

Oracle® Exalogic Elastic Cloud

Enterprise Deployment Guide for Oracle SOA Suite

Release EL X2-2, X3-2, X4-2, and X5-2

E47690-02

February 2015

Documentation for installers that describes how to install and configure Oracle SOA Suite on an Exalogic platform in an enterprise deployment.

Oracle Exalogic Elastic Cloud Enterprise Deployment Guide for Oracle SOA Suite, Release EL X2-2, X3-2, X4-2, and X5-2

E47690-02

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

Primary Author: Joe Paul (Writer), Janga Aliminati (Architect), Fermin Castro (Contributing Engineer)

Contributing Author: Peter Laquerre

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 about content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services unless otherwise set forth in an applicable agreement between you and Oracle. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services, except as set forth in an applicable agreement between you and Oracle.

Contents

Preface	xiii
Audience	xiii
Documentation Accessibility	xiii
Related Documents	xiii
Conventions	xiii
1 Overview	
1.1 About the Exalogic Enterprise Deployment Guide	1-1
1.2 Prerequisites	1-2
1.3 Benefits of Oracle Recommendations	1-2
1.3.1 Built-in Security	1-2
1.3.2 High Availability	1-2
1.4 Overview of Oracle Exalogic Configured Environment	1-2
1.4.1 Network	1-3
1.4.2 Sun ZFS Storage 7320 appliance	1-3
1.4.3 Oracle Software	1-3
2 Introduction and Planning	
2.1 Planning Your Deployment	2-1
2.1.1 Why the Deployment Topology in This Guide?	2-1
2.1.2 Alternative Deployment Topologies	2-2
2.1.2.1 Using an External Oracle HTTP Server Web Tier Instead of Oracle Traffic Director 2-2	
2.1.2.2 Using Oracle Exadata Instead of an Oracle Real Application Clusters (RAC) Database	2-2
2.1.3 Using a Worksheet to Plan for the Deployment Topology	2-3
2.2 Viewing the Oracle SOA Deployment Topology on Exalogic	2-3
2.3 Understanding the Topology Components	2-4
2.3.1 About EoIB and IPoIB Communication	2-5
2.3.2 About the Load Balancer	2-5
2.3.3 About the Web Tier	2-5
2.3.4 About the DMZ	2-6
2.3.5 About the Application Tier	2-7
2.3.5.1 Architecture Notes	2-7
2.3.6 About the Identity and Policy Stores	2-8

2.4	Hardware Requirements for the Oracle SOA on Exalogic	2-8
2.4.1	Hardware Load Balancer Requirements	2-8
2.4.2	Exalogic Machine Requirements	2-8
2.5	Software Components for an Exalogic Enterprise Deployment	2-8
2.5.1	Software Required for the Oracle SOA Deployment Topology on Exalogic	2-9
2.5.2	About Obtaining Software	2-9
2.5.3	Applying Patches and Workarounds	2-9
2.6	Road Map for the Reference Topology Installation and Configuration	2-9
2.6.1	Flow Chart of the Oracle SOA Exalogic Enterprise Deployment Process	2-10
2.6.2	Steps in the Oracle SOA Exalogic Enterprise Deployment Process	2-10

3 Configuring the Network for an Exalogic Enterprise Deployment

3.1	Overview of Preparing the Network for an Enterprise Deployment	3-1
3.2	About the Exalogic Network Configuration for the SOA Enterprise Topology	3-2
3.2.1	General Characteristics and Goals of the Exalogic Network Configuration	3-2
3.2.2	Map of the Network Interfaces Used by the Components of the SOA Topology on Exalogic	3-3
3.2.3	Explanation of the Network Interfaces Map	3-4
3.2.3.1	Communication with the Oracle Web Services Manager (OWSM) Managed Servers	3-4
3.2.3.2	Communication with the SOA Managed Servers	3-5
3.2.3.3	Communication with the Oracle Service Bus Managed Servers	3-5
3.2.3.4	Communication with the Oracle Traffic Director Instances	3-5
3.2.3.5	Communication with the WebLogic Server Administration Server	3-6
3.2.3.6	Communications with the WebLogic Server Node Manager and External Database	3-6
3.3	Configuring Virtual IP Addresses for IPoIB on Each Compute Node	3-6
3.3.1	Summary of the Required IPoIB Virtual IP Addresses	3-6
3.3.2	Creating the Virtual IP Addresses for the IPoIB Network on SOAHOST1 and SOAHOST2	3-7
3.3.3	Verifying the Required Virtual IP Addresses on the IPoIB Network	3-7
3.4	Configuring Virtual IP Addresses for EoIB on Each Compute Node	3-8
3.4.1	Summary of the Virtual IP Addresses for the EoIB Network Interfaces	3-8
3.4.2	Configuring the EoIB Network for the SOA Enterprise Topology	3-8
3.4.3	Creating the EoIB Virtual IPs for the WEBHOST1 and WEBHOST2 Compute Nodes ...	3-13
3.4.4	Verifying Connectivity Between Virtual IP Addresses	3-13
3.5	Defining the Required Hostname Resolution	3-14
3.6	Defining the Required Virtual Server Names	3-16
3.6.1	soa.mycompany.com	3-17
3.6.2	admin.mycompany.com	3-17
3.6.3	osb.mycompany.com	3-17
3.6.4	soainternal.mycompany.com	3-17
3.7	Configuring the Load Balancer	3-18
3.7.1	Load Balancer Requirements	3-18
3.7.2	Load Balancer Configuration Procedures	3-19
3.7.3	Load Balancer Configuration Details	3-19
3.8	Configuring Firewall Ports	3-21

4 Configuring Storage for an Exalogic Enterprise Deployment

4.1	Overview of Preparing Storage for an Exalogic Enterprise Deployment	4-1
4.1.1	General Information About the Exalogic Enterprise Deployment File System	4-1
4.1.2	Specific Information About the Exalogic File System	4-1
4.2	Shared Storage Recommendations for Exalogic Enterprise Deployments	4-2
4.2.1	Shared Storage Recommendations for Binary (Middleware Home) Directories	4-2
4.2.1.1	About the Binary (Middleware Home) Directories	4-2
4.2.1.2	About Using Redundant Binary (Middleware Home) Directories	4-2
4.2.2	Shared Storage Recommendations for Domain Configuration Files	4-3
4.2.2.1	About Oracle WebLogic Server Administration and Managed Server Domain Configuration Files	4-3
4.2.2.2	Shared Storage Requirements for Administration and Managed Server Domain Configuration Files	4-3
4.2.3	Shared Storage Recommendations for JMS File Stores and Transaction Logs	4-3
4.3	Directory Variables for an Oracle Fusion Middleware SOA Enterprise Deployment on Exalogic	4-4
4.4	Recommended Directory Locations for an Oracle Exalogic Enterprise Deployment	4-5
4.4.1	Shared Storage for Oracle SOA Enterprise Deployment on Exalogic	4-6
4.4.2	Private Storage for an Exalogic Enterprise Deployment	4-7
4.5	Configuring Exalogic Storage for Oracle SOA	4-8
4.5.1	Summary of the Storage Appliance Directories and Corresponding Mount Points ..	4-8
4.5.2	Prerequisite Storage Appliance Configuration Tasks	4-9
4.5.3	Creating the SOAEDG Project Using the Storage Appliance Browser User Interface (BUI)	4-9
4.5.4	Creating the Shares in the Project Using the BUI	4-10

5 Configuring the Compute Nodes for an Exalogic Enterprise Deployment

5.1	Overview of Preparing the Compute Nodes	5-1
5.2	Meeting Operating System Requirements	5-1
5.2.1	Meeting UNIX and Linux Requirements	5-2
5.2.1.1	Setting the Open File Limit	5-2
5.2.1.2	Setting Shell Limits	5-2
5.2.1.3	Increase Huge Pages Allocation	5-2
5.2.1.4	Configuring Local Hosts File	5-2
5.3	Synchronize the Node System Clock	5-3
5.4	Enabling Unicode Support	5-3
5.5	Configuring Users and Groups	5-3
5.6	Mounting the Shares for WEBHOST1 and WEBHOST2	5-4
5.7	Mounting the Shares for SOAHOST1 and SOAHOST2	5-5

6 Configuring a Database for an Exalogic Enterprise Deployment

6.1	Overview of Preparing the Database for an Enterprise Deployment	6-1
6.2	About Database Requirements	6-1
6.2.1	Database Host Requirements	6-2
6.2.2	Supported Database Versions	6-2
6.2.3	About Initialization Parameters	6-2

6.3	Creating Database Services	6-3
6.3.1	Creating Database Services for 10g and 11g Release 1 (11.1) Databases	6-3
6.3.2	Creating Database Services for 11g Release 2 (11.2) Databases	6-4
6.4	Loading the Oracle Fusion Metadata Repository in the Oracle RAC Database	6-5
6.5	Configuring SOA Schemas for Transactional Recovery Privileges	6-7
6.6	Backing Up the Database	6-8

7 Installing and Configuring Oracle Traffic Director for an Exalogic Enterprise Deployment

7.1	Overview of Installing and Configuring Oracle Traffic Director for an Exalogic Enterprise Deployment	7-1
7.2	Installing Oracle Traffic Director on WEBHOST1 and WEBHOST2	7-3
7.3	Creating and Starting the Traffic Director Administration Server	7-4
7.4	Register WEBHOST2 as an Administration Node	7-5
7.5	Creating a Configuration	7-6
7.6	Starting the Oracle Traffic Director Instances	7-7
7.7	Defining Oracle Traffic Director Virtual Servers for an Exalogic Enterprise Deployment	7-8
7.7.1	Creating an Origin-Server Pool	7-8
7.7.2	Creating the Additional Virtual Servers	7-9
7.7.3	Updating the Host Pattern Served by the SOAEXA Virtual Server	7-10
7.8	Deploying the Configuration and Testing the Virtual Server Addresses	7-11
7.9	Creating a Failover Group for Virtual Hosts	7-11
7.10	Backing the Web Tier	7-13

8 Creating a Domain for an Exalogic Enterprise Deployment

8.1	Overview of Creating a Domain	8-2
8.2	Installing Oracle Fusion Middleware	8-3
8.2.1	Installing JRockit	8-3
8.2.2	Installing WebLogic Server Using the Generic Installer	8-4
8.2.3	Installing Oracle Fusion Middleware SOA Suite	8-5
8.3	Verifying ADMINVHN in SOAHOST1	8-6
8.4	Running the Configuration Wizard on SOAHOST1 to Create a Domain	8-6
8.5	Post-Configuration and Verification Tasks	8-11
8.5.1	Creating boot.properties for the Administration Server on SOAHOST1	8-12
8.5.2	Configuring and Starting Node Manager on SOAHOST1 and SOAHOST2	8-12
8.5.2.1	Generating a properties file for Node Manager and Configuring it to use start scripts	8-12
8.5.2.2	Changing the Location of Node Manager Configuration Files	8-13
8.5.2.3	Editing the nodemanager.properties File	8-13
8.5.3	Starting the Administration Server on SOAHOST1	8-14
8.6	Associate the Domain with a Database OPSS Policy Store	8-15
8.7	Using an LDAP Authenticator (OID, OVD, OUD)	8-16
8.8	Moving the WebLogic Administrator to LDAP	8-18
8.8.1	Provisioning Admin Users and Groups in an LDAP Directory	8-19
8.8.2	Assigning the Admin Role to the Admin Group	8-20
8.8.3	Updating the boot.properties File and Restarting the System	8-21
8.9	Enabling Domain-Level Exalogic Enhancements	8-21

8.10	Validating GridLink Data Sources	8-22
8.11	Validating the Administration Server Configuration	8-22
8.12	Creating a Separate Domain Directory for Managed Servers in the Same Node as the Administration Server	8-23
8.13	Applying the Java Required Files (JRF) Template to the WSM-PM_Cluster	8-23
8.14	Disabling Host Name Verification	8-24
8.15	Starting and Validating the WLS_WSM1 Managed Server	8-25
8.16	Propagating the Domain Configuration to SOAHOST2	8-25
8.16.1	Propagating the Domain Configuration to SOAHOST2 Using the unpack Utility ..	8-26
8.16.2	Modify the Upload and Stage Directories to an Absolute Path	8-26
8.16.3	Disabling Host Name Verification for the WLS_WSM2 Managed Server	8-27
8.16.4	Starting Node Manager on SOAHOST2	8-27
8.16.5	Starting and Validating the WLS_WSM2 Managed Server	8-27
8.17	Configuring the Java Object Cache for Oracle WSM	8-28
8.18	Configuring Oracle Traffic Director for the WebLogic Domain	8-29
8.18.1	Configuring Oracle Traffic Director to Create Virtual Server Routes	8-29
8.18.2	Validating Access through Oracle Traffic Director	8-31
8.18.3	Turning on the WebLogic Plug-in Enabled Flag	8-31
8.18.4	Setting the Frontend URL for the Administration Console and Setting Redirection Preferences	8-31
8.19	Backing Up the WebLogic Domain Configuration	8-32

9 Extending the Domain for SOA Components

9.1	Overview of Extending the Domain for SOA Components	9-2
9.2	Pre-verifications for Extending the Domain for Oracle SOA Components	9-2
9.2.1	Verify Virtual IPs and Hostnames on SOAHOST1 and SOAHOST2	9-3
9.2.2	Synchronize System Clocks	9-3
9.2.3	Verifying Oracle Home Installation	9-3
9.3	Extending the Domain for SOA Components using the Configuration Wizard	9-3
9.4	Configuring Oracle Coherence for Deploying Composites	9-9
9.4.1	Enabling Communication for Deployment Using Unicast Communication	9-9
9.4.2	Specifying the Host Name Used by Oracle Coherence	9-10
9.5	Post-Configuration and Verification Tasks	9-12
9.5.1	Disabling Host Name Verification for the WLS_SOAn Managed Servers	9-12
9.5.2	Restarting the Node Manager on SOAHOST1	9-12
9.5.3	Propagating the Domain Changes to the Managed Server Domain Directory	9-13
9.5.4	Starting and Validating the WLS_SOA1 Managed Server	9-13
9.5.5	Propagating the Domain Configuration to SOAHOST2 Using the unpack Utility ..	9-14
9.5.6	Starting and Validating the WLS_SOA2 Managed Server	9-15
9.5.7	Validating GridLink Data Sources	9-16
9.6	Configuring Network Channels for HTTP and T3 Clients Through EoIB	9-16
9.6.1	Configuring Network Channels for SOA Servers on SOAHOST1 and SOAHOST2 ..	9-16
9.6.1.1	Creating an HTTP Client Channel	9-17
9.6.1.2	Creating the T3 Client Channel	9-18
9.7	Configuring Oracle Traffic Director with the Extended Domain	9-19
9.7.1	Configuring Access Through Oracle Traffic Director for the WLS_SOAn Managed Servers	9-19

9.7.1.1	Creating a New Route	9-19
9.7.2	Validating Access Through Oracle Traffic Director	9-20
9.7.3	Setting Server and HTTP URLs for SOA Servers	9-21
9.7.3.1	Webservice Local Optimization	9-22
9.8	Configuring a Default Persistence Store for Transaction Recovery	9-23
9.9	Configuring Coherence Caches for Dehydrations	9-23
9.9.1	Enabling the CacheEnabled property	9-24
9.9.2	Setting Server Properties for In-Process Coherence Cache for Dehydration	9-24
9.10	Updating SOA JVM settings	9-24
9.11	Enabling Cluster-Level Session Replication Enhancements	9-25
9.12	Configuring Oracle Adapters	9-27
9.12.1	Enabling High Availability for Oracle File and FTP Adapters	9-28
9.12.1.1	Using the Database Mutex Locking Operation	9-28
9.12.2	Enabling High Availability for Oracle JMS Adapters	9-31
9.12.3	Scaling the Oracle Database Adapter	9-32
9.13	Updating the Workflow Front End Address for Appropriate Task Display	9-32
9.14	Updating the B2B Instance Identifier for Transports	9-33
9.15	Backing Up the Oracle SOA Configuration	9-33

10 Extending the Domain to Include Oracle BPM

10.1	Overview of Extending the Domain to include Oracle BPM	10-1
10.2	Option 1: Extending a Domain to Include SOA and BPM	10-1
10.3	Option 2: Extending a SOA Domain to Include Oracle BPM	10-2
10.3.1	Running the Configuration Wizard on SOAHOST1 to Extend a SOA Domain to Include BPM	10-3
10.3.2	Propagating the Domain Configuration to the managed server directory in SOAHOST1 and to SOAHOST2	10-4
10.3.3	Starting the BPM Suite Components	10-5
10.3.4	Configuring Oracle Traffic Director for the WLS_SOAn Managed Servers	10-5
10.4	Backing Up the Oracle BPM Configuration	10-6

11 Extending a SOA Domain to Oracle Service Bus

11.1	Overview of Adding Oracle Service Bus to a SOA Domain	11-2
11.1.1	Prerequisites for Extending the SOA Domain to Include Oracle Service Bus	11-3
11.2	Installing the Required Oracle Service Bus Binaries	11-3
11.3	Verifying Virtual IP Addresses for OSB Managed Servers	11-5
11.4	Running the Configuration Wizard on SOAHOST1 to Extend a SOA Domain to Include Oracle Service Bus	11-5
11.5	Disabling Host Name Verification for the WLS_OSBn Managed Servers	11-9
11.6	Configuring Oracle Coherence for the Oracle Service Bus Result Cache	11-9
11.7	Configuring a Default Persistence Store for Transaction Recovery	11-10
11.8	Propagating the Domain Configuration to the Managed Server Directory in SOAHOST1 and to SOAHOST2	11-11
11.9	Starting the Oracle Service Bus Servers	11-12
11.10	Configuring Network Channels for HTTP and T3 Clients via EoIB	11-12
11.10.1	Creating HTTP Client Channels	11-12
11.10.2	T3 Client Channel	11-13

11.11	Validating the WLS_OSB Managed Servers	11-14
11.12	Configuring Oracle Traffic Director with the Extended Domain	11-15
11.12.1	Configuring Access Through Oracle Traffic Director for the WLS_OSBn Managed Servers	11-15
11.12.1.1	Creating a New Route	11-15
11.13	Setting the Front End HTTP Host and Port for OSB_Cluster	11-16
11.14	Validating Access Through Oracle Traffic Director and Load Balancer	11-17
11.15	High Availability for Oracle DB, File and FTP Adapters	11-17
11.16	Configuring Server Migration for the WLS_OSB Servers	11-18
11.17	Backing Up the Oracle Service Bus Configuration	11-18

12 Setting Up Node Manager for an Exalogic Enterprise Deployment

12.1	Overview of the Node Manager	12-1
12.2	Setting Up Node Manager	12-2
12.2.1	Changing the Location of Node Manager Configuration Files	12-2
12.2.2	Editing the Node Manager Property File	12-2
12.2.3	Starting Node Manager	12-3
12.3	Enabling Host Name Verification Certificates for Node Manager	12-3
12.3.1	Generating Self-Signed Certificates Using the utils.CertGen Utility	12-4
12.3.2	Creating an Identity Keystore Using the utils.ImportPrivateKey Utility	12-5
12.3.3	Creating a Trust Keystore Using the Keytool Utility	12-6
12.3.4	Configuring Node Manager to Use the Custom Keystores	12-6
12.3.5	Using a Common or Shared Storage Installation	12-7
12.3.6	Configuring Managed WebLogic Servers to Use the Custom Keystores	12-7
12.3.7	Changing the Host Name Verification Setting for the Managed Servers	12-8
12.4	Starting Node Manager	12-9

13 Configure Server Migration for an Exalogic Enterprise Deployment

13.1	Overview of Server Migration for an Exalogic Enterprise Deployment	13-1
13.2	Setting Up a User and Tablespace for the Server Migration Leasing Table	13-1
13.3	Creating a GridLink Data Source for Leasing Using the Oracle WebLogic Administration Console	13-2
13.4	Editing Node Manager's Properties File	13-2
13.5	Setting Environment and Superuser Privileges for the wlsifconfig.sh Script	13-4
13.6	Configuring Server Migration Targets	13-4
13.7	Testing the Server Migration	13-5
13.8	Backing Up the Server Migration Configuration	13-6

14 Managing the Topology for an Exalogic Enterprise Deployment

14.1	Overview of Managing the Topology	14-1
14.2	Tips for Deploying Composites and Artifacts in a SOA Enterprise Deployment Topology ..	14-2
14.3	Managing Space in the SOA Infrastructure Database	14-4
14.4	Configuring UMS Drivers	14-5
14.5	Scaling Up the Topology (Adding Managed Servers to Existing Nodes)	14-6
14.5.1	Planning for Scale Up	14-7

14.5.2	Scale-up Procedure for Oracle SOA	14-7
14.5.3	Scale-up Procedure for Oracle Service Bus	14-12
14.6	Scaling Out the Topology (Adding Managed Servers to New Nodes)	14-19
14.6.1	Prerequisites for Scaling Out the Topology	14-19
14.6.2	Scale-out Procedure for the Oracle SOA	14-20
14.6.3	Scale-out Procedure for Oracle Service Bus	14-26
14.7	Verifying Manual Failover of the Administration Server	14-35
14.7.1	Failing Over the Administration Server to a Different Node	14-35
14.7.2	Validating Access to SOAHOST2	14-36
14.7.3	Failing the Administration Server Back to SOAHOST1	14-36
14.8	Backing Up the Oracle SOA Enterprise Deployment	14-37
14.8.1	Backing Up the Database	14-37
14.8.2	Backing Up the Administration Server Domain Directory	14-37
14.8.3	Backing Up the Web Tier	14-37
14.8.4	Backing up the Middleware Home	14-38
14.9	Preventing Timeouts for SQLNet Connections	14-38
14.10	Recovering Failed BPEL and Mediator Instances	14-38
14.11	Configuring Web Services to Prevent Denial of Service and Recursive Node Attacks	14-39
14.12	Using Shared Storage for Deployment Plans and SOA Infrastructure Applications Updates	14-39
14.13	Using External BPEL Caches for Improved HAS and Performance Isolation	14-40
14.13.1	Setting the Server's bpel.cache.localStorage Property	14-40
14.13.2	Creating Cache Configuration Files and Start Scripts	14-41
14.13.3	Starting BPEL Cache Instances	14-41
14.14	Troubleshooting the Topology in an Enterprise Deployment	14-41
14.14.1	Page Not Found When Accessing soa-infra Application Through Load Balancer	14-42
14.14.2	Soa-infra Application Fails to Start Due to Deployment Framework Issues (Coherence)	14-42
14.14.3	SOA, OSB, or WMS Servers Fail to Start Due to Maximum Number of Processes Available in Database	14-43
14.14.4	Administration Server Fails to Start After a Manual Failover	14-43
14.14.5	Error While Activating Changes in Administration Console	14-44
14.14.6	SOA/OSB Server Not Failed Over After Server Migration	14-44
14.14.7	SOA/OSB Server Not Reachable From Browser After Server Migration	14-44
14.14.8	SOA Server Stops Responding after Being Active and Stressed for a Period of Time 14-45	
14.14.9	Configured JOC Port Already in Use	14-45
14.14.10	SOA or OSB Server Fails to Start	14-45
14.14.11	SOA Coherence Cluster Conflicts when Multiple Clusters Reside in the Same Node 14-46	
14.14.12	Sudo Error Occurs During Server Migration	14-46
14.14.13	Transaction Timeout Error	14-46
14.14.14	Exceeded Maximum Size Error Messages	14-47
A.1	About Multi Data Sources and Oracle RAC	A-1
A.2	Typical Procedure for Configuring Multi Data Sources for an EDG Topology	A-1
B.1	Hosts, Virtual Hosts, and Virtual IP Addresses for Oracle SOA	B-1
B.2	Directory Mapping	B-2
B.3	Port Mapping	B-2

B.4	Database Details	B-3
B.5	Web Tier Details	B-4
B.6	Application Tier Details	B-4
C.1	Viewing the Oracle SOA Deployment Topology with Oracle HTTP Server on Exalogic	C-1
C.2	Understanding the Oracle SOA with Oracle HTTP Server Topology Components	C-3
C.2.1	About the Oracle HTTP Server Instances in the Web Tier	C-3
C.2.2	About the Oracle Traffic Director Instances on the Application Tier	C-3
D.1	Creating a GridLink Data Source Using the Oracle WebLogic Server Administration Console	D-1

Preface

This preface describes the audience, contents and conventions used in the *Oracle Fusion Middleware Exalogic Enterprise Deployment Guide for Oracle SOA Suite*.

Audience

This guide is intended for system administrators who are responsible for installing and configuring Oracle Fusion Middleware enterprise deployments.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at

<http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc>.

Access to Oracle Support

Oracle customers that have purchased support have access to electronic support through My Oracle Support. For information, visit

<http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info> or visit

<http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs> if you are hearing impaired.

Related Documents

For more information, see the following documents in the Oracle Other Product One Release 7.0 documentation set or in the Oracle Other Product Two Release 6.1 documentation set:

- *Oracle Other Product One Release Notes*
- *Oracle Other Product One Configuration Guide*
- *Oracle Other Product Two Getting Started Guide*
- *Oracle Other Product Two Reference Guide*
- *Oracle Other Product Two Tuning and Performance Guide*

Conventions

The following text conventions are used in this document:

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

This chapter provides an overview of the enterprise topology for Oracle SOA Suite.

This guide describes reference enterprise topology for the Oracle SOA Infrastructure components of Oracle Fusion Middleware. It also provides detailed instructions and recommendations to create the topology by following the Exalogic enterprise deployment guidelines.

This chapter contains the following sections:

- [Section 1.1, "About the Exalogic Enterprise Deployment Guide"](#)
- [Section 1.2, "Prerequisites"](#)
- [Section 1.3, "Benefits of Oracle Recommendations"](#)
- [Section 1.4, "Overview of Oracle Exalogic Configured Environment"](#)

1.1 About the Exalogic Enterprise Deployment Guide

An Exalogic enterprise deployment is an Oracle best practices blueprint based on proven Oracle high availability, security and performance technologies and recommendations for Oracle Fusion Middleware. The high-availability best practices described in this book make up one of several components of high availability best practices for all Oracle products across the entire technology stack—Oracle Database, Oracle Fusion Middleware, Oracle Exalogic machine, and Oracle Applications.

An Oracle Fusion Middleware Exalogic enterprise deployment:

- Considers various business service level agreements (SLA) to make high-availability best practices as widely applicable as possible
- Leverages database grid servers and storage grid with low-cost storage to provide highly resilient, lower cost infrastructure
- Uses results from extensive performance impact studies for different configurations to ensure that the high-availability architecture is optimally configured to perform and scale to business needs
- Enables control over the length of time to recover from an outage and the amount of acceptable data loss from a natural disaster
- Evolves with each Oracle version and is completely independent of hardware and operating system

1.2 Prerequisites

Setup and commissioning of Oracle Exalogic machine, including initial storage and networking configuration, as described in *Oracle Fusion Middleware Exalogic Machine Owner's Guide*.

1.3 Benefits of Oracle Recommendations

The Oracle Fusion Middleware configurations discussed in this guide are designed to ensure security of all transactions, maximize hardware resources, and provide a reliable, standards-compliant system for enterprise computing with a variety of applications. The security and high availability benefits of the Oracle Fusion Middleware configurations are realized through isolation in firewall zones and replication of software components.

This section contains the following topics:

- [Section 1.3.1, "Built-in Security"](#)
- [Section 1.3.2, "High Availability"](#)

1.3.1 Built-in Security

The Exalogic enterprise deployment architectures are secure because every functional group of software components is isolated in its own tier, and all traffic is restricted by protocol and port. The following characteristics ensure security at all needed levels, as well as a high level of standards compliance:

- Even if external communication is received on port 80, it is redirected to port 443
- External communication uses the Secure Socket Layer (SSL) secure Web Protocol. This is terminated at the site's load balancer.
- Communication from external clients does not go beyond the Load Balancing Router level.
- No direct communication from the Load Balancing Router to the data tier is allowed.
- Direct communication across two firewalls at any one time is prohibited.
- If a communication begins in one firewall zone, it must end in the next firewall zone.
- All communication between components across firewalls is restricted by port and protocol, according to firewall rules.

1.3.2 High Availability

The Exalogic enterprise deployment architectures are highly available, because each component or functional group of software components is replicated on a different computer, and configured for component-level high availability without a single point of failure.

1.4 Overview of Oracle Exalogic Configured Environment

Before you start implementing the Oracle Exalogic enterprise deployment topology, you should understand the current state of the Exalogic environment.

It is assumed that you have completed all tasks described in the *Oracle Fusion Middleware Exalogic Machine Owner's Guide*, which discusses your data center site preparation, Oracle Exalogic machine commissioning, initial networking configuration including IP address assignments, and initial setup of the Sun ZFS Storage 7320 appliance.

This section describes the state of the Exalogic configured environment before Exalogic enterprise deployment.

It discusses the following topics:

- [Section 1.4.1, "Network"](#)
- [Section 1.4.2, "Sun ZFS Storage 7320 appliance"](#)
- [Section 1.4.3, "Oracle Software"](#)

1.4.1 Network

Before you start configuring the Exalogic enterprise deployment topology, you must run the Exalogic Configuration Utility to complete the following tasks, as described in the chapter "Initial Configuration of Exalogic Machine Using Oracle Exalogic Configuration Utility" in the *Oracle Fusion Middleware Exalogic Machine Owner's Guide*:

- Configuration of IP addresses for all Exalogic compute nodes and the Sun ZFS Storage 7320 appliance.
- Configuration of InfiniBand gateway switches.
- Configuration of the Cisco Ethernet management switch.
- Setup and verification of the default IP over InfiniBand (IPoIB) link spanning all compute nodes.
- Setup and verification of the default Ethernet over InfiniBand (EoIB) link for connectivity with components of the topology running on Ethernet.
- Configuration of the default InfiniBand partition that covers all of the compute nodes in Exalogic Machine.

1.4.2 Sun ZFS Storage 7320 appliance

The initial configuration of the Sun ZFS Storage 7320 appliance in your Oracle Exalogic machine is completed at the time of manufacturing. For more information about default shares (Exported File Systems), see the "Default Storage Configuration" section in the *Oracle Fusion Middleware Exalogic Machine Owner's Guide*.

1.4.3 Oracle Software

Oracle Linux 5.5 is pre-installed on each of the compute nodes in your Oracle Exalogic machine.

You must save the installation binaries, including Oracle Middleware Home, on a shared file system on the Sun ZFS Storage 7320 appliance. Before you can do so, you must configure shared storage by creating a Project and defining shares and LUNs to set up the directory structure, as necessary. Note down the mount point for such shares, so you can mount the required locations or directories from Exalogic compute nodes.

