directory for the second Oracle HTTP Server instance (`ohs2`) on `WEBHOST2`:

```
OHS_DOMAIN_HOME/config/fmwconfig/components/OHS/ohs2/moduleconf
```

6. Edit the `osb_vh.conf` file and change any references to `WEBHOST1` to `WEBHOST2` in the `<VirtualHost>` directives.

7. Restart Oracle HTTP Servers on `WEBHOST1` and `WEBHOST2`.

Example 13-1 `osb_vh.conf` file

```

<VirtualHost WEBHOST1:7777>
  ServerName https://osb.example.com:443
  ServerAdmin you@your.address
  RewriteEngine On
  RewriteOptions inherit

  <Location /sbinspection.wsil >
    WLSRequest ON
    WebLogicCluster SOAHOST1:8011,SOAHOST2:8011
    WLProxySSL ON
    WLProxySSLPassThrough ON
  </Location>

```

```

<Location /sbresource >
  WLSRequest ON
  WebLogicCluster SOAHOST1:8011,SOAHOST2:8011
  WLProxySSL ON
  WLProxySSLPassThrough ON
</Location>

<Location /osb >
  WLSRequest ON
  WebLogicCluster SOAHOST1:8011,SOAHOST2:8011
  WLProxySSL ON
  WLProxySSLPassThrough ON
</Location>

<Location /alsb >
  WLSRequest ON
  WebLogicCluster SOAHOST1:8011,SOAHOST2:8011
  WLProxySSL ON
  WLProxySSLPassThrough ON
</Location>

<Location /lwpfconsole >
  WLSRequest ON
  WebLogicCluster SOAHOST1:8011,SOAHOST2:8011
  WebLogicHost ADMINVHN
  WeblogicPort 7001
</Location>
</VirtualHost>

```

Example 13-2 admin_vh.conf file

The admin URLs should only be accessible via the admin virtual host

```

<VirtualHost WEBHOST1:7777>
  ServerName admin.example.com:80
  ServerAdmin you@your.address
  RewriteEngine On
  RewriteOptions inherit

# Admin Server and EM
<Location /console>
  WLSRequest ON
  WebLogicHost ADMINVHN
  WeblogicPort 7001
</Location>

<Location /consolehelp>
  WLSRequest ON
  WebLogicHost ADMINVHN
  WeblogicPort 7001
</Location>

<Location /em>
  WLSRequest ON
  WebLogicHost ADMINVHN
  WeblogicPort 7001
</Location>

<Location /sbconsole >
  WLSRequest ON

```

```

        WebLogicHost ADMINVHN
        WeblogicPort 7001
    </Location>

    <Location /servicebus>
        WLSRequest ON
        WebLogicHost ADMINVHN
        WeblogicPort 7001
    </Location>
</VirtualHost>

```

13.10 Configuring the WebLogic Proxy Plug-In

Set the WebLogic Plug-In Enabled parameter for the OSB cluster.

1. Log in to the Oracle WebLogic Server Administration Console.
2. In the Domain Structure pane, expand the **Environment** node.
3. Click on **Clusters**.
4. Select the OSB_Cluster cluster to which you want to proxy requests from Oracle HTTP Server.

The **Configuration: General** tab is displayed.

5. Scroll down to the Advanced section and expand it.
6. Click **Lock and Edit**.
7. Set the WebLogic Plug-In Enabled to **yes**.
8. Click **Save and Activate the changes**. Restart the OSB servers for the changes to be effective.

13.11 Validating the Oracle Service Bus URLs Through the Load Balancer

Verify the Oracle Service Bus URLs to ensure that appropriate routing and failover is working from the hardware load balancer to the HTTP Server instances to the Oracle Service Bus components.

To verify the URLs:

1. While WLS_OSB1 is running, stop WLS_OSB2 using the Oracle WebLogic Server Administration Console.
2. Access the following URL and verify the HTTP response as indicated in [Starting and Validating the WLS_OSB2 Managed Server](#):

```
https://osb.example.com/
```

3. Start WLS_OSB2 from the Oracle WebLogic Server Administration Console.
4. Stop WLS_OSB1 from the Oracle WebLogic Server Administration Console.

5. Access the same URL and verify the HTTP response as indicated in section [Starting and Validating the WLS_OSB2 Managed Server](#).

Note:

Since a front end URL has been set for the OSB_Cluster, the requests to the urls result in a re-route to the LBR, but in all cases it should suffice to verify the appropriate mount points and correct failover in Oracle HTTP Server.

6. Verify this URL using your load balancer address:

`http://osb.example.com:80/sbinspection.wsil`

You can also verify `http://osb.example.com:80/servicebus`.

13.12 Post-Configuration Tasks for Oracle Service Bus

After you install and configure Oracle Service Bus in the domain, consider the following post-configuration tasks.

See Also:

[Enabling High Availability for Oracle DB, File and FTP Adapters](#)

[Configuring Specific Oracle Service Bus Services for an Enterprise Deployment](#)

[Enabling SSL Communication Between the Oracle Service Bus Servers and the Hardware Load Balancer](#)

[Backing Up the Oracle Service Bus Configuration](#)

13.12.1 Enabling High Availability for Oracle DB, File and FTP Adapters

Oracle SOA Suite and Oracle Service Bus use the same database, File, and FTP JCA adapters.

You create the required database schemas for these adapters when you use the Oracle Repository Creation Utility before configuring Oracle SOA Suite. The database adapter does not require any configuration at the WebLogic Server resource level.

The required configuration for the other adapters is described in section [Enabling High Availability for Oracle File and FTP Adapters](#).

If you are configuring Oracle Service Bus as an extension of a SOA domain, you do not need to add to the configuration already performed for the adapters.

If you are deploying Oracle Service Bus as an extension to an Oracle Fusion Middleware Infrastructure domain (without Oracle SOA Suite), then you must do the following:

- Run RCU to seed the Oracle Service Bus database with the required adapter schemas (Select the SOA Infrastructure schema in RCU).
- Perform the steps in [Enabling High Availability for Oracle File and FTP Adapters](#).

13.12.2 Configuring Specific Oracle Service Bus Services for an Enterprise Deployment

To use IBM WebSphere MQ Connection resources and the MQ Transport in Oracle Service Bus, you must add the MQ client libraries to the classpath.

One option is to copy the required MQ libraries to the following location in the domain home directory:

`DOMAIN_HOME/lib`

This is also the case for custom assertions and JBoss integration services:

- When using JBoss initial context factory classes, make sure to include the class and any dependent classes in the `DOMAIN_HOME/lib` directory.
- Similarly, for custom assertions, create the required jar file with the assertion and add the jar to the `DOMAIN_HOME/lib` directory.

Further, to use these services in an enterprise deployment, you must add the required libraries to the Administration Server domain home (`ASERVER_HOME/lib`) and the Managed Server domain home (`MSERVER_HOME/lib`).

For more information about configuring and developing services for Oracle Service Bus, see *Developing Services with Oracle Service Bus*.

13.12.3 Enabling SSL Communication Between the Oracle Service Bus Servers and the Hardware Load Balancer

After you extend the domain with Oracle Service Bus, you should also ensure that the Administration Server and Managed Servers can access the front-end, SSL URL of the hardware load balancer.

This will allow Oracle Service Bus Web services and other services to invoke callbacks and other communications with the front-end, secure URL.

For more information, see [Enabling SSL Communication Between the Middle Tier and the Hardware Load Balancer](#).

13.12.4 Backing Up the Oracle Service Bus Configuration

It is an Oracle best practices recommendation to create a backup after successfully extending a domain or at another logical point. Create a backup after verifying that the installation so far is successful. This is a quick backup for the express purpose of immediate restoration in case of problems in later steps.

The backup destination is the local disk. You can discard this backup when the enterprise deployment setup is complete. After the enterprise deployment setup is complete, you can initiate the regular deployment-specific Backup and Recovery process.

For information about backing up your configuration, see [Performing Backups and Recoveries in the SOA Enterprise Deployments](#).

13.13 Enabling Automatic Service Migration and JDBC Persistent Stores for Oracle Service Bus

To ensure that Oracle Service Bus is configured for high availability, configure the Oracle Service Bus Managed Servers for automatic service migration for failover and zero data loss.

For more information on enabling server migration, see [Configuring Automatic Service Migration in an Enterprise Deployment](#).

For additional high availability, you can also configure your transaction logs store and JMS store in a database. For more information, see [Using JDBC Persistent Stores for TLOGs and JMS in an Enterprise Deployment](#).

Extending the Domain with Business Process Management

The following topics describe the procedures for extending the domain to include Business Process Management.

See Also:

[Configuring the Enterprise Deployment](#)

[Variables Used in This Chapter](#)

As you perform the tasks in this chapter, you will be referencing the directory variables listed in this section.

[Prerequisites for Extending the SOA Domain to Include Oracle BPM](#)

Before extending the current domain, ensure that your existing deployment meets the following prerequisites.

[Installing Oracle Business Process Management for an Enterprise Deployment](#)

The following sections describe how to install the Oracle SOA Foundation and Business Process Management software for an enterprise deployment.

[Running the Configuration Wizard on SOAHOST1 to Extend a SOA Domain to Include BPM](#)

Run the Configuration Wizard from the ORACLE_COMMON_HOME directory to extend a domain containing an Administration Server, Oracle Web Services Manager and SOA to support BPM components.

[Propagating the Extended Domain to the Domain Directories and Machines](#)

Oracle BPM Suite requires some updates to the WebLogic Server start scripts. Propagate these changes using the pack and unpack commands.

[Restarting the WLS_SOA Managed Servers with Business Process Management](#)

For configuration changes and start scripts to be effective, you must restart the WLS_SOA server to which BPM has been added.

[Adding the Enterprise Deployment Administration User to the Oracle BPM Administrators Group](#)

Before you validate the Oracle Business Process Management configuration on the Managed Server, add the enterprise deployment administration user (`weblogic_soa`) to the Business Process Management Administrators group in the LDAP directory.

[Configuring Oracle HTTP Server for Oracle Business Process Management](#)

Make the following modifications to the Oracle HTTP Server instance configuration files to ensure that the Oracle HTTP Server instances in the

Web tier can route Oracle Business Process Management requests correctly to the Oracle Business Process Management software.

[Enabling SSL Communication Between Business Process Management Servers and the Hardware Load Balancer](#)

After you extend the domain with Business Process Management, you must ensure that the Administration Server and Managed Servers can access the front-end, SSL URL of the hardware load balancer.

[Validating Access to Business Process Management Through the Hardware Load Balancer](#)

Because the cluster address for the SOA_Cluster has already been set in the previous chapter, the Business Process Management system can be verified only after the Oracle HTTP Server configuration files have been modified to route the Business Process Management context URLs to the WebLogic Servers.

[Configuring BPMJMSModule for the Oracle BPM Cluster](#)

To configure the BPMJMSModule JMS Module, you must modify some default values within the module as listed in this section.

[Backing Up the Oracle BPM Configuration](#)

It is an Oracle best practices recommendation to create a backup after successfully extending a domain or at another logical point. Create a backup after verifying that the installation so far is successful. This is a quick backup for the express purpose of immediate restoration in case of problems in later steps.

[Enabling Automatic Service Migration and JDBC Persistent Stores for Oracle Business Process Management](#)

To ensure that Oracle Business Process Management is configured for high availability, configure the Managed Servers with automatic service migration for failover and zero data loss.

14.1 Variables Used in This Chapter

As you perform the tasks in this chapter, you will be referencing the directory variables listed in this section.

The values for several directory variables are defined in [File System and Directory Variables Used in This Guide](#).

- ORACLE_HOME
- ASERVER_HOME
- MSERVER_HOME
- OHS_DOMAIN_HOME
- JAVA_HOME

In addition, you'll be referencing the following virtual IP (VIP) addresses defined in [Physical and Virtual IP Addresses Required by the Enterprise Topology](#):

- ADMINVHN

Actions in this chapter will be performed on the following host computers:

- SOAHOST1

- SOAHOST2
- WEBHOST1
- WEBHOST2

14.2 Prerequisites for Extending the SOA Domain to Include Oracle BPM

Before extending the current domain, ensure that your existing deployment meets the following prerequisites.

- Back up the installation - If you have not yet backed up the existing Fusion Middleware Home and domain, Oracle recommends backing it up now.
To back up the existing Fusion Middleware Home and domain, see [Performing Backups and Recoveries in the SOA Enterprise Deployments](#).
- Existing WL_HOME and SOA ORACLE_HOME (binaries) are installed in previous chapters on a shared storage and are available from SOAHOST1 and SOAHOST2.
- Node Manager, Admin Server, SOA Servers and WSM Servers exist and have been configured as described in previous chapters to run a SOA system.
- You do not need to run RCU to load additional schemas for BPM. These are part of the SOA repository and are loaded into the DB in the SOA chapter

14.3 Installing Oracle Business Process Management for an Enterprise Deployment

The following sections describe how to install the Oracle SOA Foundation and Business Process Management software for an enterprise deployment.

See Also:

[Starting the Installation Program](#)

[Navigating the Oracle BPM Installation Screens](#)

[Verifying the Installation](#)

14.3.1 Starting the Installation Program

To start the installation program, perform the following steps.

1. Log in to the target system.
2. Make sure certified JDK already exists on your system.
For more information, see [Installing a Supported JDK](#).
3. Go to the directory where you downloaded the installation program.
4. Launch the installation program by running the java executable from the JDK directory on your system, as shown in the examples below.

```
JAVA_HOME/bin/java -d64 -jar fmw_12.2.1.0.0_soa.jar
```

Be sure to replace JDK location in these examples with the actual JDK location on your system.

When the installation program appears, you are ready to begin the installation.

14.3.2 Navigating the Oracle BPM Installation Screens

The installation program displays a series of screens, in the order listed in [Table 14-1](#).

If you need additional help with any of the installation screens, click the screen name.

Table 14-1 Oracle Business Process Management Install Screens

Screen	Description
Installation Inventory Setup	<p>On UNIX operating systems, this screen will appear if this is the first time you are installing any Oracle product on this host. Specify the location where you want to create your central inventory. Make sure that the operating system group name selected on this screen has write permissions to the central inventory location.</p> <p>For more information about the central inventory, see "Understanding the Oracle Central Inventory" in <i>Installing Software with the Oracle Universal Installer</i>.</p>
Auto Updates	<p>Use this screen to automatically search My Oracle Support for available patches or automatically search a local directory for patches that you've already downloaded for your organization.</p>
Installation Location	<p>Use this screen to specify the location of your Oracle home directory. For the Oracle Home, specify <code>/u01/oracle/products/fmwnnnn</code>.</p> <p>For more information about Oracle Fusion Middleware directory structure, see "Selecting Directories for Installation and Configuration" in <i>Planning an Installation of Oracle Fusion Middleware</i>.</p>
Installation Type	<p>Use this screen to select the type of installation and consequently, the products and feature sets you want to install.</p> <ul style="list-style-type: none"> • Select BPM <p>NOTE: The topology in this document does not include the examples, Oracle strongly recommends that you do not install the examples into a production environment.</p>
Prerequisite Checks	<p>This screen verifies that your system meets the minimum necessary requirements.</p> <p>If there are any warning or error messages, you can refer to one of the following documents in Section 1.4.</p>

Table 14-1 (Cont.) Oracle Business Process Management Install Screens

Screen	Description
Installation Summary	<p>Use this screen to verify the installation options you selected. If you want to save these options to a response file, click Save Response File and provide the location and name of the response file. Response files can be used later in a silent installation situation.</p> <p>For more information about silent or command line installation, see "Using the Oracle Universal Installer in Silent Mode" in <i>Installing Software with the Oracle Universal Installer</i>.</p> <p>Click Install to begin the installation.</p>
Installation Progress	<p>This screen allows you to see the progress of the installation.</p> <p>Click Next when the progress bar reaches 100% complete.</p>
Installation Complete	<p>Review the information on this screen, then click Finish to dismiss the installer.</p>

14.3.3 Verifying the Installation

After you complete the installation, you can verify it by successfully completing the following tasks.

See Also:

[Reviewing the Installation Log Files](#)

[Checking the Directory Structure](#)

[Viewing the Contents of Your Oracle Home](#)

14.3.3.1 Reviewing the Installation Log Files

Review the contents of the installation log files to make sure that no problems were encountered. For a description of the log files and where to find them, see "Understanding Installation Log Files" in *Installing Software with the Oracle Universal Installer*.

14.3.3.2 Checking the Directory Structure

The contents of your installation vary based on the options you selected during the installation.

The addition of BPM will add the following directory and sub-directories to the `ORACLE_HOME/soa/bpm` directory:

```
composites
helpsets
lib
modules
```

For more information about the directory structure you should see after installation, see "What are the Key Oracle Fusion Middleware Directories?" in *Understanding Oracle Fusion Middleware. ls -lart*

14.3.3.3 Viewing the Contents of Your Oracle Home

You can also view the contents of your Oracle home using the `viewInventory` script. For more information, see "Viewing the contents of an Oracle home" in *Installing Software with the Oracle Universal Installer*.

14.4 Running the Configuration Wizard on SOAHOST1 to Extend a SOA Domain to Include BPM

Run the Configuration Wizard from the `ORACLE_COMMON_HOME` directory to extend a domain containing an Administration Server, Oracle Web Services Manager and SOA to support BPM components.

See Also:

[Starting the Configuration Wizard](#)

[Navigating the Configuration Wizard Screens to Extend the Domain](#)

14.4.1 Starting the Configuration Wizard

Note:

If you added any customizations directly to the start scripts in the domain, those will be overwritten by the configuration wizard. To customize server startup parameters that apply to all servers in a domain, you can create a file called `setUserOverrides.sh` and configure it to, for example, add custom libraries to the WebLogic Server classpath, specify additional java command line options for running the servers, or specify additional environment variables. Any customizations you add to this file are preserved during domain upgrade operations, and are carried over to remote servers when using the `pack` and `unpack` commands.

To begin domain configuration:

1. Shut down the Administration Server to prevent any configuration locks, saves, or activations from occurring during the configuration of the domain.

For more information, see the instructions for shutting down the Administration Server with Node Manager in [Starting the Configuration Wizard on SOAHOST1](#).

2. Navigate to the following directory and start the WebLogic Server Configuration Wizard.

```
ORACLE_HOME/oracle_common/common/bin
./config.sh
```

14.4.2 Navigating the Configuration Wizard Screens to Extend the Domain

Follow the instructions in this section to extend the domain for BPM.

Note:

This procedure assumes you are extending an existing domain. If your needs do not match the instructions given in the procedure, be sure to make your selections accordingly, or refer to the supporting documentation for additional details.

Domain creation and configuration includes the following tasks:

- [Task 1, "Selecting the Domain Type and Domain Home Location"](#)
- [Task 2, "Selecting the Configuration Template"](#)
- [Task 3, "Specifying the Datasource Configuration Type"](#)
- [Task 4, "Selecting Advanced Configuration"](#)
- [Task 5, "Reviewing your Configuration Specifications and Configuring the Domain"](#)
- [Task 6, "Writing Down Your Domain Home and Administration Server URL"](#)
- [Task 7, "Start the Administration Server"](#)

Task 1 Selecting the Domain Type and Domain Home Location

On the Configuration Type screen, select Update an existing domain.

In the Domain Location field, select the value of the `ASERVER_HOME` variable, which represents the complete path to the Administration Server domain home you created in [Creating the Initial Infrastructure Domain for an Enterprise Deployment](#).

For more information about the directory location variables, see [File System and Directory Variables Used in This Guide](#)

Tip:

More information about the other options on this screen can be found in Configuration Type in [Creating WebLogic Domains Using the Configuration Wizard](#).

Task 2 Selecting the Configuration Template

On the Templates screen, make sure Update Domain Using Product Templates is selected, then select the following templates:

- **Oracle BPM Suite - 12.2.1.0 [soa]**

In addition, the following additional templates should already be selected, because they were used to create the initial domain and extend it to SOA:

- Basic Weblogic Server Domain - 12.2.1.0 [wlserver]
- Oracle SOA Suite 12.2.1.0 [soa]
- Oracle Enterprise Manager - 12.2.1.0 [em]
- Oracle WSM Policy Manager - 12.2.1.0 [oracle_common]

- Oracle JRF - 12.2.1.0 [oracle_common]
- WebLogic Coherence Cluster Extension - 12.2.1.0 [wlserver]

The following templates will also be selected if you had extended the domain previously with Oracle Service Bus:

- WebLogic Advanced Web Services for JAX-RPC Extension - 12.2.1 [oracle_common]
- ODSI XQuery 2004 Components - 12.1.3.0 [oracle_common]

Note:

There is no 12.2.1 template for ODSI. The 12.1.3 template will work for your 12.2.1 configuration.

Tip:

More information about the options on this screen can be found in Templates in Creating WebLogic Domains Using the Configuration Wizard.

Task 3 Specifying the Datasource Configuration Type

All fields are pre-populated, because you already configured the domain to reference the Fusion Middleware schemas that are required for the Infrastructure domain. BPM uses the existing DataSources for SOA and no new Datasources need to be added to the domain.

Note:

Any custom datasources that were created before the extension (like LEASING datasources) will show up before this screen. Check the Datasources row and click **Next**. The test datasource screen will verify its validity. Click **Next**.

Task 4 Selecting Advanced Configuration

To complete domain configuration for the topology, do not select any additional options on the Advanced Configuration screen and Click Next. BPM applications and required artifacts will be targeted automatically to the existing SOA servers

Task 5 Reviewing your Configuration Specifications and Configuring the Domain

The Configuration Summary screen contains the detailed configuration information for the domain you are about to create. Review the details of each item on the screen and verify that the information is correct.

You can go back to any previous screen if you need to make any changes, either by using the Back button or by selecting the screen in the navigation pane.

Domain creation will not begin until you click **Update**.

Tip:

More information about the options on this screen can be found in Configuration Summary in Creating WebLogic Domains Using the Configuration Wizard.

Task 6 Writing Down Your Domain Home and Administration Server URL

The Configuration Success screen will show the following items about the domain you just configured:

- Domain Location
- Administration Server URL

You must make a note of both items as you will need them later; the domain location is needed to access the scripts used to start Administration Server, and the URL is needed to access the Administration Server.

Click **Finish** to dismiss the configuration wizard.

Task 7 Start the Administration Server

Start the Administration Server to ensure the changes you have made to the domain have been applied.

14.5 Propagating the Extended Domain to the Domain Directories and Machines

Oracle BPM Suite requires some updates to the WebLogic Server start scripts. Propagate these changes using the pack and unpack commands.

[Table 14-2](#) summarizes the steps required to propagate the changes to all the domain directories and machines.

Note that there is no need to propagate the updated domain to the WEBHOST1 and WEBHOST2 machines, because there are no changes to the Oracle HTTP Server instances on those host computers.

Table 14-2 Summary of Tasks Required to Propagate the Domain Changes to Domain Directories and Machines

Task	Description	More Information
Pack up the Extended Domain on SOAHOST1	Use the Pack command to create a new template jar file that contains the new Oracle BPM Suite Managed Servers configuration. When you pack up the domain, create a template jar file called soadomaintemplateExtSOABPM.jar.	Packing Up the Extended Domain on SOAHOST1
Unpack the Domain in the Managed Servers Directory on SOAHOST1	Unpack the template jar file in the Managed Servers directory on SOAHOST1 local storage.	Unpacking the Domain in the Managed Servers Domain Directory on SOAHOST1

Table 14-2 (Cont.) Summary of Tasks Required to Propagate the Domain Changes to Domain Directories and Machines

Task	Description	More Information
Unpack the Domain on SOAHOST2	Unpack the template jar file in the Managed Servers directory on the SOAHOST2 local storage.	Unpacking the Domain on SOAHOST2

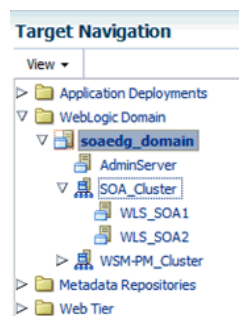
14.6 Restarting the WLS_SOA Managed Servers with Business Process Management

For configuration changes and start scripts to be effective, you must restart the WLS_SOA server to which BPM has been added.

Because BPM extends an already existing SOA system, the Administration Server and respective Node Managers are already running in SOAHOST1 and SOAHOST2.

To restart the WLS_SOA1 Managed Server:

1. Enter the following URL into a browser to display the Fusion Middleware Control login screen:
`http://ADMINVHN:7001/em`
2. Log in to Fusion Middleware Control using the Administration Server credentials.
3. In the **Target Navigation** pane, expand the domain to view the Managed Servers in the domain.



4. Select the **WLS_SOA1** Managed Server and click **Shut Down** on the Oracle WebLogic Server toolbar.

Note:

SOA Servers depend on the policy access service to be functional, so the WSM-PM Managed Servers in the domain need to be up and running and reachable before the SOA servers are started.

5. When the shutdown operation is complete, make sure the server is still selected, and then click **Start Up** in the toolbar.
6. Repeat steps 4 and 5 for the WLS_SOA2 Managed Server.

7. When the startup operation is complete, navigate to the Domain home page and verify that the WLS_SOA1 and WLS_SOA2 Managed Servers are up and running.

14.7 Adding the Enterprise Deployment Administration User to the Oracle BPM Administrators Group

Before you validate the Oracle Business Process Management configuration on the Managed Server, add the enterprise deployment administration user (`weblogic_soa`) to the Business Process Management Administrators group in the LDAP directory.

To perform this task, refer to [Configuring Roles for Administration of Oracle SOA Suite Products](#).

Note that the first time you log in to the Business Process Management Composer or Business Process Management Worklist applications, you must log in as a user that is a member of the Administrators group. After the initial login, any user can be an administration user, as long as they are granted the following roles:

```
OracleBPMComposerRolesApp/BPMComposerAdmin
```

Also, after the first login, any authenticated user should be able to access the Business Process Management applications.

14.8 Configuring Oracle HTTP Server for Oracle Business Process Management

Make the following modifications to the Oracle HTTP Server instance configuration files to ensure that the Oracle HTTP Server instances in the Web tier can route Oracle Business Process Management requests correctly to the Oracle Business Process Management software.

To enable Oracle HTTP Server to route requests to the BPM Composer and BPM Workspace console:

1. Log in to WEBHOST1 and change directory to the configuration directory for the first Oracle HTTP Server instance (`ohs1`):

```
cd OHS_DOMAIN_HOME/config/fmwconfig/components/OHS/ohs1/moduleconf
```

2. Add the following directives inside the `<VirtualHost>` tag in the `soa_vh.conf` file:

```
# BPM
<Location /bpm/composer>
  WLSRequest ON
  WebLogicCluster SOAHOST1:8001,SOAHOST2:8001
  WLProxySSL ON
  WLProxySSLPassThrough ON
</Location>

# BPM
<Location /bpm/workspace>
  WLSRequest ON
  WebLogicCluster SOAHOST1:8001,SOAHOST2:8001
  WLProxySSL ON
  WLProxySSLPassThrough ON
</Location>
```

3. Change directory to the following location so you can update the configuration file for the second Oracle HTTP Server instance (ohs2):

```
cd OHS_DOMAIN_HOME/config/fmwconfig/components/OHS/ohs2/moduleconf
```

4. Open the `soa_vh.conf` file and add the Oracle Business Process Management directives to the `<VirtualHost>` tag.
5. Restart Oracle HTTP Servers on WEBHOST1 and WEBHOST2.

14.9 Enabling SSL Communication Between Business Process Management Servers and the Hardware Load Balancer

After you extend the domain with Business Process Management, you must ensure that the Administration Server and Managed Servers can access the front-end, SSL URL of the hardware load balancer.

This will allow Business Process Management to use Web services to invoke callbacks and other communications with the front-end, secure URL.

If you already configured this communication for the Oracle SOA Suite (WLS_SOA) Managed Servers, then you should be able to validate this configuration using the validation procedures in [Validating Access to Business Process Management Through the Hardware Load Balancer](#).

If you have not yet configured SSL communication with the hardware load balancer, then see [Enabling SSL Communication Between the Middle Tier and the Hardware Load Balancer](#) before you proceed to the validation steps.

14.10 Validating Access to Business Process Management Through the Hardware Load Balancer

Because the cluster address for the SOA_Cluster has already been set in the previous chapter, the Business Process Management system can be verified only after the Oracle HTTP Server configuration files have been modified to route the Business Process Management context URLs to the WebLogic Servers.

Use the following procedure to verify the Business Process Management URLs to ensure that appropriate routing and failover is working from the hardware load balancer to the Oracle HTTP Server instances to the Business Process Management Managed Servers:

1. While the WLS_SOA2 Managed Server is running, stop the WLS_SOA1 Managed Server using the Oracle WebLogic Server Administration Console.

2. Use your Web browser to access the following URLs:

```
https://soa.example.com/bpm/composer/  
https://soa.example.com/bpm/workspace/
```

3. Log in using the `weblogic_soa` administration credentials.

You should see the BPM Composer and BPM Workspace applications ([Figure 14-1](#) and [Figure 14-2](#))

4. Start WLS_SOA1 from the Oracle WebLogic Server Administration Console.

5. Stop WLS_SOA2 from the Oracle WebLogic Server Administration Console.
6. Access the same URLs to verify that the load balancer and Oracle HTTP Server instances can route the requests to the other Managed Server.

Figure 14-1 Oracle BPM Composer

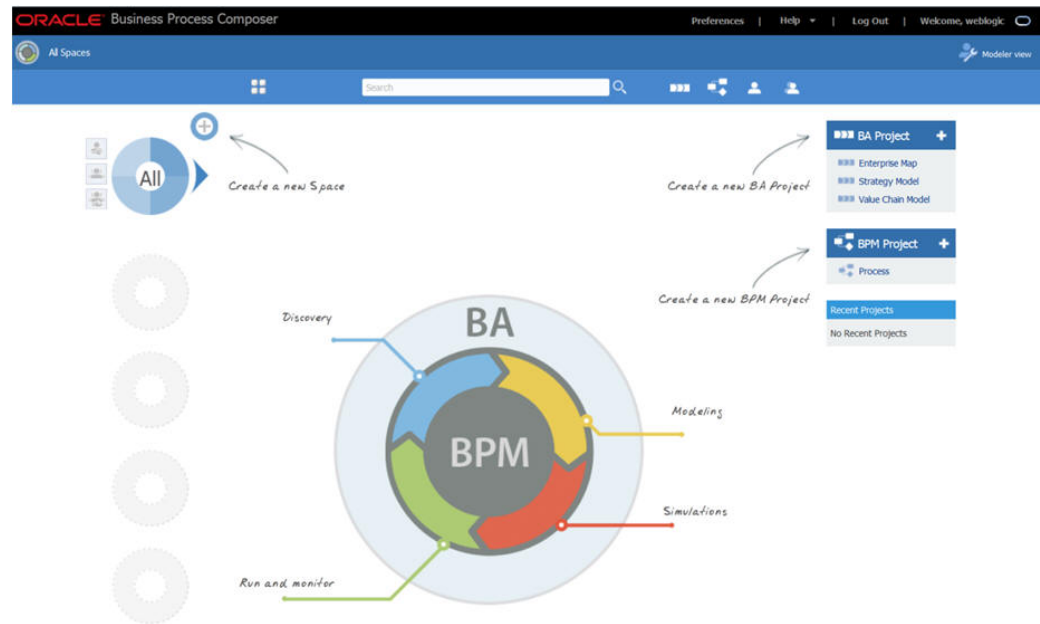
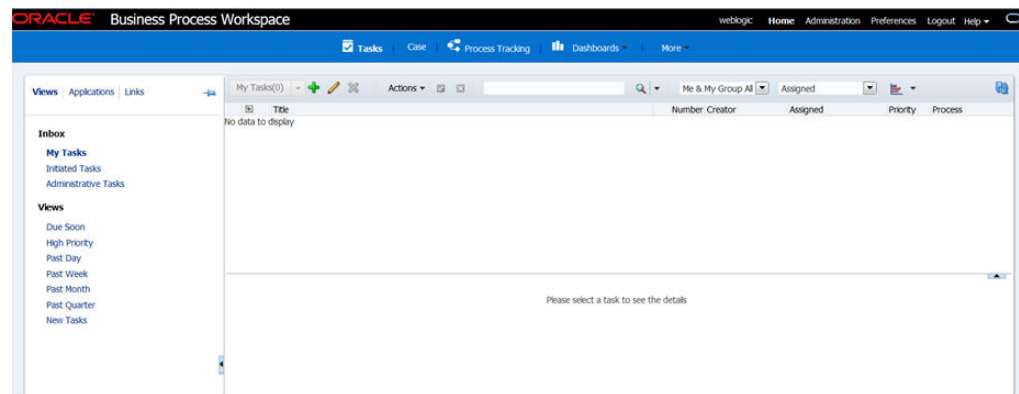


Figure 14-2 Oracle BPM Workspace



14.11 Configuring BPMJMSModule for the Oracle BPM Cluster

To configure the BPMJMSModule JMS Module, you must modify some default values within the module as listed in this section.

The BPMJMSModule JMS Module is deployed automatically when you configure Oracle Business Process Management in a Oracle WebLogic Server domain.

However, when you deploy Oracle Business Process Management Server as part of a Oracle WebLogic Server cluster, you must modify the default values for the quota and redelivery limits for specific JMS resources within the BPMJMSModule JMS module.

Specifically, you must modify the JMS topic resources listed in the following table.