Note: You can download the Oracle WebLogic 10.3.6 software from <http://edelivery.oracle.com>. Select **Oracle Fusion Middleware** as the Product Pack, **Linux x86-64** as the Platform, and **Oracle Fusion Middleware 11g Media Pack for Exalogic** as the Media Pack.

Introduction and Planning

This chapter describes and illustrates the Exalogic enterprise deployment reference topology described in this guide and helps you plan your deployment.

The key to a successful Exalogic enterprise deployment is planning and preparation. The road map for installation and configuration in this chapter directs you to the appropriate chapters for the tasks you need to perform. Use this chapter to help you plan your Oracle SOA Exalogic enterprise deployment on an Exalogic platform.

You can use [Appendix B, "Worksheet for Oracle SOA Enterprise Deployment on Exalogic Topology"](#) to help you keep track of information.

This chapter contains the following topics:

- [Planning Your Deployment](#)
- [Viewing the Oracle SOA Deployment Topology on Exalogic](#)
- [Understanding the Topology Components](#)
- [Hardware Requirements for the Oracle SOA on Exalogic](#)
- [Software Components for an Exalogic Enterprise Deployment](#)
- [Road Map for the Reference Topology Installation and Configuration](#)

2.1 Planning Your Deployment

This section provides information to help you plan the deployment of Oracle SOA on Exalogic:

- [Section 2.1.1, "Why the Deployment Topology in This Guide?"](#)
- [Section 2.1.2, "Alternative Deployment Topologies"](#)
- [Section 2.1.3, "Using a Worksheet to Plan for the Deployment Topology"](#)

2.1.1 Why the Deployment Topology in This Guide?

When planning your deployment, you should be aware that this guide provides detailed instructions for implementing the specific reference topology described in this chapter.

This topology takes advantage of key features of the Exalogic platform, including:

- The high bandwidth and performance of the Exalogic internal Infiniband (IPoIB) network fabric
- The software load balancing capabilities of Oracle Traffic Director.

This topology also takes advantage of the several optimizations specific to the Oracle SOA Suite for the Exalogic platform. For example:

- XML Parsing and transformation optimizations
- XDK optimization
- Usage of Coherence as a read-cache for BPEL process state and audit data
- Scattered reads and gathered writes optimizations
- Self-tuning Thread optimization

In this specific topology, Oracle Traffic Director is used as both a Web Listener and as a client-side load balancer between Oracle Traffic Director instances and the SOA managed servers, as well as communication between the SOA managed servers.

In this configuration, you can take advantage of the Exalogic default IPoIB network for all internal communications between the Traffic Director instances and the SOA Suite compute nodes.

Only external traffic between the Traffic Director instances and external users is on the Exalogic Ethernet over IB (EoIB) network.

2.1.2 Alternative Deployment Topologies

Besides the topologies discussed in this guide, you can consider alternative Oracle SOA topologies on Exalogic.

This guide does not provide specific instructions for implementing these alternative topologies, but consider the following when you are preparing your environment for an Oracle SOA deployment on Exalogic:

- [Using an External Oracle HTTP Server Web Tier Instead of Oracle Traffic Director](#)
- [Using Oracle Exadata Instead of an Oracle Real Application Clusters \(RAC\) Database](#)

2.1.2.1 Using an External Oracle HTTP Server Web Tier Instead of Oracle Traffic Director

As described in [Section 2.2](#), the topology in this guide uses Oracle Traffic Director as both a Web server and an internal load balancer. This configuration requires that you dedicate two compute nodes to hosting the Oracle Traffic Director instances.

An alternative topology is to use Oracle HTTP Server Web Tier as the web server and Oracle Traffic Director as the internal load balance. This topology can be used if you have Oracle HTTP Server Web Tier already running outside Exalogic. Oracle Traffic Director will be installed on the Exalogic rack.

Refer to [Appendix C, "SOA Exalogic Enterprise Topology with Oracle HTTP Server"](#) for a diagram of a typical Oracle SOA topology on Exalogic with an external Oracle HTTP Server Web tier.

2.1.2.2 Using Oracle Exadata Instead of an Oracle Real Application Clusters (RAC) Database

The reference topology in this guide provides information on using an external Real Application Clusters (RAC) database on commodity hardware as the repository for product schemas and security stores.

You can also connect to a Real Application Clusters (RAC) database on an Oracle Exadata Database Machine using the Infiniband fabric. For more information, see

"Connecting Exalogic and Exadata Machines" in the *Oracle Exalogic Elastic Cloud Multi-Rack Cabling Guide*.

2.1.3 Using a Worksheet to Plan for the Deployment Topology

The key to a successful Exalogic enterprise deployment is planning and preparation. The road map for installation and configuration in this chapter directs you to the appropriate chapters for the tasks you need to perform.

Use this chapter to help you plan your Oracle SOA Suite Exalogic enterprise deployment on an Exalogic platform.

You can also use [Appendix B, "Worksheet for Oracle SOA Enterprise Deployment on Exalogic Topology"](#) to help you keep track of information, such as host names, IP addresses, and other important information as you procure and identify the machines and resources required for this deployment.

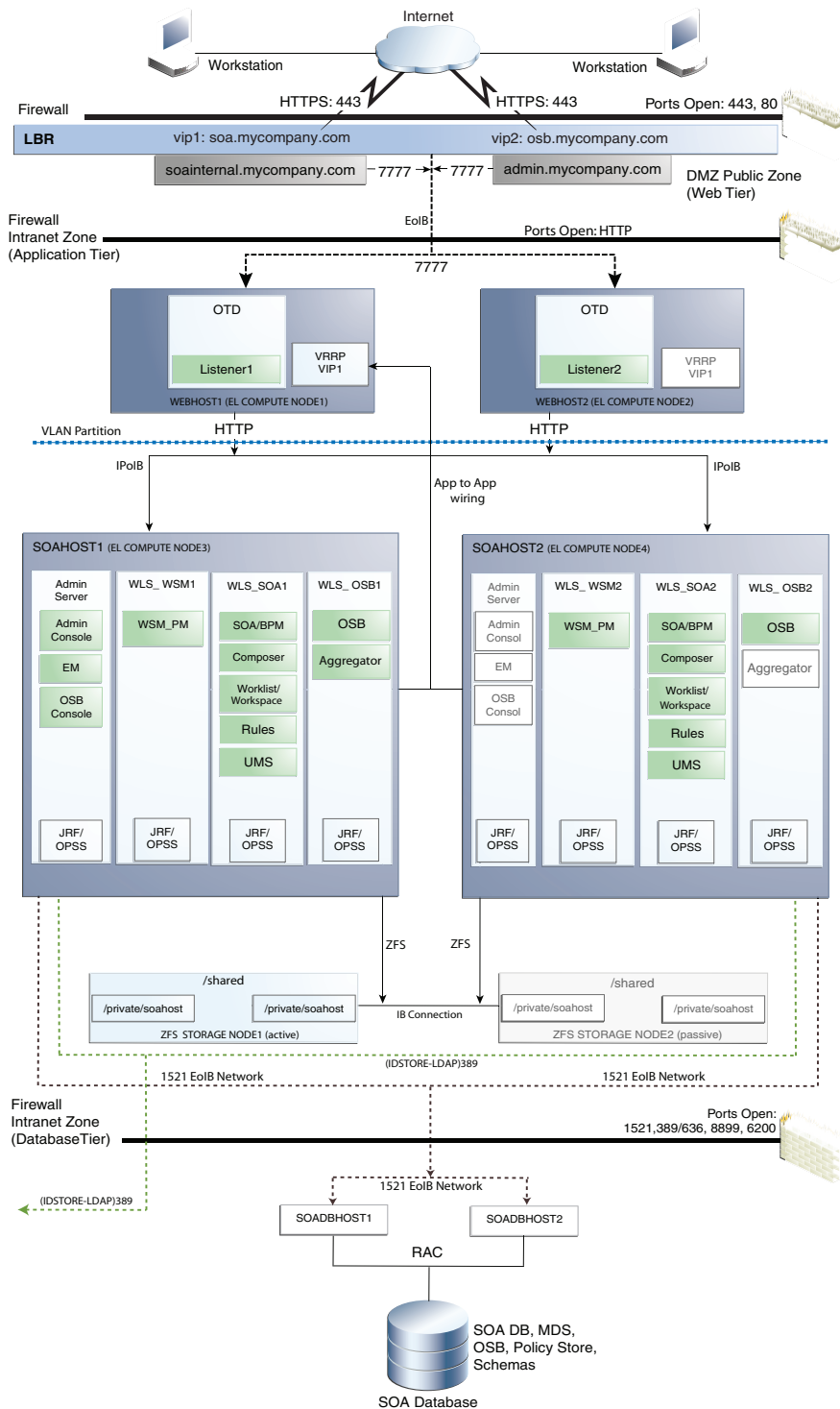
2.2 Viewing the Oracle SOA Deployment Topology on Exalogic

[Figure 2-1](#) provides a diagram of a standard, reference topology for Oracle SOA on Exalogic.

In this specific topology, the Web tier consists of Oracle Traffic Director instances, and the Exalogic machine is connected to a remote Oracle RAC database over a 10 Gb Ethernet connection.

For a detailed description of the elements of the topology, see [Section 2.3, "Understanding the Topology Components"](#).

Figure 2–1 Oracle SOA on Exalogic, Deployed with Oracle Traffic Director and an Oracle RAC Database



2.3 Understanding the Topology Components

The topologies consist of three tiers, which are described in the following sections:

- [Section 2.3.1, "About EoIB and IPoIB Communication"](#)
- [Section 2.3.2, "About the Load Balancer"](#)

- [Section 2.3.3, "About the Web Tier"](#)
- [Section 2.3.4, "About the DMZ"](#)
- [Section 2.3.5, "About the Application Tier"](#)
- [Section 2.3.6, "About the Identity and Policy Stores"](#)

2.3.1 About EoIB and IPoIB Communication

When you initially set up your Exalogic machine, the default network is running IP over Infiniband (IPoIB). For the different purposes of the topology described in this guide, you must configure Ethernet over Infiniband (EoIB) network access in addition to the IPoIB network. For more information, see "Configuring Ethernet Over InfiniBand" in the *Oracle Exalogic Elastic Cloud Machine Owner's Guide*.

The following four types of communication must be configured for the Oracle SOA enterprise deployment on Exalogic:

- For the Oracle Traffic Director hosts, the IP addresses must be EoIB addresses accessible from the load balancer. The Oracle Traffic Director IP addresses are the only addresses accessible from the DMZ network.
- For the application tier, the SOAHOST machines IP addresses must be EoIB addresses that can access the Oracle RAC database SCAN and VIP addresses. Additionally, SOA servers use IPoIB address as main listen addresses for internal invocations and for RMI interactions inside the Exalogic rack.
- Communication and routing between Oracle Traffic Director hosts and the application tier must be only over IPoIB.
- For communication between the application tier components, for example, Oracle Mediator and internal JMS destinations routing must be on IPoIB. Any front end address that is exposed ONLY for internal consumption, will use and IPoIB virtual IP on Oracle Traffic Director hosts.
- SOA Servers can also be accessed externally for RMI/JMS/T3 invocations and HTTP invocations. These take place for remote deployments, for external JMS producers and consumers and for other operations that use a listen address of the SOA servers that is available outside the Exalogic rack (EoIB).

For more information about IPoIB and EoIB network configuration, see [Section 3.2, "About the Exalogic Network Configuration for the SOA Enterprise Topology."](#)

2.3.2 About the Load Balancer

The hardware load balancer routes HTTP requests from users to the Oracle Traffic Director instances in the Web tier. The requests come in on a secure port (443) and are then routed to the Oracle Traffic Director instances via a non-secure port (7777).

The communication from the hardware load balancer to the Web tier (WEBHOST1 and WEBHOST2, in this case) is entirely over EoIB.

In addition to user traffic, the load balancer also routes Administrator requests to the administration server in the Oracle WebLogic Server domain on the application tier. This traffic is routed via a dedicated virtual server address called `admin.mycompany.com`.

2.3.3 About the Web Tier

With Exalogic, you can take advantage of Oracle Traffic Director capabilities.

In this topology, the Oracle Traffic Director instances serve two purposes:

- They receive HTTP requests coming in from the hardware load balancer (over the EoIB network) and then route those requests (over the IPoIB network) to the SOAHOSTs compute nodes on the application tier.
- They route requests from the application tier components (over the IPoIB network) including JMS (RMI) traffic, to other application tier components, such as callbacks and internal Web services invocations.

The internal application to application requests, which are routed only over the internal IPoIB network, are routed via a virtual IP address that is depicted as VIP1 in the topology diagram (Figure 2–1).

The Oracle Traffic Director instances are configured as part of a failover group. In this configuration, Oracle Traffic Director uses an implementation of the Virtual Routing Redundancy Protocol (VRRP) to provide failover capabilities. If an Oracle Traffic Director instance fails, IP addresses enabled on it are migrated to surviving instances, using VRRP. For information about configuring failover groups, see [Section 7.9, "Creating a Failover Group for Virtual Hosts."](#)

Consider these additional characteristics of the Web tier:

- WEBHOST1 and WEBHOST2 host Oracle Traffic Director. URI conditions and Host Server configuration enables requests to be proxied from the Oracle Traffic Director servers to a WebLogic Server running in the application tier.
- Oracle Traffic Director distributes the requests that it receives from clients to servers in the back end based on the specified load-balancing method, routes the requests based on specified rules, caches frequently accessed data, prioritizes traffic, and controls the quality of service.
- The Oracle Traffic Director Servers process requests received using the URLs `soa.mycompany.com`, `osb.mycompany.com` and `soainternal.mycompany.com`, and `admin.mycompany.com`. The name `admin.mycompany.com` is only resolvable inside the firewall. This prevents access to sensitive resources such as the Oracle WebLogic Server console and Oracle Enterprise Manager Fusion Middleware Control console from the public domain.

2.3.4 About the DMZ

A DMZ is a means of restricting access to components of your infrastructure to those that actually need it. In the examples in this guide, there is a public DMZ. This is where the outside world gains access to your systems. You place into this zone only those components that the outside world must access, such as the Load Balancers and Web Tiers. If users from the outside world attempts to access any servers or services below this zone, they are prevented from doing so by firewalls. The public zone is configured so that the servers in this zone can interact with the application servers in the private zone.

- The public zone—This is where the outside world gains access to your systems. You place into this zone only those components that the outside world must access, such as the Load Balancers and Web Tiers. If users from the outside world attempts to access any servers or services below this zone, they are prevented from doing so by firewalls.

The public zone is configured so that the servers in this zone can interact with the application servers in the private zone.

- The intranet zone—This is where you place servers that contain core services, such as databases. These services are very tightly controlled by the organization as they contain the most sensitive data.

By using this approach, you restrict access to information to only those components that require it. This approach is useful where you have users coming in from outside of your organization. If, instead of an extranet, you are setting up an intranet, where all communication is from trusted sources, then you might reasonably decide to do away with the public DMZ.

2.3.5 About the Application Tier

The application tier is the tier where Oracle SOA and Java EE applications are deployed. Products such as Oracle WSM, Oracle SOA Service Engines, Oracle BPEL Caches and Oracle Service Bus are deployed in this tier. Applications in this tier benefit from the High Availability support of Oracle WebLogic Server and Oracle Fusion Middleware.

In this tier, two compute nodes named SOAHOST1 and SOAHOST2 run Oracle WebLogic managed servers for running SOA components, such as BPEL Process Manager. The managed servers are configured for active-active mode.

SOAHOST1 and SOAHOST2 also run the Oracle WebLogic Server Administration Console and Oracle Enterprise Manager Fusion Middleware Control, but in an active-passive configuration. The active-passive configuration of the Administration Server is necessary because only one Administration Server can be running within a domain. In the illustration, the Administration Server on SOAHOST1 is currently in the active state, but you can failover manually to the Administration Server on SOAHOST2.

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

Applications requiring external HTTP access use Oracle Traffic Director as proxy.

Oracle WebLogic Server Console, Oracle Enterprise Manager Fusion Middleware Control console, and Oracle Fusion Middleware SOA Consoles are only accessible through a virtual host configured on the load balancer, which is only available inside the firewall.

2.3.5.1 Architecture Notes

- The Oracle WebLogic Server Administration Console and Oracle Enterprise Manager Fusion Middleware Control are always bound to the listen address of the Administration Server.
- The WebLogic Administration Server, (running both Oracle Enterprise Manager Fusion Middleware Control and Oracle WebLogic Server Administration Console), is a singleton service. It runs on only one node at a time. In the event of failure, it is restarted on a surviving node that can mount the exact same path and contents that the Admin Server used in its original node.
- The managed servers WLS_SOA1 and WLS_SOA2 are deployed in a cluster, and Oracle SOA Infrastructure, Service Engines, and Adapters are deployed to this cluster.
- The managed servers WLS_WSM1 and WLS_WSM2 are deployed in a cluster and run Oracle Web Service Policy Manager.
- The managed servers WLS_OSB1 and WLS_OSB2 are deployed in a cluster and run Oracle Service Bus.

- Additional JVMs are created and started in each of the available compute nodes (SOAHOST1 and SOAHOST2) to run Coherence Caches for BPEL Dehydration.

2.3.6 About the Identity and Policy Stores

Authorization policies are stored in Oracle Database and either Oracle Internet Directory or Oracle Unified Directory are used for authentication. For more information, see the *Oracle Exalogic Elastic Cloud Enterprise Deployment Guide for Oracle Identity Management*.

2.4 Hardware Requirements for the Oracle SOA on Exalogic

The following sections describe the hardware requirements for the SOA enterprise topologies on Exalogic:

- [Hardware Load Balancer Requirements](#)
- [Exalogic Machine Requirements](#)

2.4.1 Hardware Load Balancer Requirements

The Oracle Fusion Middleware Exalogic enterprise deployment requires a hardware load balancer to route requests to the Web tier. For information about the minimum set of features required for the load balancer in this topology, see [Section 3.7.1, "Load Balancer Requirements."](#)

2.4.2 Exalogic Machine Requirements

Oracle Exalogic is an integrated hardware and software system designed to provide a complete platform for a wide range of application types and widely varied workloads. Exalogic is designed to fully leverage an internal InfiniBand fabric that connects all of the processing, storage, memory and external network interfaces within an Exalogic machine to form a single, large computing device.

For complete information about the hardware options available for Exalogic machines, see "Exalogic Hardware Configurations" in the *Oracle Exalogic Elastic Cloud Machine Owner's Guide*.

For any of the topologies described in this guide, an Exalogic machine eighth rack can be used. For more information, see [Section 2.2, "Viewing the Oracle SOA Deployment Topology on Exalogic."](#)

You can assign the compute nodes on the Exalogic rack as follows:

- Assign two compute nodes to the Application Tier. These will be referred to as SOAHOST1 and SOAHOST2.
- For Oracle Traffic Director, assign two additional compute nodes to the Oracle Traffic Director instances. These will be referred to as WEBHOST1 and WEBHOST2.

2.5 Software Components for an Exalogic Enterprise Deployment

This section describes the software required for an Oracle SOA Exalogic enterprise deployment.

This section contains the following topics:

- [Section 2.5.1, "Software Required for the Oracle SOA Deployment Topology on Exalogic"](#)
- [Section 2.5.2, "About Obtaining Software"](#)
- [Section 2.5.3, "Applying Patches and Workarounds"](#)

2.5.1 Software Required for the Oracle SOA Deployment Topology on Exalogic

[Table 2–1](#) lists the Oracle software you need to obtain before starting the procedures in this guide.

Table 2–1 Software Versions Used

Product	Version
Oracle Database 10g or 11g	Oracle Database 10g distribution (10.2.0.4 or later SE or EE version of the database) using the AL32UTF8 character set. Oracle Database Server 11g distribution (11.1.0.7 or later SE or EE version of the database), using the AL32UTF8 character set.
Oracle Traffic Director	11.1.1.7.0
Oracle JRockit	jrockit-jdk1.6.0_29-R28.2.0-4.0.1 or newer
Oracle WebLogic Server	10.3.6.0
Oracle SOA Suite	11.1.1.7.0
Repository Creation Assistant	11.1.1.7.0
Oracle Service Bus	11.1.1.7.0

2.5.2 About Obtaining Software

For complete information about downloading Oracle Fusion Middleware software, see the *Oracle Fusion Middleware 11g Release 1 Download, Installation, and Configuration Readme* for this release, at: http://docs.oracle.com/cd/E23104_01/download_readme.htm

2.5.3 Applying Patches and Workarounds

See the Oracle Fusion Middleware Release Notes for your platform and operating system for a list of patches to apply. You **must** apply the patches to ensure that your software operates as expected.

Patches are available for download from <http://support.oracle.com>. You can find instructions for deploying each patch in the enclosed README.html file in each patch archive.

2.6 Road Map for the Reference Topology Installation and Configuration

Before beginning your Oracle SOA Exalogic enterprise deployment on Exalogic, review the flow chart in [Figure 2–2, "Flow Chart of the Oracle SOA Exalogic Enterprise Deployment Process"](#). This flow chart illustrates the high-level process for completing the Exalogic enterprise deployment documented in this guide. [Table 2–2](#) describes the steps in the flow chart and directs you to the appropriate section or chapter for each step.

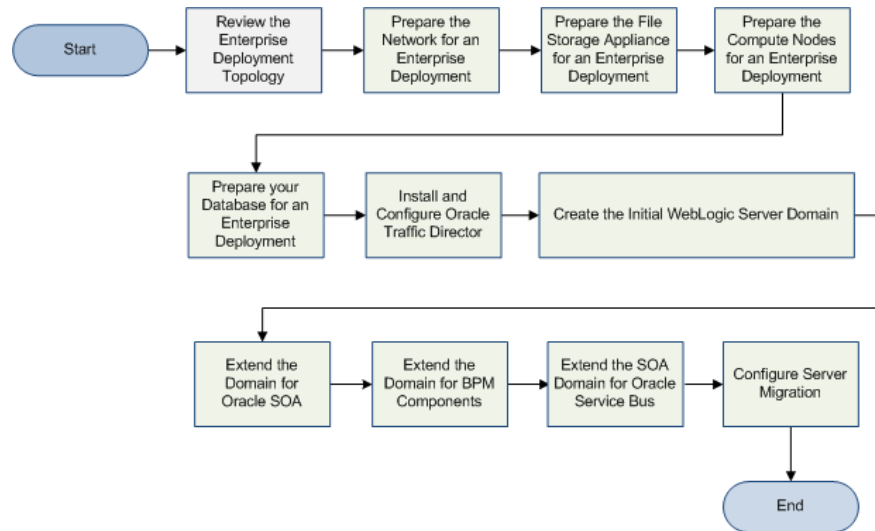
This section covers the following topics:

- [Section 2.6.1, "Flow Chart of the Oracle SOA Exalogic Enterprise Deployment Process"](#)
- [Section 2.6.2, "Steps in the Oracle SOA Exalogic Enterprise Deployment Process"](#)

2.6.1 Flow Chart of the Oracle SOA Exalogic Enterprise Deployment Process

Figure 2–2 provides a flow chart of the Oracle SOA Exalogic enterprise deployment process. Review this chart to become familiar with the steps that you must follow, based on the existing environment.

Figure 2–2 Flow Chart of the Oracle SOA Exalogic Enterprise Deployment Process



2.6.2 Steps in the Oracle SOA Exalogic Enterprise Deployment Process

Table 2–2 describes each of the steps in the Exalogic enterprise deployment process flow chart for Oracle SOA, shown in Figure 2–2. The table also provides information on where to obtain more information about each step in the process.

Table 2–2 Steps in the Oracle SOA Exalogic Enterprise Deployment Process

Step	Description	More Information
Review the Exalogic Enterprise Deployment Topology	Review the recommended topology and plan the topology best suited for your organization and applications.	Section 2.1, "Planning Your Deployment"
Prepare the Network for an Exalogic Enterprise Deployment	To prepare your network for an Exalogic enterprise deployment, understand concepts, such as virtual server names and IPs and virtual IPs, and configure your load balancer by defining virtual host names.	Chapter 3, "Configuring the Network for an Exalogic Enterprise Deployment"
Prepare your File Storage Appliance for an Exalogic Enterprise Deployment	To prepare your file system for an Exalogic enterprise deployment, review the terminology for directories and directory environment variables, and configure shared storage.	Chapter 4, "Configuring Storage for an Exalogic Enterprise Deployment"

Table 2–2 (Cont.) Steps in the Oracle SOA Exalogic Enterprise Deployment Process

Step	Description	More Information
Prepare the Compute Nodes for an Exalogic Enterprise Deployment	To prepare your servers for an Exalogic enterprise deployment, ensure that your compute nodes meet hardware and software requirements, enable Unicode support and Virtual IP Addresses, mount shared storage, configure users and groups, and, if necessary, install software onto multi-homed systems.	Chapter 5, "Configuring the Compute Nodes for an Exalogic Enterprise Deployment"
Prepare the Oracle RAC Database for an Exalogic Enterprise Deployment	To prepare an Oracle RAC database for an Exalogic enterprise deployment, review database requirements, create database services, load the metadata repository, in the Oracle RAC database, configure SOA schemas for transactional recovery privileges, and back up the database.	Chapter 6, "Configuring a Database for an Exalogic Enterprise Deployment"
Install and Configure Oracle Traffic Director on Exalogic Compute Nodes	Install and configure Oracle Traffic Director.	Chapter 7, "Installing and Configuring Oracle Traffic Director for an Exalogic Enterprise Deployment"
Create the Initial WebLogic Server Domain	Run the Configuration Wizard to create the initial WebLogic Server domain.	Chapter 8, "Creating a Domain for an Exalogic Enterprise Deployment"
Extend the Domain for Oracle SOA	Use the Configuration Wizard to extend the domain to include Oracle SOA components.	Chapter 9, "Extending the Domain for SOA Components"
Extend the Domain for BPM Components	Use the Configuration Wizard to extend the domain to include Oracle BPM.	Chapter 10, "Extending the Domain to Include Oracle BPM"
Extend the Domain for Oracle Service Bus	Use the Configuration Wizard to extend the domain to include Oracle Service Bus.	Chapter 11, "Extending a SOA Domain to Oracle Service Bus"
Configure Node Manager	Set up Node manager by enabling host name verification, starting Node Manager, and configuring WebLogic Servers to use custom keystores.	Chapter 12, "Setting Up Node Manager for an Exalogic Enterprise Deployment"
Configure Server Migration	Configure server migration for the WLS_OSBIM1, WLS_SOA1, WLS_OSB2, and WLS_SOA2 Managed Servers. The WLS_OSB1 and WLS_SOA1 Managed Server are configured to restart on SOAHOST2 should a failure occur. The WLS_OSB2 and WLS_SOA2 Managed Servers are configured to restart on SOAHOST1 should a failure occur.	Chapter 13, "Configure Server Migration for an Exalogic Enterprise Deployment"

Configuring the Network for an Exalogic Enterprise Deployment

This chapter describes the prerequisites for the Oracle SOA Infrastructure Exalogic enterprise deployment topologies.

This chapter includes the following topics:

- [Overview of Preparing the Network for an Enterprise Deployment](#)
- [About the Exalogic Network Configuration for the SOA Enterprise Topology](#)
- [Configuring Virtual IP Addresses for IPoIB on Each Compute Node](#)
- [Configuring Virtual IP Addresses for EoIB on Each Compute Node](#)
- [Defining the Required Hostname Resolution](#)
- [Defining the Required Virtual Server Names](#)
- [Configuring the Load Balancer](#)
- [Configuring Firewall Ports](#)

3.1 Overview of Preparing the Network for an Enterprise Deployment

[Table 3–1](#) summarizes the steps required to set up the network for an Enterprise Deployment on the Exalogic compute node.

This table will be revised according to the final flow of the chapter

Table 3–1 Overview of the Network Configuration Process for an Exalogic Enterprise Deployment

Task	Description	More Information
Understand the Exalogic network configuration required for the SOA enterprise topology	It is important to make sure that you have the required IPoIB and EoIB interfaces and other details before you configure the network for an enterprise deployment.	Section 3.2, "About the Exalogic Network Configuration for the SOA Enterprise Topology"
Configure the required virtual IP addresses for IPoIB	The SOA Exalogic enterprise deployment requires that specific virtual IP addresses used for IPoIB communication.	Section 3.3, "Configuring Virtual IP Addresses for IPoIB on Each Compute Node"
Configure the required virtual IP addresses for EoIB	The SOA Exalogic enterprise deployment requires that specific virtual IP addresses to be used for EoIB communication.	Section 3.4, "Configuring Virtual IP Addresses for EoIB on Each Compute Node"

Table 3–1 (Cont.) Overview of the Network Configuration Process for an Exalogic Enterprise Deployment

Task	Description	More Information
Define the required hostname resolution and virtual server names	The SOA Exalogic enterprise deployment requires hostname resolution to topologies that can sustain network changes, system relocation and disaster recovery scenarios.	Section 3.5, "Defining the Required Hostname Resolution"
Defining the required virtual server names	Ensure that the virtual server names are associated with IP addresses and are part of your DNS.	Section 3.6, "Defining the Required Virtual Server Names"
Configure the external hardware load balancer	The external hardware load balancer must be configured to accept requests from both external customers and company administrators and route them to the appropriate URLs in the topology.	Section 3.7, "Configuring the Load Balancer"
Configure the firewall ports	When you install and configure the firewalls for your topology, use this information to open only the required ports and set the proper timeouts for each port.	Section 3.8, "Configuring Firewall Ports"

3.2 About the Exalogic Network Configuration for the SOA Enterprise Topology

The following sections provide information about the Exalogic network configuration for the SOA enterprise topology:

- [Section 3.2.1, "General Characteristics and Goals of the Exalogic Network Configuration"](#)
- [Section 3.2.2, "Map of the Network Interfaces Used by the Components of the SOA Topology on Exalogic"](#)
- [Section 3.2.3, "Explanation of the Network Interfaces Map"](#)

3.2.1 General Characteristics and Goals of the Exalogic Network Configuration

When you initially set up your Exalogic system, the IP over Infiniband (IPoIB) is configured by default. In addition to IPoIB, you must manually configure Ethernet over InfiniBand (EoIB) network access for those components that are going to be exposed over ethernet out of the Exalogic rack.

An optimized Oracle Fusion Middleware system constrains communication between the various elements of the topology so it is performed over the Exalogic Infiniband network as much as possible. For example, components should listen in Infiniband interfaces to eliminate overhead in accessing the appropriate Gateways and to make use of the optimized Inifiband network.

Additionally, when the same Exalogic rack is shared with other Oracle Fusion Middleware systems, such as WebCenter and Fusion Middleware SOA, or even with other type of deployments, such as test or development, then EoIB access might require isolated VLAN connectors for SOA. VLANs can be used for this logical division of workload and for enforcing security isolation. However, the definition of such VLANs is outside the scope of this guide.

3.2.2 Map of the Network Interfaces Used by the Components of the SOA Topology on Exalogic

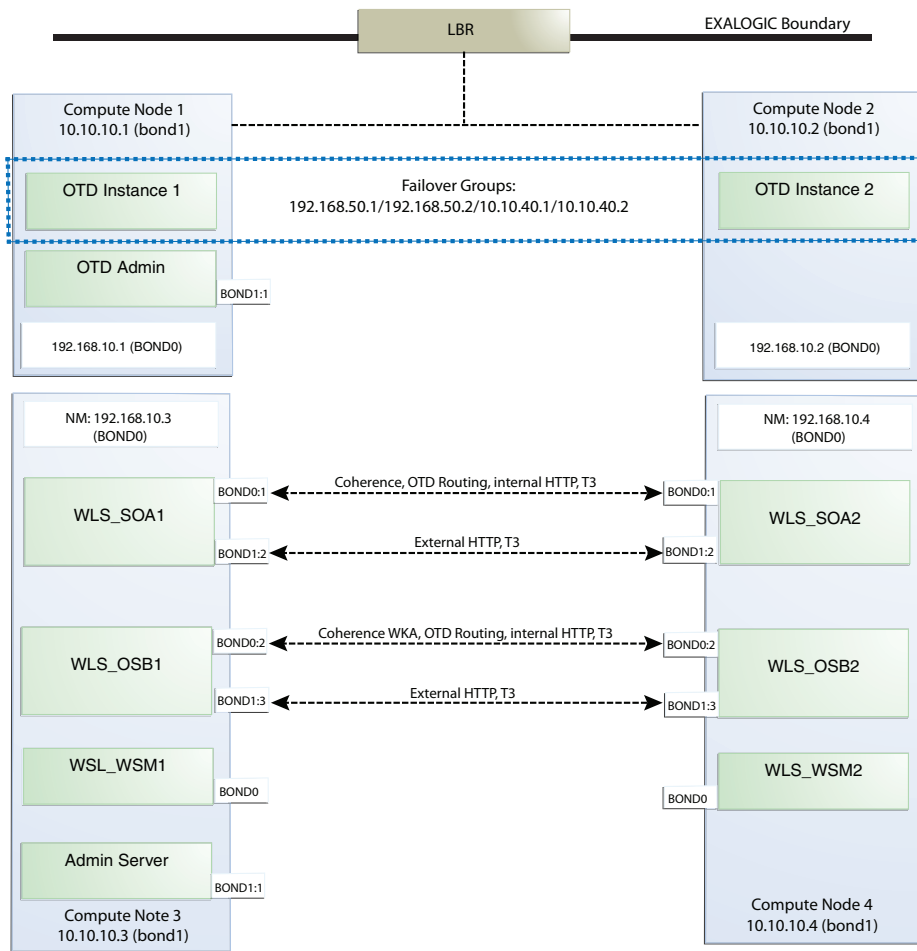
[Figure 3–1](#) describes the components of an Oracle Fusion Middleware SOA Exalogic enterprise deployment on Exalogic, and the type of interfaces and communication protocols they use.

The IP addresses used in [Figure 3–1](#) and the sections in this chapter (both internal IP addresses, such as 192.168.10.1, and external IP addresses, such as 10.10.10.1) are just examples and are used for consistency throughout this document. However, other IPs are valid. It is a good practice to follow an order and separate types of servers in IP ranges. For example, 192.168.50.x for Oracle Traffic Director, and 192.168.20.x for SOA. These ranges depend on the subnet masks they use, but will facilitate IP tracking.

For more information about the network map diagram, see the following:

- For an explanation of the network interfaces and subinterfaces shown in the network map diagram, see [Section 3.2.3, "Explanation of the Network Interfaces Map"](#).
- For a summary of the virtual IP addresses that you must assign to the IPoIB (bond0) interface and subinterfaces, as well as instructions for creating the IPoIB interfaces, see [Section 3.3, "Configuring Virtual IP Addresses for IPoIB on Each Compute Node"](#).
- For a summary of the virtual IP addresses that you must assign to the EoIB (bond1) interface and subinterfaces, as well as instructions for configuring the Exalogic EoIB network and the virtual IP address for EoIB, see [Section 3.4, "Configuring Virtual IP Addresses for EoIB on Each Compute Node"](#).
- For details about the IP addresses associated with Admin, SOA, WSM, and OSB server network interfaces, see [Table 3–4](#)

Figure 3–1 Oracle SOA Exalogic Network Map



3.2.3 Explanation of the Network Interfaces Map

The following sections describe the various network interfaces shown in Figure 3–1:

- Section 3.2.3.1, "Communication with the Oracle Web Services Manager (OWSM) Managed Servers"
- Section 3.2.3.2, "Communication with the SOA Managed Servers"
- Section 3.2.3.3, "Communication with the Oracle Service Bus Managed Servers"
- Section 3.2.3.4, "Communication with the Oracle Traffic Director Instances"
- Section 3.2.3.5, "Communication with the WebLogic Server Administration Server"
- Section 3.2.3.6, "Communications with the WebLogic Server Node Manager and External Database"

3.2.3.1 Communication with the Oracle Web Services Manager (OWSM) Managed Servers

The OWSM Managed Servers are accessed using the default bond0 interface by SOA servers and other consumers that are expected to be internal to the InfiniBand fabric. There is no need for the OWSM servers to be accessed from outside the Exalogic compute node, so there is no need for the OWSM servers to use an EoIB interface.

3.2.3.2 Communication with the SOA Managed Servers

The SOA servers listen on both EoIB and IPoIB interfaces.

Their default channel uses an IPoIB listen address. This is for optimized server-to-server invocations as well as for Coherence dehydration and deployment optimizations.

For more information about Server Migration, see [Chapter 13, "Configure Server Migration for an Exalogic Enterprise Deployment"](#).

For more information about the Virtual IP addresses assigned to each subinterface, see [Chapter 3.3, "Configuring Virtual IP Addresses for IPoIB on Each Compute Node"](#).

In addition to the IPoIB interfaces, the SOA Managed Servers also listen on EoIB virtual IP addresses so that they can be accessed externally for the following purposes:

- RMI/JMX/T3 invocations
- HTTP invocations used for remote deployment from Jdev.
- External JMS producers and consumers.
- Other operations that use the direct listen address of the SOA servers
- They may use separate channels for T3 and HTTP to isolate different types of external traffic.
- SOA servers use server migration, so they are configuring with floating IPs both for EoIB and IPoIB.

3.2.3.3 Communication with the Oracle Service Bus Managed Servers

The Oracle Service Bus servers also listen on both EoIB and IPoIB interfaces.

Their default channel uses an IPoIB listen address. This is for optimized server-to-server invocations as well as for Coherence Result caching.

Like the SOA Managed Servers, the Oracle Service Bus servers take advantage of server migration. As a result, floating IPs are configured for both EoIB and IPoIB.

In addition to the IPoIB interfaces, the Oracle Service Bus Managed Servers also listen on EoIB interfaces, so they can be accessed externally for the following purposes:

- RMI/JMX/T3 invocations
- External JMS producers and consumers.

The Oracle Service Bus servers may use separate channels for T3 and HTTP to isolate different types of traffic.

3.2.3.4 Communication with the Oracle Traffic Director Instances

The Oracle Traffic Director instances are installed on Compute Node 1 and Compute Node 2.

The Oracle Traffic Director configuration is deployed to two instances with listeners listening on ANY/*. For increased isolation, you can associate the Oracle Traffic Director EoIB interfaces (used by the front end load balancer to distributed requests between the two Oracle Traffic Director nodes) to a separate VLAN from the rest of the EoIB addresses. Listeners can be accessed both externally over the EoIB network by the front end load balancer, and internally over the IPoIB network by the application tier components.

For load balancing and high availability, four virtual IPs are mapped to Oracle Traffic Director failover groups: two for external access, one for SOA internal access, and one for OSB internal access.

The Oracle Traffic Director administration server uses an EoIB virtual IP address so it can be failed over to a different node (this failover node cannot host an Oracle Traffic Director administration node already). Note that the OTD administration server virtual IP address is not part of any failover group. To failover the OTD Admin Server, you must perform a backup and restore of the Administration Server instance (or use a shared mount) in a different node that does not contain an OTD Administration node.

3.2.3.5 Communication with the WebLogic Server Administration Server

The WebLogic Server Administration Server can listen on an EoIB or on IPoIB and be exposed to the external world by Oracle Traffic Director. The suitable approach is determined by the type of operations performed on the Administration Server. For example, if JMX management and metrics are accessed externally, as occurs in most cases, EoIB is required. Given the typical lifecycle of a SOA system and the use of the Administration Server by deployment operations from external clients, the default channel for the Administration Server should use an EoIB IP address.

Ideally, different types of traffic, such as management and deployments as opposed to runtime invocations, should be isolated from each other. For this, the Administration Server's and SOA/OSB server's EoIB addresses may be placed on a different VLANs. Whether this is required depends also on the type of management and deployment operations performed in each system. For simplicity, this guide uses a model where the SOA/OSB servers and Admin Server EoIB exist in the same VLAN and partition.

3.2.3.6 Communications with the WebLogic Server Node Manager and External Database

Network interfaces are also used for the following purposes:

- Node manager uses the default IPoIB address assigned to the compute node.
- The Database in this Exalogic enterprise deployment Topology is accessed using EoIB.

3.3 Configuring Virtual IP Addresses for IPoIB on Each Compute Node

This section provides the following sections:

- [Section 3.3.1, "Summary of the Required IPoIB Virtual IP Addresses"](#)
- [Section 3.3.2, "Creating the Virtual IP Addresses for the IPoIB Network on SOAHOST1 and SOAHOST2"](#)
- [Section 3.3.3, "Verifying the Required Virtual IP Addresses on the IPoIB Network"](#)

3.3.1 Summary of the Required IPoIB Virtual IP Addresses

For all communications over the IPoIB network, the WEBHOST compute nodes and WSM managed servers use the default `bond0` interface and the IP address assigned to this interface by default when you set up your Exalogic hardware and software.

However, as described in [Section 3.2.3.2, "Communication with the SOA Managed Servers"](#), the SOA and Oracle Service Bus Managed Servers should be configured to use subinterfaces of the `bond0` network interface.

Table 3–2 lists the Virtual IPs you must define for the SOA and Oracle Service Bus Managed Servers on SOAHOST1 and SOAHOST2.

For instructions on defining these virtual IP addresses, see Section 3.3, "Configuring Virtual IP Addresses for IPoIB on Each Compute Node."

Table 3–2 Virtual IP Addresses Associated with IPoIB Network interfaces

Interface	Address Example	Netmask Example	Used By	Default Host
BOND0:1	192.168.20.3	255.255.240.0	WLS_SOA1 (default channel)	SOAHOST1
BOND0:1	192.168.20.4	255.255.240.0	WLS_SOA2 (default channel)	SOAHOST2
BOND0:2	192.168.40.3	255.255.240.0	WLS_OSB1 (default channel)	SOAHOST1
BOND0:2	192.168.40.4	255.255.240.0	WLS_OSB2 (default channel)	SOAHOST2
BOND0:1	192.168.50.1	255.255.220.0	OTD failover group for SOA ¹	WEBHOST1
BOND0:1	192.168.50.2	255.255.220.0	OTD failover group for SOA ¹	WEBHOST2

¹ These Virtual IP addresses are managed by OTD/VRRP and do not need to be explicitly enabled with `ifconfig`.

3.3.2 Creating the Virtual IP Addresses for the IPoIB Network on SOAHOST1 and SOAHOST2

To enable each IP address listed in Table 3–2 on SOAHOST1 and SOAHOST2:

1. Use the `ifconfig` command to create the virtual IP:

```
ifconfig subinterface virtual_ip_address netmask netmask_value
```

For example, on SOAHOST1, enter the following:

```
ifconfig bond0:1 192.168.20.3 netmask 255.255.240.0
ifconfig bond0:2 192.168.40.3 netmask 255.255.240.0
```

2. For each virtual IP address you define, update the ARP caches using the following command:

```
arping -b -A -c 3 -I bond0 192.168.20.3
```

3.3.3 Verifying the Required Virtual IP Addresses on the IPoIB Network

Once you add the IP addresses, ensure that they are accessible from the Oracle Traffic Director nodes and from the other SOAHOST using the appropriate IPoIB interfaces. For example:

From SOAHOST1, run the following command:

```
ping -I bond0 192.168.20.4
```

From SOAHOST2:

```
ping -I bond0 192.168.20.3
```

3.4 Configuring Virtual IP Addresses for EoIB on Each Compute Node

Refer to the following sections for information about creating the EoIB network and the required virtual IP addresses for the SOA Exalogic enterprise deployment on Exalogic:

- [Section 3.4.1, "Summary of the Virtual IP Addresses for the EoIB Network Interfaces"](#)
- [Section 3.4.2, "Configuring the EoIB Network for the SOA Enterprise Topology"](#)
- [Section 3.4.3, "Creating the EoIB Virtual IPs for the WEBHOST1 and WEBHOST2 Compute Nodes"](#)
- [Section 3.4.4, "Verifying Connectivity Between Virtual IP Addresses"](#)

3.4.1 Summary of the Virtual IP Addresses for the EoIB Network Interfaces

[Table 3–3](#) lists the virtual IP addresses you must associate with each EoIB interface on each compute node. Each of these interfaces is shown in [Figure 3–1](#).

To define these virtual hosts, you must configure your EoIB network on Exalogic, including the Virtual Network Interface Card (vNIC) interfaces for Ethernet on each compute node. For more information, see [Section 3.4.2, "Configuring the EoIB Network for the SOA Enterprise Topology"](#).

Table 3–3 Virtual IP Addresses for the EoIB Network and Associated Network Interfaces

Interface	Address Example	Netmask Example	Used by Server	Host
BOND1:1	10.10.30.1	255.255.220.0	AdminServer (Default Channel)	SOAHOST1
BOND1:2	10.10.20.3	255.255.220.0	WLS_SOA1 (HTTP and T3 channel)	SOAHOST1
BOND1:2	10.10.20.4	255.255.220.0	WLS_SOA2 (HTTP and T3 channel)	SOAHOST2
BOND1:3	10.10.40.3	255.255.220.0	WLS_OSB1 (HTTP and T3 channel)	SOAHOST1
BOND1:3	10.10.40.4	255.255.220.0	WLS_OSB2 (HTTP and T3 channel)	SOAHOST2
BOND1:1	10.10.20.1	255.255.220.0	OTD Admin Server	WEBHOST1
BOND1:2	10.10.40.1	255.255.220.0	OTD External Failover Group ¹	WEBHOST1
BOND1:1	10.10.40.2	255.255.220.0	OSB External Failover Group 1	WEBHOST2

¹ These two EoIB virtual IP addresses are optional and are used for a VLAN separation of the Oracle Traffic Director access points, and also for faster failure detection by using VRRP OTD.

3.4.2 Configuring the EoIB Network for the SOA Enterprise Topology

Information about configuring the Ethernet over Infiniband (EoIB) network on Exalogic is available in the *Oracle Exalogic Elastic Cloud Machine Owner's Guide*. The instructions here are specific to configuring the EoIB network for the SOA Exalogic enterprise deployment.

Create the appropriate VNIC and VLAN associations, as well as EoIB Virtual IPs on SOAHOST1 and SOAHOST2.

Note: For VLAN separation of the Oracle Traffic Director access addresses, the appropriate VNICS and EoIB virtual IP addresses must be configured as access points for the Oracle Traffic Director listeners on WEBHOST1 and WEBHOST2. In addition, a separate VLAN must be created for them.

Use an SSH client, such as PuTTY, to log in to a Sun Network QDR InfiniBand Gateway Switch. Oracle recommends signing in as `ilom-admin` and running the commands through the `/SYS/Fabric_Mgmt` Linux shell target of the Oracle ILOM CLI.

To create the VNICS and VLAN associations:

1. Log in to `e101gw04` as `ilom-admin`.
2. At the command prompt, run the following:

```
listlinkup | grep Bridge
```

Example output:

```
Bridge-0 Port 0A-ETH-1 (Bridge-0-2) up (Enabled)
Bridge-0 Port 0A-ETH-2 (Bridge-0-2) down (Enabled)
Bridge-0 Port 0A-ETH-3 (Bridge-0-1) down (Enabled)
Bridge-0 Port 0A-ETH-4 (Bridge-0-1) down (Enabled)
Bridge-1 Port 1A-ETH-1 (Bridge-1-2) down (Enabled)
Bridge-1 Port 1A-ETH-2 (Bridge-1-2) down (Enabled)
Bridge-1 Port 1A-ETH-3 (Bridge-1-1) down (Enabled)
Bridge-1 Port 1A-ETH-4 (Bridge-1-1) down (Enabled)
```

From this example, identify the uplinks in the gateway to determine which of the ethernet connectors to use for creating a VNIC. For this example, the uplink would be `0A-ETH-1`.

3. Determine GUIDs of the Exalogic compute node that requires the VNIC as follows:
 - a. On the compute node that requires the VNIC, log in as `root`, and run the `ibstat` command. For example, to log in to `e101cn01` as `root`:

```
e101cn01# ibstat
CA 'mlx4_0'
  CA type: MT26428
  Number of ports: 2
  Firmware version: 2.7.8100
  Hardware version: b0
  Node GUID: 0x0021280001a0a364
  System image GUID: 0x0021280001a0a367
  Port 1:
    State: Active
    Physical state: LinkUp
    Rate: 40
    Base lid: 120
    LMC: 0
    SM lid: 6
    Capability mask: 0x02510868
    Port GUID: 0x0021280001a0a365
    Link layer: IB
```

```

Port 2:
State: Active
Physical state: LinkUp
Rate: 40
Base lid: 121
LMC: 0
SM lid: 6
Capability mask: 0x02510868
Port GUID: 0x0021280001a0a366
Link layer: IB

```

The output contains information about two ports. Identify the GUID and Base lid of the port that you want to use for creating the VNIC.

The example illustrated in this procedure uses the port with GUID 0x0021280001a0a366 and Base lid 121.

- b. On the same compute node, run the following command to view information about the active links in the InfiniBand fabric:

```
hostname# iblinkinfo.pl -R | grep hostname
```

hostname is the name of the compute node. You can also specify the bonded IPoIB address of the compute node.

For example:

```

el01cn01# iblinkinfo.pl -R | grep el01cn01
65  15[ ] == ( 4X 10.0 Gbps Active/ LinkUp)==>  121
2[ ] "el01cn01 EL-C 192.168.10.3 HCA-1" (Could be 5.0 Gbps)
64  15[ ] == ( 4X 10.0 Gbps Active/ LinkUp)==>  120
1[ ] "el01cn01 EL-C 192.168.10.3 HCA-1" (Could be 5.0 Gbps)

```

From the output of the `iblinkinfo` command, note the switch lid value (65, in first column) associated with the Base lid of the compute node port that you noted earlier (121, in the first line).

4. Determine the gateway switch that corresponds to the switch lid 65 by running the `ibswitches` command:

```

el01cn01# ibswitches
Switch  : 0x002128548042c0a0 ports 36 "SUN IB QDR GW switch el01gw03" enhanced
port 0 lid 63 lmc 0
Switch  : 0x002128547f22c0a0 ports 36 "SUN IB QDR GW switch el01gw02" enhanced
port 0 lid 6 lmc 0
Switch  : 0x00212856d0a2c0a0 ports 36 "SUN IB QDR GW switch el01gw04" enhanced
port 0 lid 65 lmc 0
Switch  : 0x00212856d162c0a0 ports 36 "SUN IB QDR GW switch el01gw05" enhanced
port 0 lid 64 lmc 0

```

lid 65 corresponds to gateway switch `el01gw04` with GUID `0x00212856d0a2c0a0`.

5. Define a dummy MAC address in the following format:

```

last3_octets_of_switchGUID :
last3_octets_of_computenode_adminIP_in_hex_format

```

For example:

- GUID of the switch: 00:21:28:56:d0:a2:c0:a0
- Last three octets: a2:c0:a0

- Administrative IP of the compute node that requires the VNIC: 192.168.10.3 (for SOAHOST1)
- Last three octets: 168.10.3 (in hexadecimal notation: a8:0A:03)
- MAC address: a2:c0:a0:a8:0A:03

Note: The dummy MAC address should be unique to the Exalogic network. Only even numbers are supported for the most significant type of the MAC address (unicast). The above address is an example only.

6. Log in as `ilom-admin` to the gateway switch (`e101gw04`) that you identified in Step 4.

7. Run the following command to associate a connector with the VLAN that will be used:

```
gwhostname# createvlan connector -vlan 0 -pkey default
```

For example:

```
e101gw04# createvlan 0A-ETH-1 -vlan 0 -pkey default
```

8. Run the following command to create a VNIC:

```
gwhostname# createvnic connector -guid compute_node_port_GUID -mac unique_mac_address -pkey default
```

For example:

```
e101gw04# createvnic 1A-ETH-3 -guid 0021280001a0a366 -mac a2:c0:a0:a8:0A:03 -pkey default -vlan 0
```

The VNIC is created.

9. To verify the VNIC, on the switch CLI, run the `showvnics` command. Grep for the hostname and verify that status is UP:

```
e101gw04# showvnics | grep e101cn01
94 UP          N 0021280001EFA4BF          e101cn01EL-C 192.168.10.3
0000 24:C0:A0:85:2F:2E 0 ffff    1A-ETH-3
```

10. On the compute node, run the following command to display the list of VNICs available on the compute node:

```
e101cn01# mlx4_vnic_info -l
```

This command displays the name of the new interface, as seen on the compute node, such as `eth4`. Note this ID.

11. Create another VNIC for the same compute node, but using a connector on a different gateway switch. Note the `ethX` ID of this VNIC too.

It is recommended that you configure the two EoIB interfaces as a bonded interface, such as `bond1`.

12. Create interface files for the VNICs on the compute node.

To ensure correct failover behavior, the name of the VNIC interface file and the value of the `DEVICE` directive in the interface file must not be based on the kernel-assigned `ethX` interface name (`eth4`, `eth5`, and so on). Instead, Oracle

recommends that the interface file name and value of the `DEVICE` directive in the interface file be derived from the `EPORT_ID` and `IOA_PORT` values:

Note: Any other unique naming scheme is also acceptable.

- a.** Run the following command to find the `EPORT_ID`:

```
#mlx4_vnic_info -i ethX | grep EPORT_ID
```

For example:

```
e101cn01#mlx4_vnic_info -i eth4 | grep EPORT_ID EPORT_ID 331
```

Note the `EPORT_ID` that is displayed, 331 in this example.

- b.** Run the following command to find the `IOA_PORT`:

```
#mlx4_vnic_info -i ethX | grep IOA_PORT
```

For example:

```
e101cn01#mlx4_vnic_info -i eth4 | grep IOA_PORT
IOA_PORT      mlx4_0:1
```

Note the number after the colon (:) is the `IOA_PORT` value that is displayed, in this case 1.

- c.** Build the interface file name and device name using the following convention:

Interface file name: `ifcfg-ethA_B`

Device name: `ethA_B`

A is the `EPORT_ID`, and B is the number after the colon (:) in the `IOA_PORT` value.

For example:

Interface file name: `ifcfg-eth331_1`

Device name: `eth331_1`

In this example, 331 is the `EPORT_ID`, and 1 is the value derived from the `IOA_PORT`.

- d.** Create the interface file for the first VNIC, `eth4` in the example, by using a text editor, such as `VI`, and save the file in the following directory:

```
/etc/sysconfig/network-scripts
```

- e.** Save the file Save the file in the following directory:

```
/etc/sysconfig/network-scripts
```

For example:

```
# more /etc/sysconfig/network-scripts/ifcfg-eth331_1
DEVICE=eth331_1
BOOTPROTO=None
ONBOOT=yes
HWADDR= a2:c0:a0:a8:0A:03
MASTER=bond1
SLAVE=yes
```

Notes:

- Make sure that the name of the interface file (`ifcfg-eth331_1` in the example) is the name derived in step 12.
- For the `DEVICE` directive, specify the device name (`eth331_1` in the example) derived in step 12.
- For the `HWADDR` directive, specify the dummy MAC address created in step 5.

- f. Create an interface file for the second VNIC, for example, `eth5`.

Be sure to name the interface file and specify the `DEVICE` directive by using a derived interface name and not the kernel-assigned name, as described earlier. In addition, be sure to specify the relevant dummy MAC address for the `HWADDR` directive.

- g. After creating the interface files, create the `ifcfg-bond1` file. If the file already exists, verify its contents

For example:

```
# more /etc/sysconfig/network-scripts/ifcfg-bond1
DEVICE=bond1
IPADDR=192.168.48.128
NETMASK=255.255.240.0
BOOTPROTO=none
USERCTL=no
TYPE=Ethernet
ONBOOT=yes
IPV6INIT=no
BONDING_OPTS="mode=active-backup miimon=100 downdelay=5000 updelay=5000"
GATEWAY=192.168.48.1
```

13. Bring up the new `bond1` interface using the `ifup` command.

Reboot the compute node for the changes to take effect.

14. Repeat these steps for the required VNICs for `SOAHOST2`, `WEBHOST1` and `WEBHOST2`.

3.4.3 Creating the EoIB Virtual IPs for the WEBHOST1 and WEBHOST2 Compute Nodes

Review [Table 3-3](#) and then use the instructions in [Section 3.3.2, "Creating the Virtual IP Addresses for the IPoIB Network on SOAHOST1 and SOAHOST2"](#) to create each of the virtual IPs in the table.

Be sure to associate the virtual IP addresses with the subinterfaces (`bond1:n`) listed in the table.

3.4.4 Verifying Connectivity Between Virtual IP Addresses

Verify the connectivity between the different virtual IP addresses: Make sure that all the added virtual IP addresses are accessible externally (specially the load balancer host), and also from the nodes themselves using the `bond1` interface.

Run the following commands from `SOAHOST1` and `SOAHOST2`:

```
ping -I bond1 10.10.30.1
ping -I bond1 10.10.20.3
```

```
ping -I bond1 10.10.20.4
ping -I bond1 10.10.40.3
ping -I bond1 10.10.40.4
```

Run the following commands from WEBHOST1 and WEBHOST2:

```
ping -I bond1 10.10.20.1
ping -I bond1 10.10.40.1
ping -I bond1 10.10.40.2
ping -I bond1 10.10.40.3
ping -I bond1 10.10.40.4
ping -I bond1 lbr_address
```

Note: The ranges 10.10.20.X, 10.10.30.x, 10.10.40.X are used to facilitate scale up or out assignments, as shown in the NetMask example. As long as virtual IPs are reachable to each other and in the same subnet, other values may be used.

3.5 Defining the Required Hostname Resolution

Appropriate hostname resolution is critical to topology designs that can sustain network changes, system relocation and disaster recovery scenarios. It is important that the required DNS (either `/etc/hosts` or central DNS server) definitions are in place and that WebLogic Servers use hostnames and virtual hostnames instead of using IPs and virtual IPs directly. Additionally, the Exalogic enterprise deployment requires a set of virtual server names for routing requests to the proper server or service within the topology through the external load balancer and the Oracle Traffic Director servers.

These virtual server names must be enabled in the corporate network. IPoIB addresses must be resolved only inside the rack's name resolution system. If multiple racks are going to be connected, to elude possible IP conflict, it is good practice to place these also in a central DNS server. Network administrators at the corporate level should enable this. Alternatively hostnames may be resolved through appropriate `/etc/hosts` file propagated through the different nodes. [Table 3–4](#) provides an example of names for the different floating IP addresses used by servers in the SOA system.

In [Table 3–4](#), virtual host names that include the suffix, "PRIV-Vn," are those mapped to IPoIB virtual IP addresses that are routed to network interfaces on the internal, Infiniband fabric. Those without the "-PRIV-Vn" suffix are mapped to EoIB virtual IP addresses that are routed to network interfaces on the external EoIB network.

Table 3–4 Hostname and Virtual IP information

Hostname Example for This Guide	IP Example and Interface	Type	Host	Bound By	Details
ADMINVHN	10.10.30.1/bond1:1	EoIB /Floating	ComputeNode3/ SOAHOST1	Administration Server	A floating IP address for the Administration Server is recommended, if you want to manually migrate the Administration Server from ComputeNode3 to ComputeNode4.

Table 3–4 (Cont.) Hostname and Virtual IP information

Hostname Example for This Guide	IP Example and Interface	Type	Host	Bound By	Details
OTDADMINV HN	10.10.20.1/bond1:1	EoIb /Floating	ComputeNode1/ WEBHOST1	OTD Administration Server	A floating IP address for the Administration Server is recommended, if you want to manually migrate the OTD Administration Server from ComputeNode1 to ComputeNode2.
WEBHOST1-PR IV	192.168.10.1/bond0	IPoIB/ Fixed	ComputeNode1/ WEBHOST1	NA	
WEBHOST2-PR IV	192.168.10.2/bond0	IPoIB/ Fixed	ComputeNode2/ WEBHOST2	NA	
WEBHOST1-PR IV-V1	192.168.50.1/bond0	IPoIB/Floa ting	ComputeNode1/ WEBHOST1	OTD SOA Internal Failover group	The IP for this VHN is managed by OTD. It is used for IPoIB routing to the WLS_SOAn servers
WEBHOST2-PR IV-V2	192.168.50.2/bond0	IPoIB/Floa ting	ComputeNode2/ WEBHOST2	OTD OSB Internal Failover group	The IP for this VHN is managed by OTD. It is used for IPoIB routing to the WLS_OSBn servers
WEBHOST1VH N1	10.10.40.1/bond1	EoIB/Floa ting	ComputeNode1/ WEBHOST1	OTD SOA/OSB External Routing Failover group	The IP for this VHN is managed by OTD. It is used as external-facing EoIB. This is the VHN that the front end load balancer adds to its pool
WEBHOST2VH N1	10.10.40.2/bond1	EoIB/Floa ting	ComputeNode2/ WEBHOST2	OTD SOA/OSB External Routing Failover group	The IP for this VHN is managed by OTD. It is used as external-facing EoIB. This is the VHN that the front end load balancer adds to its pool
SOAHOST1-PR IV	192.168.10.3/bond0	IPoIB/ Fixed	ComputeNode3/ SOAHOST1	Node Manager and WLS_WSM1	BOND0 IP used by Node Manager and WSM1 running on ComputeNode3.
SOAHOST2-PR IV	192.168.10.4/bond0	IPoIB/ Fixed	ComputeNode4/ SOAHOST2	Node Manager and WLS_WSM2	BOND0 IP used by the Node Manager and WSM2 running on ComputeNode4.
WEBHOST1-PR IV-V1	192.168.50.1/bond0: 1	IPoIB/ Floating	ComputeNode1/ WEBHOST1	OTD Instance 1 Failover Group	Initially enabled in ComputeNode1 can be failed over by OTD to ComputeNode2.
WEBHOST2-PR IV-V1	192.168.50.2/bond0: 1	IPoIB/ Floating	ComputeNode2/ WEBHOST2	OTD Instance 2 Failover Group	Initially enabled in ComputeNode1 can be failed over by OTD to ComputeNode2.