JMS Resource	Property	Description	Recommended Setting
Measurement distributed topic in a cluster configuration: dist_MeasurementTopic ic_auto	Quota	<p>This setting causes issues if a large number of messages are published to the measurement jms topic and the message consumption is relatively slow.</p> <p>When the JMS default threshold of maximum message size is reached, then additional messages cannot be published and any attempt at publishing fails with the following exception:</p> <p>ResourceAllocationException</p>	Set Quota to MeasurementQuota
Measurement distributed topic in a cluster configuration: dist_MeasurementTopic ic_auto	Redelivery Limit	<p>In a cluster configuration, this property is set to -1 by default.</p> <p>This setting causes JMS to retry sending the message until it is successfully acknowledged.</p> <p>If the measurement topic consumers cannot process messages due to a system error that causes the transaction to rollback, then the system can experience performance issues and the filling up of logs with repeated exceptions.</p>	Set the redelivery limit to three (3).
Measurement distributed topic in a cluster configuration: dist_MeasurementTopic ic_auto	Forwarding Policy	<p>A distributed measurement topic in a cluster installation is configured by default with the Forwarding Policy set to Replicated, even though this is not the best performance option for BPM analytics.</p> <p>For more information, see <i>Tuning Oracle Business Process Management in Tuning Performance</i>.</p>	Change the Forwarding Policy to Partitioned .

To modify the BPMJMSModule Resource settings:

1. Log in to the Oracle WebLogic Server Administration Console.
2. Select **Services > Messaging > JMS Modules** in the left navigation pane.
3. Click **BPMJMSModule** in the list of JMS Modules.
4. Select **dist_MeasurementTopic_auto** in the Summary of Resources table.
5. Click the **Thresholds and Quotas** tab.
6. Click **Lock and Edit**.
7. From the **Quota** drop-down menu, select **MeasurementQuota** and click **Save**.
8. Click the **Delivery Failure** tab.
9. Verify that the following fields are set to 3:
 - **Redelivery Delay Override**
 - **Redelivery Limit**
10. Click **Save**.
11. Click the **General** tab.
12. From the **Forwarding Policy** menu, select **Partitioned**.
13. Click **Save**.
14. Click **Activate Changes**.

14.12 Backing Up the Oracle BPM Configuration

It is an Oracle best practices recommendation to create a backup after successfully extending a domain or at another logical point. Create a backup after verifying that the installation so far is successful. This is a quick backup for the express purpose of immediate restoration in case of problems in later steps.

The backup destination is the local disk. You can discard this backup when the enterprise deployment setup is complete. After the enterprise deployment setup is complete, you can initiate the regular deployment-specific Backup and Recovery process.

For information about backing up your configuration, see [Performing Backups and Recoveries in the SOA Enterprise Deployments](#).

14.13 Enabling Automatic Service Migration and JDBC Persistent Stores for Oracle Business Process Management

To ensure that Oracle Business Process Management is configured for high availability, configure the Managed Servers with automatic service migration for failover and zero data loss.

For more information on enabling server migration, see [Configuring Automatic Service Migration in an Enterprise Deployment](#).

Note that if you have already configured automatic service migration for the WLS_SOA Managed Servers, then this step is not necessary.

For additional high availability, you can also configure your transaction logs store and JMS store in a database. For more information, see [Using JDBC Persistent Stores for TLOGs and JMS in an Enterprise Deployment](#).

Extending the Domain with Oracle Enterprise Scheduler

The following topics describe how to extend the enterprise deployment domain with the Oracle Enterprise Scheduler software.

See Also:

[Configuring the Enterprise Deployment](#)

[Variables Used in This Chapter](#)

As you perform the tasks in this chapter, you will be referencing the directory variables listed in this section.

[Overview of Adding Oracle Enterprise Scheduler to a SOA Domain](#)

This section provides an overview of adding Oracle Enterprise Scheduler to a SOA domain.

[Creating the Database Schemas for ESS](#)

Before you can configure an Oracle ESS server, you must install the required schemas on a certified database for use with this release of Oracle Fusion Middleware.

[Extending the SOA Domain to Include Oracle Enterprise Scheduler](#)

This section provides instructions for extending the existing enterprise deployment SOA domain with the Oracle Enterprise Scheduler.

[Configuring a Default Persistence Store for Transaction Recovery](#)

Each Managed Server uses a transaction log that stores information about committed transactions that are coordinated by the server and that may not have been completed. Oracle WebLogic Server uses this transaction log for recovery from system crashes or network failures. To leverage the migration capability of the Transaction Recovery Service for the Managed Servers within a cluster, store the transaction log in a location accessible to each Managed Server and its backup server.

[Propagating the Extended Domain to the Domain Directories and Machines](#)

After you have extended the domain with the ESS instances, and you have restarted the Administration Server on SOAHOST1, you must then propagate the domain changes to the domain directories and machines.

[Adding the ESSAdmin Role to the SOA Administrators Group](#)

Before you validate the Oracle Enterprise Scheduler configuration on the WLS_ESS1 Managed Server, add the ESSAdmin role to the enterprise deployment administration group (SOA Administrators).

[Starting WLS_ESS1 Managed Server](#)

Now that you have extended the domain, restarted the Administration Server, and propagated the domain to the other hosts, you can start the newly configured ESS servers.

[Starting and Validating the WLS_ESS2 Managed Server](#)

Follow the instructions in this section to start and validate the WLS_ESS2 Managed Server.

[Validating the Location and Creation of the Transaction Logs](#)

After WLS_ESS1 and WLS_ESS2 are up and running, verify that the transaction log directory and transaction logs were created as expected.

[Configuring Oracle HTTP Server for the WLS_ESS Managed Servers](#)

Make the following modifications to the Oracle HTTP Server instance configuration files to ensure that the Oracle HTTP Server instances in the Web tier can route Oracle Enterprise Scheduler requests correctly to the WLS_ESS Managed Servers on SOHOST1 and SOAHOST2.

[Configuring the WebLogic Proxy Plug-In](#)

Set the WebLogic Plug-In Enabled parameter for the ESS cluster.

[Validating Access to Oracle Enterprise Scheduler Through the Hardware Load Balancer](#)

Verify the URLs to ensure that appropriate routing and failover is working from the HTTP Server to the Oracle ESS components.

[Backing Up the Oracle Enterprise Scheduler Configuration](#)

It is an Oracle best practices recommendation to create a backup after successfully extending a domain or at another logical point. Create a backup after verifying that the installation so far is successful. This is a quick backup for the express purpose of immediate restoration in case of problems in later steps.

15.1 Variables Used in This Chapter

As you perform the tasks in this chapter, you will be referencing the directory variables listed in this section.

The values for several directory variables are defined in [File System and Directory Variables Used in This Guide](#).

- ORACLE_HOME
- ASERVER_HOME
- MSERVER_HOME
- OHS_DOMAIN_HOME

In addition, you'll be referencing the following virtual IP (VIP) addresses defined in [Physical and Virtual IP Addresses Required by the Enterprise Topology](#):

- ADMINVHN

Actions in this chapter will be performed on the following host computers:

- SOAHOST1
- SOAHOST2
- WEBHOST1
- WEBHOST2

15.2 Overview of Adding Oracle Enterprise Scheduler to a SOA Domain

This section provides an overview of adding Oracle Enterprise Scheduler to a SOA domain.

[Table 15-1](#) lists and describes to high-level steps for extending a SOA domain with Oracle Enterprise Scheduler.

Table 15-1 Steps for Extending a SOA Domain to Include Oracle Enterprise Scheduler

Step	Description	More Information
Create Database Schemas for ESS	Navigate the RCU screens to create the database schemas.	Creating the Database Schemas for ESS
Run the Configuration Wizard to Extend the Domain	Extend the SOA/OSB domain to contain Oracle Enterprise Scheduler components	Extending the SOA Domain to Include Oracle Enterprise Scheduler
Configure a Default Persistence Store for Transaction Recovery	To leverage the migration capability of the Transaction Recovery Service for the servers within a cluster, store the transaction log in a location accessible to a server and its backup servers.	Configuring a Default Persistence Store for Transaction Recovery
Propagate the Domain Configuration to the Managed Server Directory in SOAHOST1 and to SOAHOST2	Oracle Enterprise Scheduler requires some updates to the WebLogic Server start scripts. Propagate these changes using the pack and unpack commands.	Propagating the Extended Domain to the Domain Directories and Machines
Start the Oracle Enterprise Scheduler Servers	Oracle Enterprise Scheduler servers extend an already existing domain. As a result, the Administration Server and respective Node Managers are already running in SOAHOST1 and SOAHOST2.	Starting WLS_ESS1 Managed Server
Validate the WLS_ESS Managed Servers	Verify that the server status is reported as Running in the Admin Console and access URLs to verify status of servers.	Starting and Validating the WLS_ESS2 Managed Server

Table 15-1 (Cont.) Steps for Extending a SOA Domain to Include Oracle Enterprise Scheduler

Step	Description	More Information
Configuring Oracle HTTP Server for the WLS_ESSn Managed Servers	To enable Oracle HTTP Server to route to Oracle Enterprise Scheduler console and service, set the WebLogicCluster parameter to the list of nodes in the cluster.	Configuring Oracle HTTP Server for the WLS_ESS Managed Servers
Validating Access Through Oracle HTTP Server	Verify that the server status is reported as Running.	Validating Access to Oracle Enterprise Scheduler Through the Hardware Load Balancer
Backing up the Oracle Enterprise Scheduler	To back up the domain configuration for immediate restoration in case of failures in future procedures.	Backing Up the Oracle Enterprise Scheduler Configuration

15.3 Creating the Database Schemas for ESS

Before you can configure an Oracle ESS server, you must install the required schemas on a certified database for use with this release of Oracle Fusion Middleware.

Follow the instructions in these sections to install the schemas.

See Also:

[Starting the Repository Creation Utility \(RCU\)](#)

[Navigating the RCU Screens to Create the Enterprise Scheduler Schemas](#)

15.3.1 Starting the Repository Creation Utility (RCU)

To start the Repository Creation Utility (RCU):

1. Navigate to the `ORACLE_HOME/oracle_common/bin` directory on your system.
2. Make sure the `JAVA_HOME` environment variable is set to the location of a certified JDK on your system. The location should be up to but not including the `bin` directory. For example, if your JDK is located in `/u01/oracle/products/jdk`:

On UNIX operating systems:

```
export JAVA_HOME=/u01/oracle/products/jdk
```

3. Start RCU:

On UNIX operating systems:

```
./rcu
```

15.3.2 Navigating the RCU Screens to Create the Enterprise Scheduler Schemas

Schema creation involves the following tasks:

- [Task 1, "Introducing RCU"](#)

- [Task 2, "Selecting a Method of Schema Creation"](#)
- [Task 3, "Providing Database Connection Details"](#)
- [Task 4, "Specifying a Custom Prefix and Selecting Schemas"](#)
- [Task 5, "Specifying Schema Passwords"](#)
- [Task 6, "Specifying Custom Variables"](#)
- [Task 7, "Completing Schema Creation"](#)

Task 1 Introducing RCU

Click **Next**.

Task 2 Selecting a Method of Schema Creation

If you have the necessary permission and privileges to perform DBA activities on your database, select **System Load and Product Load Concurrently**. This procedure assumes that you have the necessary privileges.

If you do not have the necessary permission or privileges to perform DBA activities in the database, you must select **Prepare Scripts for System Load** on this screen. This option will generate a SQL script, which can be provided to your database administrator. See "Understanding System Load and Product Load" in Creating Schemas with the Repository Creation Utility.

Task 3 Providing Database Connection Details

Provide the database connection details for RCU to connect to your database. In the host name enter the scan address of your RAC DB.

Click **Next** to proceed, then click OK on the dialog window confirming that connection to the database was successful.

Task 4 Specifying a Custom Prefix and Selecting Schemas

Select **Select existing prefix** and specify the prefix you used for the original domain creation schemas.

Expand the **Oracle AS Common Schemas** and then select the **Oracle Enterprise Scheduler** in the component list.

The custom prefix is used to logically group these schemas together for use in this domain only; you must create a unique set of schemas for each domain as schema sharing across domains is not supported.

Tip:

For more information about custom prefixes, see "Understanding Custom Prefixes" in Creating Schemas with the Repository Creation Utility.

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

Tip:

You must make a note of the custom prefix you choose to enter here; you will need this later on during the domain creation process.

Click **Next** to proceed, then click **OK** on the dialog window confirming that prerequisite checking for schema creation was successful.

Task 5 Specifying Schema Passwords

Specify how you want to set the schema passwords on your database, then specify and confirm your passwords.

Tip:

You must make a note of the passwords you set on this screen; you will need them later on during the domain creation process.

Task 6 Specifying Custom Variables

Click **Next** in the Default and temporary tablespaces selection (accept defaults) and click in the Confirmation Pop up window warning about tablespaces being created.

Task 7 Completing Schema Creation

Navigate through the remainder of the RCU screens to complete schema creation. When you reach the Completion Summary screen, click **Close** to dismiss RCU.

15.4 Extending the SOA Domain to Include Oracle Enterprise Scheduler

This section provides instructions for extending the existing enterprise deployment SOA domain with the Oracle Enterprise Scheduler.

Extending the domain involves the following tasks.

See Also:

[Starting the Configuration Wizard](#)

[Navigating the Configuration Wizard Screens to Extend the Domain with Oracle Enterprise Scheduler](#)

15.4.1 Starting the Configuration Wizard

Note:

If you added any customizations directly to the start scripts in the domain, those will be overwritten by the configuration wizard. To customize server startup parameters that apply to all servers in a domain, you can create a file called `setUserOverrides.sh` and configure it, for example, add custom libraries to the WebLogic Server classpath, specify additional java command line options for running the servers, or specify additional environment variables. Any customizations you add to this file are preserved during domain upgrade operations, and are carried over to remote servers when using the **pack** and **unpack** commands.

To begin domain configuration:

1. Shut down the Administration Server to prevent any configuration locks, saves, or activations from occurring during the configuration of the domain.

For more information, see the instructions for shutting down the Administration Server with Node Manager in [Starting the Configuration Wizard on SOAHOST1](#).

2. Navigate to the following directory and start the WebLogic Server Configuration Wizard.

```
ORACLE_HOME/oracle_common/common/bin
./config.sh
```

15.4.2 Navigating the Configuration Wizard Screens to Extend the Domain with Oracle Enterprise Scheduler

In this step, you extend the domain created in [Extending the Domain with Oracle SOA Suite](#) to contain Oracle Enterprise Scheduler components.

The steps reflected in this section are very similar to the steps required to extend an Oracle Fusion Middleware Infrastructure domain directly, but some of the options, libraries, and components shown in the screens will vary.

Domain creation and configuration includes the following tasks:

- [Task 1, "Selecting the Domain Type and Domain Home Location"](#)
- [Task 2, "Selecting the Configuration Template"](#)
- [Task 3, "Specifying the Datasource Configuration Type"](#)
- [Task 4, "Specifying JDBC Component Schema Information"](#)
- [Task 5, "Providing the GridLink Oracle RAC Database Connection Details"](#)
- [Task 6, "Selecting Advanced Configuration"](#)
- [Task 7, "Configuring Managed Servers"](#)
- [Task 8, "Configuring a Cluster"](#)
- [Task 9, "Assigning Managed Servers to the Cluster"](#)
- [Task 10, "Configuring Coherence Clusters"](#)
- [Task 11, "Verifying the Existing Machines"](#)
- [Task 12, "Assigning Servers to Machines"](#)
- [Task 13, "Reviewing Your Configuration Specifications and Configuring the Domain"](#)
- [Task 14, "Start the Administration Server"](#)

Task 1 Selecting the Domain Type and Domain Home Location

On the Configuration Type screen, select **Update an existing domain**.

In the **Domain Location** field, select the value of the `ASERVER_HOME` variable, which represents the complete path to the Administration Server domain home you created in [Creating the Initial Infrastructure Domain for an Enterprise Deployment](#).

For more information about the directory location variables, see [File System and Directory Variables Used in This Guide](#)

Tip:

More information about the other options on this screen can be found in Configuration Type in *Creating WebLogic Domains Using the Configuration Wizard*.

Task 2 Selecting the Configuration Template

On the Templates screen, make sure **Update Domain Using Product Templates** is selected, then select the following templates:

Oracle Enterprise Scheduler Service Basic - 12.2.1.0 [oracle_common]

Oracle Enterprise Manager Plugin for ESS - 12.2.1.0 [em]

Click **Next**.

Task 3 Specifying the Datasource Configuration Type

Note:

Any custom datasources that were created before the extension (like LEASING datasources) will show up before this screen. Check the Datasources row and click **Next**. The test datasource screen will verify its validity. Click **Next**.

All fields are pre-populated, because you already configured the domain to reference the Fusion Middleware schemas that are required for the Infrastructure domain. Verify and ensure that credentials in all the fields are the same that you have provided while configuring Oracle Fusion Middleware Infrastructure.

Click **Get RCU Configuration** after you finish verifying the database connection information. The following output in the Connection Result Log indicates that the operation succeeded:

```
Connecting to the database server...OK
Retrieving schema data from database server...OK
Binding local schema components with retrieved data...OK
Successfully Done.
```

Tip:

More information about the RCU Data option can be found in "Understanding the Service Table Schema" in *Creating Schemas with the Repository Creation Utility*.

More information about the other options on this screen can be found in "Datasource Defaults" in *Creating WebLogic Domains Using the Configuration Wizard*.

Task 4 Specifying JDBC Component Schema Information

Select the **ESS Schema** and **ESS MDS Schema**.

When you select the schemas, the fields on the page are activated and the database connection fields are populated automatically.

Click **Convert to GridLink** and click **Next**.

Task 5 Providing the GridLink Oracle RAC Database Connection Details

On the GridLink Oracle RAC Component Schema screen, provide the information required to connect to the RAC database and component schemas, as shown in the following table.

Element	Description and Recommended Value
SCAN, Host Name, and Port	Select the SCAN check box. In the Host Name field, enter the Single Client Access Name (SCAN) Address for the Oracle RAC database. In the Port field, enter the SCAN listening port for the database (for example, 1521)
ONS Host and Port	In the ONS Host field, enter the SCAN address for the Oracle RAC database. In the Port field, enter the ONS Remote port (typically, 6200).
Enable Fan	Verify that the Enable Fan check box is selected, so the database can receive and process FAN events.

Task 6 Selecting Advanced Configuration

On the Select Advanced Configuration screen, select the following:

Managed Servers, Clusters, and Coherence

Click **Next**.

Task 7 Configuring Managed Servers

On the Managed Servers screen, add the required managed servers for Enterprise Scheduler.

- Select the automatically created server and click **Rename** to change the name to WLS_ESS1.
- Click **Add** to add another new server and enter WLS_ESS2 as the server name.
- Give servers WLS_ESS1 and WLS_ESS2 the attributes listed in [Table 15-2](#).

Click **Next**.

Name	Listen Address	Listen Port	SSL Listen Port	SSL Enabled	Server Groups
WLS_SOA1	SOAHOST1	8001	n/a	No	SOA-MGD-SVRS-ONLY
WLS_SOA2	SOAHOST2	8001	n/a	No	SOA-MGD-SVRS-ONLY
WLS_WSM1	SOAHOST1	7010	n/a	No	JRF-MAN-SVR WSMPM-MAN-SVR WSM-CACHE-SVR

Name	Listen Address	Listen Port	SSL Listen Port	SSL Enabled	Server Groups
WLS_WSM2	SOAHOST2	7010	n/a	No	JRF-MAN-SVR WSMPM-MAN-SVR WSM-CACHE-SVR
WLS_OSB1	SOAHOST1	8011	n/a	No	OSB-MGD-SVRS-ONLY
WLS_OSB2	SOAHOST2	8011	n/a	No	OSB-MGD-SVRS-ONLY
WLS_ESS1	SOAHOST1	8021	n/a	No	ESS-MGD-SVRS
WLS_ESS2	SOAHOST2	8021	n/a	No	ESS-MGD-SVRS

The WLS_SOA Managed Servers appear only if you are extending a domain where Oracle SOA Suite has been configured.

The WLS_OSB Managed Servers appear only if you are extending a domain where Oracle Service Bus has been configured.

Task 8 Configuring a Cluster

On the Configure Clusters screen, add the **Enterprise Scheduler** cluster, using the values for each property shown in the following table.

Click **Next**.

Name	Cluster Address	Front end host	Frontend HTTP Port	Frontend HTTPs
SOA_Cluster	Leave it empty.	soa.example.com	80	443
WSM-PM_Cluster	Leave it empty	Leave it empty	Leave it empty	Leave it empty
OSB_Cluster	Leave it empty	osb.example.com	80	443
ESS_Cluster	Leave it empty	soa.example.com	80	443

The SOA_Cluster cluster appears only if you are extending a domain where Oracle SOA Suite has been configured.

The OSB_Cluster cluster appears only if you are extending a domain where Oracle Service Bus has been configured.

Task 9 Assigning Managed Servers to the Cluster

On the Assign Servers to Clusters screen, assign servers to clusters as follows:

- SOA_Cluster - If you are extending a SOA domain.
 - WLS_SOA1
 - WLS_SOA2
- WSM-PM_Cluster:
 - WLS_WSM1

- WLS_WSM2
- OSB_Cluster - If you are extending an OSB domain:
 - WLS_OSB1
 - WLS_OSB2
- ESS_Cluster:
 - WLS_ESS1
 - WLS_ESS2

Click **Next**.

Task 10 Configuring Coherence Clusters

Use the Coherence Clusters screen to configure the Coherence cluster that is automatically added to the domain. Leave the port number value at 9991, as it was defined during the initial Infrastructure domain creation.

Task 11 Verifying the Existing Machines

On the Unix Machines tab, confirm that the following entries appear:

Name	Node Manager Listen Address
SOAHOST1	SOAHOST1
SOAHOST2	SOAHOST2
ADMINHOST	ADMINVHN

Leave all other fields to their default values.

Click **Next**.

Task 12 Assigning Servers to Machines

On the Assign Servers to Machines screen, assign servers to machines as follows:

- ADMINHOST:
 - AdminServer
- SOAHOST1
 - WLS_SOA1 (if extending a SOA domain)
 - WLS_WSM1
 - WLS_OSB1 (if extending an OSB domain)
 - WLS_ESS1
- SOAHOST2
 - WLS_SOA2 (if extending a SOA domain)

- WLS_WSM2
- WLS_OSB2 (if extending an OSB domain)
- WLS_ESS2

Click **Next**.

Task 13 Reviewing Your Configuration Specifications and Configuring the Domain

The Configuration Summary screen contains the detailed configuration information for the domain you are about to create. Review the details of each item on the screen and verify that the information is correct.

Click **Update**.

In the Extending Domain screen, click **Done**.

Task 14 Start the Administration Server

Start the Administration Server to ensure the changes you have made to the domain have been applied.

15.5 Configuring a Default Persistence Store for Transaction Recovery

Each Managed Server uses a transaction log that stores information about committed transactions that are coordinated by the server and that may not have been completed. Oracle WebLogic Server uses this transaction log for recovery from system crashes or network failures. To leverage the migration capability of the Transaction Recovery Service for the Managed Servers within a cluster, store the transaction log in a location accessible to each Managed Server and its backup server.

Note:

To enable migration of the Transaction Recovery Service, specify a location on a persistent storage solution that is available to other servers in the cluster. All Managed Servers in the cluster must be able to access this directory. This directory must also exist before you restart the server.

The recommended location is a dual-ported SCSI disk or on a Storage Area Network (SAN). Note that it is important to set the appropriate replication and backup mechanisms at the storage level to guarantee protection in cases of a storage failure.

This information applies for file-based transaction logs. You can also configure a database-based persistent store for translation logs. For more information, see [Using JDBC Persistent Stores for TLOGs and JMS in an Enterprise Deployment](#).

To set the location for the default persistence stores:

1. Log into the Oracle WebLogic Server Administration Console:
`ADMINVHN:7001/console`
2. In the Change Center section, click **Lock & Edit**.
3. For each of the Managed Servers in the cluster:

- a. In the Domain Structure window, expand the **Environment** node, and then click the **Servers** node.
The Summary of Servers page appears.
 - b. Click the name of the server (represented as a hyperlink) in **Name** column of the table.
The settings page for the selected server appears and defaults to the Configuration tab.
 - c. On the Configuration tab, click the **Services** tab.
 - d. In the Default Store section of the page, enter the path to the folder where the default persistent stores will store its data files.
For the enterprise deployment, use the *ORACLE_RUNTIME* directory location. This subdirectory serves as the central, shared location for transaction logs for the cluster. For more information, see [File System and Directory Variables Used in This Guide](#).
For example:

```
ORACLE_RUNTIME/domain_name/cluster_name/tlogs
```

Replace *cluster_name* with the name of the cluster you just created.
 - e. Click **Save**.
4. Click **Save** and **Activate Changes**.

Note:

You will validate the location and the creation of the transaction logs later in the configuration procedure.

15.6 Propagating the Extended Domain to the Domain Directories and Machines

After you have extended the domain with the ESS instances, and you have restarted the Administration Server on SOAHOST1, you must then propagate the domain changes to the domain directories and machines.

The following table summarizes the steps required to propagate the changes to all the domain directories and machines.

Task	Description	More Information
Pack up the Extended Domain on SOAHOST1	Use the Pack command to create a new template jar file that contains the new ESS Servers configuration. When you pack up the domain, create a template jar file called soadomaintemplateExtESS.jar.	Packing Up the Extended Domain on SOAHOST1

Task	Description	More Information
Unpack the Domain in the Managed Servers Directory on SOAHOST1	Unpack the template jar file in the Managed Servers directory on SOAHOST1 local storage.	Unpacking the Domain in the Managed Servers Domain Directory on SOAHOST1
Unpack the Domain on SOAHOST2	Unpack the template jar file in the Managed Servers directory on the SOAHOST2 local storage.	Unpacking the Domain on SOAHOST2

15.7 Adding the ESSAdmin Role to the SOA Administrators Group

Before you validate the Oracle Enterprise Scheduler configuration on the WLS_ESS1 Managed Server, add the ESSAdmin role to the enterprise deployment administration group (SOA Administrators).

To perform this task, refer to [Configuring Roles for Administration of Oracle SOA Suite Products](#).

15.8 Starting WLS_ESS1 Managed Server

Now that you have extended the domain, restarted the Administration Server, and propagated the domain to the other hosts, you can start the newly configured ESS servers.

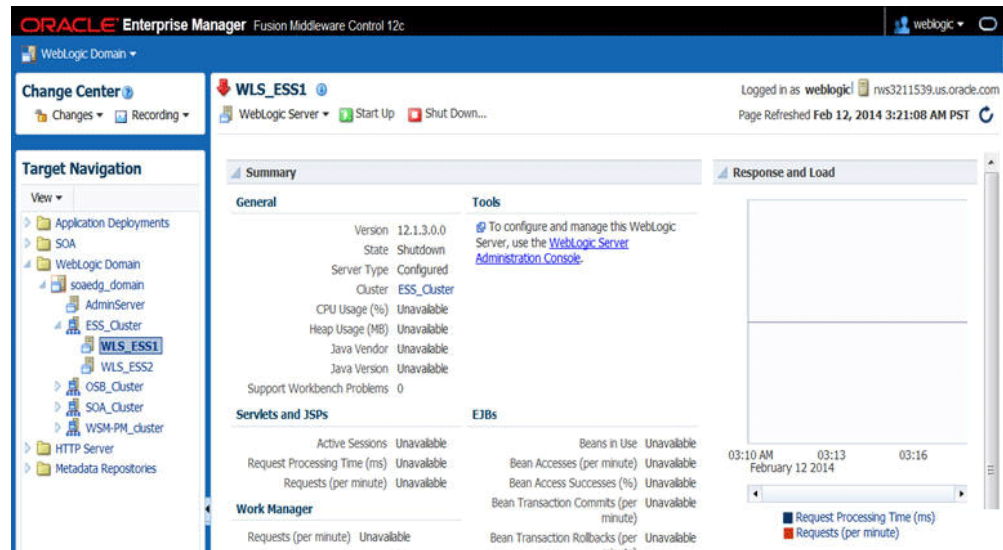
1. Enter the following URL into a browser to display the Fusion Middleware Control login screen:

```
http://ADMINVHN:7001/em
```

In this example:

- Replace ADMINVHN with the host name assigned to the ADMINVHN Virtual IP address in [Identifying and Obtaining Software Downloads for an Enterprise Deployment](#).
 - Port 7001 is the typical port used for the Administration Server console and Fusion Middleware Control. However, you should use the actual URL that was displayed at the end of the Configuration Wizard session when you created the domain.
2. Log in to Fusion Middleware Control using the Administration Server credentials.
 3. In the Target Navigation pane, expand the domain to view the Managed Servers in the domain.

Figure 15-1 WLS_ESS1 Managed Server



4. Select only the WLS_ESS1 Managed Server and click **Start Up** on the Oracle WebLogic Server tool bar.

Note:

SOA Servers depend on the policy access service to be functional. This implies that the WSM-PM servers in the domain need to be reachable before the SOA ones are started

5. When the startup operation is complete, navigate to the Domain home page and verify that the WLS_ESS1 Managed Server is up and running.
6. To verify the ESS software is configured, enter the following URL in the browser:

`http://SOAHOST1:8021/EssHealthCheck`

With the default installation, this should be the HTTP response, as shown in the following image.

ESS - Diagnostic health check service

Click on the **Check Health** button, and then log in using the `wellogic_soa` administration credentials.

The reply should report that Oracle Enterprise Scheduler (ESS) is up and running, as shown in the following image.

ESS - Diagnostic health check service

ESS is up and running

Sample job executed in 12 seconds. Job Id : 1, State: SUCCEEDED

15.9 Starting and Validating the WLS_ESS2 Managed Server

Follow the instructions in this section to start and validate the WLS_ESS2 Managed Server.

Follow similar steps as in the previous section for WLS_ESS2.

1. Log in to Fusion Middleware Control using the Administration Server credentials.
2. In the Target Navigation pane, expand the domain to view the Managed Servers in the domain.
3. Select only the WLS_ESS2 Managed Server and click **Start Up** on the Oracle WebLogic Server tool bar.
4. When the startup operation is complete, navigate to the Domain home page and verify that the WLS_ESS2 Managed Server is up and running, access the equivalent URLs for the WLS_ESS2:

`http://SOAHOST2:8021/EssHealthCheck`

Click the **Check Health** button, and then log in using the `welogic_soa` administration credentials.

The reply should report that Oracle Enterprise Scheduler is up and running, as shown in [#unique_331/unique_331_Connect_42_BABHAHGH](#).

15.10 Validating the Location and Creation of the Transaction Logs

After WLS_ESS1 and WLS_ESS2 are up and running, verify that the transaction log directory and transaction logs were created as expected.

Run the following command to verify, based on the steps you performed in [Configuring a Default Persistence Store for Transaction Recovery](#):

```
ORACLE_RUNTIME/domain_name/ESS_Cluster/tlogs
```

- `_WLS_WLS_ESS1000000.DAT`
- `_WLS_WLS_ESS2000000.DAT`

15.11 Configuring Oracle HTTP Server for the WLS_ESS Managed Servers

Make the following modifications to the Oracle HTTP Server instance configuration files to ensure that the Oracle HTTP Server instances in the Web tier can route Oracle Enterprise Scheduler requests correctly to the WLS_ESS Managed Servers on SOHOST1 and SOAHOST2.

To enable Oracle HTTP Server to route Oracle Enterprise Scheduler requests to the application tier:

1. Log in to SOAHOST1 and change directory to the configuration directory for the first Oracle HTTP Server instance (ohs1):

```
cd OHS_DOMAIN_HOME/config/fmwconfig/components/OHS/ohs1/moduleconf
```

2. Add the following directives inside the <VirtualHost> tag in the soa_vh.conf file:

```
<Location /ess >
  WLSRequest ON
  WebLogicCluster SOAHOST1:8021,SOAHOST2:8021
  WLProxySSL ON
  WLProxySSLPassThrough ON
</Location>
```

```
<Location /EssHealthCheck >
  WLSRequest ON
  WebLogicCluster SOAHOST1:8021,SOAHOST2:8021
  WLProxySSL ON
  WLProxySSLPassThrough ON
</Location>
```

3. Change directory to the following location so you can update the configuration file for the second Oracle HTTP Server instance (ohs2):

```
cd OHS_DOMAIN_HOME/config/fmwconfig/components/OHS/ohs2/moduleconf
```

4. Open the soa_vh.conf file and add the Oracle Business Process Management directives to the <VirtualHost> tag.

5. Restart Oracle HTTP Servers on WEBHOST1 and WEBHOST2.

15.12 Configuring the WebLogic Proxy Plug-In

Set the WebLogic Plug-In Enabled parameter for the ESS cluster.

1. Log in to the Oracle WebLogic Server Administration console.
2. In the Domain Structure pane, expand the **Environment** node.
3. Click on **Clusters**.
4. Select the ESS_Cluster cluster to which you want to proxy requests from Oracle HTTP Server.

The Configuration: General tab is displayed.

5. Scroll down to the Advanced section, expand it.
6. Click **Lock and Edit**.
7. Set the WebLogic Plug-In Enabled to **yes**.
8. Click **Save and Activate the Changes**.

9. Restart the ESS servers for the changes to be effective.

15.13 Validating Access to Oracle Enterprise Scheduler Through the Hardware Load Balancer

Verify the URLs to ensure that appropriate routing and failover is working from the HTTP Server to the Oracle ESS components.

For information on configuring system access through the load balancer, see Section 3.3, "Configuring the Load Balancers."

To verify the URLs:

1. While WLS_ESS1 is running, stop WLS_ESS2 using the Oracle WebLogic Server Administration Console.
2. Access the following URL from your Web browser, and verify the HTTP response as indicated in [Starting and Validating the WLS_ESS2 Managed Server](#):

```
http://soa.example.com/EssHealthCheck
```

3. Start WLS_ESS2 from the Oracle WebLogic Server Administration Console.
4. Stop WLS_ESS1 from the Oracle WebLogic Server Administration Console.

5. Verify these URLs using your load balancer address:

```
https://soa.example.com:443/EssHealthCheck
```

15.14 Backing Up the Oracle Enterprise Scheduler Configuration

It is an Oracle best practices recommendation to create a backup after successfully extending a domain or at another logical point. Create a backup after verifying that the installation so far is successful. This is a quick backup for the express purpose of immediate restoration in case of problems in later steps.

The backup destination is the local disk. You can discard this backup when the enterprise deployment setup is complete. After the enterprise deployment setup is complete, you can initiate the regular deployment-specific Backup and Recovery process.

For information about backing up your configuration, see [Performing Backups and Recoveries in the SOA Enterprise Deployments](#).

Extending the Domain with Business Activity Monitoring

The following topics describe how to extend the domain to include Oracle Business Activity Monitoring.

See Also:

[Configuring the Enterprise Deployment](#)

[Variables Used in This Chapter](#)

As you perform the tasks in this chapter, you will be referencing the directory variables listed in this section.

[Prerequisites When Adding Oracle BAM to the Domain](#)

Before you add Oracle BAM to your existing Oracle SOA Suite domain, consider the following information and prerequisites.

[Special Instructions When Configuring Oracle BAM on Separate Hosts](#)

If you choose to configure Oracle BAM on its own hardware, then you can use the instructions in this chapter, as long as you also consider the information in the following sections.

[Roadmap for Adding Oracle BAM to the Domain](#)

The table in this section lists the high-level steps for extending a SOA domain for Oracle Business Activity Monitoring.

[Extending the SOA Domain to Include Oracle Business Activity Monitoring](#)

This section provides instructions for extending the existing enterprise deployment SOA domain with the Oracle Business Activity Monitoring.

[Configuring a Default Persistence Store for Transaction Recovery](#)

Each Managed Server uses a transaction log that stores information about committed transactions that are coordinated by the server and that may not have been completed. Oracle WebLogic Server uses this transaction log for recovery from system crashes or network failures. To leverage the migration capability of the Transaction Recovery Service for the Managed Servers within a cluster, store the transaction log in a location accessible to each Managed Server and its backup server.

[Propagating the Extended Domain to the Domain Directories and Machines](#)

After you have extended the domain with the BAM instances, and you have restarted the Administration Server on SOAHOST1, you must then propagate the domain changes to the domain directories and machines.

[Adding the Enterprise Deployment Administration User to the Oracle BAM Administration Group](#)

Before you validate the Oracle BAM configuration on the Managed Server, add the enterprise deployment administration user (weblogic_soa) to the BAMAdministrators group.

[Starting WLS_BAM1 Managed Server](#)

After extending the domain, restarting the Administration Server, and propagating the domain to the other hosts, start the newly configured BAM servers.

[Starting and Validating the WLS_BAM2 Managed Server](#)

Follow the instructions in this section to start and validate WLS_BAM2.

[Configuring Oracle HTTP Server for the WLS_BAM Managed Servers](#)

Make the following modifications to the Oracle HTTP Server instance configuration files to ensure that the Oracle HTTP Server instances in the Web tier can route Oracle BAM requests correctly to the Oracle BAM software on the Oracle SOA Suite cluster.

[Configuring the WebLogic Proxy Plug-In](#)

Set the WebLogic Plug-In Enabled parameter for the BAM cluster.

[Validating Access to Oracle BAM Through the Hardware Load Balancer](#)

Verify that Oracle BAM URLs are successfully routing requests from the hardware load balancer to the Oracle HTTP Server instances to the Oracle BAM software in the middle tier.

[Enabling Automatic Service Migration and JDBC Persistent Stores for the Oracle BAM Servers](#)

To ensure that your software is configured for high availability, configure the Oracle Business Activity Monitoring Managed Servers for automatic service migration.

[Backing Up the Oracle BAM Configuration](#)

It is an Oracle best practices recommendation to create a backup after successfully extending a domain or at another logical point. Create a backup after verifying that the installation so far is successful. This is a quick backup for the express purpose of immediate restoration in case of problems in later steps.

16.1 Variables Used in This Chapter

As you perform the tasks in this chapter, you will be referencing the directory variables listed in this section.

The values for several directory variables are defined in [File System and Directory Variables Used in This Guide](#).

- ORACLE_HOME
- ASERVER_HOME
- MSERVER_HOME

- ORACLE_RUNTIME
- OHS_DOMAIN_HOME

In addition, you'll be referencing the following virtual IP (VIP) address defined in [Physical and Virtual IP Addresses Required by the Enterprise Topology](#):

- ADMINVHN

Actions in this chapter will be performed on the following host computers:

- SOAHOST1
- SOAHOST2
- WEBHOST1
- WEBHOST2

16.2 Prerequisites When Adding Oracle BAM to the Domain

Before you add Oracle BAM to your existing Oracle SOA Suite domain, consider the following information and prerequisites.

Note:

If you choose to install Oracle BAM on a separate set of host computers, then in addition to the prerequisites listed here, see [Special Instructions When Configuring Oracle BAM on Separate Hosts](#).

See Also:

[Understanding the Installation Requirements Adding Oracle BAM to the Domain](#)

[Understanding the Database Schema Requirements for Oracle BAM](#)

[Backing Up the Existing Installation](#)

16.2.1 Understanding the Installation Requirements Adding Oracle BAM to the Domain

This chapter assumes you are configuring Oracle Business Activity Monitoring on the same host computers as Oracle SOA Suite, as shown in [Figure 3-2](#).

In the default Oracle SOA Suite and Oracle Business Activity Monitoring topology, you target Oracle BAM to its own Managed Servers and its own cluster, but it shares system resources with the other Oracle SOA Suite products on SOAHOST1 and SOAHOST2. Those system resources include a shared storage device where the Oracle SOA Suite software has been installed in an existing Oracle home directory.

In the default topology, there is no need to install Oracle BAM, because Oracle BAM is included in the Oracle SOA Suite and Oracle Business Process Management distribution and is installed into the Oracle home directories when you install Oracle SOA Suite in [Understanding the SOA Enterprise Deployment Topology](#).

16.2.2 Understanding the Database Schema Requirements for Oracle BAM

The schemas required for Oracle BAM are created in the database when you run the Repository Creation Utility (RCU) to create the required Oracle SOA Suite schemas.

As a result, there is no need to run RCU specifically for Oracle BAM.

If the BAM system is being created without the other Oracle SOA Suite products and the SOA schemas creation has not been performed yet, you must use the RCU installation steps provided in [Extending the Domain with Oracle SOA Suite](#) .

16.2.3 Backing Up the Existing Installation

If you have not yet backed up the existing Fusion Middleware Home and domain, back it up now.

To back up the existing Fusion Middleware Home and domain, see [Performing Backups and Recoveries in the SOA Enterprise Deployments](#).

16.3 Special Instructions When Configuring Oracle BAM on Separate Hosts

If you choose to configure Oracle BAM on its own hardware, then you can use the instructions in this chapter, as long as you also consider the information in the following sections.

For some organizations, it might make sense to install and configure Oracle BAM on separate host computers so the Oracle BAM software can use dedicated hardware resources and can be further isolated from the other Oracle SOA Suite products.

See Also:

[Procuring Additional Host Computers for Oracle BAM](#)

[Installation Requirements When Configuring Oracle BAM on Separate Hosts](#)

[Configuration Wizard Instructions When Configuring Oracle BAM on Separate Hosts](#)

[Propagating the Domain Configuration When Configuring Oracle BAM on Separate Hosts](#)

16.3.1 Procuring Additional Host Computers for Oracle BAM

If you are configuring Oracle BAM on its own set of host computers, you must procure the additional hardware and be sure it meets the system requirements described in [Host Computer Hardware Requirements](#) and [Operating System Requirements for the Enterprise Deployment Topology](#).

You should also add the required entries to the Enterprise Deployment Workbook, as described in [Using the Enterprise Deployment Workbook](#) . For the purposes of this guide, you can refer to these host computers as BAMHOST1 and BAMHOST2.

16.3.2 Installation Requirements When Configuring Oracle BAM on Separate Hosts

If you are configuring Oracle BAM on its own set of host computers, then you should follow the same shared storage strategy you are following for the host computers where the other Oracle SOA Suite products are installed.

Note:

The Oracle home used by BAMHOST1 and BAMHOST2 must contain the exact set of software binaries used by the SOAHOST1 and SOAHOST2 hosts in the domain; otherwise, unpredictable behavior in the execution of the binaries may occur.

Depending on your shared storage strategy, one of the following sections apply if you are using separate host hardware for the Oracle BAM software:

See Also:

[Installation Requirements When Using a Separate Volume or Partition](#)

[Installation Requirements When Using a Shared Oracle Home](#)

16.3.2.1 Installation Requirements When Using a Separate Volume or Partition

If BAMHOST1 and BAMHOST2 are using separate shared storage volumes or partitions, then you must install the Infrastructure and optionally Oracle SOA Suite on those hosts. For more information, see [Shared Storage Recommendations When Installing and Configuring an Enterprise Deployment](#).

Note that the location where you install the Oracle home (which contains the software binaries) will vary, depending upon the host. To identify the proper location for you Oracle home directories, refer to the guidelines in [File System and Directory Variables Used in This Guide](#).

To install the software on BAMHOST1 and BAMHOST2, log in to each host, and perform the following tasks:

- Use the instructions in [Installing the Oracle Fusion Middleware Infrastructure in Preparation for an Enterprise Deployment](#) to create the Oracle home on the appropriate storage device and install Oracle Fusion Middleware Infrastructure.
- Optionally use the instructions in [Installing Oracle SOA Suite for an Enterprise Deployment](#) to install the Oracle SOA Suite software.

16.3.2.2 Installation Requirements When Using a Shared Oracle Home

If BAMHOST1 and BAMHOST2 are using an existing volume or partition where the Oracle Fusion Middleware Infrastructure or Oracle SOA Suite are already installed, then you must mount the volumes appropriately to BAMHOST1 and BAMHOST2. For more information, see [Mounting the Required Shared File Systems on Each Host](#). Ensure that BAMHOST1 and BAMHOST2 have access to this Oracle home, just like the rest of the hosts in the domain.

This is the preferred method of using shared storage for the enterprise deployment. For more information, see [Shared Storage Recommendations When Installing and Configuring an Enterprise Deployment](#).

After you have mounted an existing volume or partition that contains an existing Oracle home, then you should attach the Oracle home to the local Oracle Inventory on BAMHOST1 or BAMHOST2.

To attach an Oracle home in shared storage to the local Oracle Inventory, use the following command on the BAMHOSTs:

```
cd ORACLE_HOME/oui/bin/attachHome.sh
```

```
./attachHome.sh -jreLoc JAVA_HOME
```

The pack and unpack utilities is used to bootstrap the domain configuration for the WLS_BAM1 and WLS_BAM2 servers. As a result, if you have mounted an existing Oracle home with the required software already installed, then you do not need to install any software in these two hosts.

16.3.3 Configuration Wizard Instructions When Configuring Oracle BAM on Separate Hosts

If you are configuring Oracle BAM on separate host computers, then the instructions in this chapter for configuring the domain with the Configuration Wizard are slightly different.

Specifically, be sure to create additional Oracle WebLogic Server machines for BAMHOST1 and BAMHOST2, and then target the WLS_BAM1 and WLS_BAM2 Managed Servers to those machines, rather than to SOAHOST1 and SOAHOST2.

For more information, see [Task 1, "Verifying the Existing Machines"](#) and [Task 1, "Assigning Servers to Machines"](#).

16.3.4 Propagating the Domain Configuration When Configuring Oracle BAM on Separate Hosts

If you are configuring Oracle BAM on separate host computers, then the instructions in this chapter for propagating the domain to the other domain directories must be modified.

Specifically, in addition to propagating the domain to the Managed Server domain directories on SOAHOST1 and SOAHOST2, you must also unpack the domain in the local Managed Server directories for BAMHOST1 and BAMHOST2.

Note that this means you must start the Node Manager software on each BAMHOST computer before you can remotely start the WLS_BAM Managed Servers on these hosts.

16.4 Roadmap for Adding Oracle BAM to the Domain

The table in this section lists the high-level steps for extending a SOA domain for Oracle Business Activity Monitoring.

Step	Description	More Information
Run the Configuration Wizard to Extend the Domain in the Administration Server domain home	Extend the SOA domain to contain Oracle BAM components	Extending the SOA Domain to Include Oracle Business Activity Monitoring
Configure a Default Persistence Store for Transaction Recovery	To leverage the migration capability of the Transaction Recovery Service for the servers within a cluster, store the transaction log in a location accessible to a server and its backup servers.	Configuring a Default Persistence Store for Transaction Recovery

Step	Description	More Information
Propagate the Domain Configuration to the Managed Server domain directories	Oracle BAM requires some updates to the WebLogic Server start scripts. Propagate these changes using the pack and unpack commands.	Propagating the Extended Domain to the Domain Directories and Machines
Add the SOA Administrator role to the Oracle BAM Administration Group	This step allows you to use one set of credentials to access the various product-specific management utilities.	Adding the Enterprise Deployment Administration User to the Oracle BAM Administration Group
Start the Oracle BAM Servers	Oracle BAM servers extend an already existing domain. As a result, the Administration Server and respective Node Managers are already running in SOAHOST1 and SOAHOST2.	Starting WLS_BAM1 Managed Server
Validate the WLS_BAM Managed Servers	Verify that the server status is reported as Running in the Admin Console and access URLs to verify status of servers.	Starting and Validating the WLS_BAM2 Managed Server
Configuring Oracle HTTP Server for the WLS_BAMn Managed Servers	To enable Oracle HTTP Server to route to Oracle BAM, add the required directives to the Oracle HTTP Server configuration files, and set the WebLogicCluster parameter to the list of nodes in the cluster.	Configuring Oracle HTTP Server for the WLS_BAM Managed Servers
Configure the WebLogic Server Proxy Plugin	Enable the WebLogic Server Proxy Plugin for Oracle BAM.	Configuring the WebLogic Proxy Plug-In
Validating Access Through Oracle HTTP Server	Verify that the server status is reported as Running.	Validating Access to Oracle BAM Through the Hardware Load Balancer
Configure Automatic Service Migration for the Oracle BAM Servers	Service migration ensures that key pinned services can be migrated automatically to another Managed Server in the cluster if one of the Managed Servers or host computers fails. For more information about service migration, see Using Whole Server Migration and Service Migration in an Enterprise Deployment .	Configuring Automatic Service Migration for the Oracle BAM Servers
Backing up the Oracle BAM Configuration	To back up the domain configuration for immediate restoration in case of failures in future procedures.	Backing Up the Oracle BAM Configuration

16.5 Extending the SOA Domain to Include Oracle Business Activity Monitoring

This section provides instructions for extending the existing enterprise deployment SOA domain with the Oracle Business Activity Monitoring.

Extending the domain involves the following tasks.

See Also:

[Starting the Configuration Wizard](#)

[Navigating the Configuration Wizard Screens for Oracle BAM](#)

16.5.1 Starting the Configuration Wizard

Note:

If you added any customizations directly to the start scripts in the domain, those will be overwritten by the configuration wizard. To customize server startup parameters that apply to all servers in a domain, you can create a file called `setUserOverrides.sh` and configure it, for example, add custom libraries to the WebLogic Server classpath, specify additional java command line options for running the servers, or specify additional environment variables. Any customizations you add to this file are preserved during domain upgrade operations, and are carried over to remote servers when using the `pack` and `unpack` commands.

To begin domain configuration:

1. Shut down the Administration Server to prevent any configuration locks, saves, or activations from occurring during the configuration of the domain.

For more information, see the instructions for shutting down the Administration Server with Node Manager in [Starting the Configuration Wizard on SOAHOST1](#).

2. Navigate to the following directory and start the WebLogic Server Configuration Wizard.

```
ORACLE_HOME/oracle_common/common/bin
./config.sh
```

16.5.2 Navigating the Configuration Wizard Screens for Oracle BAM

In this step, you extend the domain created in [Extending the Domain with Oracle SOA Suite](#), to contain Oracle Business Activity Monitoring components.

The steps reflected in this section would be very similar if Oracle Business Activity Monitoring was extending a domain containing only an Administration Server and a WSM-PM Cluster, but some of the options, libraries and components shown in the screens could vary.

Domain creation and configuration includes the following tasks:

- [Task 1, "Selecting the Domain Type and Domain Home Location"](#)

- [Task 2, "Selecting the Configuration Template"](#)
- [Task 3, "Specifying the Datasource Configuration Type"](#)
- [Task 4, "Specifying JDBC Component Schema Information"](#)
- [Task 5, "Providing the GridLink Oracle RAC Database Connection Details"](#)
- [Task 6, "Testing the JDBC Connections"](#)
- [Task 7, "Selecting Advanced Configuration"](#)
- [Task 8, "Configuring Managed Servers"](#)
- [Task 9, "Configuring a Cluster"](#)
- [Task 10, "Assigning Managed Servers to the Cluster"](#)
- [Task 11, "Configuring Coherence Clusters"](#)
- [Task 12, "Verifying the Existing Machines"](#)
- [Task 13, "Assigning Servers to Machines"](#)
- [Task 14, "Configuring the JMS File Store"](#)
- [Task 15, "Reviewing Your Configuration Specifications and Configuring the Domain"](#)
- [Task 16, "Start the Administration Server"](#)

Task 1 Selecting the Domain Type and Domain Home Location

On the Configuration Type screen, select **Update an existing domain**.

In the **Domain Location** field, select the value of the ASERVER_HOME variable, which represents the complete path to the Administration Server domain home you created in [Creating the Initial Infrastructure Domain for an Enterprise Deployment](#).

For more information about the directory location variables, see [File System and Directory Variables Used in This Guide](#)

Tip:

More information about the other options on this screen can be found in Configuration Type in *Creating WebLogic Domains Using the Configuration Wizard*.

Task 2 Selecting the Configuration Template

On the Templates screen, make sure **Update Domain Using Product Templates** is selected, then select the following template:

Oracle Business Activity Monitoring- 12.2.1.0 [soa]

Click **Next**.

Task 3 Specifying the Datasource Configuration Type

Note:

Any custom data sources that were created before the extension (like LEASING datasources) will show up before this screen. Check the Datasources row and click **Next**. The test data source screen will verify its validity. Click **Next**.

All fields are pre-populated, because you already configured the domain to reference the Fusion Middleware schemas that are required for the Infrastructure domain. Verify and ensure that credentials in all the fields are the same that you have provided while configuring Oracle Fusion Middleware Infrastructure.

Click **Get RCU Configuration** after you finish verifying the database connection information. The following output in the Connection Result Log indicates that the operation succeeded:

```
Connecting to the database server...OK
Retrieving schema data from database server...OK
Binding local schema components with retrieved data...OK
Successfully Done.
```

Task 4 Specifying JDBC Component Schema Information

On the JDBC Component Schema page, select the following schemas:

- **BAM Schema**
- **BAM Job Sched Schema**
- **BAM Leasing Schema**
- **BAM Non JTA Schema**
- **BAM MDS Schema**

Select **Convert to Gridlink**, and then click **Next**.

Task 5 Providing the GridLink Oracle RAC Database Connection Details

On the GridLink Oracle RAC Component Schema screen, provide the information required to connect to the RAC database and component schemas, as shown in the following table.

Element	Description and Recommended Value
SCAN, Host Name, and Port	Select the SCAN check box. In the Host Name field, enter the Single Client Access Name (SCAN) Address for the Oracle RAC database. In the Port field, enter the SCAN listening port for the database (for example, 1521)
ONS Host and Port	In the ONS Host field, enter the SCAN address for the Oracle RAC database. In the Port field, enter the ONS Remote port (typically, 6200).

Element	Description and Recommended Value
Enable Fan	Verify that the Enable Fan check box is selected, so the database can receive and process FAN events.

Task 6 Testing the JDBC Connections

On the Test JDBC Data Sources screen, confirm that all connections were successful.

The connections are tested automatically. The Status column displays the results. If all connections are not successful, click Previous to return to the previous screen and correct your entries.

Click **Next** when all the connections are successful.

Task 7 Selecting Advanced Configuration

On the Select Advanced Configuration screen, select the following:

- Managed Servers, Clusters, and Coherence
- JMS File Store

Click **Next**.

Task 8 Configuring Managed Servers

On the Managed Servers screen, add the required managed servers for Oracle BAM:

- Select the automatically created server and rename it to WLS_BAM1.
- Click **Add** to add another new server and enter WLS_BAM2 as the server name.
- Select **BAM12-MGD-SVRS-ONLY** as the server group for the BAM Servers.
Deselect **BAM12-MGD-SVRS** from the list.

The configuration for the added servers should match those shown in the following table.

Name	Listen Address	Listen Port	SSL Listen Port	SSL Enabled	Server Groups
WLS_SOA1*	SOAHOST1	8001	n/a	No	SOA-MGD-SVRS-ONLY
WLS_SOA2*	SOAHOST2	8001	n/a	No	SOA-MGD-SVRS-ONLY
WLS_WSM1	SOAHOST1	7010	n/a	No	JRF-MAN-SVR WSMPM-MAN-SVR WSM-CACHE-SVR
WLS_WSM2	SOAHOST2	7010	n/a	No	JRF-MAN-SVR WSMPM-MAN-SVR WSM-CACHE-SVR

Name	Listen Address	Listen Port	SSL Listen Port	SSL Enabled	Server Groups
WLS_BAM1	SOAHOST1	9001	n/a	No	BAM12-MGD-SVRS-ONLY
WLS_BAM2	SOAHOST2	9001	n/a	No	BAM12-MGD-SVRS-ONLY

*The WLS_SOA1 and WLS_SOA2 Managed Servers are shown if you are extending a domain where Oracle SOA Suite has already been configured.

*When specifying the listen address for WLS_BAM1 and WLS_BAM2, enter the IP address for SOAHOST1 and SOAHOST2, respectively, unless you are configuring Oracle BAM on separate host computers (BAMHOST1 and BAMHOST2). If you are configuring Oracle BAM on separate hosts enter the listen addresses for BAMHOST1 and BAMHOST2.

Task 9 Configuring a Cluster

On the Configure Clusters screen, click **Add** to add the **BAM_Cluster** (leave the present cluster as they are):

Name	Cluster Address
SOA_Cluster*	Leave it empty
WSM-PM_Cluster	Leave it empty
BAM_Cluster	Leave it empty

*The SOA cluster appears only if you have already configured Oracle SOA Suite in the domain.

Click **Next**.

Task 10 Assigning Managed Servers to the Cluster

On the Assign Servers to Clusters screen, assign servers to clusters as follows:

- BAM_Cluster:
 - WLS_BAM1
 - WLS_BAM2

Click **Next**.

Task 11 Configuring Coherence Clusters

Use the Coherence Clusters screen to configure the Coherence cluster that is automatically added to the domain. Leave the port number value at 9991, as it was defined during the initial Infrastructure domain creation.

Task 12 Verifying the Existing Machines

Verify the machines that have already been created in the domain. By default, you will be targeting the new Oracle BAM Managed Servers to the SOAHOST1 and SOAHOST2 machines, respectively.

However, if you are configuring Oracle BAM on separate host computers, then you must create two new machines for the corresponding BAMHOST1 and BAMHOST2 host computers:

1. Select the **Unix Machine** tab.
2. Use the **Add** button to create two new Unix machines for BAMHOST1 and BAMHOST2.
Node Manager Listen Address to the physical IP address for BAMHOST1 and BAMHOST2.
3. Verify the port in the Node Manager Listen Port field.
The port number 5556, shown in this example, may be referenced by other examples in the documentation. Replace this port number with your own port number as needed.

Leave all other fields to their default values.

Click **Next**.

Task 13 Assigning Servers to Machines

On the Assign Servers to Machines screen, assign the new WLS_BAM1 and WLS_BAM2 servers to the SOAHOST1 and SOAHOST2 machines, respectively.

However, if you are configuring Oracle BAM on separate host computers, assign the new Oracle BAM servers to the newly created BAMHOST1 and BAMHOST2 machines, respectively.

Click **Next**.

Task 14 Configuring the JMS File Store

On the JMS File Stores screen, assign the following directory for each of the Oracle BAM persistent stores, including the 2 UMS JMS file stores created in this session:

```
ORACLE_RUNTIME/domain_name/BAM_Cluster/jms
```

Do not change the values assigned to the existing JMS file stores, which correspond to the clusters you created for previously configured products.

Task 15 Reviewing Your Configuration Specifications and Configuring the Domain

The Configuration Summary screen contains the detailed configuration information for the domain you are about to create. Review the details of each item on the screen and verify that the information is correct.

Click **Update**.

In the Extending Domain screen, click **Done**.

Task 16 Start the Administration Server

Start the Administration Server to ensure the changes you have made to the domain have been applied.

16.6 Configuring a Default Persistence Store for Transaction Recovery

Each Managed Server uses a transaction log that stores information about committed transactions that are coordinated by the server and that may not have been completed. Oracle WebLogic Server uses this transaction log for recovery from system crashes or network failures. To leverage the migration capability of the Transaction Recovery Service for the Managed Servers within a cluster, store the transaction log in a location accessible to each Managed Server and its backup server.

Note:

To enable migration of the Transaction Recovery Service, specify a location on a persistent storage solution that is available to other servers in the cluster. All Managed Servers in the cluster must be able to access this directory. This directory must also exist before you restart the server.

The recommended location is a dual-ported SCSI disk or on a Storage Area Network (SAN). Note that it is important to set the appropriate replication and backup mechanisms at the storage level to guarantee protection in cases of a storage failure.

This information applies for file-based transaction logs. You can also configure a database-based persistent store for translation logs. For more information, see [Using JDBC Persistent Stores for TLOGs and JMS in an Enterprise Deployment](#).

To set the location for the default persistence stores:

1. Log into the Oracle WebLogic Server Administration Console:
`ADMINVHN:7001/console`
2. In the Change Center section, click **Lock & Edit**.
3. For each of the Managed Servers in the cluster:
 - a. In the Domain Structure window, expand the **Environment** node, and then click the **Servers** node.
The Summary of Servers page appears.
 - b. Click the name of the server (represented as a hyperlink) in **Name** column of the table.
The settings page for the selected server appears and defaults to the Configuration tab.
 - c. On the Configuration tab, click the **Services** tab.
 - d. In the Default Store section of the page, enter the path to the folder where the default persistent stores will store its data files.

For the enterprise deployment, use the `ORACLE_RUNTIME` directory location. This subdirectory serves as the central, shared location for transaction logs for the cluster. For more information, see [File System and Directory Variables Used in This Guide](#).

For example:

`ORACLE_RUNTIME/domain_name/cluster_name/tlogs`

Replace *cluster_name* with the name of the cluster you just created.

- e. Click **Save**.
4. Click **Save and Activate Changes**.

Note:

You will validate the location and the creation of the transaction logs later in the configuration procedure.

16.7 Propagating the Extended Domain to the Domain Directories and Machines

After you have extended the domain with the BAM instances, and you have restarted the Administration Server on SOAHOST1, you must then propagate the domain changes to the domain directories and machines.

The following table summarizes the steps required to propagate the changes to all the domain directories and machines.

Task	Description	More Information
Pack up the Extended Domain on SOAHOST1	Use the Pack command to create a new template jar file that contains the new BAM Servers configuration. When you pack up the domain, create a template jar file called soadomaintemplateExtBAM.jar.	Packing Up the Extended Domain on SOAHOST1
Unpack the Domain in the Managed Servers Directory on SOAHOST1*	Unpack the template jar file in the Managed Servers directory on SOAHOST1 local storage.	Unpacking the Domain in the Managed Servers Domain Directory on SOAHOST1
Unpack the Domain on SOAHOST2	Unpack the template jar file in the Managed Servers directory on the SOAHOST2 local storage.	Unpacking the Domain on SOAHOST2

*If you are configuring Oracle BAM on separate hosts, then you would unpack the domain on BAMHOST1 and BAMHOST2, rather than on SOAHOST1 and SOAHOST2.

16.8 Adding the Enterprise Deployment Administration User to the Oracle BAM Administration Group

Before you validate the Oracle BAM configuration on the Managed Server, add the enterprise deployment administration user (`weblogic_soa`) to the `BAMAdministrators` group.

To perform this task, refer to [Configuring Roles for Administration of Oracle SOA Suite Products](#).

16.9 Starting WLS_BAM1 Managed Server

After extending the domain, restarting the Administration Server, and propagating the domain to the other hosts, start the newly configured BAM servers.

1. Enter the following URL into a browser to display the Fusion Middleware Control login screen:

`http://ADMINVHN:7001/em`

2. Log in to Fusion Middleware Control using the Administration Server credentials.
3. In the Target Navigation pane, expand the domain to view the Managed Servers in the domain.
4. Select only the WLS_BAM1 Managed Server and click **Start Up** on the Oracle WebLogic Server toolbar.

Note:

BAM Servers depend on the policy access service to be functional, so the WSM-PM Managed Servers in the domain need to be up and running and reachable before the BAM servers are started.

5. When the startup operation is complete, navigate to the Domain home page and verify that the WLS_BAM1 Managed Server is up and running.

6. To verify that the BAM software is configured properly:

- a. Enter the following URL in the browser:

`http://SOAHOST1:9001/bam/composer`

The login screen for BAM's composer appears.

If you configured Oracle BAM on separate host computers, enter BAMHOST1 in the URL, rather than SOAHOST1.

- b. Enter the `weblogic_soa` login credentials.

The BAM Composer screen appears.

7. Enter the following URL:

`http://SOAHOST1:9001/inspection.wsil/`

If you configured Oracle BAM on separate host computers, enter BAMHOST1 in the URL, rather than SOAHOST1.

You should see a response with the following list of links.