Table 3–4 (Cont.) Hostname and Virtual IP information

Hostname Example for This Guide	IP Example and Interface	Type	Host	Bound By	Details
SOAHOST1-PR IV-V1	192.168.20.3/bond0:1	IPoIB/ Floating	ComputeNode3/ SOAHOST1	WLS_SOA1 default channel	Initially enabled in ComputeNode3 can be failed over by server migration to ComputeNode4.
SOAHOST2-PR IV-V1	192.168.20.4/bond0:1	IPoIB/ Floating	ComputeNode4/ SOAHOST2	WLS_SOA2 default channel	Initially enabled in ComputeNode4 can be failed over by server migration to ComputeNode3.
SOAHOST1-PR IV-V2	192.168.40.3/bond0:3	IPoIB/ Floating	ComputeNode3/ SOAHOST1	WLS_OSB1 DEFAULT CHANNEL	Initially enabled in ComputeNode3 can be failed over by server migration to ComputeNode4.
SOAHOST2-PR IV-V2	192.168.40.4/bond0:3	IPoIB/ Floating	ComputeNode4/ SOAHOST2	WLS_OSB2 DEFAULT CHANNEL	Initially enabled in ComputeNode4 can be failed over by server migration to ComputeNode3.
SOAHOST1VH N1	10.10.20.3/bond1:2	EoIB/ Floating	ComputeNode3/ SOAHOST1	WLS_SOA1 External HTTP/T3 channel	Initially enabled in ComputeNode3 can be failed over by server migration to ComputeNode4.
SOAHOST2VH N1	10.10.20.4/bond1:2	EoIB/ Floating	ComputeNode4/ SOAHOST2	WLS_SOA2 External HTTP/T3 channel	Initially enabled in ComputeNode4 can be failed over by server migration to ComputeNode3.
SOAHOST1VH N2	10.10.40.3/bond1:3	EoIB/ Floating	ComputeNode3/ SOAHOST1	WLS_OSB1 External HTTP/T3 channel	Initially enabled in ComputeNode3 can be failed over by server migration to ComputeNode4.
SOSHST2VH N2	10.10.40.4/bond1:3	EoIB/ Floating	ComputeNode4/ SOAHOST2	WLS_OSB2 External HTTP/T3 channel	Initially enabled in ComputeNode4 can be failed over by server migration to ComputeNode3.

3.6 Defining the Required Virtual Server Names

The SOA enterprise topology uses the following virtual server names:

- [soa.mycompany.com](#)
- [admin.mycompany.com](#)
- [osb.mycompany.com](#)
- [soainternal.mycompany.com](#)

Ensure that the virtual server names are associated with IP addresses and are part of your DNS. The nodes running Oracle Fusion Middleware must be able to resolve these virtual server names.

You will define the virtual server names on the load balancer using the procedure in [Section 3.7, "Configuring the Load Balancer."](#)

Note: The virtual server names here pertain to real addresses that are available in the corporate network (and some also externally to the internet). Although these virtual servers use the same name, they are different entities from the virtual servers defined in Oracle Traffic Director. The virtual server names described here map to real IPs. The virtual server names used in OTD are just management entities defined in OTD for appropriate routing.

3.6.1 soa.mycompany.com

This virtual server name acts as the access point for all HTTP traffic to the runtime SOA components, such as soa-infra, and Workflow. Redirection of non-SSL/HTTP traffic to SSL/HTTPS is configured. Clients access this service using the address `soa.mycompany.com:443`.

3.6.2 admin.mycompany.com

This virtual server name acts as the access point for all internal HTTP traffic that is directed to administration services such as WebLogic Administration Server Console and Oracle Enterprise Manager.

The incoming traffic from clients is not SSL-enabled. Clients access this service using the address `admin.mycompany.com:80` and the requests are forwarded to port 7777 on WEBHOST1 and WEBHOST2.

3.6.3 osb.mycompany.com

This virtual server name acts as the access point for all HTTP traffic to the runtime Oracle Service Bus resources and proxy services. Redirection of non-SSL/HTTP traffic to SSL/HTTPS is configured. Clients access this service using the address `osb.mycompany.com:443`.

3.6.4 soainternal.mycompany.com

This virtual server name is used for internal invocations of SOA services. This URL is not exposed to the internet and is only accessible from the intranet. For SOA systems, users can set this while modeling composites or at runtime with the appropriate EM/MBeans, as the URL to be used for internal services invocations. Invocations like callbacks and internal WebServices, however, use an address in OTD for optimized performance on infiniband. For more information see [Section 7.7, "Defining Oracle Traffic Director Virtual Servers for an Exalogic Enterprise Deployment."](#)

The incoming traffic from clients is not SSL-enabled. Clients access this service using the address `soainternal.mycompany.com:80` which is enabled as Oracle Traffic Director Failover Groups.

3.7 Configuring the Load Balancer

Several virtual servers and associated ports must be configured on the load balancer for different types of network traffic and monitoring. These should be configured to the appropriate real hosts and ports for the services running. Also, the load balancer should be configured to monitor the real host and ports for availability so that the traffic to these 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.

This section contains the following topics:

- [Section 3.7.1, "Load Balancer Requirements"](#)
- [Section 3.7.2, "Load Balancer Configuration Procedures"](#)
- [Section 3.7.3, "Load Balancer Configuration Details"](#)

3.7.1 Load Balancer Requirements

The enterprise topologies use an external load balancer. The external load balancer must 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.
- Monitoring of ports (HTTP and HTTPS).
- The ability to configure virtual server names and ports on your external load balancer. 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 WebLogic Clusters, the load balancer must 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.
- Resource monitoring / port monitoring / process failure detection: The load balancer must be able to detect URL, service, and node failures (through notification or some other means) and to stop directing non-Oracle Net traffic to the failed node. If your external load balancer has the ability to automatically detect failures, you should use it.
- Fault tolerant mode: Oracle highly recommends configuring the load balancer to be in fault-tolerant mode.
- Oracle highly recommends configuring the load balancer virtual server to return immediately to the calling client when the back-end 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 compute node.

- SSL acceleration (this feature is recommended, but not required).
- Configure the virtual server(s) in the load balancer for the directory tier with a high value for the connection timeout for TCP connections. This value should be more than the maximum expected time over which no traffic is expected between Oracle Access Management Access Manager and the directory tier.
- Ability to preserve the Client IP Addresses: The load balancer must have the capability to insert the original client IP address of a request in an X-Forwarded-For HTTP header to preserve the Client IP Address.

3.7.2 Load Balancer Configuration Procedures

The procedures for configuring a load balancer differ, depending on the specific type of load balancer. Refer to the vendor supplied documentation for actual steps. The following steps outline the general configuration flow:

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 you create a pool of servers which 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 a Virtual Server on the load balancer. This is the address and port that receives requests used by the application. For example, to load balance Web Tier requests you would create a virtual host for `https://soa.mycompany.com:443`.
4. 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.
5. Configure SSL Termination, if applicable, for the virtual server.
6. Assign the Pool of servers created in Step 1 to the virtual server.

3.7.3 Load Balancer Configuration Details

For an Oracle SOA deployment, configure your load balancer as shown in [Table 3-5](#).

Table 3-5 Load Balancer Configuration Details

Virtual Host	Server Pool	Protocol	SSL Termination	External	Other Required Configuration/Comments
ADMIN.mycompany.com:80	WEBHOST1VHN 1.mycompany.com:7777 WEBHOST2VHN 1.mycompany.com:7777	HTTP	No	No	<ul style="list-style-type: none"> ■ Use your internal administration address as the virtual server address (for example, admin.mycompany.com). ■ Specify HTTP as the protocol ■ Enable address and port translation. ■ Enable reset of connections when services and/or nodes are down. ■ Assign the pool created in step 1 to the virtual server.

Table 3–5 (Cont.) Load Balancer Configuration Details

Virtual Host	Server Pool	Protocol	SSL Termination	External	Other Required Configuration/Comments
soa.mycompany.com:443	WEBHOST1VHN1.mycompany.com:7777 WEBHOST2VHN1.mycompany.com:7777	HTTP	No	Yes	<ul style="list-style-type: none"> ■ Use your system's frontend address as the virtual server address (for example, soa.mycompany.com). The frontend address is the externally facing host name used by your system and that will be exposed in the Internet. ■ Use port 80 and port 443. Any request that goes to port 80 (non-ssl protocol) should be redirected to port 443 (ssl protocol). ■ Enable address and port translation. ■ Enable reset of connections when services and/or nodes are down. ■ Assign the pool created in step 1 to the virtual server. ■ Create rules to filter out access to /console and /em on this virtual server.
soainternal.mycompany.com:80	WEBHOST1VHN1.mycompany.com:7777 WEBHOST2VHN1.mycompany.com:7777	HTTP	No	No	<ul style="list-style-type: none"> ■ Use your internal administration address as the virtual server address (for example, soainternal.mycompany.com). This address is not externalized. It is used by OTD. ■ Specify HTTP as the protocol ■ Enable address and port translation. ■ Enable reset of connections when services and/or nodes are down. ■ Assign the pool created in step 1 to the virtual server. ■ Optionally, create rules to filter out access to /console and /em on this virtual server.
osb.mycompany.com:443	WEBHOST1VHN1.mycompany.com:7777 WEBHOST2VHN1.mycompany.com:7777	HTTP	No	Yes	<ul style="list-style-type: none"> ■ Use port 80 and port 443. Any request that goes to port 80 (non-ssl protocol) should be redirected to port 443 (ssl protocol). ■ Enable address and port translation. ■ Enable reset of connections when services and/or nodes are down. ■ Assign the pool created in step 1 to the virtual server.

3.8 Configuring Firewall Ports

Many Oracle Fusion Middleware components and services use ports. As an administrator, you must know the port numbers used by these services, and to ensure that the same port number is not used by two services on a host.

Most port numbers are assigned during installation.

[Table 3–6](#) lists the ports used in the Oracle Exalogic deployment reference topology, including the ports that you must open on the firewalls in the topology.

Firewall notation:

- FW0 refers to the outermost firewall.
- VLAN Partition refers to the partition between the web tier and the application tier.
- FW2 refers to the firewall between the application tier and the data tier.

Table 3–6 Ports Used for the SOA Enterprise Deployment on Exalogic

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 all HTML content and the type of process model used for SOA.
Browser request	FW0	443	HTTPS / Load Balancer	Inbound	Timeout depends on all HTML content and the type of process model used for SOA.
Callbacks and Outbound invocations	VLAN Partition	80	HTTPS / Load Balancer	Outbound	Timeout depends on all HTML content and the type of process model used for SOA.
Callbacks and Outbound invocations	VLAN Partition	443	HTTPS / Load Balancer	Outbound	Timeout depends on all HTML content and the type of process model used for SOA.
Load balancer to OTD	n/a	7777	HTTP	n/a	See Section 3.7, "Configuring the Load Balancer."
SOA Server access HTTP	VLAN Partition	8001 Range: 8000 - 8010	HTTP / WLS_ SOA _n	Inbound	Timeout varies based on the type of process model used for SOA.
SOA Server access RMI/T3	F0/VLAN Partition	8003	RMI/T3/WLS_ SOA _n	Both	Timeout depends on the type of RMI/T3 invocation and also on the time the longest remote invocation operation may take.

Table 3–6 (Cont.) Ports Used for the SOA Enterprise Deployment on Exalogic

Type	Firewall	Port and Port Range	Protocol / Application	Inbound / Outbound	Other Considerations and Timeout Guidelines
Oracle Service Bus Access HTTP	VLAN Partition	8011 Range: 8011-8021	HTTP / WLS_ OSB ⁿ	Inbound	Set the timeout to a short period (5-10 seconds).
Oracle Service Bus Access RMI/T3	F0/VLAN Partition	8003	RMI/T3/WLS_ OSB ⁿ	Both	Timeout depends on the type of RMI/T3 invocation and also on the time the longest remote invocation operation may take.
Administration Console access	VLAN Partition	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 all database content and on the type of process model used for SOA.
Oracle Notification Server (ONS)	FW2	6200	ONS	Both	Required for Gridlink. An ONS server runs on each database server.
Oracle Traffic Director Administration Server Console access	VLAN Partition	8989	HTTP	Both	Tune this timeout based on the type of access to the OTD Admin Console (whether you plan to use the Console from application tier clients or clients external to the application tier)
Oracle Traffic Director Administration Node	n/a	8900	HTTP	n/a	n/a

Configuring Storage for an Exalogic Enterprise Deployment

The file system model described in this guide was chosen for maximum availability, best isolation of components, symmetry in the configuration, and facilitation of backup and disaster recovery. The rest of the guide uses this directory structure and directory terminology. Other directory layouts are possible and supported.

This chapter contains the following topics:

- [Overview of Preparing Storage for an Exalogic Enterprise Deployment](#)
- [Shared Storage Recommendations for Exalogic Enterprise Deployments](#)
- [Directory Variables for an Oracle Fusion Middleware SOA Enterprise Deployment on Exalogic](#)
- [Recommended Directory Locations for an Oracle Exalogic Enterprise Deployment](#)
- [Configuring Exalogic Storage for Oracle SOA](#)

4.1 Overview of Preparing Storage for an Exalogic Enterprise Deployment

Before you begin preparing the storage for your Exalogic enterprise deployment on Exalogic, review the following sections:

- [General Information About the Exalogic Enterprise Deployment File System](#)
- [Specific Information About the Exalogic File System](#)

4.1.1 General Information About the Exalogic Enterprise Deployment File System

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

4.1.2 Specific Information About the Exalogic File System

Each Exalogic compute node has access to the Sun ZFS Storage Appliance. The instructions in this guide assume you will be using the appliance to deploy the enterprise topology on your Exalogic compute node.

This guide assumes you have performed the initial hardware setup and configuration steps, and the Sun ZFS Storage 7320 appliance is running and available for use. For

more information, see "Configuring the Sun ZFS Storage 7320 appliance" in the *Oracle Exalogic Elastic Cloud Machine Owner's Guide*.

4.2 Shared Storage Recommendations for Exalogic Enterprise Deployments

This section contains the following topics:

- [Section 4.2.1, "Shared Storage Recommendations for Binary \(Middleware Home\) Directories"](#)
- [Section 4.2.2, "Shared Storage Recommendations for Domain Configuration Files"](#)
- [Section 4.2.3, "Shared Storage Recommendations for JMS File Stores and Transaction Logs"](#)

4.2.1 Shared Storage Recommendations for Binary (Middleware Home) Directories

The following sections describe guidelines for using shared storage for your Oracle Fusion Middleware home directories:

- [Section 4.2.1.1, "About the Binary \(Middleware Home\) Directories"](#)
- [Section 4.2.1.2, "About Using Redundant Binary \(Middleware Home\) Directories"](#)

4.2.1.1 About the Binary (Middleware Home) Directories

When you install any Oracle Fusion Middleware product, you install the product binaries into a Middleware home. The binary files installed in the Middleware home are read-only and remain unchanged unless the Middleware home is patched or upgraded to a newer version.

In a typical production environment, the Middleware home files are saved in a separate location from the domain configuration files, which you create using the Oracle Fusion Middleware Configuration Wizard.

The Middleware home for an Oracle Fusion Middleware installation contains the binaries for Oracle WebLogic Server, the Oracle Fusion Middleware common files, and any Oracle Fusion Middleware product-specific directories.

For more information about the structure and content of an Oracle Fusion Middleware home, see *Oracle Fusion Middleware Concepts*.

4.2.1.2 About Using Redundant Binary (Middleware Home) Directories

For maximum availability, Oracle recommends using redundant binary installations on shared storage.

In this model, you install two identical Middleware homes for your Oracle Fusion Middleware software on two different shares. You then mount one of the Middleware homes to one set of servers, and the other Middleware home to the remaining servers. Each Middleware home has the same mount point, so the Middleware home always has the same path, regardless of which Middleware home the server is using.

Should one Middleware home become corrupted or unavailable, only half your servers are affected. For additional protection, Oracle recommends that you disk mirror these shares.

If separate shares are not available on shared storage, Oracle recommends simulating separate shares using different directories within the same share and mounting these to the same mount location on the host side. Although this does not guarantee the

protection that multiple shares provide, it does allow protection from user deletions and individual file corruption.

4.2.2 Shared Storage Recommendations for Domain Configuration Files

The following sections describe guidelines for using shared storage for the Oracle WebLogic Server domain configuration files you create when you configure your Oracle Fusion Middleware products in an Exalogic enterprise deployment:

- [Section 4.2.2.1, "About Oracle WebLogic Server Administration and Managed Server Domain Configuration Files"](#)
- [Section 4.2.2.2, "Shared Storage Requirements for Administration and Managed Server Domain Configuration Files"](#)

4.2.2.1 About Oracle WebLogic Server Administration and Managed Server Domain Configuration Files

When you configure an Oracle Fusion Middleware product, you create or extend an Oracle WebLogic Server domain. Each Oracle WebLogic Server domain consists of a single Administration Server and one or more managed servers.

For more information about Oracle WebLogic Server domains, see *Oracle Fusion Middleware Understanding Domain Configuration for Oracle WebLogic Server*.

In an Exalogic enterprise deployment, it is important to understand that the managed servers in a domain can be configured for active-active high availability. However, the Administration Server must be active-passive, meaning that if the active instance fails, the other instance takes over.

4.2.2.2 Shared Storage Requirements for Administration and Managed Server Domain Configuration Files

Oracle recommends creating two copies of the domain configuration files:

- One copy is for the Administration Server configuration files.
This is known as the `ASERVER_HOME` directory, and you install this directory on shared storage and mount it exclusively to the host that is running the Administration Server.
In the event of the failure of that host, you can mount the directory on a different host and the Administration Server started on that host.
- The other copy is for the managed server configuration files.
This is known as the `MSERVER_HOME` directory. It resides on "private shared storage" This means that you create a storage partition for each node and mount that storage exclusively to that node.

As a result, the deployment you decide upon should conform to the requirements (if any) of the storage system. Some storage systems offer configuration options to facilitate multiple compute nodes mounting the same shared volume.

4.2.3 Shared Storage Recommendations for JMS File Stores and Transaction Logs

JMS file stores and JTA transaction logs must be placed on shared storage in order to ensure that they are available from multiple hosts for recovery in the case of a server failure or migration.

For more information about saving JMS and JTA information in a file store, see "Using the WebLogic Persistent Store" in *Oracle Fusion Middleware Configuring Server Environments for Oracle WebLogic Server*.

Note: Oracle recommends using different shares for runtime artifacts, such as JMS stores, and the domain configuration. The replication, backup and restore, and lifecycle operations on these two types of data should be different, with more aggressive preservation policies on the JMS and TLOGS artifacts than on the domain configuration.

4.3 Directory Variables for an Oracle Fusion Middleware SOA Enterprise Deployment on Exalogic

This section describes the directory variables used throughout this guide for configuring the Oracle SOA Exalogic enterprise deployment. You are not required to set these as environment variables. [Table 4–1](#) lists and describes directory variables used to identify the directories installed and configured in the guide.

Table 4–1 Directories and Directory Variables

Variable	Description	Example Location
ORACLE_BASE	This environment variable and related directory path refers to the base directory under which all Oracle products are installed.	/u01/oracle
MW_HOME	This variable and related directory path refers to the location where Oracle Fusion Middleware resides. Each MW_HOME has a WL_HOME, an ORACLE_COMMON_HOME and one or more ORACLE_HOME directories.	/u01/oracle/products/fmw
WL_HOME	This variable and related directory path contains installed files necessary to host a WebLogic Server. The WL_HOME directory is a peer of Oracle home directory and resides within the MW_HOME.	MW_HOME/wlserver_10.3
ORACLE_HOME	This variable points to the location where an Oracle Fusion Middleware product, such as Oracle Traffic Director Server or Oracle SOA Suite is installed and the binaries of that product are being used in a current procedure. In this guide, this value might be preceded by a product suite abbreviation, for example: SOA_ORACLE_HOME, OSB_ORACLE_HOME.	SOA_ORACLE_HOME: /u01/oracle/products/fmw/soa OSB_ORACLE_HOME: /u01/oracle/products/fmw/osb
WEB_ORACLE_HOME	The specific Oracle home where the Web tier (Oracle Traffic Director) software binaries have been installed.	/u02/private/oracle/products/web/web
SOA_ORACLE_HOME	The specific Oracle home where the Oracle SOA software binaries have been installed.	/u01/oracle/products/fmw/soa

Table 4–1 (Cont.) Directories and Directory Variables

Variable	Description	Example Location
OSB_ORACLE_HOME	The specific Oracle home where the Oracle Service Bus software binaries have been installed.	/u01/oracle/products/fmw/osb
ORACLE_COMMON_HOME	This variable and related directory path refer to the location where the Oracle Fusion Middleware Common Java Required Files (JRF) Libraries and Oracle Fusion Middleware Enterprise Manager Libraries are installed.	MW_HOME/oracle_common
Domain Directory	This path refers to the file system location where the Oracle WebLogic domain information (configuration artifacts) is stored. Different WebLogic Servers can use different domain directories even when in the same node.	For information, see Section 4.2, "Shared Storage Recommendations for Exalogic Enterprise Deployments."
ORACLE_INSTANCE	An Oracle instance contains one or more system components, such as Oracle Traffic Director. An Oracle instance directory contains updatable files, such as configuration files, log files, and temporary files. In this guide, this value might be preceded by a product suite abbreviation, such as WEB_ORACLE_INSTANCE.	/u02/private/config/instances/web1
JAVA_HOME	This is the location where JDK is installed. NOTE: The examples documented in this guide use JRockit. Any certified version of Java on the operating system used by the Exalogic compute nodes.	MW_HOME/jrockit_version
ASERVER_HOME	This is the primary location of the domain configuration where the Administration server is running. It is installed in the <i>ORACLE_BASE</i> directory on shared storage.	/u01/oracle/config/domains/domain_name
MSERVER_HOME	This is a copy of the domain configuration used to start and stop managed servers. It is installed in the <i>ORACLE_BASE</i> directory on the private storage volume or share.	/u02/private/oracle/config/domains/domain_name
APP_DIR	This is the primary location of the Fusion Middleware applications once they are deployed.	/u02/private/oracle/config/domain_name/applications

4.4 Recommended Directory Locations for an Oracle Exalogic Enterprise Deployment

This section describes the recommended directory structure for an Oracle SOA Exalogic enterprise deployment.

Wherever a shared storage location is directly specified, it is implied that shared storage is required for that directory. The shared storage locations are examples and

can be changed as long as the provided mount points are used. However, Oracle recommends this structure in the shared storage device for consistency and simplicity.

Note: References to the Web Tier directories and to WEBHOST1 and WEBHOST2 are included here to accommodate the topologies that include installing Oracle Traffic Director on the Exalogic compute node.

If you are using remote Oracle HTTP Server instances as your Web tier, then you will be installing the Oracle HTTP Server software and creating the Oracle HTTP Server instances on the local storage for the remote Web Tier host computers, rather than on the Sun ZFS Storage 7320 appliance.

This section includes the following topics:

- [Shared Storage for Oracle SOA Enterprise Deployment on Exalogic](#)
- [Private Storage for an Exalogic Enterprise Deployment](#)

4.4.1 Shared Storage for Oracle SOA Enterprise Deployment on Exalogic

In an Oracle SOA enterprise deployment on Exalogic, it is recommended that the shares shown in [Table 4-2](#) be created on shared Storage.

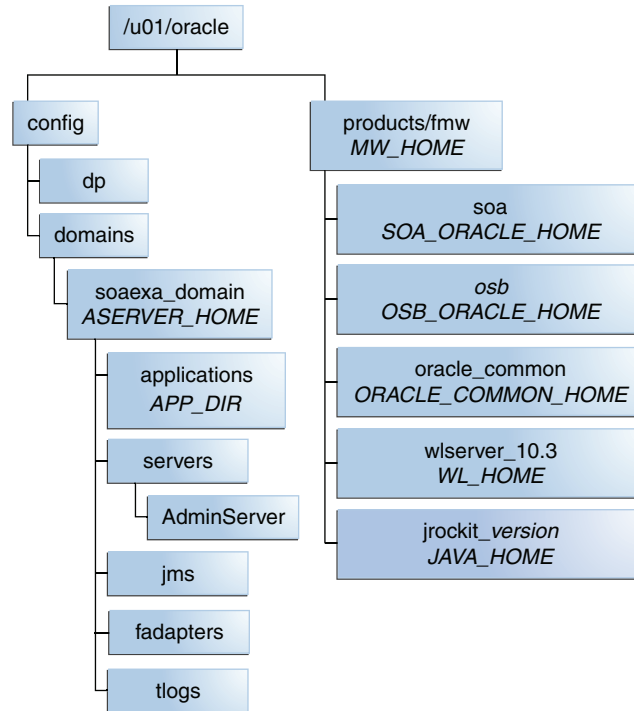
You can mount shared storage either exclusively or shared. If you mount it exclusively, it will be mounted to only one host at a time. (This is typically used for active/passive failover).

When scaling out or scaling up, you can use the shared *MW_HOME* for additional servers of the same type without performing more software installations.

Table 4-2 Shared Storage Directories

Environment Variable	Mount Point	Mounted on Hosts	Exclusive
<i>MW_HOME</i>	/u01/oracle/products/fmw	SOAHOST1, SOAHOST2	No
<i>ORACLE_BASE</i> /config	/u01/oracle/config	SOAHOST1, SOAHOST2	No

Figure 4–1 Shared Storage for an Oracle SOA Exalogic Enterprise Deployment

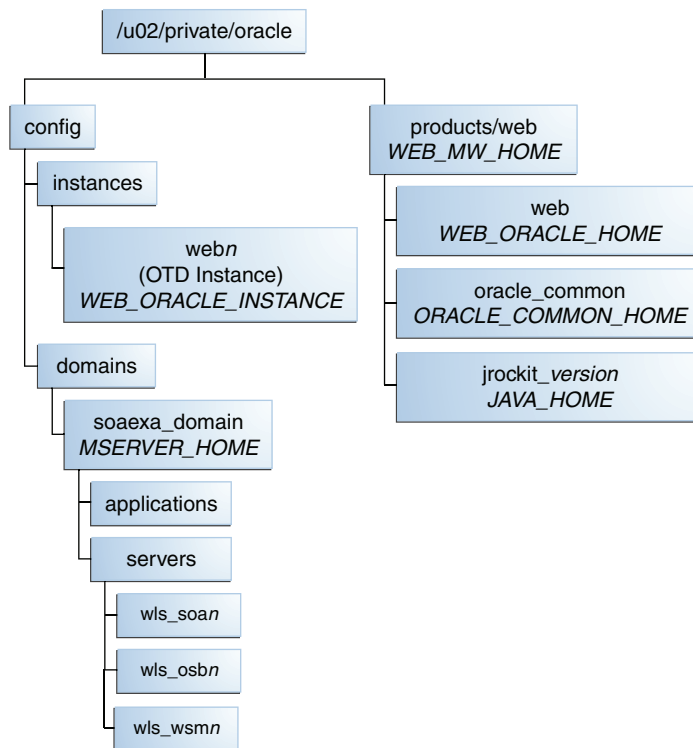


4.4.2 Private Storage for an Exalogic Enterprise Deployment

Table 4–3 shows the recommended directories to be created on private storage for an Exalogic enterprise deployment. These directories are not installed on the local disk of the compute node, but instead the mount points are used to point to a specific share on the ZFS file share for each compute node rather than the local physical disk of the compute node. They are mounted exclusively by each compute node

Table 4–3 Private Storage Directories

Tier	Environment Variable	Directory	Hosts
Web Tier	WEB_MW_HOME	/u02/private/oracle/products/web	WEBHOST1 WEBHOST2
Web Tier	WEB_ORACLE_HOME	/u02/private/oracle/products/web/web	WEBHOST1 WEBHOST2
Web Tier	WEB_ORACLE_INSTANCE	/u02/private/oracle/config/instances/webn	WEBHOST1 WEBHOST2
Web Tier	WEB_ORACLE_ADMININSTANCE	/u02/private/oracle/config/instances/OTDAS	WEBHOST1 WEBHOST2
Application Tier	MSERVER_HOME	/u02/private/oracle/config/domains/domain_name	SOAHOST1 SOAHOST2

Figure 4–2 Private Storage for Oracle SOA Exalogic Enterprise Deployment

While it is recommended that you put ORACLE_INSTANCE directories onto private storage, you can use shared storage.

4.5 Configuring Exalogic Storage for Oracle SOA

The following sections describe how to configure the Sun ZFS Storage 7320 appliance for an Exalogic enterprise deployment:

- [Section 4.5.1, "Summary of the Storage Appliance Directories and Corresponding Mount Points"](#)
- [Section 4.5.2, "Prerequisite Storage Appliance Configuration Tasks"](#)
- [Section 4.5.3, "Creating the SOAEDG Project Using the Storage Appliance Browser User Interface \(BUI\)"](#)
- [Section 4.5.4, "Creating the Shares in the Project Using the BUI"](#)

4.5.1 Summary of the Storage Appliance Directories and Corresponding Mount Points

For the Oracle SOA enterprise topology, you install all software products on the Sun ZFS Storage 7320 appliance, which is a standard hardware storage appliance available with every Exalogic compute node. No software is installed on the private storage available for each compute node.

To organize the Exalogic enterprise deployment software on the appliance, you create a new project, called `soaedg`. The shares (`/products` and `/config`) are created within this project on the appliance, so you can later mount the shares to each compute node.

To separate the product binaries from the files specific to each compute node, you create a separate share for each compute node. Each private directory is identified by the logical host name; for example, `SOAHOST1` and `SOAHOST2`.

Figure 4–3 shows the recommended physical directory structure on the Sun ZFS Storage 7320 appliance.

Table 4–4 shows how the shares on the appliance map to the mount points you will create on the compute nodes that host the Exalogic enterprise deployment software.

Figure 4–3 Physical Structure of the Shares on the Sun ZFS Storage Appliance

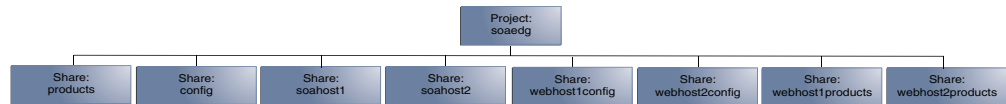


Table 4–4 Mapping the Shares on the Appliance to Mount Points on Each Compute Node

Project	Share	Mount Point	Host	Mounted On
soaedg	products	/export/soaedg/products	SOAHOST1/ SOAHOST2	/u01/oracle/products
soaedg	config	/export/soaedg/config	SOAHOST1/ SOAHOST2	/u01/oracle/config
soaedg	soahost1	/export/soaedg/configsoahost1	SOAHOST1	/u02/private/oracle/config
soaedg	soahost2	/export/soaedg/configsoahost2	SOAHOST2	/u02/private/oracle/config
soaedg	webhost1config	/export/soaedg/configwebhost1	WEBHOST1	/u02/private/oracle/config
soaedg	webhost2config	/export/soaedg/configwebhost2	WEBHOST2	/u02/private/oracle/config
soaedg	webhost1products	/export/soaedg/productswebhost1	WEBHOST1	/u02/private/oracle/products
soaedg	webhost2products	/export/soaedg/productswebhost2	WEBHOST2	/u02/private/oracle/products

4.5.2 Prerequisite Storage Appliance Configuration Tasks

The instructions in this guide assume that the Sun ZFS Storage 7320 appliance is already set up and initially configured. Specifically, it is assumed you have reviewed the following sections in the *Oracle Exalogic Elastic Cloud Machine Owner's Guide*:

- "Prerequisites"
- "Getting Started"
- "Sun ZFS Storage 7320 Appliance Overview"
- "Configuration Overview"
- "Naming Service"

4.5.3 Creating the SOAEDG Project Using the Storage Appliance Browser User Interface (BUI)

To configure the appliance for the recommended directory structure, you create a custom project, called `soaedg`, using the Sun ZFS Storage 7320 appliance Browser User Interface (BUI).

After you set up and configure the Sun ZFS Storage 7320 appliance, the appliance has a set of default projects and shares. For more information, see "Default Storage Configuration" in the *Oracle Exalogic Elastic Cloud Machine Owner's Guide*.

The instructions in this section describe the specific steps for creating a new "soaedg" project for the Exalogic enterprise deployment. For more general information about creating a custom project using the BUI, see "Creating Custom Projects" in the *Oracle Exalogic Elastic Cloud Machine Owner's Guide*.

To create a new custom project called soaedg on the Sun ZFS Storage 7320 appliance:

1. Direct your browser to the storage system BUI, using either the IP address or host name you assigned to the NET0 port as follows:

`https://ipaddress:215`

Or, for example:

`https://elsn01-priv:215`

2. Log in to the BUI using the storage administrator's user name and password. Ideally, root.
3. Access the **Projects** user interface by selecting **Configuration > STORAGE > Shares > Projects**.

The BUI displays the Project Panel.

4. Enter soaedg as the name of the new project.
5. Click the **General** tab on the project page to set project properties.

Update the mount point to meaningful names for better tracking of the shares for this project (such as `/export/soaedg`). For the purposes of the Exalogic enterprise deployment, you can accept the rest of default project properties.

For more information about the properties you can set here, see the "Project Settings" table in the *Oracle Exalogic Elastic Cloud Machine Owner's Guide*.

6. Click **Apply** on the General tab to create the soaedg project.

4.5.4 Creating the Shares in the Project Using the BUI

After you have created the soaedg project, the next step is to create the required shares within the project.

The instructions in this section describe the specific steps for creating the shares required for an Oracle SOA Exalogic enterprise deployment. For more general information about creating custom shares using the BUI, see "Creating Custom Shares" in the *Oracle Exalogic Elastic Cloud Machine Owner's Guide*.

Table 4–5 lists the shares required for all the topologies described in this guide. The table also indicates what privileges are required for each share.

Table 4–5 Shares Required on the Sun ZFS Storage 7320 appliance

Share Name	Privileges to Assign to User, Group, and Other
products	R and W (Read and Write)
config	R and W (Read and Write)
SOAHOST1	R and W (Read and Write)
SOAHOST2	R and W (Read and Write)

Create four additional shares for each of the compute nodes hosting Oracle Traffic Director, as shown in [Table 4-6](#).

Table 4-6 Shares Required When Using Oracle Traffic Director

Share Name	Privileges to Assign to User, Group, and Other
webhost1config	R and W (Read and Write)
webhost2config	R and W (Read and Write)
webhost1products	R and W (Read and Write)
webhost2products	R and W (Read and Write)

To create each share, use the following instructions, replacing the name and privileges, as described in [Table 4-5](#) and [Table 4-6](#):

1. In the Browser User Interface (BUI), access the Projects user interface by clicking **Configuration > STORAGE > Shares > Projects**. The Project Panel is displayed.
2. On the Project Panel, click `soaedg`.
3. Click the plus (+) button next to **Filesystems** to add a file system.
The Create Filesystems screen is displayed.
4. In the Create Filesystems screen, choose `soaedg` from the **Project** pull-down menu.
5. In the **Name** field, enter the name for the share.
Refer to [Table 4-5](#) and [Table 4-6](#) for the name of each share.
6. From the **Data migration source** pull-down menu, choose **None**.
7. Select the **Access** option and set the permissions for each share.
Refer to [Table 4-5](#) and [Table 4-6](#) for the permissions to assign each share.

Note: The required ACL and USER settings for appropriate restrictions on the file system must be defined specially if the mounts are using NFS V4. For details about configuring these permissions, see "Configuring NFS Version 4 (NFSv4) on Exalogic" in the *Oracle Fusion Middleware Exalogic Machine Owner's Guide*.

8. Add the appropriate mount point according to [Table 4-4](#).
9. To enforce UTF-8 encoding for all files and directories in the file system, select the **Reject non UTF-8** option.
10. From the **Case sensitivity** pull-down menu, select **Mixed**.
11. From the **Normalization** pull-down menu, select **None**.
12. Click **Apply** to create the share.
13. Add the appropriate NFS exceptions for each defined share:
 - a. Click **Protocols**.
 - b. Uncheck **inherit from project**.
 - c. Add the required hosts listed in the tables above as NFS exceptions for each share.

- d. In addition, add the required private (IPoIB) addresses for the nodes that will access the storage. these must be added as Type `network` and for the **Entity** field enter the bond0 IP address used by each compute node. For example, `192.168.10.1/32`.

Repeat the procedure for each share listed in [Table 4–5](#) and [Table 4–6](#) (if you are using Oracle Traffic Director).

Note: For information about mounting these shares, see [Section 5.6](#), "Mounting the Shares for WEBHOST1 and WEBHOST2," and [Section 5.7](#), "Mounting the Shares for SOAHOST1 and SOAHOST2."

Configuring the Compute Nodes for an Exalogic Enterprise Deployment

This chapter describes how to prepare the servers for an enterprise deployment.

It contains the following sections:

- [Overview of Preparing the Compute Nodes](#)
- [Meeting Operating System Requirements](#)
- [Synchronize the Node System Clock](#)
- [Enabling Unicode Support](#)
- [Configuring Users and Groups](#)
- [Mounting the Shares for WEBHOST1 and WEBHOST2](#)
- [Mounting the Shares for SOAHOST1 and SOAHOST2](#)

5.1 Overview of Preparing the Compute Nodes

Before you deploy Oracle Fusion Middleware on new hardware, you must set up the compute nodes you plan to use so that the Oracle Software can work in an optional fashion. Specifically, you must ensure that:

- The compute nodes are running a Linux system with the required software patches installed.
- You have configured the UNIX Kernel correctly.
- You have created Users and Groups to own the Oracle software.

The settings described in this chapter are only a guide. After using your Oracle software, you should use operating system utilities to tune the configuration to ensure that you are maximizing the potential of your servers.

5.2 Meeting Operating System Requirements

Before starting your operating provisioning you must install a certified operating system.

Note: Be sure to verify you have obtained all required patches. For more info, see [Section 2.5.3, "Applying Patches and Workarounds."](#)

5.2.1 Meeting UNIX and Linux Requirements

This section includes the following topics:

- [Section 5.2.1.1, "Setting the Open File Limit."](#)
- [Section 5.2.1.2, "Setting Shell Limits."](#)
- [Section 5.2.1.3, "Increase Huge Pages Allocation."](#)
- [Section 5.2.1.4, "Configuring Local Hosts File."](#)

5.2.1.1 Setting the Open File Limit

The minimum Open File Limit is 4096.

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 commands below.

C shell:

```
limit descriptors
```

Bash:

```
ulimit -n
```

5.2.1.2 Setting Shell Limits

To change the shell limits, login as `root` and edit the `/etc/security/limits.conf` file.

Verify that the following values, at a minimum, are met:

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

After editing the file, reboot the compute node.

5.2.1.3 Increase Huge Pages Allocation

By default huge pages are enabled in Exalogic compute nodes, verify the existing allocation by running.

```
grep Huge /proc/meminfo
```

Set the recommended Huge Page allocation to 25000.

To set the Huge Page allocation, run the following command as `root` in the compute node:

```
# echo 25000 > /proc/sys/vm/nr_hugepages
```

5.2.1.4 Configuring Local Hosts File

Before you begin the installation of the Oracle software, ensure that all your local `/ect/hosts` file is formatted like the following:

```
### Compute Node Private Interface details (IPoIB)
192.168.10.1 webhost1-priv.mycompany.com webhost1-priv
```

```

192.168.10.2 webhost2-priv.mycompany.com webhost2-priv
192.168.10.3 soahost1-priv.mycompany.com soahost1-priv
192.168.10.6 soahost2-priv.mycompany.com soahost2-priv

### SOA EDG external VIP (EoIB)
10.242.6.239 otdadmin.mycompany.com otdadmin
10.242.6.240 adminvhn.mycompany.com adminvhn
10.242.6.245 webhost1VHN1.mycompany.com webhost1VHN1
10.242.6.246 webhost2VHN1.mycompany.com webhost2VHN1
10.242.6.241 soahost1VHN1.mycompany.com soahost1VHN1
10.242.6.243 soahost1VHN2.mycompany.com soahost1VHN2
10.242.6.242 soahost2VHN1.mycompany.com soahost2VHN1
10.242.6.244 soahost2VHN2.mycompany.com soahost2VHN2

### SOA EDG private VIP (IPoIB)
192.168.30.11 webhost1-priv-V1.mycompany.com webhost1-priv-V1
192.168.30.21 webhost2-priv-V1.mycompany.com webhost2-priv-V1
192.168.20.11 soahost1-priv-V1.mycompany.com soahost1-priv-V1
192.168.20.12 soahost1-priv-V2.mycompany.com soahost1-priv-V2.
192.168.20.21 soahost2-priv-V1.mycompany.com soahost2-priv-V1
192.168.20.22 soahost2-priv-V2.mycompany.com soahost2-priv-V2

```

Note: If `soainternal.mycompany.com` and `osbinternal.mycompany.com` have DNS entries, you do not need to add to the `/etc/hosts`.

5.3 Synchronize the Node System Clock

Oracle SOA uses Quartz to maintain its jobs and schedules in the database. Synchronize the system clocks for the SOA WebLogic cluster to enable proper functioning of jobs and adapters.

You can use the NTP service to keep clocks in sync. Refer to your network administrator for the appropriate NTP server configuration.

5.4 Enabling Unicode Support

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

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 processing of any character in Unicode. Oracle SOA Suite technologies, for example, are based on Unicode.

5.5 Configuring Users and Groups

Groups

You must create the following groups on each node.

- `oinstall`
- `dba`

Users

You must create the following users on each node.

- oracle–The group that owns 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.

Notes:

- The group oinstall must have write privileges to all the file systems on shared and private 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.
 - The user and group should exist at the NIS server due to the NFSv4 mount requirement.
-

5.6 Mounting the Shares for WEBHOST1 and WEBHOST2

Define storage locations on the Sun ZFS Storage 7320 appliance for WEBHOST1 and WEBHOST2.

Note: This section is based on the assumption that you have set up NFS V4 properly for the corresponding permissions and mounts to work properly. See *Configuring NFS Version 4 (NFSv4) on Exalogic in the Oracle Fusion Middleware Exalogic Machine Owner's Guide* for more information.

To define storage locations:

1. Log in to WEBHOST1 and mount the following shares as the root user.

These directories are used as mount points for the shared and private directories required by the enterprise topology:

```
mkdir -p /u02/private/oracle/products
mkdir -p /u02/private/oracle/config
```

2. Mount the shares:

```
mount -t nfs4 -o rw,bg,hard,nointr,rsize=131072,wsiz=131072,proto=tcp
zfsHost-priv:/export/soaedg/webhost1products /u02/private/oracle/products
```

```
mount -t nfs4 zfsHost-priv:/export/soaedg/webhost1config
/u02/private/oracle/config
```

3. Log in to WEBHOST2 and create the following directories as the root user.

```
sudo root
mkdir -p /u02/private/oracle/products
mkdir -p /u02/private/oracle/config
```

4. Change the ownership of the mount points on WEBHOST1 and WEBHOST2 using the following Commands:

```
chown oracle:oinstall /u02/private/oracle/products
chown oracle:oinstall /u02/private/oracle/config
```

5. Mount the shares:

```

mount -t nfs4 -o rw,bg,hard,nointr,rsize=131072,wsiz=131072,proto=tcp
zfsHost-priv:/export/soaedg/webhost2products
/u02/private/oracle/products

mount -t nfs4 zfsHost-priv:/export/soaedg/webhost2config
/u02/private/oracle/config

```

You can now use these mount points as you install and configure your Oracle Traffic Director software in the directories.

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

```

5.7 Mounting the Shares for SOAHOST1 and SOAHOST2

Define storage locations on the Sun ZFS Storage 7320 appliance for WEBHOST1 and WEBHOST2.

To define storage locations:

1. Log in to SOAHOST1 and mount the following shares as the root user.

These directories are used as mount points for the shared and private directories required by the enterprise topology:

```

mkdir -p /u01/oracle/products
mkdir -p /u01/oracle/config
mkdir -p /u02/private/oracle/config

```

2. Mount the shares:

```

mount -t nfs4 -o rw,bg,hard,nointr,rsize=131072,wsiz=131072,proto=tcp
zfsHost-priv:/export/soaedg/products /u01/oracle/products

mount -t nfs4 -o rw,bg,hard,nointr,rsize=131072,wsiz=131072,proto=tcp
zfsHost-priv:/export/soaedg/config /u01/oracle/config

mount -t nfs4 zfsHost-priv:/export/soaedg/soahost1config
/u02/private/oracle/config

```

3. Log in to SOAHOST2 and create the following directories as the root user.

```

sudo root
mkdir -p /u01/oracle/products
mkdir -p /u01/oracle/config
mkdir -p /u02/private/oracle/config

```

4. Change the ownership of the mount points using the following commands:

```
chown oracle:oinstall /u01/oracle/products
chown oracle:oinstall /u01/oracle/config
chown oracle:oinstall /u02/private/oracle/config
```

5. Mount the shares:

```
mount -t nfs4 -o rw,bg,hard,nointr,rsize=131072,wsiz=131072,proto=tcp
zfsHost-priv:/export/soaedg/products /u01/oracle/products
```

```
mount -t nfs4 -o rw,bg,hard,nointr,rsize=131072,wsiz=131072,proto=tcp
zfsHost-priv:/export/soaedg/config /u01/oracle/products
```

```
mount -t nfs4 -o rw,bg,hard,nointr,rsize=131072,wsiz=131072,proto=tcp
zfsHost-priv:/export/soaedg/soahost2config /u02/private/oracle/config
```

You can now use these mount points as you install and configure your SOA/OSB software in the directories.

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
```

Configuring a Database for an Exalogic Enterprise Deployment

This chapter describes how to configure the Oracle SOA database repositories.

This chapter describes procedures for preparing your database for an Oracle SOA enterprise deployment. The procedures include initial setup of the database, loading the metadata repository, and backing up the database.

This chapter includes the following topics:

- [Section 6.1, "Overview of Preparing the Database for an Enterprise Deployment"](#)
- [Section 6.2, "About Database Requirements"](#)
- [Section 6.3, "Creating Database Services"](#)
- [Section 6.4, "Loading the Oracle Fusion Metadata Repository in the Oracle RAC Database"](#)
- [Section 6.5, "Configuring SOA Schemas for Transactional Recovery Privileges"](#)
- [Section 6.6, "Backing Up the Database"](#)

6.1 Overview of Preparing the Database for an Enterprise Deployment

For the SOA enterprise topology, the database contains the Oracle Fusion Middleware metadata repository, which is a collection of schemas used by various Oracle Fusion Middleware components, such as the SOA components, BAM, and UMS.

You must install the Oracle Fusion Middleware metadata repository before you can configure the Oracle Fusion Middleware components. You install the Oracle Fusion Middleware metadata repository into an existing database using the Repository Creation Utility (RCU). For the enterprise topology, a Real Application Clusters (Oracle RAC) database is highly recommended.

When you configure the SOA components, the configuration wizard will prompt you to enter the information for connecting to the database that contains the metadata repository.

6.2 About Database Requirements

Before loading the metadata repository into your database, check that the database meets the requirements described in these subsections:

- [Section 6.2.1, "Database Host Requirements"](#)
- [Section 6.2.2, "Supported Database Versions"](#)

- [Section 6.2.3, "About Initialization Parameters"](#)

6.2.1 Database Host Requirements

On the hosts SOADBHOST1 and SOADBHOST2 in the data tier, note the following requirements:

- **Oracle Clusterware**
For 11g Release 1 (11.1) for Linux, refer to the *Oracle Clusterware Installation Guide for Linux*.
- **Oracle Real Application Clusters**
For 11g Release 1 (11.1) for Linux, refer to the *Oracle Real Application Clusters Installation Guide for Linux and UNIX*. For 10g Release 2 (10.2) for Linux, refer to *Oracle Database Oracle Clusterware and Oracle Real Application Clusters Installation Guide*.
- **Automatic Storage Management (optional)**
ASM is installed for the node as a whole. Oracle recommends installing it in a separate Oracle Home from the Database Oracle Home. This option appears in the Select Configuration page. Select the Configure Automatic Storage Management option to create a separate ASM home.

6.2.2 Supported Database Versions

Oracle SOA Suite requires the presence of a supported database and schemas:

- To check if your database is certified or to see all certified databases, refer to the "Oracle Fusion Middleware 11g Release 1 (11.1.1.x)" product area on the Oracle Fusion Middleware Supported System Configurations page:
http://www.oracle.com/technology/software/products/ias/files/fusion_certification.html

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%';
```

Notes:

- Oracle SOA requires that the database used to store its metadata (either 10g or 11g) supports the AL32UTF8 character set. Check the database documentation for information on choosing a character set for the database.
 - For Oracle SOA enterprise deployments, Oracle recommends using GridLink data sources to connect to Oracle RAC databases. To use the Oracle Single Client Access Name (SCAN) feature with GridLink, the Oracle RAC database version must be Oracle Database 11gR2 (11.2 or later, Enterprise Edition).
-
-

6.2.3 About Initialization Parameters

Ensure that the following initialization parameter is set to the required minimum value. It is checked by Repository Creation Assistant.

Table 6–1 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.

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

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

Set the initialization parameter using the following command:

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

Restart the database.

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

6.3 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. In addition, the database service should be different from the default database service. For more information about connecting to Oracle databases using services, see "Overview of Connecting to Oracle Database Using Services and VIP Addresses" in the *Oracle Real Application Clusters Administration and Deployment Guide*. For complete instructions on creating and managing database services, see "Introduction to Automatic Workload Management" in the *Oracle 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**. For 10g and 11gR1 databases, use the DBMS_SERVICE package for this modification. For 11g R2 use the srvctl command utility instead.

This section includes the following topics:

- [Section 6.3.1, "Creating Database Services for 10g and 11g Release 1 \(11.1\) Databases"](#)
- [Section 6.3.2, "Creating Database Services for 11g Release 2 \(11.2\) Databases"](#)

6.3.1 Creating Database Services for 10g and 11g Release 1 (11.1) Databases

You can create and modify 10g and 11g database services using the DBMS_SERVICE package.

To create and modify database services:

1. Logon to SQL*Plus and create the service:

```
SQL*Plus "sys/password as sysdba"

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

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

```
soaedg.mycompany.com
```

Note: Enter the EXECUTE DBMS_SERVICE command shown on a single line.

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

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

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

3. Start the service:

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

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

4. Modify the service for the appropriate service goals:

```
SQL>EXECUTE DBMS_SERVICE.MODIFY_SERVICE (service_name =>
'soaedg.mycompany.com',goal => DBMS_SERVICE.GOAL_THROUGHPUT, clb_goal =>DBMS_
SERVICE.CLB_GOAL_SHORT);
```

Or

```
SQL>EXECUTE DBMS_SERVICE.MODIFY_SERVICE (service_name =>
'soaedg.mycompany.com', goal => DBMS_SERVICE.GOAL_SERVICE_TIME, clb_goal
=>DBMS_SERVICE.CLB_GOAL_SHORT);
```

6.3.2 Creating Database Services for 11g Release 2 (11.2) Databases

You can create and modify 11g Release 2 (11.2) database services using the `srvctl` utility.

To create and modify the database services:

1. Logon to SQL*Plus and create the service:

```
sqlplus "sys/password as sysdba"
```

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

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

```
soaedg.mycompany.com
```

Note: Enter the EXECUTE DBMS_SERVICE command shown on a single line.

For more information about the DBMS_SERVICE package, see the *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.mycompany.com -r soadb1,soadb2
```

3. Start the service:

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

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

4. Modify the service for the appropriate service goals:

```
srvctl modify service -d soadb -s soaedg.mycompany.com -B SERVICE_TIME -j SHORT
```

Or

```
srvctl modify service -d soadb -s soaedg.mycompany.com -B THROUGHPUT -j SHORT
```

For more information about the different service definitions, see "Load Balancing Advisory" in the *Oracle Real Application Clusters Administration and Deployment Guide*.

6.4 Loading the Oracle Fusion Metadata Repository in the Oracle RAC Database

The Repository Creation Utility (RCU) is available from the RCU DVD. The RCU used to seed the database must match the patch set level of the Oracle SOA Suite installation. This means that if you install Oracle SOA Suite 11gR1 PS6 (11.1.1.7) in this enterprise deployment, you must use RCU 11gR1 PS6 (11.1.1.7).

To load the Oracle Fusion Middleware metadata repository into a database:

1. Start Repository Creation Utility (RCU), which is available from the RCU DVD by first inserting the RCU DVD.
2. Start RCU from the *bin* directory:
`./rcu`
3. In the Welcome screen, click **Next**.
4. In the Create Repository screen, select **Create** to load component schemas into a database. Click **Next**.
5. In the Database Connection Details screen, enter the correct information for your database:

- a. **Database Type:** select **Oracle Database**.
- b. **Host Name:** Enter the name of the node that is running the database. For the Oracle RAC database, specify the virtual IP name or one of the node names as the host name: CUSTDBHOST1-VIP.

Note: You can use the RAC SCAN address as the host name. For more information about using SCAN addresses, "Using SCAN Addresses with Oracle Database 11g (11.2)" in the *Oracle Fusion Middleware High Availability Guide*.

- c. **Port:** Enter the port number for the database: 1521.
- d. **Service Name:** Enter the service name of the database in lowercase characters. For example:
`soaedg.mycompany.com`
- e. **Username:** SYS
- f. **Password:** Enter the password for the SYS user.
- g. **Role:** SYSDBA

Click **Next**.

6. If you get this warning message: The database you are connecting is with non-UTF8 charset, if you are going to use this database for multilingual support, you may have data loss. If you are not using for multilingual support you can continue, otherwise we strongly recommend using UTF-8 database.

Click **Ignore** or **Stop**.

7. In the Select Components screen, do the following:
 - a. Select **Create a New Prefix**, and enter a prefix to use for the database schemas. Example: DEV or PROD. Prefixes are used to create logical groupings of multiple repositories in a database. For more information, see *Oracle Fusion Middleware Repository Creation Utility User's Guide*.
 - b. Note the name of the schema because you will need to enter it during the procedure in [Section 6.5](#).
 - c. Select the following:
 - AS Common Schemas:
 - **Metadata Services**
 - SOA and BPM Infrastructure:

- SOA Infrastructure
- User Messaging Service

Note: Deselect Business Activity Monitoring (BAM).

Note: Oracle Service Bus required objects are created as part of the SOA_INFRA schema.

Click **Next**.

8. In the Schema Passwords screen, select **Use main schema passwords for auxiliary schemas**. In the subsequent screen refresh, enter the schema passwords for all components.
9. In the Map Tablespaces screen, choose the tablespaces for the selected components, and click **Next**.

A confirmation dialog is displayed stating that any tablespace that does not already exist in the selected schema will be created. Click **OK** to acknowledge this message.
10. In the Summary screen, click **Create**.
11. In the Completion Summary screen, click **Close**.
12. Verify that the required schemas are created by connecting to the database with the new user added:

```
sqlplus PROD_SOAINFRA/password;
```

Query the description of the CUBE_INSTANCE table for a simple verification. A table similar to the following should display:

```
SQL> desc CUBE_INSTANCE;
Name                                                    Null?    Type
-----
CIKEY                                                    NOT NULL NUMBER(38)
CREATION_DATE                                           NOT NULL TIMESTAMP(6)
...
```

6.5 Configuring SOA Schemas for Transactional Recovery Privileges

You need the appropriate database privileges to allow the Oracle WebLogic Server transaction manager to query for transaction state information and issue the appropriate commands, such as commit and rollback, during recovery of in-flight transactions after a WebLogic Server container crash.

These privileges should be granted to the owner of the soainfra schema, as determined by the RCU operations.

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>
```

6.6 Backing Up the Database

Back up the Database configuration. For more information, see [Section 14.8, "Backing Up the Oracle SOA Enterprise Deployment."](#)

Database Growth Management Strategy

An Oracle SOA Suite 11g installation 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 *Oracle Fusion Middleware Administrator's Guide for Oracle SOA and Oracle Business Process Management Suite*.

Installing and Configuring Oracle Traffic Director for an Exalogic Enterprise Deployment

This chapter describes how to install and configure Oracle Traffic Director for an Exalogic enterprise deployment.

This chapter contains the following sections:

- [Overview of Installing and Configuring Oracle Traffic Director for an Exalogic Enterprise Deployment](#)
- [Installing Oracle Traffic Director on WEBHOST1 and WEBHOST2](#)
- [Creating and Starting the Traffic Director Administration Server](#)
- [Register WEBHOST2 as an Administration Node](#)
- [Creating a Configuration](#)
- [Starting the Oracle Traffic Director Instances](#)
- [Defining Oracle Traffic Director Virtual Servers for an Exalogic Enterprise Deployment](#)
- [Deploying the Configuration and Testing the Virtual Server Addresses](#)
- [Creating a Failover Group for Virtual Hosts](#)
- [Backing the Web Tier](#)

7.1 Overview of Installing and Configuring Oracle Traffic Director for an Exalogic Enterprise Deployment

Oracle Traffic Director is a software load balancer for load balancing HTTP/S and TCP traffic to servers in the back-end. These back-end servers, which are referred to as origin servers within Oracle Traffic Director, can be application servers, web servers, or LDAP servers.

[Table 7-1](#) describes the steps for installing and configuring Oracle Traffic Director for an Exalogic enterprise deployment.

Table 7–1 Overview of Installing and Configuring Oracle Traffic Director for an Exalogic Enterprise Deployment

Task	Description	More Information
Review Oracle Traffic Director prerequisites.	For example, be sure that you have set up the required virtual IP addresses, that the user account has root permission on the storage appliance, and that you have already created the initial Oracle WebLogic Server domain for the Oracle SOA topology.	"Prerequisites" in the <i>Oracle Traffic Director Installation Guide</i>
Install the Oracle Traffic Director software on WEBHOST1 and WEBHOST2.	You install the software using the directories and mount points you created in Section 4.5, "Configuring Exalogic Storage for Oracle SOA."	Section 7.2, "Installing Oracle Traffic Director on WEBHOST1 and WEBHOST2"
Create and start an Oracle Traffic Director Administration Server.	The Oracle Traffic Director administration server hosts the administration console and command-line interface, through which you can create Oracle Traffic Director configurations, deploy them as instances on administration nodes, and manage the instances.	Section 7.3, "Creating and Starting the Traffic Director Administration Server"
Verify the installation.	Be sure that the installation was successful before you continue configuring the environment.	"Verifying the Installation" in the <i>Oracle Traffic Director Installation Guide</i>
Register WEBHOST2 as administration node.	This ensures that Oracle Traffic Director is up and running on both WEBHOST1 and WEBHOST2.	Section 7.4, "Register WEBHOST2 as an Administration Node"
Create a configuration	The configuration should route requests from the Oracle Traffic Director instances to the managed servers in the Oracle WebLogic Server domain you created in Chapter 8, "Creating a Domain for an Exalogic Enterprise Deployment." The configuration should also define the required origin-server pools to which requests should be routed.	Section 7.5, "Creating a Configuration"
Start the Oracle Traffic Director instances	Start the instances on WEBHOST1 and WEBHOST2, based on the configuration you created earlier in this procedure.	Section 7.6, "Starting the Oracle Traffic Director Instances"

Table 7–1 (Cont.) Overview of Installing and Configuring Oracle Traffic Director for an Exalogic Enterprise Deployment

Task	Description	More Information
Define the virtual servers.	Define the virtual servers required for accessing the various management tools and login screens for the topology.	Section 7.7, "Defining Oracle Traffic Director Virtual Servers for an Exalogic Enterprise Deployment"
Deploy and test the configuration.	Deploy the configuration and test the virtual server URLs to be sure you have configured the Oracle Traffic Director instances successfully.	Section 7.8, "Deploying the Configuration and Testing the Virtual Server Addresses"
Create an active-passive failover group.	Create a failover group to ensure that requests will continue to be served if WEBHOST1 or WEBHOST2 become unavailable.	Section 7.9, "Creating a Failover Group for Virtual Hosts"

7.2 Installing Oracle Traffic Director on WEBHOST1 and WEBHOST2

This section describes how to install Oracle Traffic Director software.