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```

-<inspection>
-<link referencedNamespace="http://schemas.xmlsoap.org/ws/2001/10/inspection" location="http://rws3211540.us.oracle.com:9001/inspection.wsil?appname=com.oracle.webservices.wls.bea-wls9-async-response_12.1.3">
-<abstract>
  com.oracle.webservices.wls.bea-wls9-async-response_12.1.3
</abstract>
</link>
-<link referencedNamespace="http://schemas.xmlsoap.org/ws/2001/10/inspection" location="http://rws3211540.us.oracle.com:9001/inspection.wsil?appname=usermessagingserver">
<abstract>usermessagingserver</abstract>
</link>
-<link referencedNamespace="http://schemas.xmlsoap.org/ws/2001/10/inspection" location="http://rws3211540.us.oracle.com:9001/inspection.wsil?appname=BamServer">
<abstract>BamServer</abstract>
</link>
</inspection>

```

8. Enter the following URL in the browser:

`http://SOAHOST1:9001/bam/cqservice/`

If you configured Oracle BAM on separate host computers, enter BAMHOST1 in the URL, rather than SOAHOST1.

You should get a message in the browser indicating "BAM CQService is running."

16.10 Starting and Validating the WLS_BAM2 Managed Server

Follow the instructions in this section to start and validate WLS_BAM2.

1. Log in to Fusion Middleware Control using the Administration Server credentials.
2. In the Target Navigation pane, expand the domain to view the Managed Servers in the domain.
3. Select only the WLS_BAM2 Managed Server and click **Start Up** on the Oracle WebLogic Server tool bar.
4. When the startup operation is complete, navigate to the Domain home page and verify that the WLS_BAM2 Managed Server is up and running. Access the equivalent URLs for the WLS_BAM2:

`http://SOAHOST2:9001/bam/composer`

The login screen for BAM's composer appears. Enter the login credentials. The BAM composer's menu is displayed.

5. Enter the following URL:

`http://SOAHOST2:9001/inspection.wsil/`

You should see a response with a list of links.

6. Enter the following URL in the browser:

`http://SOAHOST2:9001/bam/cqservice/`

You should get a message in the browser indicating "BAM Service is running."

Note:

If you configured Oracle BAM on separate host computers, enter *BAMHOST2* in the URL, rather than *SOAHOST2*.

16.11 Configuring Oracle HTTP Server for the WLS_BAM Managed Servers

Make the following modifications to the Oracle HTTP Server instance configuration files to ensure that the Oracle HTTP Server instances in the Web tier can route Oracle BAM requests correctly to the Oracle BAM software on the Oracle SOA Suite cluster.

Note that these instructions assume you configuring Oracle BAM on the same host as Oracle SOA Suite. If you are using separate hosts for Oracle BAM, you must modify the `WebLogicCluster` parameter in the Oracle HTTP Server configuration files to reference the `BAMHOST` computers, rather than the `SOAHOST` computers.

To enable Oracle HTTP Server to route requests to Oracle BAM:

1. Log in to `WEBHOST1` and change directory to the configuration directory for the first Oracle HTTP Server instance (`ohs1`):

```
cd OHS_DOMAIN_HOME/config/fmwconfig/components/OHS/ohs1/moduleconf
```

2. Add the following directives inside the `<VirtualHost>` tag in the `soa_vh.conf` file:

```
<Location /bam/composer >  
  WLSRequest ON  
  WebLogicCluster SOAHOST1:9001,SOAHOST2:9001  
  WLProxySSL ON  
  WLProxySSLPassThrough ON  
</Location>  
  
<Location /OracleBAMWS>  
  WLSRequest ON  
  WebLogicCluster SOAHOST1:9001,SOAHOST2:9001  
  WLProxySSL ON  
  WLProxySSLPassThrough ON  
</Location>  
  
<Location /oracle/bam/server>  
  WLSRequest ON  
  WebLogicCluster SOAHOST1:9001,SOAHOST2:9001  
  WLProxySSL ON  
  WLProxySSLPassThrough ON  
</Location>
```

3. Change directory to the following location so you can update the configuration file for the second Oracle HTTP Server instance (`ohs2`):

```
cd OHS_DOMAIN_HOME/config/fmwconfig/components/OHS/ohs2/moduleconf
```

4. Open the `soa_vh.conf` file and add the B2B directives to the `<VirtualHost>` tag.
5. Restart the Oracle HTTP Server instances on `WEBHOST1` and `WEBHOST2`.

16.12 Configuring the WebLogic Proxy Plug-In

Set the WebLogic Plug-In Enabled parameter for the BAM cluster.

1. Log in to the Oracle WebLogic Server Administration Console.

2. In the Domain Structure pane, expand the **Environment** node.
3. Click on **Clusters**.
4. Select the BAM_Cluster cluster to which you want to proxy requests from Oracle HTTP Server.

The **Configuration: General** tab is displayed.
5. Scroll down to the Advanced section and expand it.
6. Click **Lock and Edit**.
7. Set the WebLogic Plug-In Enabled to **yes**.
8. Click **Save and Activate the changes**. Restart the BAM servers for the changes to be effective.

16.13 Validating Access to Oracle BAM Through the Hardware Load Balancer

Verify that Oracle BAM URLs are successfully routing requests from the hardware load balancer to the Oracle HTTP Server instances to the Oracle BAM software in the middle tier.

You can also use this procedure test the failover of the Managed Servers where Oracle BAM is configured.

To verify the URLs:

1. While the WLS_BAM1 Managed Server is running, stop the WLS_BAM2 Managed Server, using the Oracle WebLogic Server Administration Console.
2. Access the following URL and verify the HTTP response as indicated in [Starting WLS_BAM1 Managed Server](#):

`http://soa.example.com/bam/composer`
3. Access the following URL to be sure the software is running as expected:

`https://soa.example.com/oracle/bam/server`
4. Start WLS_BAM2 from the Oracle WebLogic Server Administration Console.
5. Stop WLS_BAM1 from the Oracle WebLogic Server Administration Console.
6. Access the URL again, and verify the HTTP response is still valid, as indicated in [Starting and Validating the WLS_BAM2 Managed Server](#).

16.14 Enabling Automatic Service Migration and JDBC Persistent Stores for the Oracle BAM Servers

To ensure that your software is configured for high availability, configure the Oracle Business Activity Monitoring Managed Servers for automatic service migration.

For more information on enabling automatic service migration, see [Configuring Automatic Service Migration in an Enterprise Deployment](#).

Note:

If you are configuring Oracle BAM in its own domain, then you can use the default leasing data source (BamLeasingDatasource) when you are configuring automatic service migration for Oracle BAM.

However, in a more typical environment, where you are configuring both Oracle BAM with Oracle SOA Suite or Oracle Service Bus, then Oracle recommends you use a central automatic service migration data source, such as the LeasingDS, which is described in [Configuring Automatic Service Migration in an Enterprise Deployment](#).

For additional high availability, you can also configure your transaction logs store and JMS store in a database. For more information, see [Using JDBC Persistent Stores for TLOGs and JMS in an Enterprise Deployment](#).

16.15 Backing Up the Oracle BAM Configuration

It is an Oracle best practices recommendation to create a backup after successfully extending a domain or at another logical point. Create a backup after verifying that the installation so far is successful. This is a quick backup for the express purpose of immediate restoration in case of problems in later steps.

The backup destination is the local disk. You can discard this backup when the enterprise deployment setup is complete. After the enterprise deployment setup is complete, you can initiate the regular deployment-specific Backup and Recovery process.

For information about backing up your configuration, see [Performing Backups and Recoveries in the SOA Enterprise Deployments](#).

Extending the Domain with Oracle B2B

This chapter describes the procedures for extending the domain to include Oracle B2B.

The Oracle B2B and Healthcare distribution includes the software required to configure Oracle B2B or Oracle SOA for Healthcare.

Note:

For X12 HIPAA use cases, you can use the Oracle B2B domain extension steps described in this chapter. However, if you are a healthcare provider using HL7 documents, refer to [Extending the Domain with Oracle SOA Suite for Healthcare Integration](#), and extend the domain with the Oracle SOA Suite for Healthcare Integration software.

See Also:

[Configuring the Enterprise Deployment](#)

[Variables Used in This Chapter](#)

As you perform the tasks in this chapter, you will be referencing the directory variables listed in this section.

[Prerequisites for Extending the SOA Domain to Include Oracle B2B](#)

Before extending the current domain, ensure that your existing deployment meets the prerequisites specified in this section.

[Installing Oracle B2B for an Enterprise Deployment](#)

Use the following sections to install the Oracle Fusion Middleware Infrastructure software in preparation for configuring a new domain for an enterprise deployment.

[Running the Configuration Wizard to Extend for Oracle B2B](#)

To extend the domain to include Oracle BAM, refer to the following sections.

[Starting the B2B Suite Components](#)

For configuration changes and start scripts to be effective, you must restart the WLS_SOA server to which B2B has been added. Since B2B extends an already existing SOA system, the Administration Server and

respective Node Managers are already running in SOAHOST1 and SOAHOST2.

Updating the B2B Instance Identifier for Transports

To set up File, FTP, or Email transports in a high availability environment, set the `b2b.HAInstance` property to true.

Configuring Oracle HTTP Server for Oracle B2B

Make the following modifications to the Oracle HTTP Server instance configuration files to ensure that the Oracle HTTP Server instances in the Web tier can route Oracle B2B requests correctly to the Oracle B2B software on the Oracle SOA Suite cluster.

Adding the B2BAdmin Role to the SOA Administrators Group

Before you validate the Oracle B2B configuration on the Managed Servers, add the `B2BAdmin` administration role to the enterprise deployment administration group (`SOA Administrators`).

Validating Access to Oracle B2B Through the Load Balancer

Use the following steps to verify that the appropriate routing and failover is working from the load balancer to the HTTP Server instances to the B2B Suite Components on the Oracle SOA Suite Managed Server.

Backing Up the Configuration

It is an Oracle best practices recommendation to create a backup after successfully configuring a domain or at another logical point. Create a backup after verifying that the installation so far is successful. This is a quick backup for the express purpose of immediate restoration in case of problems in later steps.

Enabling Automatic Service Migration and JDBC Persistent Stores for Oracle B2B

In the enterprise topology, Oracle B2B is configured on the existing Oracle SOA Suite Managed Servers. If you have already configured automatic service migration for the `SOA_Cluster`, then the Oracle B2B software is already protected by automatic service migration.

17.1 Variables Used in This Chapter

As you perform the tasks in this chapter, you will be referencing the directory variables listed in this section.

The values for several directory variables are defined in [File System and Directory Variables Used in This Guide](#).

- `ORACLE_HOME`
- `ASERVER_HOME`
- `MSERVER_HOME`
- `OHS_DOMAIN_HOME`
- `JAVA_HOME`

In addition, you'll be referencing the following virtual IP (VIP) addresses defined in [Physical and Virtual IP Addresses Required by the Enterprise Topology](#):

- `ADMINVHN`

Actions in this chapter will be performed on the following host computers:

- SOAHOST1
- SOAHOST2
- WEBHOST1
- WEBHOST2

17.2 Prerequisites for Extending the SOA Domain to Include Oracle B2B

Before extending the current domain, ensure that your existing deployment meets the prerequisites specified in this section.

- Back up the installation - If you have not yet backed up the existing Fusion Middleware Home and domain, Oracle recommends backing it up now.
To back up the existing Fusion Middleware Home and domain, see [Performing Backups and Recoveries in the SOA Enterprise Deployments](#).
- There is an existing WL_HOME and SOA ORACLE_HOME (binaries) installed in previous chapters on a shared storage and available from SOAHOST1 and SOAHOST2.
- Node Manager, Admin Server, SOA Servers and WSM Servers exist and have been configured as described in previous chapters to run a SOA system.
- You do not need to run RCU to load additional schemas for B2B, these are part of the SOA repository and were loaded into the DB in the SOA chapter.

17.3 Installing Oracle B2B for an Enterprise Deployment

Use the following sections to install the Oracle Fusion Middleware Infrastructure software in preparation for configuring a new domain for an enterprise deployment.

See Also:

[Starting the Oracle B2B and Healthcare Installer on SOAHOST1](#)

[Navigating the Oracle B2B Installation Screens](#)

[Verifying the B2B or Healthcare Installation](#)

17.3.1 Starting the Oracle B2B and Healthcare Installer on SOAHOST1

To start the installation program, perform the following steps.

1. Log in to SOAHOST1.
2. Go to the directory where you downloaded the installation program.
3. Launch the installation program by invoking the `java` executable from the JDK directory on your system, as shown in the example below.

```
JAVA_HOME/bin/java -d64 -jar distribution_file_name.jar
```

In this example:

- Replace `JAVA_HOME` with the environment variable or actual JDK location on your system.
- Replace `distribution_file_name` with the actual name of the distribution jar file.

Note that if you download the distribution from the Oracle Technology Network (OTN), then the jar file is typically packaged inside a downloadable ZIP file.

To install the software required for the initial Infrastructure domain, the distribution you want to install is **fmw_12.2.1.0.0_b2bhealthcare.jar**.

For more information about the actual file names of each distribution, see [Identifying and Obtaining Software Downloads for an Enterprise Deployment](#).

When the installation program appears, you are ready to begin the installation. See [Navigating the Installation Screens](#) for a description of each installation program screen.

17.3.2 Navigating the Oracle B2B Installation Screens

[Table 17-1](#) provides description of each installation program screen.

Table 17-1 Oracle B2B Install Screens

Screen	Description
Installation Inventory Setup	<p>On UNIX operating systems, this screen will appear if this is the first time you are installing any Oracle product on this host. Specify the location where you want to create your central inventory. Make sure that the operating system group name selected on this screen has write permissions to the central inventory location.</p> <p>For more information about the central inventory, see "Understanding the Oracle Central Inventory" in <i>Installing Software with the Oracle Universal Installer</i>.</p>
Welcome	This screen introduces you to the product installer.
Auto Updates	Use this screen to automatically search My Oracle Support for available patches or automatically search a local directory for patches that you've already downloaded for your organization.
Installation Location	<p>Use this screen to specify the location of your Oracle home directory.</p> <p>For more information about Oracle Fusion Middleware directory structure, see "Selecting Directories for Installation and Configuration" in <i>Planning an Installation of Oracle Fusion Middleware</i>.</p>
Installation Type	<p>Use this screen to select the type of installation and consequently, the products and feature sets you want to install.</p> <ul style="list-style-type: none"> • Select B2B <p>NOTE: The topology in this document does not include the examples, Oracle strongly recommends that you do not install the examples into a production environment.</p>
Prerequisite Checks	<p>This screen verifies that your system meets the minimum necessary requirements.</p> <p>If there are any warning or error messages, you can refer to one of the following documents in Section 1.4.</p>

Table 17-1 (Cont.) Oracle B2B Install Screens

Screen	Description
Installation Summary	<p>Use this screen to verify the installation options you selected. If you want to save these options to a response file, click Save Response File and provide the location and name of the response file. Response files can be used later in a silent installation situation.</p> <p>For more information about silent or command line installation, see "Using the Oracle Universal Installer in Silent Mode" in <i>Installing Software with the Oracle Universal Installer</i>.</p> <p>Click Install to begin the installation.</p>
Installation Progress	<p>This screen allows you to see the progress of the installation.</p> <p>Click Next when the progress bar reaches 100% complete.</p>
Installation Complete	<p>Review the information on this screen, then click Finish to dismiss the installer.</p>

17.3.3 Verifying the B2B or Healthcare Installation

After you complete the installation, you can verify it by successfully completing the following tasks.

See Also:

[Reviewing the Installation Log Files](#)

[Checking the Directory Structure](#)

[Viewing the Contents of Your Oracle Home](#)

17.3.3.1 Reviewing the Installation Log Files

Review the contents of the installation log files to make sure that no problems were encountered. For a description of the log files and where to find them, see "Understanding Installation Log Files" in *Installing Software with the Oracle Universal Installer*.

17.3.3.2 Checking the Directory Structure

The contents of your installation vary based on the options you selected during the installation.

The addition of Oracle Healthcare will add the following directory and sub-directories:

```
ls /u01/oracle/products/fmw/soa/soa/thirdparty/edifecs/

Common
STANDARDS
XEngine
```

For more information about the directory structure you should see after installation, see "What are the Key Oracle Fusion Middleware Directories?" in *Understanding Oracle Fusion Middleware*.

17.3.3.3 Viewing the Contents of Your Oracle Home

You can also view the contents of your Oracle home using the `viewInventory` script. For more information, see "Viewing the contents of an Oracle home" in *Installing Software with the Oracle Universal Installer*.

17.4 Running the Configuration Wizard to Extend for Oracle B2B

To extend the domain to include Oracle BAM, refer to the following sections.

See Also:

[Starting the Configuration Wizard](#)

[Navigating the Configuration Wizard Screens for Oracle B2B](#)

17.4.1 Starting the Configuration Wizard

Note:

If you added any customizations directly to the start scripts in the domain, those will be overwritten by the configuration wizard. To customize server startup parameters that apply to all servers in a domain, you can create a file called `setUserOverrides.sh` and configure it, for example, add custom libraries to the WebLogic Server classpath, specify additional java command line options for running the servers, or specify additional environment variables. Any customizations you add to this file are preserved during domain upgrade operations, and are carried over to remote servers when using the `pack` and `unpack` commands.

To begin domain configuration:

1. Shut down the Administration Server to prevent any configuration locks, saves, or activations from occurring during the configuration of the domain.

For more information, see the instructions for shutting down the Administration Server with Node Manager in [Starting the Configuration Wizard on SOAHOST1](#).

2. Navigate to the following directory and start the WebLogic Server Configuration Wizard.

```
ORACLE_HOME/oracle_common/common/bin
./config.sh
```

17.4.2 Navigating the Configuration Wizard Screens for Oracle B2B

Follow the instructions in this section to extend the domain for B2B.

Note:

You can use the same procedure described in this section to extend an existing domain. If your needs do not match the instructions given in the procedure, be sure to make your selections accordingly, or refer to the supporting documentation for additional details.

Domain creation and configuration includes the following tasks:

- [Task 1, "Selecting the Domain Type and Domain Home Location"](#)
- [Task 2, "Selecting the Configuration Template"](#)
- [Task 3, "Specifying the Datasource Configuration Type"](#)
- [Task 4, "Selecting Advanced Configuration"](#)
- [Task 5, "Reviewing Your Configuration Specifications and Configuring the Domain"](#)
- [Task 6, "Writing Down Your Domain Home and Administration Server URL"](#)
- [Task 7, "Start the Administration Server"](#)

Task 1 Selecting the Domain Type and Domain Home Location

On the Configuration Type screen, select **Update an existing domain**.

In the Domain Location field, select the value of the `ASERVER_HOME` variable, which represents the complete path to the Administration Server domain home you created in [Creating the Initial Infrastructure Domain for an Enterprise Deployment](#).

For more information about the directory location variables, see [File System and Directory Variables Used in This Guide](#)

Tip:

More information about the other options on this screen can be found in Configuration Type in [Creating WebLogic Domains Using the Configuration Wizard](#).

Task 2 Selecting the Configuration Template

On the Templates screen, make sure **Update Domain Using Product Templates** is selected, then select the following templates:

- **Oracle B2B - 12.2.1.0 [soa]**

In addition, the following additional templates should already be selected, because they were used to create the initial domain and extend it to SOA:

- Basic Weblogic Server Domain - 12.2.1.0 [wlserver]
- Oracle SOA Suite 12.2.1.0 [soa]
- Oracle Enterprise Manager - 12.2.1.0 [em]
- Oracle WSM Policy Manager - 12.2.1.0 [oracle_common]

- Oracle JRF - 12.2.1.0 [oracle_common]
- WebLogic Coherence Cluster Extension - 12.2.1.0 [wlserver]

Tip:

More information about the options on this screen can be found in Templates in Creating WebLogic Domains Using the Configuration Wizard.

Task 3 Specifying the Datasource Configuration Type

All fields are pre-populated, because you already configured the domain to reference the Fusion Middleware schemas that are required for the Infrastructure domain. B2B uses the existing DataSources for SOA and no new Datasources need to be added to the domain.

Note:

Any custom data sources that were created before the extension (like LEASING data sources) will show up before this screen. Check the Datasources row and click **Next**. The test data source screen will verify its validity. Click **Next**.

Task 4 Selecting Advanced Configuration

To complete domain configuration for the topology, do not select any additional options on the Advanced Configuration screen and Click **Next**. B2B applications and required artifacts will be targeted automatically to the existing SOA servers

Task 5 Reviewing Your Configuration Specifications and Configuring the Domain

The Configuration Summary screen contains the detailed configuration information for the domain you are about to create. Review the details of each item on the screen and verify that the information is correct.

You can go back to any previous screen if you need to make any changes, either by using the **Back** button or by selecting the screen in the navigation pane.

Domain creation will not begin until you click **Domain Update**.

Tip:

More information about the options on this screen can be found in Configuration Summary in Creating WebLogic Domains Using the Configuration Wizard.

Task 6 Writing Down Your Domain Home and Administration Server URL

The Configuration Success screen will show the following items about the domain you just configured:

- Domain Location
- Administration Server URL

You must make a note of both items as you will need them later; the domain location is needed to access the scripts used to start the Administration Server, and the URL is needed to access the Administration Server.

Click **Finish** to dismiss the configuration wizard.

Task 7 Start the Administration Server

Start the Administration Server to ensure the changes you have made to the domain have been applied.

17.5 Starting the B2B Suite Components

For configuration changes and start scripts to be effective, you must restart the WLS_SOA server to which B2B has been added. Since B2B extends an already existing SOA system, the Administration Server and respective Node Managers are already running in SOAHOST1 and SOAHOST2.

To start the added B2B components, restart the Managed Servers:

1. Log into the Oracle WebLogic Server Administration Console at:

`http://ADMINVHN:7001/em`

In this example:

Replace *ADMINVHN* with the host name assigned to the ADMINVHN Virtual IP address in [Identifying and Obtaining Software Downloads for an Enterprise Deployment](#).

Port 7001 is the typical port used for the Administration Server console and Fusion Middleware Control. However, you should use the actual URL that was displayed at the end of the Configuration Wizard session when you created the domain.

2. In the Domain Structure window, expand the **Environment** node, then select **Servers**.
The Summary of Servers page appears.
3. Click the **Control** tab.
4. Select **WLS_SOA1** from the Servers column of the table.
5. Click **Shutdown**. Wait for the shutdown to complete (refresh the WebLogic Server Console page to verify shutdown status).
6. Click **Start**.
7. Repeat steps 2 through 6 for WLS_SOA2.

17.6 Updating the B2B Instance Identifier for Transports

To set up File, FTP, or Email transports in a high availability environment, set the `b2b.HAInstance` property to true.

To do this follow these steps:

1. Log in to Oracle Enterprise Manager Fusion Middleware Control with the user name and password specified for the domain administration.
2. Display the Target Navigation pane, by clicking the target navigation icon near the left top corner of the screen.

3. In the navigation tree, expand **SOA**, and then right click the `soa-infra(server_name)`, and select the **SOA Administration**, and then **B2B Server Properties** from the context menu.

If there are multiple `soa-infra (server_name)`, add the property only once.

4. Click **More B2B Configuration Properties**.

B2BConfig b2b should already be selected.

5. Click the **Operations** tab.
6. Click **addProperty** in the list on the right.
7. In the Key field enter **b2b.HAInstance**.

8. In the value field enter **true**.

This property is stored in MDS and needs to be created only once for the cluster.

9. Click **Invoke**.

After you define high availability properties, you can view them on the Attributes tab. To view the properties, click the **Attributes** tab and then click **Properties**. Expand the Element nodes in the Value table to verify the property names and values.

17.7 Configuring Oracle HTTP Server for Oracle B2B

Make the following modifications to the Oracle HTTP Server instance configuration files to ensure that the Oracle HTTP Server instances in the Web tier can route Oracle B2B requests correctly to the Oracle B2B software on the Oracle SOA Suite cluster.

To enable Oracle HTTP Server to route requests to Oracle B2B Console and to Oracle B2B services:

1. Log in to WEBHOST1 and change directory to the configuration directory for the first Oracle HTTP Server instance (ohs1):

```
cd OHS_DOMAIN_HOME/config/fmwconfig/components/OHS/ohs1/moduleconf
```

2. Add the following directives inside the `<VirtualHost>` tag in the `soa_vh.conf` file:

```
# B2B
<Location /b2bconsole>
  WLSRequest ON
  WebLogicCluster SOAHOST1:8001,SOAHOST2:8001
</Location>

# B2B
<Location /b2b>
  WLSRequest ON
  WebLogicCluster SOAHOST1:8001,SOAHOST2:8001
</Location>
```

3. Restart the ohs1 instance:

- a. Change directory to the following location:

```
cd OHS_DOMAIN_HOME/bin
```

- b. Enter the following commands to stop and start the instance:

```
./stopComponent.sh ohs1
./startComponent.sh ohs1
```

4. Log in to WEBHOST2 and copy the `soa_vh.conf` file to the configuration directory for the second Oracle HTTP Server instance (ohs_2):

```
OHS_DOMAIN_HOME/config/fmwconfig/components/OHS/ohs2/moduleconf
```

5. Edit the `soa_vh.conf` file to change any references to WEBHOST1 to WEBHOST2.

6. Restart the ohs2 instance:

- a. Change directory to the following location:

```
cd OHS_DOMAIN_HOME/bin
```

- b. Enter the following commands to stop and start the instance:

```
./stopComponent.sh ohs2
./startComponent.sh ohs2
```

17.8 Adding the B2BAdmin Role to the SOA Administrators Group

Before you validate the Oracle B2B configuration on the Managed Servers, add the B2BAdmin administration role to the enterprise deployment administration group (SOA Administrators).

To perform this task, refer to [Configuring Roles for Administration of Oracle SOA Suite Products](#).

17.9 Validating Access to Oracle B2B Through the Load Balancer

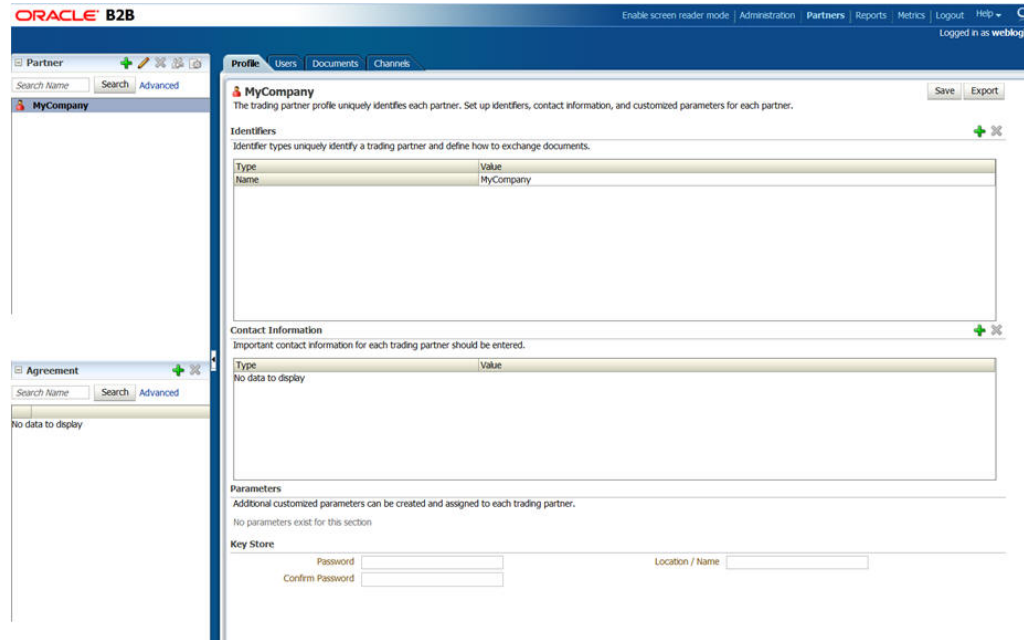
Use the following steps to verify that the appropriate routing and failover is working from the load balancer to the HTTP Server instances to the B2B Suite Components on the Oracle SOA Suite Managed Server.

1. Enter the following URL to access the Oracle B2B Console through the load balancer:

```
https://soa.example.com/b2bconsole
```

2. Log in using `weblogic_soa` user. You should see the Oracle B2B Partner, Agreement, and Profile screen, as shown in [Figure 17-1](#).

Figure 17-1 Oracle B2B Partner, Agreement, and Profile Screen



3. Enter the following URL to access the Oracle B2B Web services endpoint:

`https://soa.example.com/b2b/services`

You should see the links to the different B2B endpoints test.

Figure 17-2 Oracle B2B Services

Welcome to the Oracle B2B Services

Service	TestClient	WSDLs
NativeToXMLServiceAsString	Test	WSDL
XMLToNativeServiceAsString	Test	WSDL
GetTPAConfigService	Test	WSDL
fpInternalDCsAndCalloutsService	Test	WSDL
fpListMapSetNameService	Test	WSDL
fpListChannelsService	Test	WSDL
IsTPASetupService	Test	WSDL
OutboundService	Test	WSDL
fpGetAllDocumentService	Test	WSDL
fpGetSchemaService	Test	WSDL
fpCreateChannelConfService	Test	WSDL
fpGetChannelDetailsService	Test	WSDL
TranslateService	Test	WSDL
NativeToXMLService	Test	WSDL
XMLToNativeService	Test	WSDL

17.10 Backing Up the Configuration

It is an Oracle best practices recommendation to create a backup after successfully configuring a domain or at another logical point. Create a backup after verifying that

the installation so far is successful. This is a quick backup for the express purpose of immediate restoration in case of problems in later steps.

The backup destination is the local disk. You can discard this backup when the enterprise deployment setup is complete. After the enterprise deployment setup is complete, you can initiate the regular deployment-specific Backup and Recovery process.

For information about backing up your configuration, see [Performing Backups and Recoveries for an Enterprise Deployment](#).

17.11 Enabling Automatic Service Migration and JDBC Persistent Stores for Oracle B2B

In the enterprise topology, Oracle B2B is configured on the existing Oracle SOA Suite Managed Servers. If you have already configured automatic service migration for the `SOA_Cluster`, then the Oracle B2B software is already protected by automatic service migration.

If you have not configured automatic service migration for the Managed Servers where Oracle B2B is configured, then see [Configuring Automatic Service Migration in an Enterprise Deployment](#).

In addition, you can optionally configure database-based persistent stores for Oracle B2B. For more information, see [Using JDBC Persistent Stores for TLOGs and JMS in an Enterprise Deployment](#).

Extending the Domain with Oracle SOA Suite for Healthcare Integration

The following topics describe the procedures for extending the domain to include Oracle SOA Suite for healthcare integration (Oracle Healthcare).

See Also:

[Configuring the Enterprise Deployment](#)

[Variables Used When Extending the Domain for Oracle Healthcare](#)

As you perform the tasks in this chapter, you will be referencing the directory variables listed in this section.

[About Oracle SOA Suite for Healthcare Integration in an Enterprise Deployment](#)

Oracle SOA Suite for healthcare integration uses several features of Oracle SOA Suite to help you design, create, and manage applications that process healthcare data.

[Prerequisites for Extending the Domain to Include Oracle Healthcare](#)

Before extending the domain with Oracle Healthcare, note the following important prerequisites that are specific Oracle Healthcare.

[Installing Oracle Healthcare for an Enterprise Deployment](#)

Use the following sections to install the Oracle Fusion Middleware Infrastructure software in preparation for configuring a new domain for an enterprise deployment.

[Running the Configuration Wizard for Oracle Healthcare](#)

To extend the domain to include Oracle Healthcare, refer to the following sections.

[Starting the Healthcare Components](#)

This topic explains how to start the Oracle Healthcare components you have configured in the domain. The procedure requires you to restart the Managed Server where Oracle Healthcare has been configured. This ensures that the configuration changes and startup scripts are updated and validated correctly. Because you are extending an existing domain, the Administration Server and respective Node Managers are already running in on both application server hosts.

[Updating the B2B Instance Identifier and MLLP High Availability Mode](#)

To set up File, FTP, or Email transports in a high availability environment, set the `b2b.HAInstance` property to `true`.

[Disabling Connection Factory Affinity for Optimum Load Balancing](#)

To avoid the possibility of all the load affecting one Oracle Healthcare server, Oracle recommends that you perform the following steps.

[Configuring Oracle HTTP Server for Oracle Healthcare](#)

Make the following modifications to the Oracle HTTP Server instance configuration files to ensure that the Oracle HTTP Server instances in the Web tier can route Oracle Healthcare requests correctly to the Oracle SOA Suite cluster.

[Adding the B2BAdmin Role to the SOA Administrators Group](#)

Before you validate the Oracle B2B configuration on the Managed Servers, add the `B2BAdmin` administration role to the enterprise deployment administration group (`SOA Administrators`).

[Validating Access to Oracle Healthcare Through the Load Balancer](#)

Use the following steps to verify that the appropriate routing and failover is working from the load balancer to the HTTP Server instances to the Oracle Healthcare Components on the Oracle SOA Suite Managed Server.

[Backing Up the Configuration](#)

It is an Oracle best practices recommendation to create a backup after successfully configuring a domain or at another logical point. Create a backup after verifying that the installation so far is successful. This is a quick backup for the express purpose of immediate restoration in case of problems in later steps.

[Enabling Automatic Service Migration for Oracle Healthcare](#)

In the enterprise topology, Oracle Healthcare is configured on the existing Oracle SOA Suite Managed Servers. If you have already configured automatic service migration for the `SOA_Cluster`, then the Oracle Healthcare software is already protected by automatic service migration.

18.1 Variables Used When Extending the Domain for Oracle Healthcare

As you perform the tasks in this chapter, you will be referencing the directory variables listed in this section.

The values for several directory variables are defined in [File System and Directory Variables Used in This Guide](#).

- `ORACLE_HOME`
- `ASERVER_HOME`
- `MSERVER_HOME`
- `OHS_DOMAIN_HOME`
- `JAVA_HOME`

In addition, you'll be referencing the following virtual IP (VIP) addresses defined in [Physical and Virtual IP Addresses Required by the Enterprise Topology](#):

- ADMINVHN

Actions in this chapter will be performed on the following host computers:

- SOAHOST1
- SOAHOST2
- WEBHOST1
- WEBHOST2

18.2 About Oracle SOA Suite for Healthcare Integration in an Enterprise Deployment

Oracle SOA Suite for healthcare integration uses several features of Oracle SOA Suite to help you design, create, and manage applications that process healthcare data.

For more information about Oracle SOA Suite for Healthcare Integration, see “Introduction to Oracle SOA Suite for Healthcare Integration” in the *Healthcare Integration User’s Guide for Oracle SOA Suite*.

When you configure Oracle SOA Suite for Healthcare Integration in an enterprise deployment, Oracle recommends that you configure Healthcare in its own domain. For more information about this recommendation, see “B2B and Healthcare Domain Topology Best Practices” in the *Healthcare Integration User’s Guide for Oracle SOA Suite*.

Typically, you configure a single domain for Oracle SOA Suite for Healthcare Integration, and a separate domain for Oracle B2B. Both these domain topologies are described and supported in this enterprise deployment guide. For more information, see the following:

- [Flow Charts and Roadmaps for Implementing the Primary Oracle SOA Suite Enterprise Topologies](#)
- [Building Your Own Oracle SOA Suite Enterprise Topology](#)

18.3 Prerequisites for Extending the Domain to Include Oracle Healthcare

Before extending the domain with Oracle Healthcare, note the following important prerequisites that are specific Oracle Healthcare.

- You can extend an existing domain with the Oracle Healthcare software only if the domain was configured using the Oracle SOA Suite domain configuration template. You cannot extend an Oracle BPM, Oracle BAM, or Oracle OSB domain with Oracle Healthcare.

This chapter assumes you have performed the steps in [Extending the Domain with Oracle SOA Suite](#).

- When you installed the Oracle SOA Infrastructure schema, as part of the Oracle SOA Suite installation and configuration, you should have set the **Healthcare Integration** RCU custom variable to YES. When you enter YES as the value for the Healthcare Integration custom variable, RCU creates additional materialized views in the database, which are required by the Healthcare Integration User Interface.

If you entered NO, you can perform these additional schema configuration tasks later by running the following SQL script on the database. This script is installed in the Oracle Fusion Middleware Oracle home when you select the Healthcare with B2B installation type:

```
ORACLE_HOME/soa/common/sql/soainfra/sql/oracle/b2b_mv.sql
```

In addition, ensure that your existing deployment meets the following general prerequisites for extending the domain

- Back up the installation - If you have not yet backed up the existing Fusion Middleware Home and domain, Oracle recommends backing it up now.
To back up the existing Fusion Middleware Home and domain, see [Performing Backups and Recoveries in the SOA Enterprise Deployments](#).
- Verify that you have installed the Infrastructure and SOA software binaries in an Oracle home on shared storage and they are available from SOAHOST1 and SOAHOST2.
- You have already configured Node Manager, Administration Server, SOA Servers, and WSM Servers as described in previous chapters to run a SOA system. Optionally, you may have already configured Server migration, transaction logs, coherence, and all other configuration steps for the SOA System.
- If you haven't done so already, verify that the system clocks on each host computer are synchronized. You can do this by running the date command as simultaneously as possible on the hosts in each cluster.

Alternatively, there are third-party and open-source utilities you can use for this purpose.

18.4 Installing Oracle Healthcare for an Enterprise Deployment

Use the following sections to install the Oracle Fusion Middleware Infrastructure software in preparation for configuring a new domain for an enterprise deployment.

See Also:

[Starting the Oracle B2B and Healthcare Installer on SOAHOST1](#)

[Navigating the Installation Screens for Oracle Healthcare Installation](#)

[Verifying the B2B or Healthcare Installation](#)

18.4.1 Starting the Oracle B2B and Healthcare Installer on SOAHOST1

To start the installation program, perform the following steps.

1. Log in to SOAHOST1.
2. Go to the directory where you downloaded the installation program.
3. Launch the installation program by invoking the java executable from the JDK directory on your system, as shown in the example below.