Note: Be sure that you are not logged in as root user before installing or performing any action on Oracle Traffic Director.

Note: Be sure to verify you have obtained all required patches. For more information, see [Section 2.5.3, "Applying Patches and Workarounds."](#)

To install Oracle Traffic Director:

1. Extract the contents of the installer zip file to a directory on WEBHOST1.
2. Change directory to the `Disk1` subdirectory in the directory in which you unzipped the installer.
3. Run the following command:

```
./runInstaller
```

4. Follow the instructions on the screen to install the software.

When the Specify Installation Location screen appears, enter the value of the `WEB_ORACLE_HOME` variable in the **Oracle Home Directory** field.

The recommended directory location for the `WEB_ORACLE_HOME` is listed in [Table 4–3](#).

If you need help with any of the other options on the installer screens, click **Help**, or refer to "Installing Oracle Traffic Director in Graphical Mode" in the *Oracle Traffic Director Installation Guide*.

Note: You can ignore the errors derived from missing i386 libraries.

5. Repeat steps 1 through 5 on WEBHOST2.

7.3 Creating and Starting the Traffic Director Administration Server

After you install Oracle Traffic Director on WEBHOST1 and WEBHOST2, you can then create an Oracle Traffic Director administration server.

For more information, see "Managing the Administration Server" in the *Oracle Traffic Director Administrator's Guide*

To create the Oracle Traffic Director administration server on WEBHOST1 run the **tadm** command from the `WEB_ORACLE_HOME/bin` directory, as follows:

1. On WEBHOST1 enter the following command:

```
WEB_ORACLE_HOME/bin/tadm configure-server --host=otdadminvhn --port=8989
--user=admin
--instance-home=WEB_ORACLE_ADMININSTANCE
```

```
WEB_ORACLE_ADMININSTANCE
```

Where:

- `WEB_ORACLE_HOME` the Oracle Home location you entered in the Oracle Traffic Director installer.
- `WEB_ORACLE_INSTANCE` is the recommended value listed in [Table 4-3](#).
- `otdadminvhn` is the virtual hostname to be sued for the Oracle Traffic Director administration server and console.

For example:

```
WEB_ORACLE_HOME/web/bin/tadm configure-server
--port=8989 --user=otd_admin
--instance-home=/u01/private/oracle/config/adminHA
--host=scan0309-1.mycompany.com
```

2. Enter the administrator password.

You will later use this password to log in to the Oracle Traffic Director administration console.

A prompt to re-enter the administrator password is displayed, as follows:

```
Please enter admin-user-password again>
```

3. Confirm the administrator password by entering it again.

An Administration Server instance of Oracle Traffic Director is created and deployed on the local host in a directory named `admin-server` within the `WEB_ORACLE_INSTANCE` directory that you specified in step 1.

4. Start the Administration Server by running the following command on WEBHOST1:

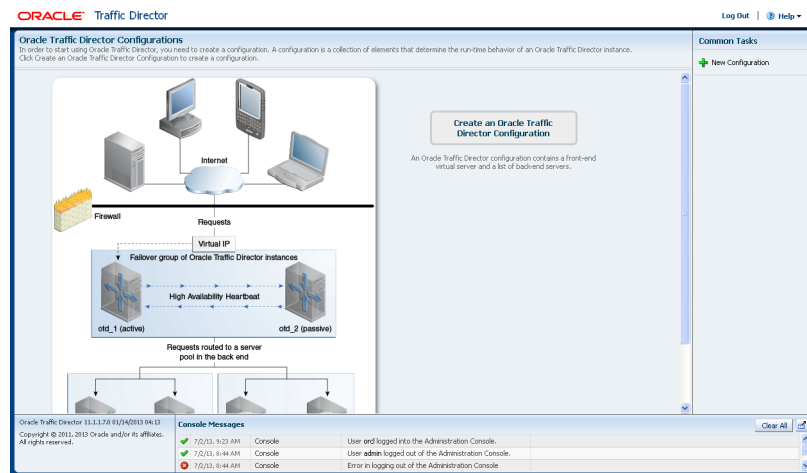
```
WEB_ORACLE_ADMININSTANCE/admin-server/bin/startserv
```

5. Login to the Administration Server using the following URL:

```
https://OTDADMINVHN:8989
```

Use the password provided above and verify that you can see the Oracle Traffic Director main page:

Figure 7–1 Oracle Traffic Director Main Page



7.4 Register WEBHOST2 as an Administration Node

This section assumes you have installed Oracle Traffic Director, started the Administration Server, and verified the installation.

Two Administration Nodes are created using IPoIB addresses in WEBHOST1 and WEBHOST2 (WEBHOST1-PRIV and WEBHOST2-PRIV). See [Table 4–3](#) for the appropriate IP reference.

You can now register both WEBHOST2 with the Oracle Traffic Director administration server using the `tadm` command from the following directory:

```
WEB_ORACLE_HOME/bin
```

Note: WEBHOST1 is registered when you create the administration server.

To register WEBHOST2 with the Oracle Traffic Director administration Server:

1. On the WEBHOST2, run the `configure-server` command to register the host with the remote Administration Server as an administration node.

```
./tadm configure-server --user=admin --port=8989 --host=OTDADMINVHN
--admin-node --node-host=WEBHOST2-PRIV --node-port=8900
--instance-home=WEB_ORACLE_INSTANCE
```

Where:

- `WEB_ORACLE_HOME` is the path to the Oracle Traffic Director Oracle home on WEBHOST2.
- `WEB_ORACLE_INSTANCE` is the recommended directory path listed in [Table 4–3](#), "Private Storage Directories".

For more information, see "configure-server" in the *Oracle Traffic Director Command-Line Reference* or use the `configure-server --help` command to see an explanation of the command line options.

The following prompt appears after you run `configure-server` command:

This command creates an Administration Node and register it with the

following remote Administration Server: `https://OTDADMINVHN.mycompany.com`

Enter admin-user password>

2. Enter the admin-user password for the Oracle Traffic Director Administration Server.

The `configure-server` command attempts to connect to the remote administration server by using the specified administration server host, port, user, and password. The Administration Server on WEBHOST1 must be up and running.

If this is the first time that the host on which you are creating the administration node is attempting to connect to the administration server, the server certificate of the administration server is displayed.

3. Enter `y` to trust the certificate.

The following message is displayed:

```
OTD-70215 The administration node has been configured successfully.
The node can be started by executing:
WEB_ORACLE_INSTANCE/admin-server/bin/startserv
```

After you start the administration node, you can create instances of Oracle Traffic Director configurations on the administration node. Note that on each administration node, you can create only one instance of a configuration.

7.5 Creating a Configuration

The next step in installing and configuring Oracle Traffic Director for an Exalogic enterprise deployment is to create a configuration that will route requests to a server pool that consists of the managed servers in your Oracle WebLogic Server domain.

When creating a new configuration, you are required to provide the host and port information for the origin server, which in turn automatically creates (and names) an origin-server pool called **origin-server-pool-1**. This is the default origin-server pool and this pool can be found when you click the Server Pools option in the administration console. You cannot rename the default origin-server pool.

To create a configuration named SOAEXA by using the administration console:

1. Log in to the administration console using the following URL:

`https://OTDADMINVHN:8989`

2. In the Common Tasks pane, click **New Configuration**.

The New Configuration wizard starts.

Figure 7–2 New Configuration Wizard

The screenshot shows a configuration wizard form with the following elements:

- Name:** A text input field with a red asterisk indicating it is required. Below the field is a note: "Configuration name should not contain spaces, invalid characters or non-ASCII characters."
- Server User:** A text input field. Below the field is a note: "Instances of this configuration run with this UNIX user ID. The user ID should either be root or should belong to the group svrtch."
- Origin Server Type:** A group of radio buttons:
 - HTTP
 - HTTPS (HTTP over SSL)
 - TCP (Example: LDAP, T3, SSL Tunneling)
 Below the radio buttons is a note: "Specifies the type of requests handled by the origin servers."

3. In the Step 1 Configuration Information screen, enter the following information:
 - **Name:** SOAEXA
 - **Server User:** oracle
 - **Origin Server Type:** Make sure **HTTP** is selected.
4. In the Step 2 Listener Information screen, change the port to *7777*. Accept the other default values and click **Next**.
5. In the Step 3 Server Pool Information screen:
 - a. In the **Origin Servers: Host:** field, enter *SOAHOST1-PRIV-V1*, the port *8001*, and click **Add Server**.
 - b. Enter *SOAHOST2-PRIV-V1* and the required port, click **Add Server** and click **Next**.
6. In the Step 4 Deployment Information screen, select **WEBHOST2-PRIV node** and the **Administration Server Node** for deployment.
The Review screen appears.
7. Review the information and click **Create Configuration**.
The Results screen appears.

After the configuration is created, the Results screen of the New Configuration wizard displays a message confirming successful creation of the configuration. If you chose to create instances of the configuration, then a message confirming successful creation of the instances is also displayed.
8. Click **Close** on the Results screen.

In the New Configuration wizard, if you chose not to create an instance of the configuration, the message **Undeployed Configuration** is displayed, indicating that the configuration that you just created is yet to be deployed.

7.6 Starting the Oracle Traffic Director Instances

You can start or restart Oracle Traffic Director instances using the OTD Administration Console

To start Oracle Traffic Director instances:

1. Log in to the administration console using the following URL:
`https://OTDADMINVHN:8989`
2. Click the **Configurations** button that is situated at the upper left corner of the page.
A list of the available configurations is displayed.
3. Select the configuration for which you want to start the instance.
4. In the navigation pane, select **Instances**.
5. Click the **Start/Restart** button for the instance that you want to start.

Note: To start or restart *all* instances of the selected configuration, click **Start/Restart Instances** in the Common Tasks pane. To stop all instances of the configuration, click **Stop Instances**.

7.7 Defining Oracle Traffic Director Virtual Servers for an Exalogic Enterprise Deployment

To create and configure virtual servers using the administration console, you first create the pools used by them and then use the administration console to create the virtual servers and assign these pools in the creation process. You must also update the appropriate host patterns served by each virtual server. Assign the required routes for these virtual servers in each pertaining chapter in the book.

For more information, see "Creating a Virtual Server" in the *Oracle Traffic Director Administrator's Guide*. By default, when the SOAEXA configuration was created, a new virtual server for the external soa.mycompany.com access was created (named soaexa). In this section the following additional virtual servers are created for the Oracle SOA configuration:

- admin.mycompany.com
- soainternal.mycompany.com
- osb.mycompany.com
- osbinternal.mycompany.com

This section contains the following topics:

- [Section 7.7.1, "Creating an Origin-Server Pool"](#)
- [Section 7.7.2, "Creating the Additional Virtual Servers"](#)
- [Section 7.7.3, "Updating the Host Pattern Served by the SOAEXA Virtual Server"](#)

7.7.1 Creating an Origin-Server Pool

To create and configure virtual servers using the administration console complete the following steps:

Create the following origin-server pools using the administration console:

- admin-pool
- soa-pool
- osb-pool

Note: A pool for the soa.mycompany.com virtual servers named **origin-server-pool1** is created when the configuration is created.

To create an origin-server pool by using the administration console, do the following:

1. Log in to the administration console using the following URL:
`https://OTDADMINVHN:8989`
2. Click the **Configurations** button that is situated at the upper left corner of the page.
A list of the available configurations is displayed.
3. Select the **SOAEXA** configuration for which you want to create a virtual server.
4. In the Common Tasks pane, click **New Server Pool**.

The New Origin-Server Pool wizard starts.

Figure 7–3 New Origin-Server Pool Wizard

- In the Step 1: Server Pool Information screen, create the following Origin Server Pools using the information in [Table 7–2](#) and clicking **add**.

Table 7–2 Origin-Server Pools and Origin Servers

Origin-Server Pool	Origin Server Type	Origin Servers
admin-pool	HTTP	ADMINVHN.mycompany.com:7001
osb-pool	HTTP	SOAHOST1-PRIV-V2.mycompany.com:8011 SOAHOST2-PRIV-V2.mycompany.com:8011

After the origin-server pool is created, the Results screen of the New Origin-Server Pool wizard displays a message confirming successful creation of the origin-server pool.

- Click **Close** on the Results screen.
 - The details of the origin-server pool that you just created are displayed on the Origin-Server Pools page.
 - In addition, the **Deployment Pending** message is displayed at the top of the main pane. You can either deploy the updated configuration immediately by clicking **Deploy Changes**, or you can do so later after making further changes as described in [Section 7.8, "Deploying the Configuration and Testing the Virtual Server Addresses."](#)

7.7.2 Creating the Additional Virtual Servers

Create the following additional virtual servers:

- admin.mycompany.com
- soainternal.mycompany.com
- osb.mycompany.com
- osbinternal.mycompany.com

Note: A virtual server `soaexa` for `soa.mycompany.com` was implicitly created when the configuration was created (`origin.server-pool1`).

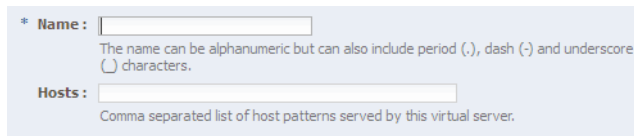
To create a virtual server using the administration console:

- Log in to the administration console using the following URL:

`https://OTDADMINVHN:8989`

2. Click the **Configurations** button that is situated at the upper left corner of the page.
A list of the available configurations is displayed.
3. Select the configuration for which you want to create a virtual server.
4. In the Common Tasks pane, click **New Virtual Server**.
The New Virtual Server wizard starts.

Figure 7–4 New Virtual Server Wizard



5. Create three virtual servers and enter the information in [Table 7–3](#).

Table 7–3 Virtual Server Information

Name	Host Served	Pool
admin.mycompany.com	admin.mycompany.com	admin-pool
soainternal.mycompany.com	WEBHOST1-PRIV-V1	origin-server-pool-1
osb.mycompany.com	osb.mycompany.com	osb-pool
osbinternal.mycompany.com	WEBHOST2-PRIV-V1	osb-pool

6. Select the default (7777) listener.
After the virtual server is created, the Results screen of the New Virtual Server wizard displays a message confirming successful creation of the virtual server.
7. Click **Close** on the Results screen.
 - The details of the virtual server that you just created are displayed on the Virtual Servers page.
 - In addition, the **Deployment Pending** message is displayed at the top of the main pane. You can either deploy the updated configuration immediately by clicking **Deploy Changes**, or you can do so later after making further changes, as described in [Section 7.8, "Deploying the Configuration and Testing the Virtual Server Addresses."](#)

7.7.3 Updating the Host Pattern Served by the SOAEXA Virtual Server

Once the virtual servers are created, the host patterns served by each virtual server are used for appropriate request routing. Update the HOSTS field for the soa.mycompany.com virtual server, clicking on the virtual server name and updating the HOSTS value to soa.mycompany.com. Verify the values in [Table 7–4](#) for the different virtual servers.

Table 7–4 Virtual Server and Host Names

Name	Host
SOAEXA	soa.mycompany.com
admin.mycompany.com	admin.mycompany.com

Table 7–4 (Cont.) Virtual Server and Host Names

Name	Host
soainternal.mycompany.com	WEBHOST1-PRIV-V1
osbinternal.mycompany.com	WEBHOST2-PRIV-V1

Routes for the virtual servers will be created in each of the domain creation and extension chapters.

7.8 Deploying the Configuration and Testing the Virtual Server Addresses

Deploy the configuration to create an instance of it on an administration node. When you deploy a configuration, the running instances are reconfigured to reflect the configuration changes.

To deploy a configuration using the administration console:

1. Log in to the administration console using the following URL:

```
https://OTDADMINVHN:8989
```

2. Click the **Configurations** button at the upper left corner of the page.

A list of the available configurations is displayed.

3. Select the **SOAEXA** configuration.

4. Click **Deploy**.

A message is displayed confirming that the updated configuration was successfully deployed.

5. Click **Close**.

Verify the Deployment

Verify the deployment by accessing the following URL:

```
https://OTDADMINVHN.mycompany.com:8989
```

7.9 Creating a Failover Group for Virtual Hosts

When a request is sent to one of the virtual hosts `admin.mycompany.com`, `osb.mycompany.com`, and `soa.mycompany.com`, the front end load balancer redirects the request to the IP addresses it has been configured to load balance requests to. This IP address is enabled on one of the OTD instances but it can be "migrated" to another OTD instance should a failure occur. You can ensure high availability of Oracle Traffic Director instances by combining two Oracle Traffic Director instances in a failover group represented by one or two virtual IP (VIP) addresses.

You do this by creating an active-passive failover group for the IP address. This failover group lists a primary and a number of secondary instances.

The steps below show you how to create failover groups for the IP addresses associated with the different virtual servers in the configuration.

This SOA enterprise deployment on Exalogic uses four failover groups. One failover group is created for the listen address that is used for load balancing internal/IPoIB requests of the SOA servers. Another failover groups is created for the listen address that is used for load balancing internal/IPoIB requests of the OSB servers. Another

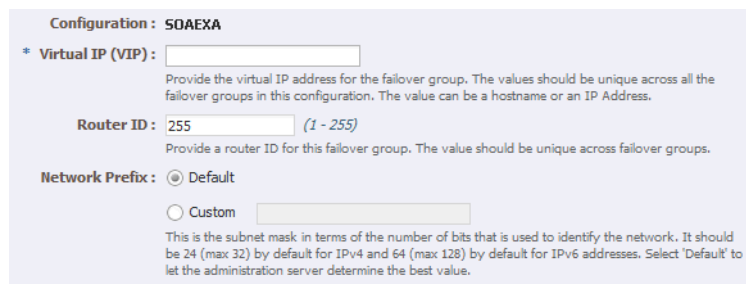
two failover groups are created for the EoIB (externally accessible) addresses used by the OTD listeners in each node.

For more information about creating failover groups or other high availability configurations for Oracle traffic Director, see "Configuring Oracle Traffic Director for High Availability" in the *Oracle Traffic Director Administrator's Guide*.

To create a failover group:

1. Log in to the administration console using the following URL:
 https://OTDADMINVHN:8989
2. Click the **Configurations** button at the upper left corner of the page.
 A list of the available configurations appears.
3. Select the configuration for which you want to create a failover group.
4. In the navigation pane, select **Failover Groups**.
 The Failover Groups page is displayed.
5. Click **New Failover Group**.
 The New Failover Group wizard is displayed.

Figure 7-5 New Failover Group Wizard



6. In the **Virtual IP (VIP)** field, enter the virtual IP address associated with soainternal.mycompany.com (WEBHOST1-PRIV-V1). Check **Custom**, and provide the number of bits to identify the subnet where the IP lives, and click **Next**.

Note: The virtual IP addresses should belong to the same subnet as that of the main NIC that is provided in step 7.

7. In the Step 2: Failover Nodes Information screen, select the Primary and Backup nodes, (OTD Administration Server, WEBHOST2-PRIV-V1), select also the NIC for the virtual IP (typically bond0) and click **Next**.
 The details of the failover group that you just created are displayed on the Failover Groups page.
8. Click **Close** on the Results screen.
 The details of the failover group that you just created are displayed on the Failover Groups page. Repeat the steps to create the appropriate failover groups for the required hostnames according to the values in [Table 7-5](#).
 The router ID for each failover group is unique.

Table 7-5 Failover Groups Node and Network Interface Information

Failover Group	Primary Node	Primary Network Interface	Backup Node	Backup Network Interface
WEBHOST1-PRIV-V1	OTD Administration Server	BOND0	WEBHOST2-PRIV	BOND0
WEBHOST2-PRIV-V1	WEBHOST2-PRIV	BOND0	OTD Administration Server	BOND0
WEBHOST1-VHN1	OTD Administration Server	BOND1	WEBHOST2-VHN1	BOND1
WEBHOST2-VHN1	WEBHOST2-PRIV	BOND1	OTD Administration Server	BOND1

Note: The failover groups for the external virtual IP addresses are optional since the load balancer fails over requests between the two Oracle Traffic Director instances, but they will provide faster failure detection and failover than the typical load balancer monitors

Note: A message may be displayed indicating that the failover group could not be started in the involved nodes due to insufficient privileges. To overcome this log to each node as root and run the following command:

```
WEB_ORACLE_HOME/bin/tadm start-failover --instance-home=WEB_INSTANCE_HOME/ --config=SOAEXA
```

7.10 Backing the Web Tier

Back up the Oracle Traffic director configuration. For more information, see [Section 14.8, "Backing Up the Oracle SOA Enterprise Deployment."](#)

Creating a Domain for an Exalogic Enterprise Deployment

This chapter describes how to create a domain using the Configuration Wizard, Oracle WebLogic Server Administration Console, Oracle Enterprise Manager, and Oracle WSM Policy Manager. You can extend the domain to add SOA components such as Oracle BPM and Oracle Service Bus. To create the domain, you must first create the appropriate Middleware Home and Oracle Homes with the required binaries and libraries to run Oracle SOA.

Note: Before starting the setup process, read the *Oracle Fusion Middleware Release Notes* at the following URL:

http://docs.oracle.com/cd/E28280_01/relnotes.htm

This chapter contains the following sections:

- Section 8.1, "Overview of Creating a Domain"
- Section 8.2, "Installing Oracle Fusion Middleware"
- Section 8.3, "Verifying ADMINVHN in SOAHOST1"
- Section 8.4, "Running the Configuration Wizard on SOAHOST1 to Create a Domain"
- Section 8.5, "Post-Configuration and Verification Tasks"
- Section 8.6, "Associate the Domain with a Database OPSS Policy Store"
- Section 8.7, "Using an LDAP Authenticator (OID, OVD, OUD)"
- Section 8.8, "Moving the WebLogic Administrator to LDAP"
- Section 8.9, "Enabling Domain-Level Exalogic Enhancements"
- Section 8.10, "Validating GridLink Data Sources"
- Section 8.11, "Validating the Administration Server Configuration"
- Section 8.12, "Creating a Separate Domain Directory for Managed Servers in the Same Node as the Administration Server"
- Section 8.13, "Applying the Java Required Files (JRF) Template to the WSM-PM_Cluster"
- Section 8.14, "Disabling Host Name Verification"
- Section 8.15, "Starting and Validating the WLS_WSM1 Managed Server"

- [Section 8.16, "Propagating the Domain Configuration to SOAHOST2"](#)
- [Section 8.17, "Configuring the Java Object Cache for Oracle WSM"](#)
- [Section 8.18, "Configuring Oracle Traffic Director for the WebLogic Domain"](#)
- [Section 8.19, "Backing Up the WebLogic Domain Configuration"](#)

8.1 Overview of Creating a Domain

[Table 8–1](#) lists the steps for creating a WebLogic domain, including post-configuration tasks.

Table 8–1 Steps for Creating a WebLogic Domain

Step	Description	More Information
Install Oracle Fusion Middleware Software	Install the required Oracle Fusion Middleware software for the Exalogic enterprise deployment reference topology for Oracle SOA	Section 8.2, "Installing Oracle Fusion Middleware"
Verify ADMINVHN on SOAHOST1	Associate the administration server with a virtual hostname, ADMINVHN for the SOAHOST1 hostname.	Section 8.3, "Verifying ADMINVHN in SOAHOST1"
Create a WebLogic Domain	Run the Configuration Wizard to create WebLogic domain.	Section 8.4, "Running the Configuration Wizard on SOAHOST1 to Create a Domain"
Post-Configuration and Verification Tasks	Follow the instructions for post-configuration and validation tasks.	Section 8.5, "Post-Configuration and Verification Tasks"
Associate the Domain with a Database Policy Store	Create a data source for OPSS database access.	Section 8.6, "Associate the Domain with a Database OPSS Policy Store"
Enable Domain-Level Exalogic Enhancements	Enable domain-level Exalogic enhancements.	Section 8.9, "Enabling Domain-Level Exalogic Enhancements"
Validate GridLink Data Sources	Verify that the GridLink data sources are correctly configured and that the ONS setup is correct	Section 8.10, "Validating GridLink Data Sources"
Validate the Administration Server Configuration	Validate the configuration by logging into the Oracle WebLogic Server Administration Console and verifying the managed servers and the cluster are listed	Section 8.11, "Validating the Administration Server Configuration"
Create a Separate Domain Directory for Managed Servers	Use the pack and unpack commands to separate the domain directory used by the Administration Server.	Section 8.12, "Creating a Separate Domain Directory for Managed Servers in the Same Node as the Administration Server"
Apply the JRF Template to the WSM-PM_CLUSTER	Target a number of resources not included in the WebLogic server installation to the WSM-PM_Cluster.	Section 8.13, "Applying the Java Required Files (JRF) Template to the WSM-PM_Cluster"
Disable Hostname Verification	Disable host name verification while setting up and validating the topology, and enable it again once the Exalogic enterprise deployment topology configuration is complete.	Section 8.14, "Disabling Host Name Verification"
Start and Validate the WLS_WSM1 Managed Server	Start the managed server and check to confirm that it is running properly.	Section 8.15, "Starting and Validating the WLS_WSM1 Managed Server"

Table 8–1 (Cont.) Steps for Creating a WebLogic Domain

Step	Description	More Information
Propagate the Domain Configuration to SOAHOST2	Propagate the start scripts and classpath configuration from the Administration Server's domain directory to the managed server domain directory.	Section 8.16, "Propagating the Domain Configuration to SOAHOST2"
Configure the Oracle Traffic Director with the WebLogic domain	Configure the Oracle Traffic Director with the WebLogic domain and validate the configuration.	Section 8.18, "Configuring Oracle Traffic Director for the WebLogic Domain"
Back Up the Domain	Back up the newly configured WebLogic domain.	Section 8.19, "Backing Up the WebLogic Domain Configuration"

After you create and configure this domain, you can extend it to include Oracle SOA components or Oracle BAM, as other chapters describe.

8.2 Installing Oracle Fusion Middleware

This section describes how to install the required Oracle Fusion Middleware software for the Exalogic enterprise deployment reference topology for Oracle SOA. The main software components to be installed consist of the Oracle WebLogic Server Home (WL_HOME) and Oracle Home (ORACLE_HOME). You install Oracle Fusion Middleware in at least two storage locations for redundancy.

Note: Before starting the setup process, read the release notes for additional installation and deployment information. They are available on the Oracle Fusion Middleware Documentation Library.

This section covers the following topics:

- [Section 8.2.1, "Installing JRockit"](#)
- [Section 8.2.2, "Installing WebLogic Server Using the Generic Installer"](#)
- [Section 8.2.3, "Installing Oracle Fusion Middleware SOA Suite"](#)

8.2.1 Installing JRockit

Install JRockit on SOAHOST1. SOAHOST2 will use the same mount points.

To install JRockit:

1. Download the version of JRockit for your platform from:

```
http://www.oracle.com/technetwork/middleware/jrockit/downloads/index.html
```

2. Add execute permissions to JRockit. For example:

```
chmod +x jrockit-1.6.0_29-R28.2.0-4.0.1-linux-x64.bin
```

3. Start the JRockit installer by issuing the command:

```
./jrockit-version.bin
```

For example:

```
./jrockit-1.6.0_29-R28.2.0-4.0.1-linux-x64.bin
```

4. On the Welcome Screen, click **Next**.
5. On the Choose Product Installation Directories screen, enter the Product Installation Directory, which is in the Middleware Home.
6. On the Optional Components Screen, click **Next**.
7. On the Installation Complete screen, click **Done**.

8.2.2 Installing WebLogic Server Using the Generic Installer

Install WebLogic Server on SOAHOST1. SOAHOST2 uses the same mount points.

To install WebLogic Server:

1. Download the Oracle WebLogic Server Generic Installer from the following site:

<http://edelivery.oracle.com>

2. Add JRockit to your path. For example, on Linux, enter:

```
export PATH=IAM_MW_HOME/jrockit-jdk1.6.0_29-R28.2.0-4.0.1/bin:$PATH
```

3. Check the version of Java with the following command:

```
java -version
```

Verify that the 64-bit version appears if you have a 64-bit operating system.

4. Start the WebLogic installer by entering one of the following command:

```
java -d64 -jar wls1036_generic.jar
```

5. On the Welcome screen, click **Next**.
6. In the Choose Middleware Home Directory screen, select **Create a new Middleware Home** then enter the following for **Middleware Home Directory**:

```
ORACLE_BASE/product/fmw
```
7. Click **Next**.
8. In the Register for Security Updates screen, enter your contact information so that you can be notified of security updates, and click **Next**.
9. In the Choose Install Type screen, select **Custom** and click **Next**.
10. In the Choose Products and Components screen, deselect **Evaluation database** and click **Next**.
11. In the JDK Selection screen, select only Oracle JRockit 1.6.0_<version> SDK then click **Next**.
12. In the Choose Product Installation Directories screen, accept the directories

```
ORACLE_BASE/fmw/wlserver_10.3
```

 and

```
ORACLE_BASE/fmw/coherence_3.7
```

 then click **Next**.
13. In the Installation Summary screen, click **Next**.
14. In the Installation Complete screen, clear the **Run Quickstart** check box and click **Done**.
15. Validate the installation by verifying that the following directories and files are in the `MW_HOME` directory:

- `coherence_version`

- jrockit-jdkversion
- modules
- registry.xml
- utils
- domain-registry.xml
- logs
- ocm.rsp
- registry.dat
- wlserver_10.3

8.2.3 Installing Oracle Fusion Middleware SOA Suite

Install Oracle Fusion Middleware SOA Suite on SOAHOST1. SOAHOST2 uses the same mount points.

To install Oracle Fusion Middleware SOA Suite on SOAHOST1 and SOAHOST2:

1. On Linux platforms, if the `/etc/oraInst.loc` file exists, check that its contents are correct. Specifically, check that the inventory directory is correct and that you have write permissions for that directory. If the `/etc/oraInst.loc` file does not exist, you can skip this step.
2. Start the installer for Oracle Fusion Middleware SOA Suite from Disk 1 of the installation media:


```
./runInstaller
```
3. When the installer prompts you for a JRE/JDK location, enter the Oracle SDK location created in the Oracle WebLogic Server installation, for example, `ORACLE_BASE/product/fmw/jrockit-jdk1.6.0_version`.
4. In the Specify Inventory Directory screen, enter `HOME/oraInventory`, where `HOME` is the home directory of the user performing the installation. This is the recommended location.
5. Enter the OS group for the user performing the installation then select **OK**.
6. Follow the instructions to run `/createCentralInventory.sh` as root, then click **OK**.

Note: The Specify Inventory Directory screen appears only on a UNIX operating system for the first installation by Oracle Universal Installer. The installer uses the inventory directory to keep track of all Oracle products installed on the machine.

7. In the Welcome screen, click **Next**.
8. In the Install Software Updates screen, choose **Skip Software Updates** and click **Next**.
9. In the Prerequisite Checks screen, verify that all checks complete successfully, and click **OK**.
10. In the Specify Installation Location screen, enter the installation location for Oracle Fusion Middleware SOA Suite. Select the previously-installed Oracle Middleware

Home from the drop-down list. For the Oracle Home directory, enter the directory name (`soa`).

8.3 Verifying ADMINVHN in SOAHOST1

Please note that this step is required for failover of the Administration Server, regardless of whether or not SOA is installed.

You are associating the Administration Server with a virtual hostname (ADMINVHN). This Virtual Host Name must be mapped to the appropriate virtual IP (VIP1) either by a DNS Server or by a custom `/etc/hosts` entry. Check that ADMINVHN is available according to your name resolution system, (DNS server, `/etc/hosts`), in the required nodes in your SOA topology. The virtual IP (VIP1) that is associated to this Virtual Host Name (ADMINVHN) must be enabled in SOAHOST1. External to the Exalogic rack, this VHN must be reachable because this is an EoIB address that typically needs to be accessible to external JMX, JMS, and RMI clients.

8.4 Running the Configuration Wizard on SOAHOST1 to Create a Domain

Run the Configuration Wizard from the Oracle Common home directory to create a domain containing the Administration Server and Oracle Web Services Manager. Later, you will extend the domain to contain SOA components.

To create a domain:

1. Ensure that the database where you installed the repository is running. For Oracle RAC databases, all instances should be running, so that the validation check later in the procedure is more reliable.
2. Change directory to the location of the Configuration Wizard. This is within the SOA home directory. From SOAHOST1:

```
cd ORACLE_COMMON_HOME/common/bin
```

3. Start the Oracle Fusion Middleware Configuration Wizard:

```
./config.sh
```

4. In the Welcome screen, select **Create a New WebLogic Domain**, and click **Next**.
5. The Select Domain Source screen appears ([Figure 8-1](#)).

Figure 8–1 Select Domain Source Screen

Select Domain Source

ORACLE

Generate a domain configured automatically to support the following products:

- Oracle BPM Suite for developers - 11.1.1.0 [soa]
- Oracle BPM Suite - 11.1.1.0 [soa]
- Oracle SOA Suite for developers - 11.1.1.0 [soa]
- Oracle SOA Suite - 11.1.1.0 [soa]
- Oracle Service Bus OWSM Extension - 11.1.1.7 [osb]
- Oracle Enterprise Manager - 11.1.1.0 [oracle_common]
- Oracle Service Bus for developers - 11.1.1.7 [osb]
- Oracle Service Bus - 11.1.1.7 [osb]
- WebLogic Advanced Web Services for JAX-RPC Extension - 10.3.6.0 [wlserver_10.3]
- Oracle Business Activity Monitoring - 11.1.1.0 [soa]
- SOA Bridge Portlet Container - 11.1.1.0 [soa]
- Oracle WSM Policy Manager - 11.1.1.0 [oracle_common]
- Oracle JRF WebServices Asynchronous services - 11.1.1.0 [oracle_common]
- Oracle JRF - 11.1.1.0 [oracle_common]

Base this domain on an existing template

Template location:

In the Select Domain Source screen, do the following:

- Select **Generate a domain configured automatically to support the following products**.
- Select the following products:
 - **Basic WebLogic Server Domain - 10.3.6.0 [wlserver_10.3]** (this should be selected automatically)
 - **Oracle Enterprise Manager - 11.1.1.0 [oracle_common]**
 - **Oracle WSM Policy Manager 11.1.1.0 [oracle_common]**
 - **Oracle JRF - 11.1.1.0 [oracle_common]**

If you accidentally deselect some of the targets, make sure that the following selections are made in this screen:

- Oracle Enterprise Manager
- Oracle WSM Policy Manager
- Oracle JRF

Click **Next**.

6. In the Specify Domain Name and Location screen, enter the domain name (soaexa_domain).

Make sure that the domain directory matches the directory and shared storage mount point recommended in [Section 4.2, "Shared Storage Recommendations for Exalogic Enterprise Deployments."](#) Enter the following for the domain directory:

```
/u01/oracle/config/domains
```

And the following for the application directory, which should be in shared storage:

```
/u01/oracle/config/domains/applications
```

7. Click **Next**.
8. In the Configure Administrator Username and Password screen, enter the username and password to be used for the domain's administrator.
Click **Next**.
9. In the Configure Server Start Mode and JDK screen, do the following:
 - For WebLogic Domain Startup Mode, select **Production Mode**.
 - For JDK Selection, select **JROCKIT SDK1.6.0_<version>**.
 Click **Next**.
10. In the Configure JDBC Components Schema screen, do the following:
 - Select the OWSM MDS schema.
 - For the Oracle RAC configuration for component schemas, select **Convert to GridLink**.
 Click **Next**. The Configure Gridlink RAC Component Schema screen appears (Figure 8–2).

Figure 8–2 RAC Component Schema Screen

Configure GridLink RAC Component Schema

Note: Change only the input fields below that you wish to modify and values will be applied to all selected rows.

Driver: *Oracle's Driver (Thin) for GridLink Connections; Versior

Service Name:

Username:

Password:

Enable FAN:

Enable SSL:

Wallet File:

Wallet Password:

Service Listener	Port	Protocol
slc00erv-v.mycompany.com	1521	TCP
slc00erw-v.mycompany.com	1521	TCP

ONS Host	Port
slc00erterw-r.mycompany.com	6200

RAC Component Schema	Service Name	Schema Owner	Schema Password
<input checked="" type="checkbox"/> OWSM MDS Schema	srv2_panda.mycompany.com	SOAEDGE_X_MDS	*****

Buttons: Exit, Help, Previous, Next

11. Enter values for the following fields, specifying the connect information for the Oracle RAC database that was seeded with RCU:
 - **Driver:** Select **Oracle's driver (Thin) for GridLinkConnections, Versions:10 and later**.
 - **Service Name:** Enter the service name of the database using lowercase characters. For example:
soaedg.mycompany.com.

- **Username:** Enter the database schema owner name of the corresponding component.
- **Password:** Enter the password for the database schema owner.
- Select **Enable FAN**
- Make sure **Enable SSL** is unchecked (alternatively if ssl is selected for ONS notifications to be encrypted, provide the appropriate wallet and wallet password).
- **Service listener:** Enter the SCAN address and port for the RAC database being used. You can identify this 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.mycompany.com:1521

Note: For Oracle Database 11g Release 1 (11.1), use the virtual IP and port of each database's instance listener, for example:

```
custdbhost1-vip.mycompany.com (port 1521)
```

and

```
custdbhost2-vip.mycompany.com (1521)
```

-
- **ONS Host:** Enter the SCAN address for the Oracle RAC database and the ONS remote port as reported by the database:

```
[orcl@db-scan1 ~]$ srvctl config nodeapps -s
```

```
ONS exists: Local port 6100, remote port 6200, EM port 2016
```

Note: For Oracle Database 11g Release 1 (11.1), use the hostname and port of each database's ONS service, for example

```
custdbhost1.mycompany.com (port 6200)
```

and

```
custdbhost2.mycompany.com (6200)
```

-
12. In the Test JDBC Data Sources screen, the connections are tested automatically. The **Status** column displays the results. Ensure that all connections were successful. If not, click **Previous** to return to the previous screen and correct your entries.

Click **Next** when all the connections are successful.

13. In the Select Advanced Configuration screen, select the following:

- **Administration Server**
- **Managed Servers, Clusters and Machines**

- **Deployment and Services**

Click **Next**.

14. In the Configure the Administration Server screen, enter the following values:

- Name: **AdminServer**
- Listen Address: enter ADMINVHN.
- Listen Port: **7001**
- SSL listen port: **N/A**
- SSL enabled: **unchecked**

Click **Next**.

15. In the Configure Managed Servers screen, click **Add** to add the following managed servers:

Table 8–2 Managed Servers

Name	Listen Address	Listen Port	SSL Listen Port	SSL Enabled
WLS_WSM1	SOAHOST1-PRIV	7010	n/a	No
WLS_WSM2	SOAHOST2-PRIV	7010	n/a	No

Click **Next**.

16. In the Configure Clusters screen, Click **Add** to add the following clusters:

Table 8–3 Clusters

Name	Cluster Messaging Mode	Multicast Address	Multicast Port	Cluster Address
WSM-PM_Cluster	unicast	n/a	n/a	Leave it empty.

Click **Next**.

17. In the Assign Servers to Clusters screen, assign servers to the **WSM-PM_Cluster** as follows:

- WLS_WSM
- WLS_WSM2

Click **Next**.

18. In the Configure Machines screen, click the **Unix Machine** tab and then click **Add** to add the following machines:

Note: "Name" can be any unique string. "Node Manager Listen Address" must be a resolvable host name.

Table 8–4 Machines

Name	Node Manager Listen Address
SOAHOST1	SOAHOST1-PRIV
SOAHOST2	SOAHOST2-PRIV

Table 8–4 (Cont.) Machines

Name	Node Manager Listen Address
ADMINHOST	localhost

Leave all other fields to their default values.

Note: The machine name does not need to be a valid host name or listen address; it is just a unique identifier of a Node Manager location

Click **Next**.

19. In the Assign Servers to Machines screen, assign servers to machines as follows:

- **SOAHOST1:** WLS_WSM1
- **SOAHOST2:** WLS_WSM2
- **ADMINHOST:** AdminServer

Click **Next**.

20. In the **Target Deployments to Clusters or Servers** screen, make sure that the **wsm-pm** application is targeted to the **WSM-PM_Cluster** only. Target the library **oracle.wsm.seedpolicies** to **WSM-PM_Cluster**. Make sure that all other deployments are targeted to the **AdminServer** and click **Next**.

21. In the **Target Services to Clusters or Servers** screen, select the following:

- On the left, select **WSM-PM_Cluster**. On the right, select **JDBC System Resource** (this automatically selects all the wsm datasources (mds-owsm)).
- On the left, select **Admin Server**. On the right, select **JDBC System Resource** (this automatically selects all the wsm datasources (mds-owsm)).

All JDBC system resources should be targeted to both the **Admin Server** and **WSM-PM_Cluster**.

- Make sure that all the remaining services are targeted to the **Admin Server**.
- Click **Next**.

For information on targeting applications and resources, see "Appendix B, Targeting Applications and Resources to Servers" in the *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle SOA Suite*.

22. In the Configuration Summary screen, click **Create**.

23. In the Create Domain screen, click **Done**.

8.5 Post-Configuration and Verification Tasks

After configuring the domain with the configuration Wizard, follow these instructions for post-configuration and verification.

This section includes the following topics:

- [Section 8.5.1, "Creating boot.properties for the Administration Server on SOAHOST1"](#)
- [Section 8.5.2, "Configuring and Starting Node Manager on SOAHOST1 and SOAHOST2"](#)

- [Section 8.5.3, "Starting the Administration Server on SOAHOST1"](#)

8.5.1 Creating boot.properties for the Administration Server on SOAHOST1

Create a `boot.properties` file for the Administration Server on SOAHOST1. This is a required step that enables you to start the Administration Server using Node Manager.

To create a `boot.properties` file for the Administration Server:

1. Create the following directory structure:

```
mkdir -p /u01/oracle/config/domains/domain_name/servers/AdminServer/security
```

2. In a text editor, create a file called `boot.properties` in the last directory created in the previous step, and enter the following lines in the file:

```
username=<adminuser>  
password=<password>
```

Note: When you start the Administration Server, the username and password entries in the file get encrypted. You start the Administration Server in [Section 8.5.3, "Starting the Administration Server on SOAHOST1."](#)

For security reasons, you want to minimize the 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.

8.5.2 Configuring and Starting Node Manager on SOAHOST1 and SOAHOST2

Oracle recommends placing the Node Manager configuration and log files in a separate directory from the default directory that uses the Middleware Home.

This section includes the following topics:

- [Section 8.5.2.1, "Generating a properties file for Node Manager and Configuring it to use start scripts"](#)
- [Section 8.5.2.2, "Changing the Location of Node Manager Configuration Files"](#)
- [Section 8.5.2.3, "Editing the nodemanager.properties File"](#)

8.5.2.1 Generating a properties file for Node Manager and Configuring it to use start scripts

To start Node Manager on SOAHOST1:

1. Run the `setNMProps.sh` script located in the following directory:

```
ORACLE_COMMON_HOME/common/bin
```

Set the `StartScriptEnabled` property to 'true' before starting Node Manager:

```
cd ORACLE_COMMON_HOME/common/bin  
./setNMProps.sh
```

Note: You must use the `StartScriptEnabled` property to avoid class loading failures and other problems. For more information, see "Incomplete Policy Migration After Failed Restart of SOA Server" in the *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle SOA Suite*.

2. Start Node Manager:

```
cd WL_HOME/server/bin
export JAVA_OPTIONS="-DDomainRegistrationEnabled=true"
./startNodeManager.sh
```

Note: It is important that you set `-DDomainRegistrationEnabled=true` whenever a Node Manager that manages the AdminServer starts. If there is no AdminServer on this machine and this machine is not an AdminServer failover node, you can start the Node Manager using the following command from SOAHOST1:

```
./startNodeManager.sh
```

3. Stop the Node Manager process by running the following commands:

```
ps -eaf |grep NodeManager
```

Example output:

```
user 10597 472 7 10:40 pts/3 00:00:00 java
weblogic.NodeManager
```

Run the `kill` command to stop the Node Manager process, as in the following example:

```
kill 10597
```

8.5.2.2 Changing the Location of Node Manager Configuration Files

You must create a new directory for Node Manager configuration and log files outside the `MW_HOME` directory, and perform all Node Manager configuration tasks from this directory. Ensure that you do not make any configuration changes to Node Manager files located in the Oracle WebLogic home directory.

To change the location of Node Manager configuration files, see [Section 12.2.1, "Changing the Location of Node Manager Configuration Files."](#):

8.5.2.3 Editing the `nodemanager.properties` File

Node Manager properties define configuration settings for a Java-based Node Manager process. You must specify Node Manager properties on the command line or define them in the `nodemanager.properties` file, located at `/u02/private/oracle/config/nodemanager` on `SOAHOST1` and `SOAHOST2`. Values you enter on the command line override the values in `nodemanager.properties`.

[Table 8–5](#) lists the Node Manager properties that you must change for `SOAHOST1` and `SOAHOST2`.

Table 8–5 Node Manager Properties for SOAHOST1 and SOAHOST2

Properties	Value
StartScriptEnabled	Set the value to "true" to start a server. For more information, see the section "Configuring Node Manager to Use Start and Stop Scripts" in the <i>Oracle Fusion Middleware Node Manager Administrator's Guide for Oracle WebLogic Server</i> .
DomainsFile	/u02/private/oracle/config/nodemanager/nodemanager.domains
ListenAddress	For SOAHOST1: SOAHOST1-PRIV For SOAHOST2: SOAHOST2-PRIV
NodeManagerHome	/u02/private/oracle/config/nodemanager/
LogFile	/u02/private/oracle/config/nodemanager.log
DomainRegistrationEnabled	Set the value to "true".

Note: For more information about `nodemanager.properties`, see "Reviewing `nodemanager.properties`" in the *Oracle Fusion Middleware Node Manager Administrator's Guide for Oracle WebLogic Server*.

You must restart Node Manager for changes to take effect.

For SOAHOST1:

```
cd /u02/private/oracle/config/nodemanager
./startNodeManager.sh
```

For SOAHOST2:

```
cd /u02/private/oracle/config/nodemanager
./startNodeManager.sh
```

8.5.3 Starting the Administration Server on SOAHOST1

The Administration Server is started and stopped using Node Manager. However, the first start of the Administration Server with Node Manager, requires changing the defaulted username and password that are set for Node Manager by the Configuration Wizard. Therefore, use the start script for the Administration Server for the first start.

Steps 1-4 are required for the first start operation; subsequent starts require only step 4.

To start the Administration Server using Node Manager:

1. Start the Administration Server using the start script in the domain directory on SOAHOST1:

```
cd /u01/oracle/config/domains/domain_name/bin
./startWebLogic.sh
```

2. Use the Administration Console to update the Node Manager credentials.

- a. In a browser, go to the following URL:

```
http://ADMINVHN:7001/console
```

- b. Log in as the administrator.
 - c. Click **Lock & Edit**.
 - d. Click **domain_name**, (**Security**) tab, **General**, and then expand the **Advanced** options at the bottom.
 - e. Enter a new username for Node Manager, or make a note of the existing one and update the Node Manager password.
 - f. Click **Save** and **Activate Changes**.
3. Stop the Administration Server process by using **CTRL-C** in the shell where it was started, or by process identification and kill in the OS.
 4. Start WLST and connect to Node Manager with **nmconnect** and the credentials set in the previous steps and start the Administration Server using **nmstart**. Enter the Node Manager Username and password that you entered in step 2e.

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

Once you are in the WLST shell:

```
wls:/offline>nmConnect('nodemanager_username', 'nodemanager_password',
'SOAHOST1-PRIV', '5556', 'domain_name', 'ASERVER_HOME/')
```

```
wls:/nm/domain_name nmStart('AdminServer')
```

Note: This username and password are used only to authenticate connections between Node Manager and clients. They are independent of the server admin ID and password and are stored in the `nm_password.properties` file located in the following directory:

```
u01/oracle/config/domain_name/config/nodemanager
```

5. Restart Node Manager after changing the user and password.

8.6 Associate the Domain with a Database OPSS Policy Store

Associate the domain with a Database OPSS Policy Store.

Before re-association, back up the following configuration files:

- `ASERVER_HOME/config/config.xml`
- `ASERVER_HOME/config/fmwconfig/jps-config.xml`
- `ASERVER_HOME/config/fmwconfig/system-jazn-data.xml`

Back up the `boot.properties` file for the Administration Server in the following directory:

```
ASERVER_HOME/servers/AdminServer/security
```

To re-associate the policy stores with a database:

1. Create a data source for OPSS database access. Use `jdbc/OPSS` as the name and point the connection pool to the OPSS schema that you created using RCU.

Use [Appendix D, "Creating a GridLink Data Source."](#) to create a GridLink data source for OPSS database access using the Oracle WebLogic Administration Console.

For an a GridLink OPSS database access, use the following names:

- Datasource Name: **OPSS**
 - JNDI Name: **jdbc/OPSS**
 - Database User Name: **PROD_OPSS**
2. From SOAHOST1, start the wlst shell:

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

3. Use the wlst command to connect to the WebLogic Administration Server:

```
connect('AdminUser','AdminUserPassword','t3://hostname:port')
```

4. Run the reassociateSecurityStore command:

```
reassociateSecurityStore(domain="domainName",
serverType="DB_ORACLE",datasourceName="datasourceName",
jpsroot="jpsRoot",admin="adminAcnt",
password="passWord")
```

For example:

```
reassociateSecurityStore(domain='soaexa_domain',
serverType='DB_ORACLE',datasourceName='jdbc/OPSS',jpsroot='cn=jpsRoot')
```

For more information on this command, see "reassociateSecurityStore" in the *Oracle Fusion Middleware Application Security Guide*.

5. The reassociateSecurityStore command returns output indicating that data is migrated, has been tested, and that audit store re-association is done.
6. Restart the Administration Server using Node Manager. To restart the Administration Server:

- a. Access ./wlst.sh in the following directory:

```
cd ORACLE_COMMON_HOME/common/bin
```

- b. Run the following command:

```
wls:/offline>nmConnect('nodemanager_username','nodemanager_password',
'SOAHOST1-PRIV','5556','domain_name','ASERVER_HOME')
```

```
wls:/nm/domain_name nmKill('AdminServer')
```

- c. Once you stop the administration server, following command to start it:

```
wls:/nm/domain_name nmStart('AdminServer')
```

8.7 Using an LDAP Authenticator (OID, OVD, OUD)

This section describes how to create the LDAP authenticator using the WebLogic Server Administration Console.

Prerequisites

Before you create the LDAP authenticator, 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
```

Back up the `boot.properties` file for the Administration Server in the following directory:

```
ASERVER_HOME/servers/AdminServer/security
```

To configure the credential store to use LDAP:

1. Log in to the WebLogic Server Console.
2. Click the **Security Realms** link on the left navigational bar.
3. Click the **myrealm** default realm entry to configure it.
4. Open the **Providers** tab within the realm.
5. Observe that there is a `DefaultAuthenticator` provider configured for the realm.
6. Click **Lock & Edit**.
7. Click the **New** button to add a new provider.
8. Enter a name for the provider such as **OIDAuthenticator** or **OVDAuthenticator** depending on whether Oracle Internet Directory or Oracle Virtual Directory will be used.
9. Select the **OracleInternetDirectoryAuthenticator**, **OracleVirtualDirectoryAuthenticator** or **LDAPAuthenticbicator** type from the list of authenticators depending on whether Oracle Internet Directory, Oracle Virtual Directory or Oracle Unified Directory will be used and click **OK**.

Note: The table below applies to OID and is an example, for OUD the default ports and other properties vary.

10. In the Providers screen, click the newly created Authenticator.
11. Set the control flag to **SUFFICIENT**. This indicates that if a user can be authenticated successfully by this authenticator, then it should accept that authentication and should not continue to invoke any additional authenticators. If the authentication fails, it will fall through to the next authenticator in the chain. Make sure all subsequent authenticators also have their control flag set to **SUFFICIENT**; in particular, check the `DefaultAuthenticator` and set that to **SUFFICIENT**.
12. Click **Save** to save this setting.
13. Open the **Provider Specific** tab to enter the details for the LDAP server.
14. Enter the details specific to your LDAP server, as shown in the following table:

Parameter	Value	Value Description
Host	For example: <code>oid.mycompany.com</code>	The LDAP server's server ID.
Port	For example: 636	The LDAP server's port number.

Parameter	Value	Value Description
Principal	For example: cn=orcladmin	The LDAP user DN used to connect to the LDAP server.
Credential	NA	The password used to connect to the LDAP server
SSL Enabled	Checked	Specifies whether SSL protocol is used when connecting to LDAP server.
User Base DN	For example: cn=users,dc=us,dc= mycompany,dc=com	Specify the DN under which your Users start.
Group Base DN	For example: cn=groups,dc=us,dc= =mycompany,dc=com	Specify the DN that points to your Groups node.
Use Retrieved User Name as Principal	Checked	Must be turned on.

Click **Save** when done.

15. Click **Activate Changes** to propagate the changes.

Reorder Authenticator

Reorder the OID/OVD/LOUD Authenticator and Default Authenticator and ensure that the control flag for each authenticator is set in the following order:

To set the order of the Authenticators:

1. Log in to WebLogic Console, if not already logged in.
2. Click **Lock & Edit**.
3. Navigate to **SecurityRealms**, then the default realm name, and then **Providers**.
4. Reorder the OID/OVD Authenticator, and Default Authenticator by ensuring that the control flag for each authenticator is set as follows:
 - OID LDAP Authenticator /OVD LDAP Authenticator/LDAP Authenticator: SUFFICIENT
 - Default Authenticator: SUFFICIENT
5. Click **OK**.
6. Click **Activate Changes** to propagate the changes.
7. Restart the Administration Server and all managed servers.

8.8 Moving the WebLogic Administrator to LDAP

This section provides details for provisioning a new administrator user and group for managing the Oracle Fusion Middleware SOA Suite Enterprise Deployment WebLogic Domain. This section describes the following tasks:

- [Section 8.8.1, "Provisioning Admin Users and Groups in an LDAP Directory"](#)
- [Section 8.8.2, "Assigning the Admin Role to the Admin Group"](#)
- [Section 8.8.3, "Updating the boot.properties File and Restarting the System"](#)

8.8.1 Provisioning Admin Users and Groups in an LDAP Directory

As mentioned in the introduction to this section, users and groups from multiple WebLogic domains may be provisioned in a central LDAP user store. In such a case, there is a possibility that one WebLogic admin user may have access to all the domains within an enterprise. Oracle does not recommend this. To avoid one WebLogic admin user having access to all the domains, the users and groups provisioned must have a unique distinguished name within the directory tree. For the SOA enterprise deployment WebLogic domain described in this guide, the admin user and group are provisioned with the DNs below:

- Admin User DN:

```
cn=weblogic_soa,cn=Users,dc=us,dc=mycompany,dc=com
```

- Admin Group DN:

```
cn=SOA Administrators,cn=Groups,dc=us,dc=mycompany,dc=com
```

To provision the admin user and admin group in Oracle Internet Directory:

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=us, dc=mycompany, dc=com
orclsamaccountname: weblogic_soa
givenname: weblogic_soa
sn: weblogic_soa
userpassword: password
mail: weblogic_soa
objectclass: top
objectclass: person
objectclass: organizationalPerson
objectclass: inetorgperson
objectclass: orcluser
objectclass: orcluserV2
uid: weblogic_soa
cn: weblogic_soa
description: Admin User for the SOA Domain
```

2. Run the `ldapadd` command located under the `ORACLE_HOME/bin` directory to provision the user in Oracle Internet Directory.

Note: The `ORACLE_HOME` used here is the `ORACLE_HOME` for the Identity Management installation where Oracle Internet Directory resides. The `ORACLE_HOME` environment variable must be set for the `ldapadd` command to succeed.

Note: For OUD, refer to the `import-ldiff` command reference in the OUD documentation.

For example (the command is shown as two lines in the example below for readability purposes, but you should enter the command on a single line):

```
OIDHOST1> ORACLE_HOME/bin/ldapadd -h oid.mycompany.com -p 389 -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=us, dc=mycompany, dc=com
displayname: SOA Administrators
objectclass: top
objectclass: groupOfUniqueNames
objectclass: orclGroup
uniquemember: cn=weblogic_soa, cn=users, dc=us, dc=mycompany, dc=com
cn: SOA Administrators
description: Administrators Group for the SOA Domain
```

4. Run the `ldapadd` command located under the `ORACLE_HOME/bin/` directory to provision the group in Oracle Internet Directory (the command is shown as two lines in the example below for readability purposes, but you should enter the command on a single line):

```
OIDHOST1> ORACLE_HOME/bin/ldapadd -h oid.mycompany.com -p 389 -D
cn="orcladmin" -w <password> -c -v -f admin_group.ldif
```

8.8.2 Assigning the Admin Role to the Admin Group

After adding the users and groups to Oracle Internet Directory, the group must be assigned the Admin role within the WebLogic domain security realm. This enables all users that belong to the group to be administrators for that domain.

To assign the Admin role to the Admin group:

1. Log into the WebLogic Administration Server Console.
2. In the left pane of the console, click **Security Realms**.
3. On the Summary of Security Realms page, click **myrealm** under the Realms table.
4. On the Settings page for myrealm, click the **Roles & Policies** tab.
5. On the Realm Roles page, expand the **Global Roles** entry under the **Roles** table. This brings up the entry for **Roles**. Click on the **Roles** link to bring up the Global Roles page.
6. On the Global Roles page, click the **Admin** role to bring up the Edit Global Role page:
 - a. On the Edit Global Roles page, under the **Role Conditions** table, click the **Add Conditions** button.
 - b. On the Choose a Predicate page, select **Group** from the drop down list for predicates and click **Next**.
 - c. On the Edit Arguments Page, specify **SOA Administrators** in the **Group Argument** field and click **Add**.
7. Click **Finish** to return to the Edit Global Rule page.
8. The **Role Conditions** table now shows the SOA Administrators Group as an entry.
9. Click **Save** to finish adding the Admin Role to the SOA Administrators Group.
10. Validate that the changes were successful by bringing up the WebLogic Administration Server Console using a web browser. Log in using the credentials for the `weblogic_soa` user.

Note: Each SOA application has its own predefined roles and groups defined for administration and monitoring. By default, the "Administrator" group allows these operations. However, the "Administrator" group may be too broad. For example, you may not want B2B Administrators to be WebLogic Server Domain Administrators where SOA is running. Therefore, you may wish to create a more specific group, such as "SOA Administrators." In order for the different applications to allow the SOA Administrator group to administer the different systems, you must add the required roles to the SOA Administrator group. For example, for B2B's Administration, add the B2BAdmin role to the SOA Administrators group, for Worklistapp's administration, add the SOAdmin role. Refer to each component's specific roles for the required roles in each case.