```
JAVA_HOME/bin/java -d64 -jar distribution_file_name.jar
```

In this example:

- Replace *JAVA_HOME* with the environment variable or actual JDK location on your system.
- Replace *distribution_file_name* with the actual name of the distribution jar file.

Note that if you download the distribution from the Oracle Technology Network (OTN), then the jar file is typically packaged inside a downloadable ZIP file.

To install the software required for the initial Infrastructure domain, the distribution you want to install is `fmw_12.2.1.0.0_b2bhealthcare.jar`.

For more information about the actual file names of each distribution, see [Identifying and Obtaining Software Downloads for an Enterprise Deployment](#).

When the installation program appears, you are ready to begin the installation. See [Navigating the Installation Screens](#) for a description of each installation program screen.

18.4.2 Navigating the Installation Screens for Oracle Healthcare Installation

[Table 17-1](#) provides description of each installation program screen.

Table 18-1 Oracle B2B and Healthcare InstallScreens

Screen	Description
Installation Inventory Setup	<p>On UNIX operating systems, this screen will appear if this is the first time you are installing any Oracle product on this host. Specify the location where you want to create your central inventory. Make sure that the operating system group name selected on this screen has write permissions to the central inventory location.</p> <p>For more information about the central inventory, see "Understanding the Oracle Central Inventory" in <i>Installing Software with the Oracle Universal Installer</i>.</p>
Auto Updates	<p>Use this screen to automatically search My Oracle Support for available patches or automatically search a local directory for patches that you've already downloaded for your organization.</p>
Installation Location	<p>Use this screen to specify the location of your Oracle home directory.</p> <p>For more information about Oracle Fusion Middleware directory structure, see "Selecting Directories for Installation and Configuration" in <i>Planning an Installation of Oracle Fusion Middleware</i>.</p>
Installation Type	<p>Use this screen to select the type of installation and consequently, the products and feature sets you want to install.</p> <ul style="list-style-type: none"> • Select Healthcare <p>NOTE: The topology in this document does not include the examples, Oracle strongly recommends that you do not install the examples into a production environment.</p>
Prerequisite Checks	<p>This screen verifies that your system meets the minimum necessary requirements.</p>

Table 18-1 (Cont.) Oracle B2B and Healthcare InstallScreens

Screen	Description
Installation Summary	<p>Use this screen to verify the installation options you selected. If you want to save these options to a response file, click Save Response File and provide the location and name of the response file. Response files can be used later in a silent installation situation.</p> <p>For more information about silent or command line installation, see "Using the Oracle Universal Installer in Silent Mode" in <i>Installing Software with the Oracle Universal Installer</i>.</p> <p>Click Install to begin the installation.</p>
Installation Progress	<p>This screen allows you to see the progress of the installation.</p> <p>Click Next when the progress bar reaches 100% complete.</p>
Installation Complete	<p>Review the information on this screen, then click Finish to dismiss the installer.</p>

18.4.3 Verifying the B2B or Healthcare Installation

After you complete the installation, you can verify it by successfully completing the following tasks.

See Also:

[Reviewing the Installation Log Files](#)

[Checking the Directory Structure](#)

[Viewing the Contents of Your Oracle Home](#)

18.4.3.1 Reviewing the Installation Log Files

Review the contents of the installation log files to make sure that no problems were encountered. For a description of the log files and where to find them, see "Understanding Installation Log Files" in *Installing Software with the Oracle Universal Installer*.

18.4.3.2 Checking the Directory Structure

The contents of your installation vary based on the options you selected during the installation.

The addition of Oracle Healthcare will add the following directory and sub-directories:

```
ls /u01/oracle/products/fmw/soa/soa/thirdparty/edifecs/

Common
STANDARDS
XEngine
```

For more information about the directory structure you should see after installation, see "What are the Key Oracle Fusion Middleware Directories?" in *Understanding Oracle Fusion Middleware*.

18.4.3.3 Viewing the Contents of Your Oracle Home

You can also view the contents of your Oracle home using the `viewInventory` script. For more information, see "Viewing the contents of an Oracle home" in *Installing Software with the Oracle Universal Installer*.

18.5 Running the Configuration Wizard for Oracle Healthcare

To extend the domain to include Oracle Healthcare, refer to the following sections.

See Also:

[Starting the Configuration Wizard](#)

[Navigating the Configuration Wizard Screens for Oracle Healthcare](#)

18.5.1 Starting the Configuration Wizard

Note:

If you added any customizations directly to the start scripts in the domain, those will be overwritten by the configuration wizard. To customize server startup parameters that apply to all servers in a domain, you can create a file called `setUserOverrides.sh` and configure it, for example, add custom libraries to the WebLogic Server classpath, specify additional java command line options for running the servers, or specify additional environment variables. Any customizations you add to this file are preserved during domain upgrade operations, and are carried over to remote servers when using the `pack` and `unpack` commands.

To begin domain configuration:

1. Shut down the Administration Server to prevent any configuration locks, saves, or activations from occurring during the configuration of the domain.

For more information, see the instructions for shutting down the Administration Server with Node Manager in [Starting the Configuration Wizard on SOAHOST1](#).

2. Navigate to the following directory and start the WebLogic Server Configuration Wizard.

```
ORACLE_HOME/oracle_common/common/bin
./config.sh
```

18.5.2 Navigating the Configuration Wizard Screens for Oracle Healthcare

Follow the instructions in this section to extend the domain for Oracle Healthcare.

Note:

You can use the same procedure described in this section to extend an existing domain. If your needs do not match the instructions given in the procedure, be sure to make your selections accordingly, or refer to the supporting documentation for additional details.

Domain creation and configuration includes the following tasks:

- [Task 1, "Selecting the Domain Type and Domain Home Location"](#)
- [Task 2, "Selecting the Configuration Template"](#)
- [Task 3, "Specifying the Datasource Configuration Type"](#)
- [Task 4, "Selecting Advanced Configuration"](#)
- [Task 5, "Reviewing Your Configuration Specifications and Configuring the Domain"](#)
- [Task 6, "Writing Down Your Domain Home and Administration Server URL"](#)
- [Task 7, "Start the Administration Server"](#)

Task 1 Selecting the Domain Type and Domain Home Location

On the Configuration Type screen, select **Update an existing domain**.

In the Domain Location field, select the value of the `ASERVER_HOME` variable, which represents the complete path to the Administration Server domain home you created in [Creating the Initial Infrastructure Domain for an Enterprise Deployment](#).

For more information about the directory location variables, see [File System and Directory Variables Used in This Guide](#)

Tip:

More information about the other options on this screen can be found in Configuration Type in [Creating WebLogic Domains Using the Configuration Wizard](#).

Task 2 Selecting the Configuration Template

On the Templates screen, make sure **Update Domain Using Product Templates** is selected, then select the following templates:

- **Oracle SOA Suite for healthcare integration — 12.2.1.0 [soa]**

Selecting this template automatically selects the following as a dependency:

Oracle B2B — 12.2.1.0 [soa]

In addition, the following additional templates should already be selected, because they were used to create the initial domain and extend it to SOA:

- Basic Weblogic Server Domain - 12.2.1.0 [wlserver]
- Oracle SOA Suite 12.2.1.0 [soa]

- Oracle Enterprise Manager - 12.2.1.0 [em]
- Oracle WSM Policy Manager - 12.2.1.0 [oracle_common]
- Oracle JRF - 12.2.1.0 [oracle_common]
- WebLogic Coherence Cluster Extension - 12.2.1.0 [wlserver]

Tip:

More information about the options on this screen can be found in Templates in Creating WebLogic Domains Using the Configuration Wizard.

Task 3 Specifying the Datasource Configuration Type

All fields are pre-populated, because you already configured the domain to reference the Fusion Middleware schemas that are required for the Infrastructure domain. Oracle Healthcare uses the existing DataSources for SOA and no new Datasources need to be added to the domain.

Note:

Any custom data sources that were created before the extension (like LEASING data sources) will show up before this screen. Check the Datasources row and click **Next**. The test data source screen will verify its validity. Click **Next**.

Task 4 Selecting Advanced Configuration

To complete domain configuration for the topology, do not select any additional options on the Advanced Configuration screen and Click **Next**. Oracle Healthcare applications and required artifacts will be targeted automatically to the existing SOA servers

Task 5 Reviewing Your Configuration Specifications and Configuring the Domain

The Configuration Summary screen contains the detailed configuration information for the domain you are about to create. Review the details of each item on the screen and verify that the information is correct.

You can go back to any previous screen if you need to make any changes, either by using the **Back** button or by selecting the screen in the navigation pane.

Domain creation will not begin until you click **Domain Update**.

Tip:

More information about the options on this screen can be found in Configuration Summary in Creating WebLogic Domains Using the Configuration Wizard.

Task 6 Writing Down Your Domain Home and Administration Server URL

The Configuration Success screen will show the following items about the domain you just configured:

- Domain Location

- Administration Server URL

You must make a note of both items as you will need them later; the domain location is needed to access the scripts used to start the Administration Server, and the URL is needed to access the Administration Server.

Click **Finish** to dismiss the configuration wizard.

Task 7 Start the Administration Server

Start the Administration Server to ensure the changes you have made to the domain have been applied.

18.6 Starting the Healthcare Components

This topic explains how to start the Oracle Healthcare components you have configured in the domain. The procedure requires you to restart the Managed Server where Oracle Healthcare has been configured. This ensures that the configuration changes and startup scripts are updated and validated correctly. Because you are extending an existing domain, the Administration Server and respective Node Managers are already running in on both application server hosts.

To start the Healthcare components, restart the Managed Servers:

1. Log into the Oracle WebLogic Server Administration Console at:


`http://ADMINVHN:7001/em`

In this example:

- Replace ADMINVHN with the host name assigned to the ADMINVHN Virtual IP address in [Identifying and Obtaining Software Downloads for an Enterprise Deployment](#).
 - Port 7001 is the typical port used for the Administration Server console and Fusion Middleware Control. However, you should use the actual URL that was displayed at the end of the Configuration Wizard session when you created the domain.
2. In the Domain Structure window, expand the **Environment** node, then select **Servers**.
The Summary of Servers page appears.
 3. Click the **Control** tab.
 4. Select **WLS_SOA1** from the **Servers** column of the table.
 5. Click **Shutdown**.
Wait for the shutdown to complete. Refresh the Console page to verify shutdown status.
 6. Select **WLS_SOA1** from the Servers column of the table.
 7. Click **Start**.
 8. Repeat steps 4 through 7 for the WLS_SOA2 Managed Server.

18.7 Updating the B2B Instance Identifier and MLLP High Availability Mode

To set up File, FTP, or Email transports in a high availability environment, set the `b2b.HAInstance` property to `true`.

1. Log in to Oracle Enterprise Manager Fusion Middleware Control with the user name and password specified for the domain administration.
2. If not already displayed, click the Target Navigation icon  in the top left corner of the page to display the **Target Navigation** pane.
3. Expand SOA, and then right click on the `soa-infra(server_name)`, and select the SOA Administration, and then B2B Server Properties.

If there are multiple `soa-infra` instances, then add the property only once.

4. Click **More B2B Configuration Properties...**

When you click this link, Fusion Middleware Control displays the MBean browser, so you can make modifications to specific B2B MBean properties.

5. Click the `b2b` MBean.

`B2BConfig b2b` should already be selected.

6. Click the **Operations** tab.
7. Click **addProperty** in the list of operations.
8. In the **Key** field enter `b2b.HAInstance`.
9. In the **Value** field enter `true`.

This property is stored in the MDS repository and must be created only once for the cluster.

10. Click **Invoke**.

A message should display, confirming that the property has been added.

11. Click **Return** to return to the B2B Properties list.
12. Click the **Operations** tab.
13. Click **addProperty** to add another property.
14. In the **Key** field enter `b2b.MLLP_HA_Mode`.
15. In the **Value** field enter `true`.
16. Click **Invoke**.

After you define high availability properties, you can view them on the Attributes tab. To view the properties, click the **Attributes** tab and then click **Properties**. Expand the Element nodes in the Value table to verify the property names and values.

18.8 Disabling Connection Factory Affinity for Optimum Load Balancing

To avoid the possibility of all the load affecting one Oracle Healthcare server, Oracle recommends that you perform the following steps.

1. Log in to the WebLogic Administration Console Control with the user name and password specified for the domain administration.
2. In the left navigation tree, expand **Services**, and then **Messaging**.
3. Click **JMS Modules**.
4. Click **SOAJMSModule** on the table.
5. Click **B2BEventQueueConnectionFactory**.
6. Click the **Load Balance** tab.
7. Click **Lock and Edit** on the Change Center Menu.
8. Clear the **Server Affinity Enabled** check box.
9. Click **Save and Activate the changes**.

18.9 Configuring Oracle HTTP Server for Oracle Healthcare

Make the following modifications to the Oracle HTTP Server instance configuration files to ensure that the Oracle HTTP Server instances in the Web tier can route Oracle Healthcare requests correctly to the Oracle SOA Suite cluster.

To enable Oracle HTTP Server to route requests to Oracle Healthcare:

1. Log in to WEBHOST1 and change directory to the configuration directory for the first Oracle HTTP Server instance (ohs1):

```
cd OHS_DOMAIN_HOME/config/fmwconfig/components/OHS/ohs1/moduleconf
```

2. Add the following directives inside the `<VirtualHost>` tag in the `soa_vh.conf` file:

```
<Location /healthcare>
  SetHandler weblogic-handler
  WebLogicCluster SOAHOST1:8001,SOAHOST2:8001
  WLProxySSL ON
  WLProxySSLPassThrough ON
</Location>
```

3. Update the `soainternal_vh.conf` file with the following directive:

```
<Location /healthcare>
  SetHandler weblogic-handler
  WebLogicCluster SOAHOST1:8001,SOAHOST2:8001
  WLProxySSL ON
  WLProxySSLPassThrough ON
</Location>
```

4. Restart the ohs1 instance:
 - a. Change directory to the following location:

```
cd OHS_DOMAIN_HOME/bin
```


- b. Enter the following commands to stop and start the instance:

```
./stopComponent.sh ohs1
./startComponent.sh ohs1
```

5. Log in to WEBHOST2 and copy the soa_vh.conf file and soainternal_vh.conf file to the configuration directory for the second Oracle HTTP Server instance (ohs_2):

```
OHS_DOMAIN_HOME/config/fmwconfig/components/OHS/ohs2/moduleconf
```

6. Edit the soa_vh.conf and soainternal_vh.conf files to change any references to WEBHOST1 to WEBHOST2.

7. Restart the ohs2 instance:

- a. Change directory to the following location:

```
cd OHS_DOMAIN_HOME/bin
```

- b. Enter the following commands to stop and start the instance:

```
./stopComponent.sh ohs2
./startComponent.sh ohs2
```

18.10 Adding the B2BAdmin Role to the SOA Administrators Group

Before you validate the Oracle B2B configuration on the Managed Servers, add the B2BAdmin administration role to the enterprise deployment administration group (SOA Administrators).

To perform this task, refer to [Configuring Roles for Administration of Oracle SOA Suite Products](#).

18.11 Validating Access to Oracle Healthcare Through the Load Balancer

Use the following steps to verify that the appropriate routing and failover is working from the load balancer to the HTTP Server instances to the Oracle Healthcare Components on the Oracle SOA Suite Managed Server.

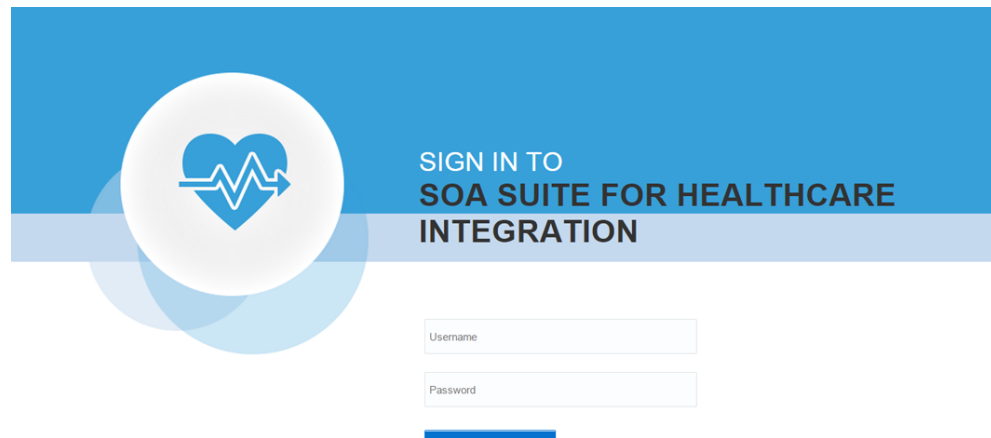
Note:

When you previously configured the Oracle SOA Suite Managed Servers, you should have enabled the WebLogic Plug-in. If have not performed this task, then see [Configuring the WebLogic Proxy Plug-In](#).

1. Enter the following URL to access the Oracle Healthcare User Interface through the load balancer:

```
https://soa.example.com/healthcare
```

You should see the Oracle Healthcare User Interface screen.



2. Log in using `weblogic_soa` user.

For more information, see “Using the Oracle SOA Suite for Healthcare Integration User Interface” in the *Healthcare Integration User’s Guide for Oracle SOA Suite*.

18.12 Backing Up the Configuration

It is an Oracle best practices recommendation to create a backup after successfully configuring a domain or at another logical point. Create a backup after verifying that the installation so far is successful. This is a quick backup for the express purpose of immediate restoration in case of problems in later steps.

The backup destination is the local disk. You can discard this backup when the enterprise deployment setup is complete. After the enterprise deployment setup is complete, you can initiate the regular deployment-specific Backup and Recovery process.

For information about backing up your configuration, see [Performing Backups and Recoveries for an Enterprise Deployment](#).

18.13 Enabling Automatic Service Migration for Oracle Healthcare

In the enterprise topology, Oracle Healthcare is configured on the existing Oracle SOA Suite Managed Servers. If you have already configured automatic service migration for the `SOA_Cluster`, then the Oracle Healthcare software is already protected by automatic service migration.

If you have not configured automatic service migration for the Managed Servers where Oracle Healthcare is configured, then see [Configuring Automatic Service Migration in an Enterprise Deployment](#).

For additional high availability, you can also configure your transaction logs store and JMS store in a database. For more information, see [Using JDBC Persistent Stores for TLOGs and JMS in an Enterprise Deployment](#).

Part IV

Common Configuration and Management Procedures for an Enterprise Deployment

The following topics contain configuration and management procedures that are required or recommended for a typical enterprise deployment.

See Also:

[Common Configuration and Management Tasks for an Enterprise Deployment](#)

[Using Whole Server Migration and Service Migration in an Enterprise Deployment](#)

[Configuring Single Sign-On for an Enterprise Deployment](#)

This chapter describes how to configure the Oracle HTTP Server WebGate to enable single sign-on with Oracle Access Manager.

Common Configuration and Management Tasks for an Enterprise Deployment

The following topics describe configuration and management tasks you will likely need to perform on the enterprise deployment environment.

See Also:

[Common Configuration and Management Procedures for an Enterprise Deployment](#)

[Configuration and Management Tasks for All Enterprise Deployments](#)

This section provides information about some typical configuration and management tasks you are likely need to perform on an Oracle Fusion Middleware enterprise deployment.

[Configuration and Management Tasks for an Oracle SOA Suite Enterprise Deployment](#)

This section describes some of the key configuration and management tasks you will likely need to perform on an Oracle SOA Suite enterprise deployment.

19.1 Configuration and Management Tasks for All Enterprise Deployments

This section provides information about some typical configuration and management tasks you are likely need to perform on an Oracle Fusion Middleware enterprise deployment.

See Also:

[Verifying Manual Failover of the Administration Server](#)

In case a host computer fails, you can fail over the Administration Server to another host. The following sections provide the steps to verify the

failover and failback of the Administration Server from SOAHOST1 and SOAHOST2.

[Enabling SSL Communication Between the Middle Tier and the Hardware Load Balancer](#)

This section describes how to enable SSL communication between the middle tier and the hardware load balancer.

[Configuring Roles for Administration of an Enterprise Deployment](#)

This section provides a summary of products with specific administration roles and provides instructions to add a product-specific administration role to the Enterprise Deployment Administration group.

[Using JDBC Persistent Stores for TLOGs and JMS in an Enterprise Deployment](#)

The following topics provide guidelines for when to use JDBC persistent stores for transaction logs (TLOGs) and JMS. They also provide the procedures you can use to configure the persistent stores in a supported database.

[Performing Backups and Recoveries for an Enterprise Deployment](#)

This section provides guidelines for making sure you back up the necessary directories and configuration data for an Oracle SOA Suite enterprise deployment.

19.1.1 Verifying Manual Failover of the Administration Server

In case a host computer fails, you can fail over the Administration Server to another host. The following sections provide the steps to verify the failover and failback of the Administration Server from SOAHOST1 and SOAHOST2.

Assumptions:

- The Administration Server is configured to listen on ADMINVHN, and not on localhost or ANY address.

For more information about the ADMINVHN virtual IP address, see [Reserving the Required IP Addresses for an Enterprise Deployment](#).

- These procedures assume that the Administration Server domain home (ASERVER_HOME) has been mounted on both host computers. This ensures that the Administration Server domain configuration files and the persistent stores are saved on the shared storage device.
- The Administration Server is failed over from SOAHOST1 to SOAHOST2, and the two nodes have these IPs:
 - SOAHOST1: 100.200.140.165
 - SOAHOST2: 100.200.140.205
 - ADMINVHN : 100.200.140.206. This is the Virtual IP where the Administration Server is running, assigned to ethX:Y, available in SOAHOST1 and SOAHOST2.
- Oracle WebLogic Server and Oracle Fusion Middleware components have been installed in SOAHOST2 as described in the specific configuration chapters in this guide.

Specifically, both host computers use the exact same path to reference the binary files in the Oracle home.

The following topics provide details on how to perform a test of the Administration failover procedure.

See Also:

[Failing Over the Administration Server When Using a Per Host Node Manager](#)

The following procedure shows how to fail over the Administration Server to a different node (SOAHOST2). Note that even after failover, the Administration Server will still use the same Oracle WebLogic Server *machine* (which is a logical machine, not a physical machine).

[Validating Access to the Administration Server on SOAHOST2 Through Oracle HTTP Server](#)

After you perform a manual failover of the Administration Server, it is important to verify that you can access the Administration Server, using the standard administration URLs.

[Failing the Administration Server Back to SOAHOST1 When Using a Per Host Node Manager](#)

After you have tested a manual Administration Server failover, and after you have validated that you can access the administration URLs after the failover, you can then migrate the Administration Server back to its original host.

19.1.1.1 Failing Over the Administration Server When Using a Per Host Node Manager

The following procedure shows how to fail over the Administration Server to a different node (SOAHOST2). Note that even after failover, the Administration Server will still use the same Oracle WebLogic Server *machine* (which is a logical machine, not a physical machine).

This procedure assumes you've configured a per host Node Manager for the enterprise topology, as described in [Creating a Per Host Node Manager Configuration](#). For more information, see [About the Node Manager Configuration in a Typical Enterprise Deployment](#).

To fail over the Administration Server to a different host:

1. Stop the Administration Server on SOAHOST1.
2. Stop the Node Manager on SOAHOST1.

If you started the per host Node Manager using the procedure earlier in this guide, then return to the terminal window where you started the Node Manager and press Ctrl/C to stop the process.

3. Migrate the ADMINVHN virtual IP address to the second host:
 - a. Run the following command as root on SOAHOST1 (where X:Y is the current interface used by ADMINVHN):

```
/sbin/ifconfig ethX:Y down
```

- b. Run the following command as root on SOAHOST2:

```
/sbin/ifconfig <interface:index> ADMINVHN netmask <netmask>
```

For example:

```
/sbin/ifconfig eth0:1 100.200.140.206 netmask 255.255.255.0
```

Note:

Ensure that the netmask and interface to be used to match the available network configuration in SOAHOST2.

4. Update the routing tables using `arping`, for example:

```
/sbin/arping -q -U -c 3 -I eth0 100.200.140.206
```

5. From SOAHOST1, change directory to the Node Manager home directory:

```
cd NM_HOME
```

6. Edit the `nodemanager.domains` file and remove the reference to `MSERVER_HOME`.

The resulting entry in the SOAHOST1 `nodemanager.domains` file should appear as follows:

```
soaedg_domain=MSERVER_HOME;
```

7. From SOAHOST2, change directory to the Node Manager home directory:

```
cd NM_HOME
```

8. Edit the `nodemanager.domains` file and add the reference the `MSERVER_HOME`.

The resulting entry in the SOAHOST2 `nodemanager.domains` file should appear as follows:

```
soaedg_domain=MSERVER_HOME;ASERVER_HOME
```

9. Start the Node Manager on SOAHOST1 and restart the Node Manager on SOAHOST2.

10. Start the Administration Server on SOAHOST2.

11. Test that you can access the Administration Server on SOAHOST2 as follows:

- a. Ensure that you can access the Oracle WebLogic Server Administration Console using the following URL:

```
http://ADMINVHN:7001/console
```

- b. Check that you can access and verify the status of components in Fusion Middleware Control using the following URL:

```
http://ADMINVHN:7001/em
```

19.1.1.2 Validating Access to the Administration Server on SOAHOST2 Through Oracle HTTP Server

After you perform a manual failover of the Administration Server, it is important to verify that you can access the Administration Server, using the standard administration URLs.

From the load balancer, access the following URLs to ensure that you can access the Administration Server when it is running on SOAHOST2:

- <http://admin.example.com/console>

This URL should display the WebLogic Server Administration console.

- <http://admin.example.com/em>

This URL should display Oracle Enterprise Manager Fusion Middleware Control.

19.1.1.3 Failing the Administration Server Back to SOAHOST1 When Using a Per Host Node Manager

After you have tested a manual Administration Server failover, and after you have validated that you can access the administration URLs after the failover, you can then migrate the Administration Server back to its original host.

This procedure assumes you've configured a per host Node Manager for the enterprise topology, as described in [Creating a Per Host Node Manager Configuration](#). For more information, see [About the Node Manager Configuration in a Typical Enterprise Deployment](#).

1. Stop the Administration Server on SOAHOST2.
2. Stop the Node Manager on SOAHOST2.
3. Run the following command as root on SOAHOST2.

```
/sbin/ifconfig ethZ:N down
```

4. Run the following command as root on SOAHOST1:

```
/sbin/ifconfig ethX:Y 100.200.140.206 netmask 255.255.255.0
```

Note:

Ensure that the netmask and interface to be used match the available network configuration in SOAHOST1

5. Update the routing tables using `arping` on SOAHOST1:

```
/sbin/arping -q -U -c 3 -I ethX 100.200.140.206
```
6. From SOAHOST2, change directory to the Node Manager home directory:

```
cd NM_HOME
```
7. Edit the `nodemanager.domains` file and remove the reference to `MSERVER_HOME`.
8. From SOAHOST1, change directory to the Node Manager home directory:

```
cd NM_HOME
```
9. Edit the `nodemanager.domains` file and add the reference the `MSERVER_HOME`.
10. Start the Node Manager on SOAHOST2 and restart the Node Manager on SOAHOST1.
11. Start the Administration Server on SOAHOST1.

12. Test that you can access the Oracle WebLogic Server Administration Console using the following URL:

`http://ADMINVHN:7001/console`

13. Check that you can access and verify the status of components in the Oracle Enterprise Manager using the following URL:

`http://ADMINVHN:7001/em`

19.1.2 Enabling SSL Communication Between the Middle Tier and the Hardware Load Balancer

This section describes how to enable SSL communication between the middle tier and the hardware load balancer.

Note:

This steps are applicable if the hardware load balancer is configured with SSL and the front end address of the system has been secured accordingly.

See Also:

[When is SSL Communication Between the Middle Tier and Load Balancer Necessary?](#)

[Generating Self-Signed Certificates Using the `utils.CertGen` Utility](#)

[Creating an Identity Keystore Using the `utils.ImportPrivateKey` Utility](#)

[Creating a Trust Keystore Using the `Keytool` Utility](#)

[Importing the Load Balancer Certificate into the Trust Store](#)

[Adding the Updated Trust Store to the Oracle WebLogic Server Start Scripts](#)

[Configuring WebLogic Servers to Use the Custom Keystores](#)

[Testing Composites Using SSL Endpoints](#)

19.1.2.1 When is SSL Communication Between the Middle Tier and Load Balancer Necessary?

In an enterprise deployment, there are scenarios where the software running on the middle tier must access the front-end SSL address of the hardware load balancer. In these scenarios, an appropriate SSL handshake must take place between the load balancer and the invoking servers. This handshake is not possible unless the Administration Server and Managed Servers on the middle tier are started using the appropriate SSL configuration.

For example, in an Oracle SOA Suite enterprise deployment, the following examples apply:

- Oracle Business Process Management requires access to the front-end load balancer URL when it attempts to retrieve role information through specific Web services.
- Oracle Service Bus performs invocations to endpoints exposed in the Load Balancer SSL virtual servers.

- Oracle SOA Suite composite applications and services often generate callbacks that need to perform invocations using the SSL address exposed in the load balancer.
- Finally, when you test a SOA Web services endpoint in Oracle Enterprise Manager Fusion Middleware Control, the Fusion Middleware Control software running on the Administration Server must access the load balancer front-end to validate the endpoint.

19.1.2.2 Generating Self-Signed Certificates Using the `utils.CertGen` Utility

This section describes the procedure for creating self-signed certificates on SOAHOST1. Create these certificates using the network name or alias of the host.

The directory where keystores and trust keystores are maintained must be on shared storage that is accessible from all nodes so that when the servers fail over (manually or with server migration), the appropriate certificates can be accessed from the failover node. Oracle recommends using central or shared stores for the certificates used for different purposes (for example, SSL set up for HTTP invocations). In this case, SOAHOST2 uses the cert directory created for SOAHOST1certificates.

For information on using trust CA certificates instead, see the information about configuring identity and trust in *Oracle Fusion Middleware Securing Oracle WebLogic Server*.

About Passwords

The passwords used in this guide are used only as examples. Use secure passwords in a production environment. For example, use passwords that include both uppercase and lowercase characters as well as numbers.

To create self-signed certificates:

1. Set up your environment by running the `WL_HOME/server/bin/setWLSEnv.sh` script:
2. Verify that the `CLASSPATH` environment variable is set:

```
echo $CLASSPATH
```

3. Verify that the shared configuration directory folder has been created and mounted to shared storage correctly as described in [Preparing the File System for an Enterprise Deployment](#).

For example, use the following command to verify that the shared configuration directory is available to each host:

```
df -h | grep -B1 SHARED_CONFIG_DIR
```

Replace `SHARED_CONFIG_DIR` with the actual path to your shared configuration directory.

You can also do a listing of the directory to ensure it is available to the host:

```
ls -al SHARED_CONFIG_DIR
```

4. Create the keystore home folder structure if does not already exist.

For example:

```
cd SHARED_CONFIG_DIR
mkdir keystores
```

```
chown oracle:oinstall keystores
chmod 750 keystores
```

5. Change directory to the keystore home:

```
cd KEYSTORE_HOME
```

6. Run the `utils.CertGen` tool to create the certificates for both the physical host names and the virtual host names used by servers in the node.

Syntax:

```
java utils.CertGen key_passphrase cert_file_name key_file_name [export |
domestic] [hostname]
```

Examples:

```
java utils.CertGen password ADMINVHN.example.com_cert \
ADMINVHN.example.com_key domestic ADMINVHN.example.com
```

```
java utils.CertGen password SOAHOST1.example.com_cert \
SOAHOST1.example.com_key domestic SOAHOST1.example.com
```

19.1.2.3 Creating an Identity Keystore Using the `utils.ImportPrivateKey` Utility

This section describes how to create an Identity Keystore on `SOAHOST1.example.com`.

In previous sections you have created certificates and keys that reside in a shared storage. In this section, the certificate and private key for both `SOAHOST1` and `ADMINVHN` are imported into a new Identity Store. Make sure that you use a different alias for each of the certificate/key pair imported.

Note:

The Identity Store is created (if none exists) when you import a certificate and the corresponding key into the Identity Store using the `utils.ImportPrivateKey` utility.

1. Import the certificate and private key for `ADMINVHN` and `SOAHOST1` into the Identity Store. Make sure that you use a different alias for each of the certificate/key pair imported.

Syntax:

```
java utils.ImportPrivateKey
  -certfile cert_file
  -keyfile private_key_file
  [-keyfilepass private_key_password]
  -keystore keystore
  -storepass storepass
  [-storetype storetype]
  -alias alias
  [-keypass keypass]
```

Note:

Default keystore_type is jks.

Examples:

```
Java utils.ImportPrivateKey
-certfile KEYSTORE_HOME/ADMINVHN.example.com_cert.pem
-keyfile KEYSTORE_HOME/ADMINVHN.example.com_key.pem
-keyfilepass password
-keystore appIdentityKeyStore.jks
-storepass password
-alias ADMINVHN
-keypass password
```

```
java utils.ImportPrivateKey
-certfile KEYSTORE_HOME/SOAHOST1.example.com_cert.pem
-keyfile KEYSTORE_HOME/SOAHOST1.example.com_key.pem
-keyfilepass password
-keystore appIdentityKeyStore.jks
-storepass password
-alias SOAHOST1
-keypass password
```

2. Repeat the above step for all the remaining hosts used in the system (for example, SOAHOST2).

19.1.2.4 Creating a Trust Keystore Using the Keytool Utility

To create the Trust Keystore on SOAHOST1.example.com.

1. Copy the standard java keystore to create the new trust keystore since it already contains most of the root CA certificates needed.

Oracle does not recommend modifying the standard Java trust key store directly. Copy the standard Java keystore CA certificates located under the WL_HOME/server/lib directory to the same directory as the certificates. For example:

```
cp WL_HOME/server/lib/cacerts KEYSTORE_HOME/appTrustKeyStore.jks
```

2. Use the keytool utility to change the default password.

The default password for the standard Java keystore is `changeit`. Oracle recommends always changing the default password, as follows:

```
keytool -storepasswd -new NewPassword -keystore TrustKeyStore -storepass
Original_Password
```

For example:

```
keytool -storepasswd -new password -keystore appTrustKeyStore.jks -storepass
changeit
```

3. Import the CA certificate into the appTrustKeyStore using the keytool utility.

The CA certificate `CertGenCA.der` is used to sign all certificates generated by the `utils.CertGen` tool and is located at `WL_HOME/server/lib` directory.

Use the following syntax to import the certificate:

```
keytool -import -v -noprompt -trustcacerts -alias AliasName -file CAFileLocation -  
keystore KeyStoreLocation -storepass KeyStore_Password
```

For example:

```
keytool -import -v -noprompt -trustcacerts -alias clientCACert -file WL_HOME/  
server/lib/CertGenCA.der -keystore appTrustKeyStore.jks -storepass password
```

19.1.2.5 Importing the Load Balancer Certificate into the Trust Store

For the SSL handshake to behave properly, the load balancer's certificate needs to be added to the WLS servers trust store. For adding it, follow these steps:

1. Access the site on SSL with a browser (this add the server's certificate to the browser's repository).
2. From the browser's certificate management tool, export the certificate to a file that is on the server's file system (with a file name like `soa.example.com`).
3. Use the keytool to import the load balancer's certificate into the truststore:

```
keytool -import -file soa.example.com -v -keystore appTrustKeyStore.jks
```

19.1.2.6 Adding the Updated Trust Store to the Oracle WebLogic Server Start Scripts

To ensure each server can access the updated trust store, edit the `setDomainEnv.sh` script in each of the domain home directories in the enterprise deployment.

1. Log in to SOAHOST1 and open the following file with a text editor:

```
ASERVER_HOME/bin/setDomainEnv.sh
```

2. Replace reference to the existing `DemoTrustStore` entry with the location of the `appTrustKeyStore.jks` file.

Note that all the values for `EXTRA_JAVA_PROPERTIES` must be on one line in the file, followed by the export command on a new line. For example:

```
EXTRA_JAVA_PROPERTIES="{EXTRA_JAVA_PROPERTIES}  
-Djavax.net.ssl.trustStore=KEYSTORE_HOME/appTrustKeyStore.jks..."  
export EXTRA_JAVA_PROPERTIES
```

3. Make the same change to the `setDomainEnv.sh` file in the `MSERVER_HOME/bin` directory on SOAHOST1 and SOAHOST2.

Alternatively, you can use the `setUserOverrides.sh` file to place these options, this way they will not get overwritten when the domain's scripts are extended with the configuration wizard or updated with unpack operations. For more information, see "Customizing Domain Wide Server Parameters" in *Administering Server Startup and Shutdown for Oracle WebLogic Server*.

19.1.2.7 Configuring WebLogic Servers to Use the Custom Keystores

Configure the WebLogic Servers to use the custom keystores using the Oracle WebLogic Server Administration Console. Complete this procedure for the Administration Server and the Managed Servers that require access to the front end LBR on SSL.

To configure the identity and trust keystores:

1. Log in to the Administration Console, and click **Lock & Edit**.
2. In the left pane, expand **Environment**, and select **Servers**.
3. Click the name of the server for which you want to configure the identity and trust keystores.
4. Select **Configuration**, and then **Keystores**.
5. In the **Keystores** field, click **Change**, and select **Custom Identity and Custom Trust** method for storing and managing private keys/digital certificate pairs and trusted CA certificates, and click Save.
6. In the Identity section, define attributes for the identity keystore.
 - Custom Identity Keystore: Enter the fully qualified path to the identity keystore:
`KEYSTORE_HOME/appIdentityKeyStore.jks`
 - Custom Identity Keystore Type: Leave this field blank, it defaults to JKS.
 - Custom Identity Keystore Passphrase: Enter the password Keystore_Password you provided in [Creating an Identity Keystore Using the utils.ImportPrivateKey Utility](#)

This attribute may be optional or required depending on the type of keystore. All keystores require the passphrase in order to write to the keystore. However, some keystores do not require the passphrase to read from the keystore. WebLogic Server reads only from the keystore, so whether or not you define this property depends on the requirements of the keystore.
7. In the Trust section, define properties for the trust keystore:
 - Custom Trust Keystore: Enter the fully qualified path to the trust keystore:
`KEYSTORE_HOME/appTrustKeyStore.jks`
 - Custom Trust Keystore Type: Leave this field blank, it defaults to JKS.
 - Custom Trust Keystore Passphrase: The password you provided in as New_Password in "Creating a Trust Keystore Using the Keytool Utility."

As mentioned in the previous step, this attribute may be optional or required depending on the type of keystore.
8. Click **Save**.
9. To activate these changes, in the Change Center of the Administration Console, click **Activate Changes**.
10. Click **Lock & Edit**.
11. Select **Configuration**, then **SSL**.

12. In the Private Key Alias field, enter the alias you used for the host name the managed server listens on.

In the Private Key Passphrase and the Confirm Private Key Passphrase fields, enter the password for the keystore that you created in [Creating an Identity Keystore Using the `utils.ImportPrivateKey` Utility](#)
13. Click **Save**.
14. Click **Activate Changes** in the Administration Console's Change Center to make the changes take effect.
15. Restart the server for which the changes have been applied. The fact that servers can be restarted using the Administration Console/Node Manager is a good verification that the communication between Node Manager, Administration Server and the managed servers is correct.

19.1.2.8 Testing Composites Using SSL Endpoints

Once SSL has been enabled, composites endpoints can be verified on SSL from Oracle Enterprise Manager FMW Control. To test a SSL endpoint follow this steps:

1. Enter the following URL into a browser to display the Fusion Middleware Control login screen:

```
http://ADMINVHN:7001/em
```

In this example:

- Replace ADMINVHN with the host name assigned to the ADMINVHN Virtual IP address in [Identifying and Obtaining Software Downloads for an Enterprise Deployment](#).
 - Port 7001 is the typical port used for the Administration Server console and Fusion Middleware Control. However, you should use the actual URL that was displayed at the end of the Configuration Wizard session when you created the domain.
2. Log in to Fusion Middleware Control using the Administration Server credentials.
 3. From the tree on the left, expand SOA. Click **soa-infra(WLS_SOAn)** and select the partition that the composite was deployed to.
 4. Select the composite.
 5. From the right pane, click on the Test tab.
 6. In the WSDL or WADL address, replace `http://soa.example.com:80` with `soa.example.com`.
 7. Click **Parse WSDL or WADL**.

8. Verify that the Endpoint URL shown is SSL.
9. Test the composite. If the response is as expected for the web service, the SSL communication between the Administration Server and the Load Balancer has been configured properly.

19.1.3 Configuring Roles for Administration of an Enterprise Deployment

This section provides a summary of products with specific administration roles and provides instructions to add a product-specific administration role to the Enterprise Deployment Administration group.

Each enterprise deployment consists of multiple products. Some of the products have specific administration users, roles, or groups that are used to control administration access to each product.

However, for an enterprise deployment, which consists of multiple products, you can use a single LDAP-based authorization provider and a single administration user and group to control access to all aspects of the deployment. For more information about creating the authorization provider and provisioning the enterprise deployment administration user and group, see [Creating a New LDAP Authenticator and Provisioning a New Enterprise Deployment Administrator User and Group](#).

To be sure that you can manage each product effectively within the single enterprise deployment domain, you must understand which products require specific administration roles or groups, you must know how to add any specific product administration roles to the single, common enterprise deployment administration group, and if necessary, you must know how to add the enterprise deployment administration user to any required product-specific administration groups.

For more information, see the following topics.

See Also:

[Summary of Products with Specific Administration Roles](#)

[Summary of Oracle SOA Suite Products with Specific Administration Groups](#)

[Adding a Product-Specific Administration Role to the Enterprise Deployment Administration Group](#)

[Adding the Enterprise Deployment Administration User to a Product-Specific Administration Group](#)

19.1.3.1 Summary of Products with Specific Administration Roles

The following table lists the Fusion Middleware products that have specific administration roles, which must be added to the enterprise deployment administration group (SOA Administrators), which you defined in the LDAP Authorization Provider for the enterprise deployment.

Use the information in the following table and the instructions in [Adding a Product-Specific Administration Role to the Enterprise Deployment Administration Group](#) to add the required administration roles to the enterprise deployment Administration group.

Product	Application Stripe	Administration Role to be Assigned
Oracle Web Services Manager	wsm-pm	policy.updater
SOA Infrastructure	soa-infra	SOAAdmin
Oracle Service Bus	Service_Bus_Console	MiddlewareAdministrator
Enterprise Scheduler Service	ESSAPP	ESSAdmin
Oracle B2B	b2bui	B2BAdmin

19.1.3.2 Summary of Oracle SOA Suite Products with Specific Administration Groups

[Table 19-1](#) lists the Oracle SOA Suite products that need to use specific administration groups.

For each of these components, the common enterprise deployment Administration user must be added to the product-specific Administration group; otherwise, you won't be able to manage the product resources using the enterprise manager administration user that you created in [Provisioning an Enterprise Deployment Administration User and Group](#).

Use the information in [Table 19-1](#) and the instructions in [Adding the Enterprise Deployment Administration User to a Product-Specific Administration Group](#) to add the required administration roles to the enterprise deployment Administration group.

Table 19-1 Oracle SOA Suite Products with a Product-Specific Administration Group


Product	Product-Specific Administration Group
Oracle Business Activity Monitoring	BAMAdministrators
Oracle Business Process Management	Administrators
Oracle Service Bus Integration	IntegrationAdministrators

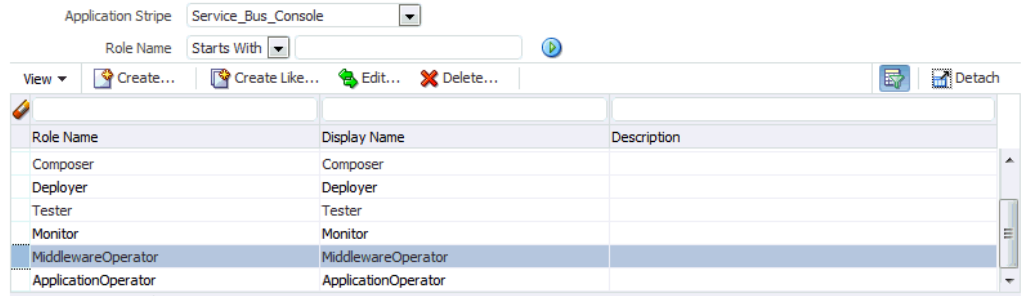
19.1.3.3 Adding a Product-Specific Administration Role to the Enterprise Deployment Administration Group



For products that require a product-specific administration role, use the following procedure to add the role to the enterprise deployment administration group:

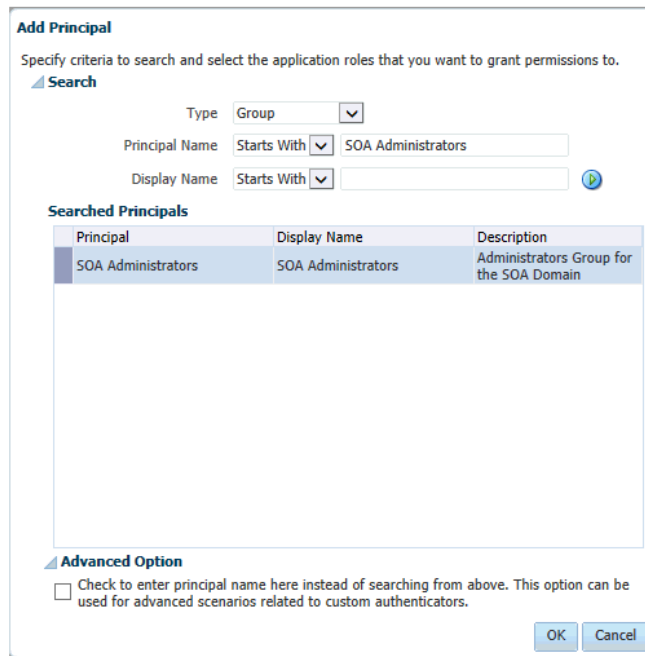
1. Use the Oracle WebLogic Server Administration Server credentials to log in to Oracle Enterprise Manager Fusion Middleware Control.

These are the credentials you created when you initially configured the domain and created the Oracle WebLogic Server Administration user name (typically, `weblogic_soa`) and password.
2. From the **WebLogic Domain** menu, select **Security**, and then **Application Roles**.
3. For each production-specific application role, select the corresponding application stripe from the **Application Stripe** drop-down menu.

4. Click Search Application Roles icon  to display all the application roles available in the domain.
5. Select the row for the application role you are adding to the enterprise deployment administration group.



6. Click the Edit icon  to edit the role.
7. Click the Add icon  on the Edit Application Role page.
8. In the Add Principal dialog box, select **Group** from the **Type** drop-down menu.
9. Search for the enterprise deployment administrators group, by entering the group name (for example, SOA Administrators) in the **Principal Name Starts With** field and clicking the right arrow to start the search.
10. Select the administrator group in the search results and click **OK**.



11. Click **OK** on the Edit Application Role page.

19.1.3.4 Adding the Enterprise Deployment Administration User to a Product-Specific Administration Group

For products with a product-specific administration group, use the following procedure to add the enterprise deployment administration user (`weblogic_soa`) to the group. This will allow you to manage the product using the enterprise manager administrator user:

1. Create an ldif file called `product_admin_group.ldif` similar to the one shown below:

```
dn: cn=product-specific_group_name, cn=groups, dc=us, dc=oracle, dc=com
displayname: product-specific_group_display_name
objectclass: top
objectclass: groupOfUniqueNames
objectclass: orclGroup
uniquemember: cn=weblogic_soa, cn=users, dc=us, dc=oracle, dc=com
cn: product-specific_group_name
description: Administrators Group for the OSB Domain
```

In this example, replace `product-specific_group_name` with the actual name of the product administrator group, as shown in [Table 19-1](#).

Replace `product-specific_group_display_name` with the display name for the group that appears in the management console for the LDAP server and in the Oracle WebLogic Server Administration Console.

2. Use the ldif file to add the enterprise deployment administrator user to the product-specific administration group.

For Oracle Unified Directory:

```
OID_INSTANCE_HOME/bin/ldapmodify -a
-D "cn=Administrator"
-X
-p 1389
-f product_admin_group.ldif
```

For Oracle Internet Directory:

```
OID_ORACLE_HOME/bin/ldapadd -h oid.example.com
-p 389
-D cn="orcladmin"
-w <password>
-c
-v
-f product_admin_group.ldif
```

19.1.4 Using JDBC Persistent Stores for TLOGs and JMS in an Enterprise Deployment

The following topics provide guidelines for when to use JDBC persistent stores for transaction logs (TLOGs) and JMS. They also provide the procedures you can use to configure the persistent stores in a supported database.

See Also:

[About JDBC Persistent Stores for JMS and TLOGs](#)

Oracle Fusion Middleware supports both database-based and file-based persistent stores for Oracle WebLogic Server transaction logs (TLOGs) and JMS. Before deciding on a persistent store strategy for your

environment, consider the advantages and disadvantages of each approach.

[Performance Impact of the TLOGs and JMS Persistent Stores](#)

One of the primary considerations when selecting a storage method for Transaction Logs and JMS persistent stores is the potential impact on performance. This topic provides some guidelines and details to help you determine the performance impact of using JDBC persistent stores for TLOGs and JMS.

[Roadmap for Configuring a JDBC Persistent Store for TLOGs](#)

The following topics describe how to configure a database-based persistent store for JMS.

[Roadmap for Configuring a JDBC Persistent Store for JMS](#)

The following topics describe how to configure a database-based persistent store for JMS.

[Creating a User and Tablespace for TLOGs](#)

Before you can create a database-based persistent store for transaction logs, you must create a user and tablespace in a supported database.

[Creating a User and Tablespace for JMS](#)

Before you can create a database-based persistent store for JMS, you must create a user and tablespace in a supported database.

[Creating GridLink Data Sources for TLOGs and JMS Stores](#)

Before you can configure database-based persistent stores for JMS and TLOGs, you must create two data sources: one for the TLOGs persistent store and one for the JMS persistent store.

[Assigning the TLOGs JDBC store to the Managed Servers](#)

After you create the tablespace and user in the database, and you have created the datasource, you can then assign the TLOGs persistence store to each of the required Managed Servers.

[Creating a JDBC JMS Store](#)

After you create the JMS persistent store user and table space in the database, and after you create the data source for the JMS persistent store, you can then use the Administration Console to create the store.

[Assigning the JMS JDBC store to the JMS Servers](#)

After you create the JMS tablespace and user in the database, create the JMS datasource, and create the JDBC store, then you can then assign the JMS persistence store to each of the required JMS Servers.

[Creating the Required Tables for the JMS JDBC Store](#)

The final step in using a JDBC persistent store for JMS is to create the required JDBC store tables. Perform this task before restarting the Managed Servers in the domain.

19.1.4.1 About JDBC Persistent Stores for JMS and TLOGs

Oracle Fusion Middleware supports both database-based and file-based persistent stores for Oracle WebLogic Server transaction logs (TLOGs) and JMS. Before deciding

on a persistent store strategy for your environment, consider the advantages and disadvantages of each approach.

Note:

Regardless of which storage method you choose, Oracle recommends that for transaction integrity and consistency, you use the same type of store for both JMS and TLOGs.

When you store your TLOGs and JMS data in an Oracle database, you can take advantage of the replication and high availability features of the database. For example, you can use OracleData Guard to simplify cross-site synchronization. This is especially important if you are deploying Oracle Fusion Middleware in a disaster recovery configuration.

Storing TLOGs and JMS data in a database also means you don't have to identify a specific shared storage location for this data. Note, however, that shared storage is still required for other aspects of an enterprise deployment. For example, it is necessary for Administration Server configuration (to support Administration Server failover), for deployment plans, and for adapter artifacts, such as the File/FTP Adapter control and processed files.

If you are storing TLOGs and JMS stores on a shared storage device, then you can protect this data by using the appropriate replication and backup strategy to guarantee zero data loss, and you will potentially realize better system performance. However, the file system protection will always be inferior to the protection provided by an Oracle Database.

For more information about the potential performance impact of using a database-based TLOGs and JMS store, see [Performance Impact of the TLOGs and JMS Persistent Stores](#).

19.1.4.2 Performance Impact of the TLOGs and JMS Persistent Stores

One of the primary considerations when selecting a storage method for Transaction Logs and JMS persistent stores is the potential impact on performance. This topic provides some guidelines and details to help you determine the performance impact of using JDBC persistent stores for TLOGs and JMS.

Performance Impact of Transaction Logs Versus JMS Stores

For transaction logs, the impact of using a JDBC store is relatively small, because the logs are very transient in nature. Typically, the effect is minimal when compared to other database operations in the system.

On the other hand, JMS database stores can have a higher impact on performance if the application is JMS intensive. For example, the impact of switching from a file-based to database-based persistent store is very low when you are using the SOA Fusion Order Demo (a sample application used to test Oracle SOA Suite environments), because the JMS database operations are masked by many other SOA database invocations that are much heavier.

Factors that Affect Performance

There are multiple factors that can affect the performance of a system when it is using JMS DB stores for custom destinations. The main ones are:

- Custom destinations involved and their type

- Payloads being persisted
- Concurrency on the SOA system (producers on consumers for the destinations)

Depending on the effect of each one of the above, different settings can be configured in the following areas to improve performance:

- Type of data types used for the JMS table (using raw vs. lobs)
- Segment definition for the JMS table (partitions at index and table level)

Impact of JMS Topics

If your system uses Topics intensively, then as concurrency increases, the performance degradation with an Oracle RAC database will increase more than for Queues. In tests conducted by Oracle with JMS, the average performance degradation for different payload sizes and different concurrency was less than 30% for Queues. For topics, the impact was more than 40%. Consider the importance of these destinations from the recovery perspective when deciding whether to use database stores.

Impact of Data Type and Payload Size

When choosing to use the RAW or SecureFiles LOB data type for the payloads, consider the size of the payload being persisted. For example, when payload sizes range between 100b and 20k, then the amount of database time required by SecureFiles LOB is slightly higher than for the RAW data type.

More specifically, when the payload size reach around 4k, then SecureFiles tend to require more database time. This is because 4k is where writes move out-of-row. At around 20k payload size, SecureFiles data starts being more efficient. When payload sizes increase to more than 20k, then the database time becomes worse for payloads set to the RAW data type.

One additional advantage for SecureFiles is that the database time incurred stabilizes with payload increases starting at 500k. In other words, at that point it is not relevant (for SecureFiles) whether the data is storing 500k, 1MB or 2MB payloads, because the write is asynchronous, and the contention is the same in all cases.

The effect of concurrency (producers and consumers) on the queue's throughput is similar for both RAW and SecureFiles until the payload sizes reeach 50K. For small payloads, the effect on varying concurrency is practically the same, with slightly better scalability for RAW. Scalability is better for SecureFiles when the payloads are above 50k.

Impact of Concurrency, Worker Threads, and Database Partitioning

Concurrency and worker threads defined for the persistent store can cause contention in the RAC database at the index and global cache level. Using a reverse index when enabling multiple worker threads in one single server or using multiple Oracle WebLogic Server clusters can improve things. However, if the Oracle Database partitioning option is available, then global hash partition for indexes should be used instead. This reduces the contention on the index and the global cache buffer waits, which in turn improves the response time of the application. Partitioning works well in all cases, some of which will not see significant improvements with a reverse index.

19.1.4.3 Roadmap for Configuring a JDBC Persistent Store for TLOGs

The following topics describe how to configure a database-based persistent store for JMS.

1. [Creating a User and Tablespace for TLOGs](#)

2. [Creating GridLink Data Sources for TLOGs and JMS Stores](#)
3. [Assigning the TLOGs JDBC store to the Managed Servers](#)

19.1.4.4 Roadmap for Configuring a JDBC Persistent Store for JMS

The following topics describe how to configure a database-based persistent store for JMS.

1. [Creating a User and Tablespace for JMS](#)
2. [Creating GridLink Data Sources for TLOGs and JMS Stores](#)
3. [Creating a JDBC JMS Store](#)
4. [Assigning the JMS JDBC store to the JMS Servers](#)
5. [Creating the Required Tables for the JMS JDBC Store](#)

19.1.4.5 Creating a User and Tablespace for TLOGs

Before you can create a database-based persistent store for transaction logs, you must create a user and tablespace in a supported database.

1. Create a tablespace called `tlogs`.

For example, log in to SQL*Plus as the `sysdba` user and run the following command:

```
SQL> create tablespace tlogs
      logging datafile 'DB_HOME/oradata/orcl/tlogs.dbf'
      size 32m autoextend on next 32m maxsize 2048m extent management local;
```

2. Create a user named `TLOGS` and assign to it the `tlogs` tablespace.

For example:

```
SQL> create user TLOGS identified by password;

SQL> grant create table to TLOGS;

SQL> grant create session to TLOGS;

SQL> alter user TLOGS default tablespace tlogs;

SQL> alter user TLOGS quota unlimited on tlogs;
```

19.1.4.6 Creating a User and Tablespace for JMS

Before you can create a database-based persistent store for JMS, you must create a user and tablespace in a supported database.

1. Create a tablespace called `jms`.

For example, log in to SQL*Plus as the `sysdba` user and run the following command:

```
SQL> create tablespace jms
      logging datafile 'DB_HOME/oradata/orcl/jms.dbf'
      size 32m autoextend on next 32m maxsize 2048m extent management local;
```

2. Create a user named `JMS` and assign to it the `jms` tablespace.

For example:

```
SQL> create user JMS identified by password;
```

```
SQL> grant create table to JMS;
```

```
SQL> grant create session to JMS;
```

```
SQL> alter user JMS default tablespace jms;
```

```
SQL> alter user TLOG quota unlimited on jms;
```

19.1.4.7 Creating GridLink Data Sources for TLOGs and JMS Stores

Before you can configure database-based persistent stores for JMS and TLOGs, you must create two data sources: one for the TLOGs persistent store and one for the JMS persistent store.

For an enterprise deployment, you should use GridLink data sources for your TLOGs and JMS stores. To create a GridLink data source:

1. Log in to the Oracle WebLogic Server Administration Console.
2. If you have not already done so, in the **Change Center**, click **Lock & Edit**.
3. In the **Domain Structure** tree, expand **Services**, then select **Data Sources**.
4. On the Summary of Data Sources page, click **New** and select **GridLink Data Source**, and enter the following:
 - Enter a logical name for the data source in the **Name** field.
For the TLOGs store, enter TLOG; for the JMS store, enter JMS.
 - Enter a name for **JNDI**.
For the TLOGs store, enter jdbc/tlogs; for the JMS store, enter jdbc/jms.
 - For the Database Driver, select **Oracle's Driver (Thin) for GridLink Connections Versions: Any**.
 - Click **Next**.
5. In the Transaction Options page, clear the **Supports Global Transactions** check box, and then click **Next**.

Supports Global Transactions
6. In the GridLink Data Source Connection Properties Options screen, select **Enter individual listener information** and click **Next**.
7. Enter the following connection properties:
 - **Service Name:** Enter the service name of the database with lowercase characters. For a GridLink data source, you must enter the Oracle RAC service name. For example:
soaedg.example.com

- **Host Name and Port:** Enter the SCAN address and port for the RAC database, separated by a colon. For example:

db-scan.example.com:1521

Click **Add** to add the host name and port to the list box below the field.

Enter host and port of each listener separated by colon and click the add button. In the case of a RAC DB listener, specify the SCAN address.

Host and Port:

You can identify the SCAN address by querying the appropriate parameter in the database using the TCP Protocol:

```
SQL>show parameter remote_listener;
```

NAME	TYPE	VALUE
remote_listener	string	db-scan.example.com

Note:

For Oracle Database 11g Release 1 (11.1), use the virtual IP and port of each database instance listener, for example:

dbhost1-vip.mycompany.com (port 1521)

and

dbhost2-vip.mycompany.com (1521)

- **Database User Name:** Enter the following:
For the TLOGs store, enter TLOG; for the JMS persistent store, enter JMS.
 - **Password:** Enter the password you used when you created the user in the database.
 - **Confirm Password:** Enter the password again and click **Next**.
8. On the Test GridLink Database Connection page, review the connection parameters and click **Test All Listeners**.

Here is an example of a successful connection notification:

```
Connection test for
jdbc:oracle:thin:@(DESCRIPTION=(ADDRESS_LIST=(ADDRESS=(PROTOCOL=TCP)(HOST=db-
scan.example.com)
(PORT=1521))))(CONNECT_DATA=(SERVICE_NAME=soaedg.example.com))) succeeded.
```

Click **Next**.

9. In the ONS Client Configuration page, do the following:

- Select **FAN Enabled** to subscribe to and process Oracle FAN events.
- Enter here also the SCAN address: ONS remote port for the RAC database and the ONS remote port as reported by the database (example below) and click **Add**:

```
[orcl@db-scan1 ~]$ srvctl config nodeapps -s
```

```
ONS exists: Local port 6100, remote port 6200, EM port 2016
```

- Click **Next**.

Note:

For Oracle Database 11g Release 1 (11.1), use the hostname and port of each database's ONS service, for example:

```
custdbhost1.example.com (port 6200)
```

and

```
custdbhost2.example.com (6200)
```

10. On the Test ONS Client Configuration page, review the connection parameters and click **Test All ONS Nodes**.

Here is an example of a successful connection notification:

```
Connection test for db-scan.example.com:6200 succeeded.
```

Click **Next**.

11. In the Select Targets page, select the cluster that will be using the persistent store, and then select **All Servers in the cluster**.

12. Click **Finish**.

13. To activate these changes, in the Change Center of the Administration Console, click **Activate Changes**.

19.1.4.8 Assigning the TLOGs JDBC store to the Managed Servers

After you create the tablespace and user in the database, and you have created the datasource, you can then assign the TLOGs persistence store to each of the required Managed Servers.

1. Login in to the Oracle WebLogic Server Administration Console.
2. In the **Change Center**, click **Lock and Edit**.
3. In the Domain Structure tree, expand **Environment**, then **Servers**.
4. Click the name of the Managed Server you want to use the TLOGs store.
5. Select the **Configuration > Services** tab.
6. Under **Transaction Log Store**, select **JDBC** from the **Type** menu.

7. From the **Data Source** menu, select the data source you created for the TLOGs persistence store.
8. In the **Prefix Name** field, specify a prefix name to form a unique JDBC TLOG store name for each configured JDBC TLOG store
9. Click **Save**.
10. Repeat steps 3 to 7 for each of the additional Managed Servers in the cluster.
11. To activate these changes, in the Change Center of the Administration Console, click **Activate Changes**.

19.1.4.9 Creating a JDBC JMS Store

After you create the JMS persistent store user and table space in the database, and after you create the data source for the JMS persistent store, you can then use the Administration Console to create the store.

1. Log in to the Oracle WebLogic Server Administration Console.
2. If you have not already done so, in the **Change Center**, click **Lock & Edit**.
3. In the **Domain Structure** tree, expand **Services**, then select **Persistent Store**.
4. Click **New**, and then click **JDBC Store**.
5. Enter a persistent store name that easily relates it to the pertaining JMS servers that will be using it.
6. Specify a unique prefix qualifying the installation and cluster and associate it with the data source you created for the JMS persistent store.
7. Target the store to the entity that will host the JTA services.

In the case of a server using service migration, this will be the cluster that the server belongs to.
8. To activate these changes, in the Change Center of the Administration Console, click **Activate Changes**.

19.1.4.10 Assigning the JMS JDBC store to the JMS Servers

After you create the JMS tablespace and user in the database, create the JMS datasource, and create the JDBC store, then you can then assign the JMS persistence store to each of the required JMS Servers.

1. Login in to the Oracle WebLogic Server Administration Console.
2. In the **Change Center**, click **Lock and Edit**.
3. In the Domain Structure tree, expand **Environment**, then **Messaging**, then **JMS Servers**.
4. Click the name of the JMS Server that you want to use the persistent store.
5. From the **Persistent Store** menu, select the JMS persistent store you created earlier.

6. Click **Save**.
7. Repeat steps 3 to 6 for each of the additional JMS Servers in the cluster.
8. To activate these changes, in the Change Center of the Administration Console, click **Activate Changes**.

19.1.4.11 Creating the Required Tables for the JMS JDBC Store

The final step in using a JDBC persistent store for JMS is to create the required JDBC store tables. Perform this task before restarting the Managed Servers in the domain.

1. Use the JAR utility supplied with the JDK to extract the DDL files to the `/weblogic/store/io/jdbc/ddl` directory using the following command:

```
jar xf com.bea.core.store.jdbc_1.0.0.0.jar /weblogic/store/io/jdbc/ddl
```

Note:

If you omit the `weblogic/store/io/jdbc/ddl` parameter, then the entire jar file is extracted.

2. Review the information in [Performance Impact of the TLOGs and JMS Persistent Stores](#), and edit the DDL file, accordingly.

For example, for an optimized schema definition that uses both secure files and hash partitioning, create a `jms_custom.ddl` file in the `ORACLE_RUNTIME` directory (or any other directory on shared storage accessible from all servers) with the following content.

```
CREATE TABLE $TABLE (
  id      int not null,
  type    int not null,
  handle  int not null,
  record  blob not null,
  PRIMARY KEY (ID) USING INDEX GLOBAL PARTITION BY HASH (ID) PARTITIONS 8)
LOB (RECORD) STORE AS SECUREFILE (ENABLE STORAGE IN ROW);
```

This example can be compared to the default schema definition for JMS stores, where the RAW data type is used without any partitions for indexes.

Note that the number of partitions should be a power of two. This will ensure that each partition will be a similar size. The recommended number of partitions will vary depending on the expected table or index growth. You should have your database administrator (DBA) analyze the growth of the tables over time and adjust the tables accordingly. For more information, see the Oracle Database VLDB and Partitioning Guide.

3. Use the Administration Console to edit the existing JDBS Store you created earlier; create the table that will be used for the JMS data:
 - a. Login in to the Oracle WebLogic Server Administration Console.
 - b. In the **Change Center**, click **Lock and Edit**.
 - c. In the Domain Structure tree, expand **Services**, then **Persistent Stores**.
 - d. Click the persistent store you created earlier.

- e. Under the **Advanced** options, enter `ORACLE_RUNTIME/jms_custom.ddl` in the **Create Table from DDL File** field.
 - f. Click **Save**.
 - g. To activate these changes, in the Change Center of the Administration Console, click **Activate Changes**.
4. Restart the Managed Servers.

19.1.5 Performing Backups and Recoveries for an Enterprise Deployment

This section provides guidelines for making sure you back up the necessary directories and configuration data for an Oracle SOA Suite enterprise deployment.

Note:

Some of the static and run-time artifacts listed in this section are hosted from Network Attached Storage (NAS). If possible, backup and recover these volumes from the NAS filer directly rather than from the application servers.

For general information about backing up and recovering Oracle Fusion Middleware products, see the following sections in *Administering Oracle Fusion Middleware*:

- "Backing Up Your Environment"
- "Recovering Your Environment"

[Table 19-2](#) lists the static artifacts to back up in a typical Oracle SOA Suite enterprise deployment.

Table 19-2 Static Artifacts to Back Up in the Oracle SOA Suite Enterprise Deployment

Type	Host	Tier
Database Oracle home	DBHOST1 and DBHOST2	Data Tier
Oracle Fusion Middleware Oracle home	WEBHOST1 and WEBHOST2	Web Tier
Oracle Fusion Middleware Oracle home	SOAHOST1 and SOAHOST2	Application Tier
Installation-related files	WEBHOST1, WEHOST2, and shared storage	N/A

[Table 19-3](#) lists the runtime artifacts to back up in a typical Oracle SOA Suite enterprise deployment.

Table 19-3 Run-Time Artifacts to Back Up in the Oracle SOA Suite Enterprise Deployment

Type	Host	Tier
Administration Server domain home (ASERVER_HOME)	SOAHOST1 and SOAHOST2	Application Tier

Table 19-3 (Cont.) Run-Time Artifacts to Back Up in the Oracle SOA Suite Enterprise Deployment

Type	Host	Tier
Application home (APPLICATION_HOME)	SOAHOST1 and SOAHOST2	Application Tier
Oracle RAC databases	DBHOST1 and DBHOST2	Data Tier
Scripts and Customizations	SOAHOST1 and SOAHOST2	Application Tier
Deployment Plan home (DEPLOY_PLAN_HOME)	SOAHOST1 and SOAHOST2	Application Tier

19.2 Configuration and Management Tasks for an Oracle SOA Suite Enterprise Deployment

This section describes some of the key configuration and management tasks you will likely need to perform on an Oracle SOA Suite enterprise deployment.

See Also:

[Deploying Oracle SOA Suite Composite Applications to an Enterprise Deployment](#)

[Using Shared Storage for Deployment Plans and SOA Infrastructure Applications Updates](#)

[Managing Database Growth in an Oracle SOA Suite Enterprise Deployment](#)

19.2.1 Deploying Oracle SOA Suite Composite Applications to an Enterprise Deployment

Oracle SOA Suite applications are deployed as composites, consisting of different kinds of Oracle SOA Suite components. SOA composite applications include the following:

- Service components such as Oracle Mediator for routing, BPEL processes for orchestration, BAM processes for orchestration (if Oracle BAM Suite is also installed), human tasks for workflow approvals, spring for integrating Java interfaces into SOA composite applications, and decision services for working with business rules.
- Binding components (services and references) for connecting SOA composite applications to external services, applications, and technologies.

These components are assembled into a single SOA composite application.

When you deploy an Oracle SOA Suite composite application to an Oracle SOA Suite enterprise deployment, be sure to deploy each composite to a specific server or cluster address and not to the load balancer address (`soa.example.com`).

Deploying composites to the load balancer address often requires direct connection from the deployer nodes to the external load balancer address. As a result, you will have to open additional ports in the firewalls.

For more information about Oracle SOA Suite composite applications, see the following sections in *Administering Oracle SOA Suite and Oracle Business Process Management Suite*:

- "Deploying SOA Composite Applications"
- "Monitoring SOA Composite Applications"
- "Managing SOA Composite Applications"

19.2.2 Using Shared Storage for Deployment Plans and SOA Infrastructure Applications Updates

When redeploying a SOA infrastructure application or resource adapter within the SOA cluster, the deployment plan along with the application bits should be accessible to all servers in the cluster.

SOA applications and resource adapters are installed using nostage deployment mode. Because the administration sever does not copy the archive files from their source location when the nostage deployment mode is selected, each server must be able to access the same deployment plan.

To ensure deployment plan location is available to all servers in the domain, use the Deployment Plan home location described in [File System and Directory Variables Used in This Guide](#) and represented by the DEPLOY_PLAN_HOME variable in the *Enterprise Deployment Workbook*.

19.2.3 Managing Database Growth in an Oracle SOA Suite Enterprise Deployment

When the amount of data in the Oracle SOA Suite database grows very large, maintaining the database can become difficult, especially in an Oracle SOA Suite enterprise deployment where potentially many composite applications are deployed.

For more information, review the following sections in *Administering Oracle SOA Suite and Oracle Business Process Management Suite*:

- "Developing a Database Growth Management Strategy"
- "Managing Database Growth"

Using Whole Server Migration and Service Migration in an Enterprise Deployment

The following topics describe Oracle WebLogic Server Whole Server migration and Oracle WebLogic Server Automatic Service Migration. They also explain how these features can be used in an Oracle Fusion Middleware enterprise topology.

See Also:

[Common Configuration and Management Procedures for an Enterprise Deployment](#)

[About Whole Server Migration and Automatic Service Migration in an Enterprise Deployment](#)

Oracle WebLogic Server provides a migration framework that is integral part of any highly available environment. The following sections provide more information about how this framework can be used effectively in an enterprise deployment.

[Creating a GridLink Data Source for Leasing](#)

Both Whole Server Migration and Automatic Service Migration require a data source for the leasing table, which is a tablespace created automatically as part of the Oracle WebLogic Server schemas by the Repository Creation Utility (RCU).

[Configuring Whole Server Migration for an Enterprise Deployment](#)

After you have prepared your domain for whole server migration or automatic service migration, you can then configure Whole Server Migration for specific Managed Servers within a cluster. See the following topics for more information.

[Configuring Automatic Service Migration in an Enterprise Deployment](#)

To configure automatic service migration for specific services in an enterprise deployment, refer to the topics in this section.

20.1 About Whole Server Migration and Automatic Service Migration in an Enterprise Deployment

Oracle WebLogic Server provides a migration framework that is integral part of any highly available environment. The following sections provide more information about how this framework can be used effectively in an enterprise deployment.

See Also:

[Understanding the Difference Between Whole Server and Service Migration](#)

[Implications of Using Whole Server Migration or Service Migration in an Enterprise Deployment](#)

Understanding Which Products and Components Require Whole Server Migration and Service Migration

20.1.1 Understanding the Difference Between Whole Server and Service Migration

The Oracle WebLogic Server migration framework supports two distinct types of automatic migration:

- **Whole Server Migration**, where the Managed Server instance is migrated to a different physical system upon failure.

Whole server migration provides for the automatic restart of a server instance, with all of its services, on a different physical machine. When a failure occurs in a server that is part of a cluster that is configured with server migration, the server is restarted on any of the other machines that host members of the cluster.

For this to happen, the servers need to use a floating IP as listen address and the required resources (transactions logs and JMS persistent stores) must be available on the candidate machines.

For more information, see "Whole Server Migration" in *Administering Clusters for Oracle WebLogic Server*.

- **Service Migration**, where specific services are moved to a different Managed Server within the cluster.

To understand service migration, it's important to understand *pinned services*.

In a WebLogic Server cluster, most subsystem services are hosted homogeneously on all server instances in the cluster, enabling transparent failover from one server to another. In contrast, pinned services, such as JMS-related services, the JTA Transaction Recovery Service, and user-defined singleton services, are hosted on individual server instances within a cluster—for these services, the WebLogic Server migration framework supports failure recovery with service migration, as opposed to failover.

For more information, see "Understanding the Service Migration Framework" in *Administering Clusters for Oracle WebLogic Server*.

20.1.2 Implications of Using Whole Server Migration or Service Migration in an Enterprise Deployment

When a server or service is started in another system, the required resources (such as services data and logs) must be available to both the original system and to the failover system; otherwise, the service cannot resume the same operations successfully on the failover system.

For this reason, both whole server and service migration require that all members of the cluster have access to the same transaction and JMS persistent stores (whether the persistent store is file-based or database-based).

This is another reason why shared storage is important in an enterprise deployment. When you properly configure shared storage, you ensure that in the event of a manual failover (Administration Server failover) or an automatic failover (whole server migration or service migration), both the original machine and the failover machine can access the same file store with no change in service.

In the case of an automatic service migration, when a pinned service needs to be resumed, the JMS and JTA logs that it was using before failover need to be accessible.

In addition to shared storage, Whole Server Migration requires the procurement and assignment of a virtual IP address (VIP). When a Managed Server fails over to another machine, the VIP is automatically reassigned to the new machine.

Note that service migration does not require a VIP.

20.1.3 Understanding Which Products and Components Require Whole Server Migration and Service Migration

The following table summarizes the products and components you should configure for whole server migration and the products you should configure for Automatic Service Migration.

Note that the table lists the recommended best practice. It does not preclude you from using Whole Server or Automatic Server Migration for those components that support it.

Component	Whole Server Migration (WSM)	Automatic Service Migration (ASM)
Oracle Web Services Manager (OWSM)	NO	NO
Oracle SOA Suite	NO	YES
Oracle Service Bus	NO	YES
Oracle Business Process Management	NO	YES
Enterprise Enterprise Scheduler	NO	NO
Oracle Business Activity Monitoring	NO	YES
Oracle B2B	NO	YES

20.2 Creating a GridLink Data Source for Leasing

Both Whole Server Migration and Automatic Service Migration require a data source for the leasing table, which is a tablespace created automatically as part of the Oracle WebLogic Server schemas by the Repository Creation Utility (RCU).

For an enterprise deployment, you should use create a GridLink data source:

1. Log in to the Oracle WebLogic Server Administration Console.
2. If you have not already done so, in the **Change Center**, click **Lock & Edit**.
3. In the **Domain Structure** tree, expand **Services**, then select **Data Sources**.
4. On the Summary of Data Sources page, click **New** and select **GridLink Data Source**, and enter the following:
 - Enter a logical name for the data source in the **Name** field. For example, **Leasing**.

- Enter a name for **JNDI**. For example, `jdbc/leasing`.
 - For the Database Driver, select **Oracle's Driver (Thin) for GridLink Connections Versions: Any**.
 - Click **Next**.
5. In the Transaction Options page, clear the **Supports Global Transactions** check box, and then click **Next**.

Supports Global Transactions

6. In the GridLink Data Source Connection Properties Options screen, select **Enter individual listener information** and click **Next**.

7. Enter the following connection properties:

- **Service Name:** Enter the service name of the database with lowercase characters. For a GridLink data source, you must enter the Oracle RAC service name. For example:

`soaedg.example.com`

- **Host Name and Port:** Enter the SCAN address and port for the RAC database, separated by a colon. For example:

`db-scan.example.com:1521`

Click **Add** to add the host name and port to the list box below the field.

Enter host and port of each listener separated by colon and click the add button. In the case of a RAC DB listener, specify the SCAN address.

Host and Port:

db-scan.example.com:1521

You can identify the SCAN address by querying the appropriate parameter in the database using the TCP Protocol:

```
SQL>show parameter remote_listener;
```

NAME	TYPE	VALUE

remote_listener	string	db-scan.example.com

Note:

For Oracle Database 11g Release 1 (11.1), use the virtual IP and port of each database instance listener, for example:

```
dbhost1-vip.mycompany.com (port 1521)
```

and

```
dbhost2-vip.mycompany.com (1521)
```

For Oracle Database 10g, use multi data sources to connect to an Oracle RAC database. For information about configuring multi data sources see [Using Multi Data Sources with Oracle RAC](#).

- **Database User Name:** Enter the following:

```
FMW1221_WLS_RUNTIME
```

In this example, FMW1221 is the prefix you used when you created the schemas as you prepared to configure the initial enterprise manager domain.

Note that in previous versions of Oracle Fusion Middleware, you had to manually create a user and tablespace for the migration leasing table. In Fusion Middleware 12c (12.2.1), the leasing table is created automatically when you create the WLS schemas with the Repository Creation Utility (RCU).

- **Password:** Enter the password you used when you created the WLS schema in RCU.
 - **Confirm Password:** Enter the password again and click **Next**.
8. On the Test GridLink Database Connection page, review the connection parameters and click **Test All Listeners**.

Here is an example of a successful connection notification:

```
Connection test for
jdbc:oracle:thin:@(DESCRIPTION=(ADDRESS_LIST=(ADDRESS=(PROTOCOL=TCP)(HOST=db-
scan.example.com)
(PORT=1521)))(CONNECT_DATA=(SERVICE_NAME=soaedg.example.com))) succeeded.
```

Click **Next**.

9. In the ONS Client Configuration page, do the following:

- Select **FAN Enabled** to subscribe to and process Oracle FAN events.
- Enter the SCAN address in the **ONS Host and Port** field, and then click **Add** and click **Add**:

This value should be the ONS host and ONS remote port for the RAC database. To find the ONS remote port for the database, you can use the following command on the database host:

```
[orcl@db-scan1 ~]$ srvctl config nodeapps -s
```

```
ONS exists: Local port 6100, remote port 6200, EM port 2016
```

- Click **Next**.

Note:

For Oracle Database 11g Release 1 (11.1), use the hostname and port of each database's ONS service, for example:

custdbhost1.example.com (port 6200)

and

custdbhost2.example.com (6200)

10. On the Test ONS Client Configuration page, review the connection parameters and click **Test All ONS Nodes**.

Here is an example of a successful connection notification:

```
Connection test for db-scan.example.com:6200 succeeded.
```

Click **Next**.

11. In the Select Targets page, select the cluster that you are configuring for Whole Server Migration or Automatic Service Migration, and then select **All Servers in the cluster**.

12. Click **Finish**.

13. Click **Activate Changes**.

20.3 Configuring Whole Server Migration for an Enterprise Deployment

After you have prepared your domain for whole server migration or automatic service migration, you can then configure Whole Server Migration for specific Managed Servers within a cluster. See the following topics for more information.

See Also:

[Editing the Node Manager's Properties File to Enable Whole Server Migration](#)

[Setting Environment and Superuser Privileges for the wlsifconfig.sh Script](#)

[Configuring Server Migration Targets](#)

[Testing Whole Server Migration](#)

20.3.1 Editing the Node Manager's Properties File to Enable Whole Server Migration

Use the section to edit the Node Manager properties file on the two nodes where the servers are running.

1. Locate and open the following file with a text editor:

```
MSERVER_HOME/nodemanager/nodemanager.properties
```

2. If not done already, set the `StartScriptEnabled` property in the `nodemanager.properties` file to `true`.

This is required to enable Node Manager to start the managed servers.

3. Add the following properties to the `nodemanager.properties` file to enable server migration to work properly:

- Interface

```
Interface=eth0
```

This property specifies the interface name for the floating IP (`eth0`, for example).

Note:

Do not specify the sub interface, such as `eth0:1` or `eth0:2`. This interface is to be used without the `:0`, or `:1`.

The Node Manager's scripts traverse the different `:X` enabled IPs to determine which to add or remove. For example, the valid values in Linux environments are `eth0`, `eth1`, or, `eth2`, `eth3`, `ethn`, depending on the number of interfaces configured.

- NetMask

```
NetMask=255.255.255.0
```

This property specifies the net mask for the interface for the floating IP.

- UseMACBroadcast

```
UseMACBroadcast=true
```

This property specifies whether or not to use a node's MAC address when sending ARP packets, that is, whether or not to use the `-b` flag in the arping command.

4. Restart the Node Manager.
5. Verify in the output of Node Manager (the shell where the Node Manager is started) that these properties are in use. Otherwise, problems may occur during migration. The output should be similar to the following:

```
...
SecureListener=true
LogCount=1
eth0=*,NetMask=255.255.255.0
...
```

20.3.2 Setting Environment and Superuser Privileges for the `wlsifconfig.sh` Script

Use this section to set the environment and superuser privileges for the `wlsifconfig.sh` script, which is used to transfer IP addresses from one machine to another during migration. It must be able to run `ifconfig`, which is generally only available to superusers.

For more information about the `wlsifconfig.sh` script, see "Configuring Automatic Whole Server Migration" in *Administering Clusters for Oracle WebLogic Server*.

Refer to the following sections for instructions on preparing your system to run the `wlsifconfig.sh` script.

See Also:

[Setting the PATH Environment Variable for the wlsifconfig.sh Script](#)

[Granting Privileges to the wlsifconfig.sh Script](#)

20.3.2.1 Setting the PATH Environment Variable for the wlsifconfig.sh Script

Ensure that the commands listed in the following table are included in the PATH environment variable for each host computers.

File	Directory Location
wlsifconfig.sh	MSERVER_HOME/bin/server_migration
wlscontrol.sh	WL_HOME/common/bin
nodemanager.domains	MSERVER_HOME/nodemanager

20.3.2.2 Granting Privileges to the wlsifconfig.sh Script

Grant sudo privilege to the operating system user (for example, oracle) with no password restriction, and grant execute privilege on the /sbin/ifconfig and /sbin/arping binaries.

Note:

For security reasons, sudo should be restricted to the subset of commands required to run the wlsifconfig.sh script.

Ask the system administrator for the sudo and system rights as appropriate to perform this required configuration task.

The following is an example of an entry inside */etc/sudoers* granting sudo execution privilege for oracle to run ifconfig and arping:

```
Defaults:oracle !requiretty
oracle ALL=NOPASSWD: /sbin/ifconfig,/sbin/arping
```

20.3.3 Configuring Server Migration Targets

To configure migration in a cluster:

1. Log in to the Oracle WebLogic Server Administration Console.
2. In the Domain Structure window, expand **Environment** and select **Clusters**. The Summary of Clusters page appears.
3. Click the cluster for which you want to configure migration in the Name column of the table.
4. Click the **Migration** tab.
5. Click **Lock & Edit**.

6. Verify that **Database** is the selected leasing mechanism.
If necessary, select **Database** as the leasing mechanism.
7. Under **Candidate Machines For Migratable Server**, in the Available field, select the Managed Servers in the cluster and click the right arrow to move them to **Chosen**.
8. Select the Leasing data source that you created in [Creating a GridLink Data Source for Leasing](#).
9. Click **Save**.
10. Set the Candidate Machines for Server Migration. You must perform this task for all of the managed servers as follows:
 - a. In Domain Structure window of the Oracle WebLogic Server Administration Console, expand **Environment** and select **Servers**.
 - b. Select the server for which you want to configure migration.
 - c. Click the **Migration** tab.
 - d. Select **Automatic Server Migration Enabled** and click **Save**.
This enables the Node Manager to start a failed server on the target node automatically.
For information on targeting applications and resources, see [Using Multi Data Sources with Oracle RAC](#).
 - e. In the **Available** field, located in the Migration Configuration section, select the machines to which to allow migration and click the right arrow.
In this step, you are identifying host to which the Managed Server should failover if the current host is unavailable. For example, for the Managed Server on the HOST1, select HOST2; for the Managed Server on HOST2, select HOST1.

Tip:
Click **Customize this table** in the Summary of Servers page, move Current Machine from the Available Window to the Chosen window to view the machine on which the server is running. This is different from the configuration if the server is migrated automatically.
11. Click **Activate Changes**.
12. Restart the Administration Server and the servers for which server migration has been configured.

20.3.4 Testing Whole Server Migration

Perform the steps in this section to verify that automatic whole server migration is working properly.

To test from Node 1:

1. Stop the managed server process.

```
kill -9 pid
```

pid specifies the process ID of the managed server. You can identify the *pid* in the node by running this command:

```
ps -ef | grep WLS_SOA1
```

2. Watch the Node Manager console (the terminal window where you performed the kill command): you should see a message indicating that the managed server's floating IP has been disabled.
3. Wait for the Node Manager to try a second restart of the Managed Server. Node Manager waits for a period of 30 seconds before trying this restart.
4. After node manager restarts the server and before it reaches "RUNNING" state, kill the associated process again.

Node Manager should log a message indicating that the server will not be restarted again locally.

Note:

The number of restarts required is determined by the `RestartMax` parameter in the following configuration file:

```
MSERVER_HOME/servers/WLS_SOA1/data/nodemanager/startup.properties
```

The default value is `RestartMax=2`.

To test from Node 2:

1. Watch the local Node Manager console. After 30 seconds since the last try to restart the managed server on Node 1, Node Manager on Node 2 should prompt that the floating IP for the managed server is being brought up and that the server is being restarted in this node.
2. Access a product URL using same IP address. If the URL is successful, then the migration was successful.

Verification From the Administration Console

You can also verify migration using the Oracle WebLogic Server Administration Console:

1. Log in to the Administration Console.
2. Click **Domain** on the left console.
3. Click the **Monitoring** tab and then the **Migration** subtab.

The Migration Status table provides information on the status of the migration.

Note:

After a server is migrated, to fail it back to its original machine, stop the managed server from the Oracle WebLogic Administration Console and then start it again. The appropriate Node Manager starts the managed server on the machine to which it was originally assigned.

20.4 Configuring Automatic Service Migration in an Enterprise Deployment

To configure automatic service migration for specific services in an enterprise deployment, refer to the topics in this section.

Note:

This SOA Suite feature is part of Oracle Integration Continuous Availability. Please refer to the Oracle Fusion Middleware Licensing Information for more details on Oracle SOA Suite for Middleware Options.

See Also:

[Setting the Leasing Mechanism and Data Source for an Enterprise Deployment Cluster](#)

[Changing the Migration Settings for the Managed Servers in the Cluster](#)

[About Selecting a Service Migration Policy](#)

[Setting the Service Migration Policy for Each Managed Server in the Cluster](#)

[Restarting the Managed Servers and Validating Automatic Service Migration](#)

[Failing Back Services After Automatic Service Migration](#)

20.4.1 Setting the Leasing Mechanism and Data Source for an Enterprise Deployment Cluster

Before you can configure automatic service migration, you must verify the leasing mechanism and data source that will be used by the automatic service migration feature:

Note:

The following procedure assumes you have already created the Leasing data source, as described in [Creating a GridLink Data Source for Leasing](#).

1. Log in to the Oracle WebLogic Server Administration Console.
2. Click **Lock & Edit**.
3. In the Domain Structure window, expand **Environment** and select **Clusters**.
The Summary of Clusters page appears.
4. In the **Name** column of the table, click the cluster for which you want to configure migration.
5. Click the **Migration** tab.
6. Verify that **Database** is selected in the **Migration Basis** drop-down menu.

7. From the **Data Source for Automatic Migration** drop-down menu, select the Leasing data source that you created in [Creating a GridLink Data Source for Leasing](#).
8. Click **Save**.
9. Activate changes.

20.4.2 Changing the Migration Settings for the Managed Servers in the Cluster

After you set the leasing mechanism and data source for the cluster, you can then enable automatic JTA migration for the Managed Servers that you want to configure for service migration. Note that this topic applies only if you are deploying JTA services as part of your enterprise deployment.

To change the migration settings for the Managed Servers in each cluster:

1. If you haven't already, log in to the Administration Console, and click **Lock & Edit**.
2. In the Domain Structure pane, expand the **Environment** node and then click **Servers**.

The Summary of Servers page appears.

3. Click the name of the server you want to modify in **Name** column of the table.

The settings page for the selected server appears and defaults to the Configuration tab.

4. Click the **Migration** tab.
5. From the **JTA Migration Policy** drop-down menu, select **Failure Recovery**.
6. In the **JTA Candidate Servers** section of the page, select the Managed Servers in the **Available** list box, and then click the move button to move them into the **Chosen** list box.
7. In the **JMS Candidate Servers** section of the page, select the Managed Servers in the **Available** list box, and then click the move button to move them into the **Chosen** list box.
8. Click **Save**.

20.4.3 About Selecting a Service Migration Policy

When you configure Automatic Service Migration, you select a Service Migration Policy for each cluster. This topic provides guidelines and considerations when selecting the Service Migration Policy.

For example, products or components running singletons or using Path services can benefit from the **Auto-Migrate Exactly-Once** policy. With this policy, if at least one Managed Server in the candidate server list is running, the services hosted by this migratable target will be active somewhere in the cluster if servers fail or are administratively shut down (either gracefully or forcibly). This can cause multiple homogenous services to end up in one server on startup.

When you are using this policy, you should monitor the cluster startup to identify what servers are running on each server. You can then perform a manual failback, if necessary, to place the system in a balanced configuration.

Other Fusion Middleware components are better suited for the **Auto-Migrate Failure-Recovery Services** policy. With this policy, the services hosted by the migratable target will start only if the migratable target's User Preferred Server (UPS) is started.

Based on these guidelines, the following policies are recommended for an Oracle SOA Suite enterprise topology:

- SOA_Cluster: **Auto-Migrate Failure-Recovery Services**
- OSB_Cluster: **Auto-Migrate Exactly-Once Services**
- BAM_Cluster: **Auto-Migrate Exactly-Once Services**

For more information, see *Policies for Manual and Automatic Service Migration in Administering Clusters for Oracle WebLogic Server*.

20.4.4 Setting the Service Migration Policy for Each Managed Server in the Cluster

After you modify the migration settings for each server in the cluster, you can then identify the services and set the migration policy for each Managed Server in the cluster, using the WebLogic Administration Console:

1. If you haven't already, log in to the Administration Console, and click **Lock & Edit**.
2. In the Domain Structure pane, expand **Environment**, then expand **Clusters**, then select **Migratable Targets**.
3. Click the name of the first Managed Server in the cluster.
4. Click the **Migration** tab.
5. From the **Service Migration Policy** drop-down menu, select the appropriate policy for the cluster.

For more information, see [About Selecting a Service Migration Policy](#).

6. Click **Save**.
7. Repeat steps 2 through 6 for each of the additional Managed Servers in the cluster.
8. Activate the changes.
9. Restart the Managed Servers in the cluster.

20.4.5 Restarting the Managed Servers and Validating Automatic Service Migration

After you configure automatic service migration for your cluster and Managed Servers, validate the configuration, as follows:

1. If you haven't already, log in to the Administration Console.
2. In the Domain Structure pane, select **Environment**, then **Clusters**, and restart the cluster you just configured for automatic service migration.
3. In the Domain Structure pane, expand **Environment**, and then expand **Clusters**.
4. Click **Migratable Targets**.
5. Click the **Control** tab.

The console displays a list of migratable targets and their current hosting server.

6. In the Migratable Targets table, select a row for the one of the migratable targets.
7. Note the value in the **Current Hosting Server** column.
8. Use the operating system command line to stop the the first Managed Server.

Use the following command to kill the Managed Server Process and simulate a crash scenario:

```
kill -9 pid
```

In this example, replace *pid* with the process ID (PID) of the Managed Server. You can identify the PID by running the following UNIX command:

```
ps -ef | grep managed_server_name
```

Note that after you kill the process, the Managed Server might be configured to start automatically after you initially kill the process. In this case, you must kill the second process using the `kill -9` command again.

9. Watch the terminal window (or console) where the Node Manager is running.

You should see a message indicating that the selected Managed Server has failed. The message will be similar to the following:

```
<INFO> <domain_name> <server_name>  
<The server 'server_name' with process id 4668 is no longer alive; waiting for  
the process to die.>  
<INFO> <domain_name> <server_name>  
<Server failed so attempting to restart (restart count = 1)>.
```

10. Return to the Oracle WebLogic Server Administration Console and refresh the table of migratable targets; verify that the migratable targets are transferred to the remaining, running Managed Server in the cluster:
 - Verify that the Current Hosting Server for the process you killed is now updated to show that it has been migrated to a different host.
 - Verify that the value in the **Status of Last Migration** column for the process is "Succeeded".
11. Open and review the log files for the Managed Servers that are now hosting the services; look for any JTA or JMS errors.

Note:

For JMS tests, it is a good practice to get message counts from destinations and make sure that there are no stuck messages in any of the migratable targets:

For example, for uniform distributed destinations (UDDs):

- a. Access the JMS Subdeployment module in the Administration Console:
In the Domain Structure pane, select **Services**, then **Messaging**, and then **JMS Modules**.
 - b. Click the JMS Module.
 - c. Click the destination in the **Summary of Resources** table. destination->Select monitoring and get the Messages Total and Messages Pending Counts
 - d. Select the Monitoring tab, and review the **Messages Total** and **Messages Pending** values in the **Destinations** table.
-

20.4.6 Failing Back Services After Automatic Service Migration

When Automatic Service Migration occurs, Oracle WebLogic Server does not support failing back services to their original server when a server is back online and rejoins the cluster.

As a result, after the Automatic Service Migration migrates specific JMS services to a backup server during a fail-over, it does not migrate the services back to the original server after the original server is back online. Instead, you must migrate the services back to the original server manually.

To fail back a service to its original server, follow these steps:

1. If you have not already done so, in the Change Center of the Administration Console, click **Lock & Edit**.
2. In the Domain Structure tree, expand **Environment**, expand **Clusters**, and then select **Migratable Targets**.
3. To migrate one or more migratable targets at once, on the Summary of Migratable Targets page:
 - a. Click the **Control** tab.
 - b. Use the check boxes to select one or more migratable targets to migrate.
 - c. Click **Migrate**.
 - d. Use the **New hosting server** drop-down to select the original Managed Server.
 - e. Click **OK**.

A request is submitted to migrate the JMS-related service and the configuration edit lock is released. In the Migratable Targets table, the Status of Last Migration column indicates whether the requested migration has succeeded or failed.

4. To migrate a specific migratable target, on the Summary of Migratable Targets page:
 - a. Select the migratable target to migrate.
 - b. Click the **Control** tab.
 - c. Reselect the migratable target to migrate.
 - d. Click **Migrate**.
 - e. Use the New hosting server drop-down to select a new server for the migratable target.
 - f. Click **OK**.

Configuring Single Sign-On for an Enterprise Deployment

This chapter describes how to configure the Oracle HTTP Server WebGate to enable single sign-on with Oracle Access Manager.

See Also:

[Common Configuration and Management Procedures for an Enterprise Deployment](#)

[About Oracle HTTP Server Webgate](#)

Oracle HTTP Server WebGate is a Web server plug-in that intercepts HTTP requests and forwards them to an existing Oracle Access Manager instance for authentication and authorization.

[General Prerequisites for Configuring Oracle HTTP Server 12c Webgate](#)

Before you can configure Oracle HTTP Server 12c WebGate, you must have installed and configured a certified version of Oracle Access Manager.

[Enterprise Deployment Prerequisites for Configuring OHS 12c Webgate](#)

When you are configuring Oracle HTTP Server Webgate to enable Single Sign-On for an enterprise deployment, consider the prerequisites mentioned in this section.

[Configuring Oracle HTTP Server 12c WebGate for an Enterprise Deployment](#)

Perform the following steps to configure Oracle HTTP Server 12c WebGate for Oracle Access Manager on both WEBHOST1 and WEBHOST2.

[Registering the Oracle HTTP Server 12c WebGate with Oracle Access Manager](#)

You can register the WebGate agent with Oracle Access Manager by using the Oracle Access Manager Administration console.

[Setting Up the WebLogic Server Authentication Providers](#)

To set up the WebLogic Server authentication providers, back up the configuration files, set up the Oracle Access Manager Identity Assertion Provider and set the order of providers.

[Configuring Oracle ADF and OPSS Security with Oracle Access Manager](#)

Some Oracle Fusion Middleware management consoles use Oracle Application Development Framework (Oracle ADF) security, which can integrate with Oracle Access Manager Single Sign On (SSO). These applications can take advantage of Oracle Platform Security Services (OPSS) SSO for user authentication, but you must first configure the domain-level `jsp-config.xml` file to enable these capabilities.

21.1 About Oracle HTTP Server Webgate

Oracle HTTP Server WebGate is a Web server plug-in that intercepts HTTP requests and forwards them to an existing Oracle Access Manager instance for authentication and authorization.

For Oracle Fusion Middleware 12c, the WebGate software is installed as part of the Oracle HTTP Server 12c software installation.

For more extensive information about WebGates, see “Registering and Managing OAM 11g Agents” in the *Administrator’s Guide for Oracle Access Management*.

21.2 General Prerequisites for Configuring Oracle HTTP Server 12c Webgate

Before you can configure Oracle HTTP Server 12c WebGate, you must have installed and configured a certified version of Oracle Access Manager.

At the time this document was published, the supported versions of Oracle Access Manager were 11g Release 2 (11.1.2.2) and 11g Release 2 (11.1.2.3). For the most up-to-date information, see the certification document for your release on the *Oracle Fusion Middleware Supported System Configurations* page.

Note:

For production environments, it is highly recommended that you install Oracle Access Manager in its own environment and not on the machines that are hosting the enterprise deployment.

For more information about Oracle Access Manager, see the latest Oracle Identity and Access Management documentation, which you can find in the **Middleware** documentation on the [Oracle Help Center](#).

21.3 Enterprise Deployment Prerequisites for Configuring OHS 12c Webgate

When you are configuring Oracle HTTP Server Webgate to enable Single Sign-On for an enterprise deployment, consider the prerequisites mentioned in this section.

- Oracle recommends that you deploy Oracle Access Manager as part of a highly available, secure, production environment. For more information about deploying Oracle Access Manager in an enterprise environment, see the Enterprise Deployment Guide for your version of Oracle Identity and Access Management.
- To enable single sign-on for the WebLogic Server Administration Console and the Oracle Enterprise Manager Fusion Middleware Control, you must add a central LDAP-provisioned administration user to the directory service that Oracle Access Manager is using (for example, Oracle Internet Directory or Oracle Unified Directory). For more information about the required user and groups to add to the LDAP directory, follow the instructions in [Creating a New LDAP Authenticator and Provisioning Enterprise Deployment Users and Group](#).

21.4 Configuring Oracle HTTP Server 12c WebGate for an Enterprise Deployment

Perform the following steps to configure Oracle HTTP Server 12c WebGate for Oracle Access Manager on both WEBHOST1 and WEBHOST2.

In the following procedure, replace the directory variables, such as *OHS_ORACLE_HOME* and *OHS_CONFIG_DIR*, with the values, as defined in [File System and Directory Variables Used in This Guide](#).

1. Perform a complete backup of the Web Tier domain.

2. Change directory to the following location in the Oracle HTTP Server Oracle home:

```
cd OHS_ORACLE_HOME/webgate/ohs/tools/deployWebGate/
```

3. Run the following command to create the WebGate Instance directory and enable WebGate logging on OHS Instance:

```
./deployWebGateInstance.sh -w OHS_CONFIG_DIR -oh OHS_ORACLE_HOME
```

4. Verify that a webgate directory and subdirectories was created by the `deployWebGateInstance` command:

```
ls -lart OHS_CONFIG_DIR/webgate/
total 6
drwxr-x---+ 8 orcl oinstall 20 Oct  2 07:14 ..
drwxr-xr-x+ 4 orcl oinstall  4 Oct  2 07:14 .
drwxr-xr-x+ 3 orcl oinstall  3 Oct  2 07:14 tools
drwxr-xr-x+ 3 orcl oinstall  4 Oct  2 07:14 config
```

5. Run the following command to ensure that the `LD_LIBRARY_PATH` environment variable contains *OHS_ORACLE_HOME/lib* directory path:

```
export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:OHS_ORACLE_HOME/lib
```

6. Change directory to the following directory

```
OHS_ORACLE_HOME/webgate/ohs/tools/setup/InstallTools
```

7. Run the following command from the `InstallTools` directory.

```
./EditHttpConf -w OHS_CONFIG_DIR -oh OHS_ORACLE_HOME -o
output_file_name
```

This command:

- Copies the `apache_webgate.template` file from the Oracle HTTP Server Oracle home to a new `webgate.conf` file in the Oracle HTTP Server configuration directory.
- Updates the `httpd.conf` file to add one line, so it includes the `webgate.conf`.
- Generates a WebGate configuration file. The default name of the file is `webgate.conf`, but you can use a custom name by using the `output_file` argument to the command.

21.5 Registering the Oracle HTTP Server 12c WebGate with Oracle Access Manager

You can register the WebGate agent with Oracle Access Manager by using the Oracle Access Manager Administration console.

For more information, see "Registering an OAM Agent Using the Console" in the *Oracle Fusion Middleware Administrator's Guide for Oracle Access Management*.

For more information, see the following topics:

See Also:

[Locating and Preparing the RREG Tool](#)

[About RREG In-Band and Out-of-Band Mode](#)

[Updating the Standard Properties in the OAM11gRequest.xml File](#)

[Updating the Protected, Public, and Excluded Resources for an Enterprise Deployment](#)

[Running the RREG Tool](#)

[Files and Artifacts Generated by RREG](#)

[Copying Generated Artifacts to the Oracle HTTP Server WebGate Instance Location](#)

[Restarting the Oracle HTTP Server Instance](#)

21.5.1 Locating and Preparing the RREG Tool

To set up the RREG tool, complete the following steps:

1. Log in to one of the Oracle Access Manager hosts in the Application tier.
2. Change directory to the following directory in the Oracle Access Manager Oracle home:

```
OAM_ORACLE_HOME/oam/server/rreg/client
```

In this example, *OAM_ORACLE_HOME* refers to the Oracle home on the system where the Oracle Access Manager software was installed.

Note: If you do not have privileges or access to the Oracle Access Manager server, then you can use out-of-band mode to generate the required files and register the WebGate with Oracle Access Manager. For more information, see [About RREG In-Band and Out-of-Band Mode](#).

3. Open the `oamreg.sh` file and set the following environment variables in the file, as follows:

- Set `OAM_REG_HOME` to the absolute path to the directory in which you extracted the contents of RREG archive.

Set `JDK_HOME` to the absolute path of the directory in which a supported JDK is installed on your machine.

21.5.2 About RREG In-Band and Out-of-Band Mode

You can run the RREG Tool in one of two modes: in-band and out-of-band.

Use **in-band** mode when you have the privileges to access the Oracle Access Manager server and run the RREG tool yourself from the Oracle Access Manager Oracle home. You can then copy the generated artifacts and files to the Web server configuration directory after you run the RREG Tool.

Use **out-of-band** mode if you do *not* have privileges or access to the Oracle Access Manager server. For example, in some organizations, only the Oracle Access Manager server administrators have privileges access the server directories and perform administration tasks on the server. In out-of-band mode, the process can work as follows:

1. The Oracle Access Manager server administrator provides you with a copy of the RREG archive file (RREG.tar.gz), which the server administrator can find in the location described in [Locating and Preparing the RREG Tool](#).
2. Untar the RREG.tar.gz file that was provided to you by the server administrator.

For example:

```
gunzip RREG.tar.gz
tar -xvf RREG.tar
```

After you unpack the RREG archive, you can find the tool for registering the agent in the following location:

```
RREG_HOME/bin/oamreg.sh
```

In this example, *RREG_Home* is the directory in which you extracted the contents of RREG archive.

3. Use the instructions in [Updating the Standard Properties in the OAM11gRequest.xml File](#) to update the OAM11GRequest.xml file, and send the completed OAM11GRequest.xml file to the Oracle Access Manager server administrator.
4. The Oracle Access Manager server administrator then uses the instructions in [Running the RREG Tool in Out-Of-Band Mode](#) to run the RREG Tool and generate the *AgentID_response.xml* file.
5. The Oracle Access Manager server administrator sends the *AgentID_response.xml* file to you.
6. Use the instructions in [Running the RREG Tool in Out-Of-Band Mode](#) to run the RREG Tool with the *AgentID_response.xml* file and generate the required artifacts and files on the client system.

21.5.3 Updating the Standard Properties in the OAM11gRequest.xml File

Before you can register the Webgate agent with Oracle Access Manager, you must update some required properties in the OAM11gRequest.xml file.

Note:

If you plan to use the default values for most of the parameters in the provided XML file, then you can use the shorter version (OAM11gRequest_short.xml, in which all non-listed fields will take a default value.

To perform this task:

1. If you are using in-band mode, then change directory to the following location in the directory:

`OAM_ORACLE_HOME/oam/server/rreg/client`

If you are using out-of-band mode, then change directory to the location where you unpacked the RREG archive.

2. Make a copy of the OAM11gRequest.xml file template.
3. Review the properties listed in the file, and then update your copy of the OAM11gRequest.xml file to make sure the properties reference the host names and other values specific to your environment.

OAM11gRequest.xml Property	Set to...
serverAddress	The host and the port of the Administration Server for the Oracle Access Manager domain.
agentName	Any custom name for the agent. Typically, you use a name that identifies the Fusion Middleware product you are configuring for single sign-on.
applicationDomain	A value that identifies the Web tier host and the FMW component you are configuring for single sign-on.
security	The security mode of the Oracle Access Manager server, which can be open, simple, or certificate mode. For an enterprise deployment, Oracle recommends simple mode, unless additional requirements exist to implement custom security certificates for the encryption of authentication and authorization traffic. In most cases, avoid using open mode, because in open mode, traffic to and from the Oracle Access Manager server is not encrypted. For more information using certificate mode or about Oracle Access Manager supported security modes in general, see Securing Communication Between OAM Servers and WebGates in the <i>Administrator's Guide for Oracle Access Management</i> .
cachePragmaHeader	private
cacheControlHeader	private
ipValidation	0 <code><ipValidation>0<ipValidation></code>

ipValidationExceptions	<p>The IP address of the front-end load balancer. For example:</p> <pre><ipValidationExceptions> <ipAddress>130.35.165.42</ipAddress> </ipValidation></pre>
agentBaseUrl	<p>The host and the port of the machine on which Oracle HTTP Server 12c WebGate is installed.</p>

21.5.4 Updating the Protected, Public, and Excluded Resources for an Enterprise Deployment

When you set up an Oracle Fusion Middleware environment for single sign-on, you identify a set of URLs that you want Oracle Access Manager to protect with single sign-on. You identify these using specific sections of the `OAM11gRequest.xml` file. To identify the URLs:

1. If you haven't already opened `OAM11gRequest.xml` file for editing, locate and open the file in a text editor.

For more information, see the following:

- [Locating and Preparing the RREG Tool](#)
- [Updating the Standard Properties in the OAM11gRequest.xml File](#)

2. Remove the sample entries from the file, and then enter the list of protected, public, and excluded resources in the appropriate sections of the file, as shown in the following example.

Note:

If you are using Oracle Access Manager 11g Release 2 (11.1.2.2) or later, then note that the entries with the wildcard syntax ("`.../*`") are included for backward compatibility with previous versions of Oracle Access Manager.

```
<protectedResourcesList>
  <resource>/integration/worklistapp</resource>
  <resource>/integration/worklistapp/.../*</resource>
  <resource>/workflow/sdpmessagingsca-ui-worklist</resource>
  <resource>/workflow/sdpmessagingsca-ui-worklist/.../*</resource>
  <resource>/b2bconsole</resource>
  <resource>/b2bconsole/.../*</resource>
  <resource>/sdpmessaging/userprefs-ui</resource>
  <resource>/sdpmessaging/userprefs-ui/.../*</resource>
  <resource>/workflow/DefaultToDoTaskFlow</resource>
  <resource>/workflow/DefaultToDoTaskFlow/.../*</resource>
  <resource>/DefaultToDoTaskFlow</resource>
  <resource>/DefaultToDoTaskFlow/.../*</resource>
  <resource>/ess</resource>
  <resource>/ess/.../*</resource>
  <resource>/EssHealthCheck</resource>
  <resource>/EssHealthCheck/.../*</resource>
  <resource>/em</resource>
```

```

        <resource>/em/.../*</resource>
        <resource>/console</resource>
        <resource>/console/.../*</resource>
        <resource>/servicebus</resource><!-- (For OSB systems only) -->
        <resource>/servicebus/.../*</resource><!-- (For OSB systems only) -->
        <resource>/lwpfconsole</resource><!-- (For OSB systems only) -->
        <resource>/lwpfconsole/.../*</resource><!-- (For OSB systems only) -->
        <resource>/soa/composer</resource>
        <resource>/soa/composer/.../*</resource>
        <resource>/OracleBAM </resource><!-- (For BAM systems only) -->
        <resource>/OracleBAM/.../*</resource><!-- (For BAM systems only) -->
        <resource>/oracle/bam/server</resource><!-- (For BAM systems only) -->
        <resource>/oracle/bam/server/.../*</resource><!-- (For BAM systems only)
-->
        <resource>/bam/composer </resource><!-- (For BAM systems only) -->
        <resource>/bam/composer/.../*</resource><!-- (For BAM systems only) -->
        <resource>/bpm/composer</resource> <!-- (For BPM systems only) -->
        <resource>/bpm/composer/.../*</resource> <!-- (For BPM systems only) -->
        <resource>/bpm/workspace</resource><!-- (For BPM systems only) -->
        <resource>/bpm/workspace/.../*</resource><!-- (For BPM systems only) -->
        <resource>/soa-infra</resource>
        <resource>/soa-infra/deployer</resource>
        <resource>/soa-infra/deployer/.../*</resource>
        <resource>/soa-infra/events/edn-db-log</resource>
        <resource>/soa-infra/events/edn-db-log/.../*</resource>
        <resource>/soa-infra/cluster/info</resource>
        <resource>/soa-infra/cluster/info/.../*</resource>
        <resource>/inspection.wsil</resource>
        <resource>/healthcare/.../*</resource><!-- (For HC systems only) -->
        <resource>/healthcare</resource><!-- (For HC systems only) -->
    </protectedResourcesList>
    <publicResourcesList>
        <resource>/soa-infra/directWSDL</resource>
        <resource>/sbinspection.wsil</resource><!-- (For OSB systems only) -->
    </publicResourcesList>
    <excludedResourcesList>
        <resource>/wsm-pm</resource>
        <resource>/wsm-pm/.../*</resource>
        <resource>/soa-infra</resource>
        <resource>/soa-infra/services/.../*</resource>
        <resource>/OracleBAMWS</resource> <!-- (For BAM systems only) -->
        <resource>/OracleBAMWS/.../*</resource><!-- (For BAM systems only) -->
        <resource>/ucs/messaging/webservice</resource>
        <resource>/ucs/messaging/webservice/.../*</resource>
        <resource>/sbconsole</resource><!-- (For OSB systems only) -->
        <resource>/sbconsole/.../*</resource><!-- (For OSB systems only) -->
        <resource>/sbresource</resource><!-- (For OSB systems only) -->
        <resource>/sbresource/.../*</resource><!-- (For OSB systems only) -->
        <resource>/integration/services/.../*</resource>
        <resource>/integration/services</resource>
        <resource>/b2b/services</resource>
        <resource>/b2b/services/.../*</resource>
    </excludedResourcesList>

```

3. Save and close the OAM11GRequest.xml file.

21.5.5 Running the RREG Tool

The following topics provide information about running the RREG tool to register your Oracle HTTP Server Webgate with Oracle Access Manager.

See Also:

[Running the RREG Tool in In-Band Mode](#)

[Running the RREG Tool in Out-Of-Band Mode](#)

21.5.5.1 Running the RREG Tool in In-Band Mode

To run the RREG Tool in in-band mode:

1. Change directory to the RREG home directory.

If you are using in-band mode, the RREG directory is inside the Oracle Access Manager Oracle home:

```
OAM_ORACLE_HOME/oam/server/rreg/client
```

If you are using out-of-band mode, then the RREG home directory is the location where you unpacked the RREG archive.

2. Change directory to the bin directory inside the RREG home directory:

```
cd RREG_HOME/bin/
```

3. Set the permissions of the `oamreg.sh` command so you can execute the file:

```
chmod +x oamreg.sh
```

4. Run the following command:

```
./oamreg.sh inband input/OAM11GRequest.xml
```

In this example:

- It is assumed the edited `OAM11GRequest.xml` file is located in the `RREG_HOME/input` directory.
- The output from this command will be saved to the following directory:

```
RREG_HOME/output/
```

The following example shows a sample RREG session:

```
Welcome to OAM Remote Registration Tool!
Parameters passed to the registration tool are:
Mode: inband
Filename: /u01/oracle/products/fmw/iam_home/oam/server/rreg/client/rreg/input/
OAM11GWCCDomainRequest.xml
Enter admin username:weblogic_idm
Username: weblogic_idm
Enter admin password:
Do you want to enter a Webgate password?(y/n):
n
Do you want to import an URIs file?(y/n):
n
```

```
-----
Request summary:
OAM11G Agent Name:WCC1221_EDG_AGENT
URL String:null
Registering in Mode:inband
Your registration request is being sent to the Admin server at: http://
host1.example.com:7001
-----
```

```
Jul 08, 2015 7:18:13 PM oracle.security.jps.util.JpsUtil disableAudit
INFO: JpsUtil: isAuditDisabled set to true
Jul 08, 2015 7:18:14 PM oracle.security.jps.util.JpsUtil disableAudit
INFO: JpsUtil: isAuditDisabled set to true
Inband registration process completed successfully! Output artifacts are created in
the output folder.
```

21.5.5.2 Running the RREG Tool in Out-Of-Band Mode

To run the RREG Tool in out-of-band mode on the Oracle Access Manager server, the Oracle Access Manager server administrator uses the following command:

```
RREG_HOME/bin/oamreg.sh outofband input/OAM11GRequest.xml
```

In this example:

- Replace *RREG_HOME* with the location where the RREG archive file was unpacked on the server.
- The edited *OAM11GRequest.xml* file is located in the *RREG_HOME/input* directory.
- The RREG Tool saves the output from this command (the *AgentID_response.xml* file) to the following directory:

```
RREG_HOME/output/
```

The Oracle Access Manager server administrator can then send the *AgentID_response.xml* to the user who provided the *OAM11GRequest.xml* file.

To run the RREG Tool in out-of-band mode on the Web server client machine, use the following command:

```
RREG_HOME/bin/oamreg.sh outofband input/AgentID_response.xml
```

In this example:

- Replace *RREG_HOME* with the location where you unpacked the RREG archive file on the client system.
- The *AgentID_response.xml* file, which was provided by the Oracle Access Manager server administrator, is located in the *RREG_HOME/input* directory.
- The RREG Tool saves the output from this command (the artifacts and files required to register the Webgate software) to the following directory on the client machine:

```
RREG_HOME/output/
```

21.5.6 Files and Artifacts Generated by RREG

The files that get generated by the RREG Tool vary, depending on the security level you are using for communications between the WebGate and the Oracle Access Manager server. For more information about the supported security levels, see *Securing Communication Between OAM Servers and WebGates* in the *Administrator's Guide for Oracle Access Management*.

Note that in this topic any references to *RREG_HOME* should be replaced with the path to the directory where you ran the RREG tool. This is typically the following

directory on the Oracle Access Manager server, or (if you are using out-of-band mode) the directory where you unpacked the RREG archive:

```
OAM_ORACLE_HOME/oam/server/rreg/client
```

The following table lists the artifacts that are always generated by the RREG Tool, regardless of the Oracle Access Manager security level.

File	Location
<code>cwallet.sso</code>	<code>RREG_HOME/output/Agent_ID/</code>
<code>ObAccessClient.xml</code>	<code>RREG_HOME/output/Agent_ID/</code>

The following table lists the additional files that are created if you are using the SIMPLE or CERT security level for Oracle Access Manager:

File	Location
<code>aaa_key.pem</code>	<code>RREG_HOME/output/Agent_ID/</code>
<code>aaa_cert.pem</code>	<code>RREG_HOME/output/Agent_ID/</code>
<code>password.xml</code>	<code>RREG_HOME/output/Agent_ID/</code>

Note that the `password.xml` file contains the obfuscated global passphrase to encrypt the private key used in SSL. This passphrase can be different than the passphrase used on the server.

You can use the files generated by RREG to generate a certificate request and get it signed by a third-party Certification Authority. To install an existing certificate, you must use the existing `aaa_cert.pem` and `aaa_chain.pem` files along with `password.xml` and `aaa_key.pem`.

21.5.7 Copying Generated Artifacts to the Oracle HTTP Server WebGate Instance Location

After the RREG Tool generates the required artifacts, manually copy the artifacts from the `RREG_Home/output/agent_ID` directory to the Oracle HTTP Server configuration directory on the Web tier host.

The location of the files in the Oracle HTTP Server configuration directory depends upon the Oracle Access Manager security mode setting (OPEN, SIMPLE, or CERT).

The following table lists the required location of each generated artifact in the Oracle HTTP configuration directory, based on the security mode setting for Oracle Access Manager. In some cases, you might have to create the directories if they do not exist already. For example, the wallet directory might not exist in the configuration directory.

Note:

For an enterprise deployment, Oracle recommends simple mode, unless additional requirements exist to implement custom security certificates for the encryption of authentication and authorization traffic. The information about using open or certification mode is provided here as a convenience.

Avoid using open mode, because in open mode, traffic to and from the Oracle Access Manager server is not encrypted.

For more information using certificate mode or about Oracle Access Manager supported security modes in general, see *Securing Communication Between OAM Servers and WebGates* in the *Administrator's Guide for Oracle Access Management*.

File	Location When Using OPEN Mode	Location When Using SIMPLE Mode	Location When Using CERT Mode
wallet/cwallet.sso	OHS_CONFIG_DIR/ webgate/config/ wallet	OHS_CONFIG_DIR/ webgate/config/ wallet/	OHS_CONFIG_DIR/ webgate/config/ wallet/
ObAccessClient.xml	OHS_CONFIG_DIR/ webgate/config	OHS_CONFIG_DIR/ webgate/config/	OHS_CONFIG_DIR/ webgate/config/
password.xml	N/A	OHS_CONFIG_DIR/ webgate/config/	OHS_CONFIG_DIR/ webgate/config/
aaa_key.pem	N/A	OHS_CONFIG_DIR/ webgate/config/ simple/	OHS_CONFIG_DIR/ webgate/config/
aaa_cert.pem	N/A	OHS_CONFIG_DIR/ webgate/config/ simple/	OHS_CONFIG_DIR/ webgate/config/

Note: When you copy ObAccessClient.xml to a new directory, you will need to delete the old ObAccessClient.xml file from the first Oracle HTTP Server location cache location on WEBHOST1:

OHS_DOMAIN_HOME/servers/ohs1/cache/config/

And you must perform the similar step for the second Oracle HTTP Server instance on WEBHOST2:

OHS_DOMAIN_HOME/servers/ohs2/cache/config/

21.5.8 Restarting the Oracle HTTP Server Instance

For information about restarting the Oracle HTTP Server Instance, see "Restarting Oracle HTTP Server Instances by Using WLST" in *Administering Oracle HTTP Server*.

If you have configured Oracle HTTP Server in a WebLogic Server domain, you can also use Oracle Fusion Middleware Control to restart the Oracle HTTP Server

Instances. For more information, see "Restarting Oracle HTTP Server Instances by Using Fusion Middleware Control" in *Administrator's Guide for Oracle HTTP Server*.

21.6 Setting Up the WebLogic Server Authentication Providers

To set up the WebLogic Server authentication providers, back up the configuration files, set up the Oracle Access Manager Identity Assertion Provider and set the order of providers.

The following topics assumes that you have already configured the LDAP authenticator by following the steps in [Creating a New LDAP Authenticator and Provisioning Enterprise Deployment Users and Group](#). If you have not already created the LDAP authenticator, then do so before continuing with this section.

See Also:

[Backing Up Configuration Files](#)

[Setting Up the Oracle Access Manager Identity Assertion Provider](#)

[Setting the Order of Providers](#)

21.6.1 Backing Up Configuration Files

To be safe, you should first back up the relevant configuration files:

```
ASERVER_HOME/config/config.xml
ASERVER_HOME/config/fmwconfig/jps-config.xml
ASERVER_HOME/config/fmwconfig/system-jazn-data.xml
```

Also back up the `boot.properties` file for the Administration Server:

```
ASERVER_HOME/servers/AdminServer/security/boot.properties
```

21.6.2 Setting Up the Oracle Access Manager Identity Assertion Provider

Set up an Oracle Access Manager identity assertion provider in the Oracle WebLogic Server Administration Console.

To set up the Oracle Access Manager identity assertion provider:

1. Log in to the WebLogic Server Administration Console, if not already logged in.
2. Click **Lock & Edit**.
3. Click **Security Realms** in the left navigation bar.
4. Click the **myrealm** default realm entry.
5. Click the **Providers** tab.
6. Click **New**, and select the asserter type **OAMIdentityAsserter** from the drop-down menu.
7. Name the asserter (for example, *OAM ID Asserter*) and click **OK**.
8. Click the newly added asserter to see the configuration screen for the Oracle Access Manager identity assertion provider.
9. Set the control flag to **REQUIRED**.

10. Select both the **ObSSOCookie** and **OAM_REMOTE_USER** options under Chosen types.
11. Click **Save** to save the settings.
12. Click **Activate Changes** to propagate the changes.
13. Restart the Administration Server and Managed Servers.

21.6.3 Setting the Order of Providers

Set the order of identity assertion and authentication providers in the WebLogic Server Administration Console.

To set the order of the providers:

1. Log in to the WebLogic Server Administration Console, if not already logged in.
2. Click **Lock & Edit**.
3. Click **Security Realms** in the left navigation bar.
4. Click the **myrealm** default realm entry.
5. Click the **Providers** tab.
6. Reorder the Oracle Access Manager identity assertion provider, the LDAP authentication provider, and the default authentication provider by ensuring that the control flag for each provider is set as follows:
 - Oracle Access Manager identity assertion provider: **REQUIRED**
 - LDAP authentication provider: **SUFFICIENT**
 - Default authentication provider (DefaultAuthenticator): **SUFFICIENT**
7. Click **OK**.
8. Click **Activate Changes** to propagate the changes.
9. Restart the Administration Server, Managed Servers, and system components.

21.7 Configuring Oracle ADF and OPSS Security with Oracle Access Manager

Some Oracle Fusion Middleware management consoles use Oracle Application Development Framework (Oracle ADF) security, which can integrate with Oracle Access Manager Single Sign On (SSO). These applications can take advantage of Oracle Platform Security Services (OPSS) SSO for user authentication, but you must first configure the domain-level `jps-config.xml` file to enable these capabilities.

The domain-level `jps-config.xml` file is located in the following location after you create an Oracle Fusion Middleware domain:

```
DOMAIN_HOME/config/fmwconfig/jps-config.xml
```

Note:

The domain-level `jps-config.xml` should not be confused with the `jps-config.xml` that is deployed with custom applications.

To update the OPSS configuration to delegate in SSO actions in Oracle Access Manager, complete the following steps:

1. Change directory to the following directory:

```
cd ORACLE_COMMON_HOME/common/bin
```

2. Start the WebLogic Server Scripting Tool (WLST):

```
./wlst.sh
```

3. Connect to the Administration Server, using the following WLST command:

```
connect('admin_user','admin_password','admin_url')
```

For example:

```
connect('weblogic','mypassword','t3://ADMINVHN:7001')
```

4. Execute the `addOAMSSOProvider` command, as follows:

```
addOAMSSOProvider(loginuri="/${app.context}/adfAuthentication", logouturi="/oamssso/logout.html")
```

The following table defines the expected value for each argument in the `addOAMSSOProvider` command.

Argument	Definition
loginuri	<p>Specifies the URI of the login page</p> <p>Note: For ADF security enabled applications, "<code><context-root>/adfAuthentication</code>" should be provided for the 'loginuri' parameter.</p> <p>For example:</p> <pre>/\${app.context}/adfAuthentication</pre> <p>Here is the flow:</p> <ol style="list-style-type: none"> a. User accesses a resource that has been protected by authorization policies in OPSS, for example. b. If the user is not yet authenticated, ADF redirects the user to the URI configured in 'loginuri'. c. Access Manager, should have a policy to protect the value in 'loginuri': for example, "<code><context-root>/adfAuthentication</code>". d. When ADF redirects to this URI, Access Manager displays a Login Page (depending on the authentication scheme configured in Access Manager for this URI).

logouturi	Specifies the URI of the logout page Notes: <ul style="list-style-type: none">• For ADF security enabled applications, <code>logouturi</code> should be configured based on logout guidelines in “Configuring Centralized Logout for Sessions Involving 11g WebGates” in the <i>Administrator’s Guide for Oracle Access Management</i>.• When using WebGate 11g, the value of the <code>logouturi</code> should be sought from the 11g WebGate Administrator.• When using WebGate 10g, the value of <code>logouturi</code> should be <code>/oamssso/logout.html</code>.
autologinuri	Specifies the URI of the autologin page. This is an optional parameter.

5. Disconnect from the Administration Server:

`disconnect()`

6. Restart the Administration Server and all the Managed Servers.

Using Multi Data Sources with Oracle RAC

Oracle recommends using GridLink data sources when developing new Oracle RAC applications. However, if you are using legacy applications and databases that do not support GridLink data sources, refer to the information in this appendix.

This appendix provides information about multi data sources and Oracle RAC and procedure for configuring multi data sources for an Enterprise Deployment.

See Also:

[About Multi Data Sources and Oracle RAC](#)

A multi data source provides an ordered list of data sources to use to satisfy connection requests.

[Typical Procedure for Configuring Multi Data Sources for an Enterprise Deployment](#)

You configure data sources when you configure a domain. If you want to use Multi Data Sources instead of GridLink data sources, replace the GridLink instructions with the instructions provided in this section.

A.1 About Multi Data Sources and Oracle RAC

A multi data source provides an ordered list of data sources to use to satisfy connection requests.

Normally, every connection request to this kind of multi data source is served by the first data source in the list. If a database connection test fails and the connection cannot be replaced, or if the data source is suspended, a connection is sought sequentially from the next data source on the list.

For more information about configuring Multi Data Sources with Oracle RAC, see "Using Multi Data Sources with Oracle RAC" in the *Oracle Fusion Middleware Configuring and Managing JDBC Data Sources for Oracle WebLogic Server*.

A.2 Typical Procedure for Configuring Multi Data Sources for an Enterprise Deployment

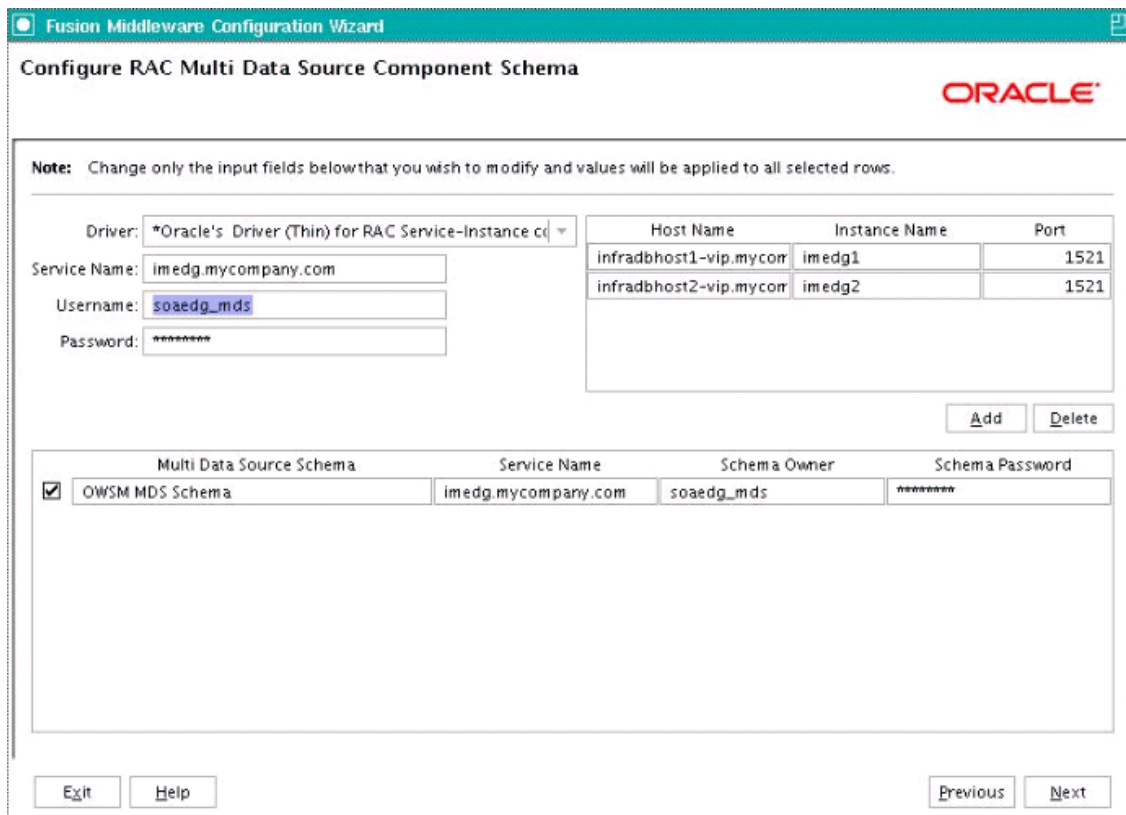
You configure data sources when you configure a domain. If you want to use Multi Data Sources instead of GridLink data sources, replace the GridLink instructions with the instructions provided in this section.

For example, when you are configuring the initial Administration domain for an Enterprise Deployment reference topology, you use the configuration wizard to define the characteristics of the domain, as well as the data sources.

The procedures for configuring the topologies in this Enterprise Deployment Guide include specific instructions for defining GridLink data sources with Oracle RAC. If you want to use Multi Data Sources instead of GridLink data sources, replace the GridLink instructions with the following:

1. In the Configure JDBC Component Schema screen:
 - a. Select the appropriate schemas.
 - b. For the RAC configuration for component schemas, **Convert to RAC multi data source**.
 - c. Ensure that the following data source appears on the screen with the schema prefix when you ran the Repository Creation Utility.
 - d. Click **Next**.
2. The Configure RAC Multi Data Sources Component Schema screen appears (Figure A-1).

Figure A-1 Configure RAC Multi Data Source Component Schema Screen



In this screen, do the following:

- a. Enter values for the following fields, specifying the connect information for the Oracle RAC database that was seeded with RCU.
 - **Driver:** Select **Oracle driver (Thin) for RAC Service-Instance connections, Versions:10, 11**.
 - **Service Name:** Enter the service name of the database.
 - **Username:** Enter the complete user name (including the prefix) for the schemas.
 - **Password:** Enter the password to use to access the schemas.

Index

D

data sources, [A-2](#)

P

PROCESSES parameter for database, [9-3](#)

R

RAC database, [A-2](#)

V

virtual servers, [5-2](#)