8.8.3 Updating the boot.properties File and Restarting the System

The `boot.properties` file for the Administration Server should be updated with the WebLogic admin user created in Oracle Internet Directory. Follow the steps below to update the `boot.properties` file:

1. On SOAHOST1, go the following directory:

```
cd $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. Enter the following lines in the file:

```
username=weblogic_soa
password=password
```

4. Save the file.
5. Stop the Administration Server using the following command:

```
wls:/nm/domain_name>nmKill("AdminServer")
```

6. Start the Administrator Server using the procedure in [Section 8.5.3, "Starting the Administration Server on SOAHOST1."](#)

8.9 Enabling Domain-Level Exalogic Enhancements

To enable domain-level Exalogic enhancements, complete the following steps:

1. Log into the Oracle WebLogic Server Administration Console.
2. Click **Lock & Edit**.
3. Select the domain name in the left navigation pane. The Settings for *Domainname* screen appear. Click the **General** tab.
4. In your domain home page, select **Enable Exalogic Optimizations** then click **Save**.
5. Activate the changes.
6. Stop and start your domain.

8.10 Validating GridLink Data Sources

When the servers are started, verify that the GridLink data sources are correctly configured and that the ONS setup is correct. Perform this procedure for every GridLink data source created.

To validate the GridLink data sources configuration:

1. Log on to the Oracle WebLogic Administration Console.
2. In the **Domain Structure** tree, expand **Services**, and select **Data Sources**.
3. Click one of the new data sources.
4. Click the **Monitoring** tab and select one of the servers.
5. Click the **Statistics** tab and select one of the servers.
6. Click the **ONS** tab, and then click the **Testing** tab.
7. Select the server and click **Test ONS**.

If both tests are successful, the configuration is correct. If the ONS test fails, verify that the ONS service is running in the RAC database nodes:

```

orcl@db-scan1 ~]$ srvctl status scan_listener
SCAN Listener LISTENER_SCAN1 is enabled
SCAN listener LISTENER_SCAN1 is running on node db-scan1
SCAN Listener LISTENER_SCAN2 is enabled
SCAN listener LISTENER_SCAN2 is running on node db-scan2
SCAN Listener LISTENER_SCAN3 is enabled
SCAN listener LISTENER_SCAN3 is running on node db-scan2

[orcl@db-scan1 ~]$ srvctl config nodeapps -s
ONS exists: Local port 6100, remote port 6200, EM port 2016

[orcl@db-scan1 ~]$ srvctl status nodeapps | grep ONS
ONS is enabled
ONS daemon is running on node: db-scan1
ONS daemon is running on node: db-scan2

```

Run the ONS test from every WebLogic server that uses the data source.

8.11 Validating the Administration Server Configuration

To ensure that the Administration Server for the domain you have created is properly configured, validate the configuration by logging into the Oracle WebLogic Server Administration Console and verifying the managed servers and the cluster are listed, and log into Oracle Enterprise Manager.

To verify that the Administration Server is properly configured:

1. In a browser, go to the following URL:


```
http://ADMINVHN:7001/console
```
2. Log in as the administrator.
3. Verify that the WLS_WSM1 and WLS_WSM2 managed servers are listed.
4. Verify that the WSM-PM_Cluster cluster is listed.
5. Check that you can access Oracle Enterprise Manager at the following URL:

```
http://ADMINVHN:7001/em
```

6. Log in to EM Console with the username and password you specified in [Section 8.5.1, "Creating boot.properties for the Administration Server on SOAHOST1."](#)

8.12 Creating a Separate Domain Directory for Managed Servers in the Same Node as the Administration Server

Use the `pack` and `unpack` commands to separate the domain directory used by the Administration Server from the domain directory used by the managed server in *SOAHOST1* as [Section 4.4, "Recommended Directory Locations for an Oracle Exalogic Enterprise Deployment"](#) recommends.

Before running the `unpack` script, be sure the following directory exists:

```
/u02/private/oracle/config/domains
```

To create a separate domain directory:

1. Run the `pack` command on *SOAHOST1* to create a template pack as follows:

```
cd ORACLE_COMMON_HOME/common/bin

./pack.sh -managed=true -domain=ASERVER_HOME
-template=soadomaintemplate.jar -template_name=soa_domain_template
```

2. If the *ASERVER_HOME* directory does not exist, create the directory:

```
mkdir -p /u02/private/oracle/config/domains/soaedg_domain
```

3. Run the `unpack` command on *SOAHOST1* to unpack the template in the managed server domain directory as follows:

```
cd ORACLE_COMMON_HOME/common/bin

./unpack.sh -domain=MSERVER_HOME
-template=soadomaintemplate.jar -app_dir=APP_DIR
```

Note: You must have write permissions on the following directory before running the `unpack` command:

```
/u02/private/oracle/config
```

8.13 Applying the Java Required Files (JRF) Template to the WSM-PM_Cluster

After the domain is created with the Configuration Wizard, you must target a number of resources not included in the WebLogic server installation to the *WSM-PM_Cluster*.

To target these resources:

1. Log in to Oracle Enterprise Manager Fusion Middleware Control using the following URL:

```
http://ADMINVHN:7001/em
```

Use the username and password you specified in [Section 8.5.1, "Creating boot.properties for the Administration Server on SOAHOST1."](#)

2. On the navigation tree on the left, expand **Farm_<domain_name>, WebLogic Domain**, and then **<domain_name>**, and select **WSM-PM_Cluster**.
3. Click **Apply JRF Template** on the right.
4. Wait for the confirmation message to appear on the screen.
This message should confirm that the JRF Template has been successfully applied to the WSM-PM_Cluster cluster.
5. Repeat the steps for the Administration Server.
Expand **Farm_<domain_name>, WebLogic Domain**, and then **<domain_name>**, and select **Admin server**.

8.14 Disabling Host Name Verification

This step is required because you have not set up the appropriate certificates to authenticate the different nodes with the Administration Server (see [Chapter 12, "Setting Up Node Manager for an Exalogic Enterprise Deployment"](#)). Because you have not configured the server certificates, you receive errors when managing the different WebLogic Servers. To avoid these errors, disable host name verification while setting up and validating the topology, and enable it again once the Exalogic enterprise deployment topology configuration is complete as [Chapter 12, "Setting Up Node Manager for an Exalogic Enterprise Deployment"](#) describes.

To disable host name verification:

1. Log in to Oracle WebLogic Server Administration Console using the following URL:
`http://ADMINVHN:7001/console`
2. Click **Lock & Edit**.
3. Expand the **Environment** node in the Domain Structure window.
4. Click **Servers**. The Summary of Servers page appears.
5. Select **AdminServer(admin)** in the Names column of the table. The Settings page for AdminServer(admin) appear.
6. Click the **SSL** tab.
7. Click **Advanced**.
8. Set Hostname Verification to **None**.
9. Click **Save**.
10. Repeat steps 5 to 9 for the WLS_WSM1 server.
11. Save and activate the changes.
12. Restart the Administration Server for the changes to take effect.

To restart the Administration Server:

- a. In the Summary of Servers screen, select the **Control** tab.
- b. Select **AdminServer(admin)** in the table and then click **Shutdown**.
- c. Start the Administration Server again using the procedure in [Section 8.5.3, "Starting the Administration Server on SOAHOST1."](#)

8.15 Starting and Validating the WLS_WSM1 Managed Server

After configuring the managed server, start it and check to confirm that it is running properly. You can start the managed server and check its status by using the Oracle WebLogic Server Administration Console.

To start the WLS_WSM1 managed server and check that it is configured correctly:

1. Start the WLS_WSM1 managed server using the Oracle WebLogic Server Administration Console as follows:
 - a. Expand the **Environment** node in the Domain Structure window.
 - b. Choose **Servers**. The Summary of Servers page appears.
 - c. Click the **Control** tab.
 - d. Select **WLS_WSM1** and then click **Start**.
2. Verify that the server status is reported as **Running** in the Admin 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. See [Section 14.14, "Troubleshooting the Topology in an Enterprise Deployment."](#) for possible causes.
3. Access the following URL:

`http://SOAHOST1-priv:7010/wsm-pm`

Note: Because SOAHOST1-PRIV-V1 is a private/infiniband address, you must use browser from within the Exalogic cells to open the `http://SOAHOST1-priv:7010/wsm-pm` URL.

4. Click **Validate Policy Manager**.

If the configuration is correct, a list of policies and assertion templates available in the data store appear. If the configuration is not correct, no policies or assertion templates appear.

8.16 Propagating the Domain Configuration to SOAHOST2

After completing the configuration of SOAHOST1, propagate the configuration to SOAHOST2 using the unpack utility, and then validate the propagated configuration.

This section includes the following topics:

- [Section 8.16.1, "Propagating the Domain Configuration to SOAHOST2 Using the unpack Utility"](#)
- [Section 8.16.2, "Modify the Upload and Stage Directories to an Absolute Path"](#)
- [Section 8.16.3, "Disabling Host Name Verification for the WLS_WSM2 Managed Server"](#)
- [Section 8.16.4, "Starting Node Manager on SOAHOST2"](#)
- [Section 8.16.5, "Starting and Validating the WLS_WSM2 Managed Server"](#)

8.16.1 Propagating the Domain Configuration to SOAHOST2 Using the unpack Utility

Propagate the domain configuration using the unpack utility. Before running the unpack script, be sure the following directory exists as [Section 4.4, "Recommended Directory Locations for an Oracle Exalogic Enterprise Deployment"](#) recommends:

```
/u02/private/oracle/config/
```

To propagate the domain configuration:

1. Run the following command on SOAHOST1 to copy the template file created previously.

```
cd ORACLE_COMMON_HOME/common/bin
scp soadomaintemplate.jar oracle@SOAHOST2:/ORACLE_COMMON_HOME/common/bin
```

2. Run the unpack command from the ORACLE_COMMON_HOME/common/bin directory, not from the WL_HOME/common/bin directory on SOAHOST2 to unpack the propagated template.

```
cd ORACLE_COMMON_HOME/common/bin

./unpack.sh -domain=MSERVER_HOME
-template=soadomaintemplate.jar -app_dir=APP_DIR
```

Note: The configuration steps provided in this Exalogic enterprise deployment topology are documented with the assumption that a local (per node) domain directory is used for each managed server.

8.16.2 Modify the Upload and Stage Directories to an Absolute Path

After creating the domain and unpacking to the private directory, update the upload and stage directories for WLS_WSM1, WLS_WSM2 and the Administration Server. These directories are defaulted to:

```
./servers/AdminServer/upload
```

and

```
./servers/server_name/stage
```

As a result, these default directory paths create issues for remote deployments and for deployments using the stage mode.

To avoid these issues, update the upload directory to:

```
ASERVER_HOME/servers/AdminServer/upload
```

And update the stage directory to:

```
MSERVER_HOME/servers/manage_server_name/stage
```

Update these directory paths for all servers.

To update these directories:

1. Access the Administration Console.
2. In the left navigation tree, expand **Domain**, and then **Environment**.
3. Click on **Servers**, and then the server's name.

4. Under the **Configuration**, and then **Deployments** section, change the **Upload** and **Stage** directories.

8.16.3 Disabling Host Name Verification for the WLS_WSM2 Managed Server

For the Exalogic enterprise deployment described in this guide, you set up the appropriate certificates to authenticate the different nodes with the Administration Server after you have completed the procedures to extend the domain for Oracle SOA Suite. You must disable the host name verification for the WLS_SOA1 and WLS_SOA2 managed servers to avoid errors when managing the different WebLogic Server instances. For more information, see [Section 8.14, "Disabling Host Name Verification."](#)

You enable host name verification again once the Exalogic enterprise deployment topology configuration is complete. For more information, see [Section 12.3, "Enabling Host Name Verification Certificates for Node Manager."](#)

8.16.4 Starting Node Manager on SOAHOST2

After you propagate the domain configuration and disable host name verification, start Node Manager using the `startNodeManager.sh` script.

Note: A prerequisite for starting Node Manager is updating the Node Manager location, which [Section 8.5.2.2, "Changing the Location of Node Manager Configuration Files"](#) describes.

To start Node Manager on SOAHOST2, run the following:

```
cd /u02/private/oracle/config/nodemanager
./startNodeManager.sh
```

8.16.5 Starting and Validating the WLS_WSM2 Managed Server

Use the Administration Console to start and validate the WLS_WSM2 managed server.

To start the WLS_WSM2 managed server and check that it is configured correctly:

1. Start the WLS_WSM2 managed server using the Administration Console.
2. Verify that the server status is reported as **Running** in the Admin 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. See [Section 14.14, "Troubleshooting the Topology in an Enterprise Deployment."](#) for possible causes.
3. Access the following URL:

```
http://SOAHOST2-priv:7010/wsm-pm
```

Note: Because SOAHOST2-PRIV-V1 is a private/infiniband address, you must use browser from within the Exalogic cells to open the `http://SOAHOST2-priv:7010/wsm-pm` URL.

4. Click validate policy manager.

8.17 Configuring the Java Object Cache for Oracle WSM

Configure the Java Object Cache (JOC) among all the servers running Oracle WSM. This procedure is optional, but increases the performance of Oracle WSM by keeping a local cache instead of having to search for a cache.

Use JOC for MDS updates in B2B if you are planning to change the delivery channels for B2B agreements frequently.

Configure the Java Object Cache using the `configure-joc.py` script in the following directory:

```
MW_HOME/oracle_common/bin/
```

This is a Python script that runs in WLST online mode and expects the Administration Server to be up and running.

Use ports in the 9988 to 9998 range when configuring JOC ports for Oracle products.

To configuring the Java Object Cache for Oracle WSM:

1. Connect to the Administration Server on *SOAHOST1* using the command-line Oracle WebLogic Scripting Tool (WLST), for example:

```
/MW_HOME/oracle_common/common/bin/wlst.sh
connect('weblogic_userr','weblogic_password','t3://ADMINVHN:7001')
```

When prompted, enter the server URL (`t3://ADMINVHN:7001`) and the Oracle WebLogic Server administrator user name and password.

2. After connecting to the Administration Server using `wlst`, start the script using the `execfile` command, for example:

```
wls:/mydomain/serverConfig> execfile('MW_HOME/oracle_
common/bin/configure-joc.py')
```

3. Configure JOC for all the managed servers for a given cluster.

Enter 'y' when the script prompts whether you want to specify a cluster name, and also specify the cluster name and discover port, when prompted. This discovers all the managed servers for the given cluster and configure the JOC. The discover port is common for the entire JOC configuration across the cluster. For example:

```
Do you want to specify a cluster name (y/n) <y>
Enter Cluster Name : WSM-PM_Cluster
Enter Discover Port : 9991
```

Here is a walkthrough for using `configure-joc.py` for HA environments:

```
execfile('MW_HOME/oracle_common/bin/configure-joc.py')
.
Enter Hostnames (eg host1,host2) : SOAHOST1-PRIV,SOAHOST2-PRIV
.
Do you want to specify a cluster name (y/n) <y>y
.
Enter Cluster Name : WSM-PM_Cluster
.
Enter Discover Port : 9991
.
Enter Distribute Mode (true|false) <true> : true
.
Do you want to exclude any server(s) from JOC configuration (y/n) <n> n
```


4. After configuring the Java Object Cache using the `wlst` commands or `configure-joc.py` script, restart all affected managed servers for the configurations to take effect.

The script can also be used to perform the following optional JOC configurations:

- Configure JOC for all specified managed servers.

Enter 'n' when the script prompts whether you want to specify a cluster name, and also specify the managed server and discover port, when prompted. For example:

```
Do you want to specify a cluster name (y/n) <y>n
Enter Managed Server and Discover Port (eg WLS_WSM1:9998, WLS_WSM1:9998) : WLS_
WSM1:9991,WLS_WSM2:9991
```

- Exclude JOC configuration for some managed servers.

The script allows you to specify the list of managed servers for which the JOC configuration "DistributeMode" will be set to 'false'. Enter 'y' when the script prompts whether you want to exclude any servers from JOC configuration, and enter the managed server names to be excluded, when prompted. For example:

```
Do you want to exclude any server(s) from JOC configuration (y/n) <n>y
Exclude Managed Server List (eg Server1,Server2) : WLS_WSM1,WLS_WSM3
```

- Disable the distribution mode for all managed servers.

The script allows you to disable the distribution to all the managed servers for a specified cluster. Specify 'false' when the script prompts for the distribution mode. By default, the distribution mode is set to 'true'.

Verify JOC configuration using the CacheWatcher utility. See *Oracle Fusion Middleware High Availability Guide*.

You can configure the Java Object Cache (JOC) using the **HA Power Tools** tab in the Oracle WebLogic Administration Console as described in the *Oracle Fusion Middleware High Availability Guide*.

8.18 Configuring Oracle Traffic Director for the WebLogic Domain

This section describes tasks for configuring Oracle Traffic Director for the WebLogic Domain, and for verifying the configuration.

This section includes the following topics:

- [Section 8.18.1, "Configuring Oracle Traffic Director to Create Virtual Server Routes"](#)
- [Section 8.18.2, "Validating Access through Oracle Traffic Director"](#)
- [Section 8.18.3, "Turning on the WebLogic Plug-in Enabled Flag"](#)
- [Section 8.18.4, "Setting the Frontend URL for the Administration Console and Setting Redirection Preferences"](#)

8.18.1 Configuring Oracle Traffic Director to Create Virtual Server Routes

The required virtual servers for the SOA EDG topology were created in [Chapter 7, "Installing and Configuring Oracle Traffic Director for an Exalogic Enterprise Deployment."](#) For this procedure, add the routes so that those virtual servers route to the appropriate servers only when specific URLs are used.

To create the required virtual server routes for Oracle Traffic Director:

1. Log into the Administration Console using the following URL:
`https://OTDADMINVHN:8989`
2. Click **Configurations** to open a list of available configurations.
3. Select the configuration you want to configure routes for.
4. In the Navigation pane, expand Virtual Servers, expand the `admin.mycompany.com` virtual server, and select **Routes** to open the Routes page, which lists the routes that are currently defined for the virtual server.

Continue to the following procedure.

To create a new route:

1. Click **New Route**.
2. In the Step 1: Route Properties screen, enter `admin-route` in the **Name** field.
3. In the **Origin Server Pool** drop-down menu, select **admin-pool** and click **Next**.
4. In the Step2: Condition Information screen, select the `$uri` variable from the Variable/Function drop-down list. Select the Operator (`= ~`). And enter `/console` in the Value field.

Note: You cannot use a joiner (and/or) for the first expression in the sequence.

5. Click **OK** and click **Plus** to add the next expression.

Note: You can now select the joiner 'or'.

6. Select **\$uri** as the Variable/Function, `= ~` as the Operator, and `/em` in the Value field. Click **OK**.

Figure 8–3 New Route Condition Information



7. Click **Next** and then click **Create Route**.

The new route appears on the Routes page and a Deployment Pending message appears in the main pane. You can deploy the updated configuration immediately by selecting **Deploy Changes** or wait until you make changes. See the topic [Section 7.8, "Deploying the Configuration and Testing the Virtual Server Addresses"](#) for more information.

To create a new route in the `soainternal` virtual server, repeat the preceding steps and conditions. You can copy the rules from the first route you created and use `origin-server-pool1`.

8.18.2 Validating Access through Oracle Traffic Director

In the Administration Console, verify that the server status is `Running`. If the server status is `Starting` or `Resuming`, wait for the server status to change to `Started`. If you see another status, such as `Admin` or `Failed`, check the server output log files for errors. See [Section 14.14, "Troubleshooting the Topology in an Enterprise Deployment."](#)

Test the WebLogic Server Administration Console and Enterprise Manager Fusion Middleware Control using the `admin.mycompany.com` URL (through the pertaining load balancer). To do this, verify using the following URLs:

```
http://admin.mycompany.com/console
```

```
http://admin.mycompany.com/em
```

8.18.3 Turning on the WebLogic Plug-in Enabled Flag

For security purposes, and since the load balancer terminates SSL requests, turn on the WebLogic plug-in enabled flag for the domain after you configure SSL for the load balancer.

To turn on the WebLogic plug-in enabled flag:

1. Log on to the Administration Console.
2. Click on the domain name in the navigation tree on the left.
3. Click on the **Web Applications** tab.
4. Click **Lock & Edit**.
5. Select the **WebLogic Plugin Enabled** check box.
6. Save and activate the changes.

8.18.4 Setting the Frontend URL for the Administration Console and Setting Redirection Preferences

When you access the Oracle WebLogic Server Administration Console using a load balancer, changing the Administration Server's frontend URL is required so that the user's browser is redirected to the appropriate load balancer address.

The Oracle WebLogic Server Administration Console application tracks changes made to ports, channels and security using the console. When changes made through the console are activated, the console validates its current listen address, port and protocol. If the listen address, port and protocol are still valid, the console redirects the HTTP request replacing the host and port information with the Administration Server's listen address and port.

To change the Administration Server's frontend URL:

1. Log in to Oracle WebLogic Server Administration Console.
2. Click **Lock & Edit**.
3. Expand the **Environment** node in the Domain Structure window.
4. Click **Servers** to open the Summary of Servers page.
5. Select **Admin Server** in the **Names** column of the table. The Settings page for AdminServer(admin) appears.
6. Click the **Protocols** tab.
7. Click the **HTTP** tab.
8. Set the **Frontend Host** to **admin.mycompany.com** and the **Frontend HTTP Port** to **80** (modify accordingly if HTTPS is used for the admin URL).
9. Save and activate the changes.
10. Disable tracking on configuration changes in the Oracle WebLogic Server Administration Console so that the console does not trigger the reload of configuration pages when activation of changes occurs.
 - a. Log in to the Oracle WebLogic Server Administration Console.
 - b. Click the **preferences** link in the banner.
 - c. Click the **shared preferences** tab.
 - d. Deselect the **follow configuration changes** check box.

8.19 Backing Up the WebLogic Domain Configuration

Back up the WebLogic domain configuration. For more information, see [Section 14.8, "Backing Up the Oracle SOA Enterprise Deployment."](#)

Extending the Domain for SOA Components

This chapter describes how to use the Configuration Wizard to extend the domain to include SOA components. You created in the domain in [Chapter 8, "Creating a Domain for an Exalogic Enterprise Deployment."](#)

Note: Before starting the setup process, read the *Oracle Fusion Middleware Release Notes* at the following URL:

http://docs.oracle.com/cd/E28280_01/relnotes.htm

SOA components use server migration, allowing SOA managed servers to be migrated from one host to another, so that if a node hosting one of the servers fails, the service can continue on another node. For information about configuring server migration, see [Chapter 13, "Configure Server Migration for an Exalogic Enterprise Deployment."](#)

This chapter contains the following sections:

- [Section 9.1, "Overview of Extending the Domain for SOA Components."](#)
- [Section 9.2, "Pre-verifications for Extending the Domain for Oracle SOA Components."](#)
- [Section 9.3, "Extending the Domain for SOA Components using the Configuration Wizard."](#)
- [Section 9.4, "Configuring Oracle Coherence for Deploying Composites."](#)
- [Section 9.5, "Post-Configuration and Verification Tasks."](#)
- [Section 9.6, "Configuring Network Channels for HTTP and T3 Clients Through EoIB."](#)
- [Section 9.7, "Configuring Oracle Traffic Director with the Extended Domain."](#)
- [Section 9.8, "Configuring a Default Persistence Store for Transaction Recovery."](#)
- [Section 9.9, "Configuring Coherence Caches for Dehydrations."](#)
- [Section 9.10, "Updating SOA JVM settings."](#)
- [Section 9.11, "Enabling Cluster-Level Session Replication Enhancements."](#)
- [Section 9.12, "Configuring Oracle Adapters."](#)
- [Section 9.13, "Updating the Workflow Front End Address for Appropriate Task Display."](#)
- [Section 9.14, "Updating the B2B Instance Identifier for Transports."](#)
- [Section 9.15, "Backing Up the Oracle SOA Configuration."](#)

9.1 Overview of Extending the Domain for SOA Components

Extend the WebLogic domain to include Oracle SOA components. [Table 9–1](#) lists the steps for configuring Oracle SOA and other tasks required for extending the domain for Oracle SOA components.

Table 9–1 Steps for Extending the Domain for SOA Components

Step	Description	More Information
Prepare for extending the Domain for SOA Components	Verify that the appropriate virtual IP mapping for each of the hostnames on the two SOA Machines is available.	Section 9.2, "Pre-verifications for Extending the Domain for Oracle SOA Components"
Extend the Domain for SOA Components	Extend the WebLogic domain you created in Chapter 8 .	Section 9.3, "Extending the Domain for SOA Components using the Configuration Wizard"
Configure Oracle Coherence for Deploying Composites	Configure Oracle Coherence in order to use unicast communication for deploying composites for Oracle BPEL Dehydration.	Section 9.4, "Configuring Oracle Coherence for Deploying Composites"
Post-Configuration and Verification Tasks	Follow these instructions for post-configuration and validation tasks.	Section 9.5, "Post-Configuration and Verification Tasks"
Configure Additional Network Channels for T3 clients.	If your HTTP clients and T3 clients use the 10 Gb Ethernet network, you must create additional network channels for the SOA Servers on SOAHOST1 and SOAHOST2.	Section 9.6, "Configuring Network Channels for HTTP and T3 Clients Through EoIB"
Configure Oracle Traffic Director for the Extended Domain	Configure Oracle Traffic Director to route to the SOA servers for the appropriate URLs.	Section 9.7, "Configuring Oracle Traffic Director with the Extended Domain"
Configure a Default Persistence Store	Configure a default persistence store for transaction recovery.	Section 9.8, "Configuring a Default Persistence Store for Transaction Recovery"
Configure Oracle Coherence for Deploying Composites	Configure Oracle Coherence in order to use unicast communication for deploying composites.	Section 9.9, "Configuring Coherence Caches for Dehydrations"
Configure Oracle Adapters	Enable high availability for Oracle File and FTP Adapters, enable high availability for Oracle JMS Adapters, and scale the Oracle Database Adapter.	Section 9.12, "Configuring Oracle Adapters"
Update the B2B Instance Identifier for Transports	Set up File, FTP, or Email transports in a high availability environment.	Section 9.14, "Updating the B2B Instance Identifier for Transports"
Back Up the SOA Configuration	Back up the newly extended domain configuration.	Section 9.15, "Backing Up the Oracle SOA Configuration"

9.2 Pre-verifications for Extending the Domain for Oracle SOA Components

Before you run the Configuration Wizard to extend the domain, verify that the appropriate virtual IP mapping for each of the hostnames on the two SOA Machines is available. Also, verify that the clocks are synchronized in the nodes where you will run SOA.

This section includes the following topics:

- [Section 9.2.1, "Verify Virtual IPs and Hostnames on SOAHOST1 and SOAHOST2"](#)
- [Section 9.2.2, "Synchronize System Clocks"](#)
- [Section 9.2.3, "Verifying Oracle Home Installation"](#)

9.2.1 Verify Virtual IPs and Hostnames on SOAHOST1 and SOAHOST2

Make sure you can ping from each of the cells and from the OTD nodes the following hostnames:

- SOAHOST1-PRIV-V1
- SOAHOST2-PRIV-V1
- SOAHOST1VHN1
- SOAHOST2VHN1

9.2.2 Synchronize System Clocks

Verify that clocks are in sync by running as simultaneously as possible a "date" command in the two nodes in the cluster.

9.2.3 Verifying Oracle Home Installation

This chapter is based on the assumption that you have installed WL_HOME and MW_HOME (binaries) containing Oracle Fusion Middleware SOA on shared storage, and they are available from SOAHOST1 and SOAHOST2. This chapter is also based on the assumption that you have already configured Node Manager, Admin Server and WSM Servers as described in previous chapters. Validate the installation by verifying that the following directories and files appear in the ORACLE_HOME directory after installing both Oracle WebLogic Sever and Oracle Fusion Middleware for SOA:

- coherence_X.X
- jrocket-jdkY.Y
- modules
- oracle_common
- registry.xml
- utils
- domain-reistry.xml
- logs
- ocm.rsp
- registry.dat
- soa
- wlserver_10.3

9.3 Extending the Domain for SOA Components using the Configuration Wizard

Use the Configuration Wizard to extend the domain created in [Chapter 8, "Creating a Domain for an Exalogic Enterprise Deployment,"](#) to contain SOA components.

Note: If you have not backed up the domain created in [Chapter 8, "Creating a Domain for an Exalogic Enterprise Deployment,"](#) back up the current domain before extending it for SOA components. You may use the backup to recover in case any errors are made in the domain extension. See "Backing Up Your Environment" in the *Oracle Fusion Middleware Administrator's Guide*.

To extend the domain using the Configuration Wizard:

1. Change directory to the location of the Configuration Wizard. This is within the SOA home directory on SOAHOST1. Oracle recommends having all database instances up.

```
ORACLE_COMMON_HOME/common/bin
```

2. Start the Configuration Wizard.

```
./config.sh
```

3. In the Welcome screen, select **Extend an existing WebLogic domain**, and click **Next**.

4. In the WebLogic Domain Directory screen, select the following WebLogic domain directory

```
ASERVER_HOME
```

Click **Next**.

5. In the Select Extension Source screen, do the following:
 - a. Select **Extend my domain automatically to support the following added products**.

- b. Select the following products: **Oracle SOA Suite 11.1.1.0**

The following products should already be selected, and grayed out. They were selected when you created the domain in [Chapter 8.4, "Running the Configuration Wizard on SOAHOST1 to Create a Domain."](#)

Basic WebLogic Server Domain

Oracle Enterprise Manager

Oracle WSM Policy Manager

Oracle JRF

Click **Next**.

6. If you see a "Conflict Detected" message that Oracle JRF is already defined in the domain, select the **Keep Existing Component** option and click **OK**.

7. In the Configure Gridlink RAC Data Sources screen

- a. Select the OPSS datasource created for Policy Store re-association.
 - b. Deselect **Enable SSI**.
 - c. Click **Next**.

8. In the Test Datasource screen, click **Next**.

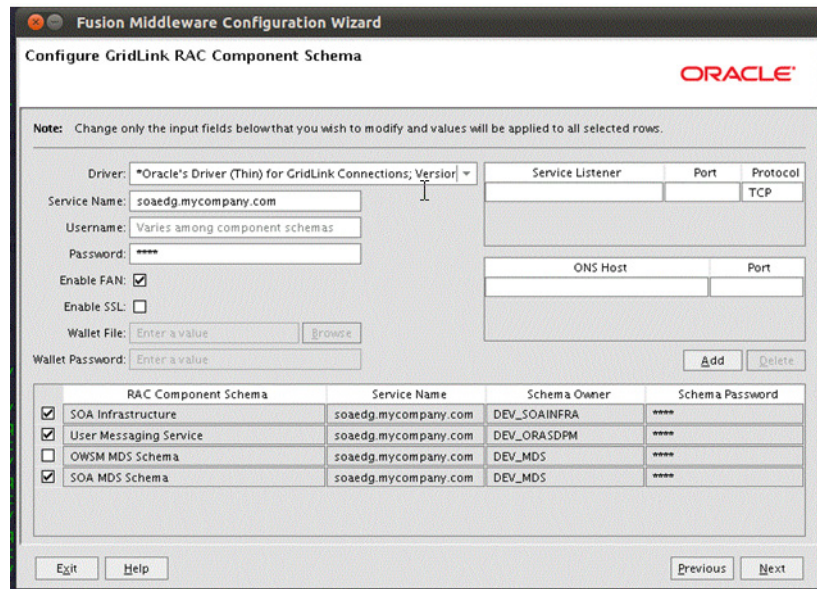
9. In the Configure JDBC Components Schema screen, do the following:

- Select the **SOA Infrastructure, User Messaging Service, and SOA MDS Schema.**
- For the Oracle RAC configuration for component schemas, select **Convert to GridLink**

Click **Next**.

10. The Configure Gridlink RAC Component Schema screen appears (Figure 9–1).

Figure 9–1 Configure GridLink RAC Component Schema Screen



Enter values for the following fields, specifying the connect information for the Oracle RAC database that was seeded with RCU:

- **Driver:** Select **Oracle's driver (Thin) for GridLinkConnections, Versions:10 and later.**
- **Service Name:** Enter the service name of the database using lowercase characters. For example:
soaedg.mycompany.com.
- **Username:** Enter the database schema owner name of the corresponding component.
- **Password:** Enter the password for the database schema owner.
- Select **Enable FAN**
- Make sure **Enable SSL** is unchecked (alternatively if ssl is selected for ONS notifications to be encrypted, provide the appropriate wallet and wallet password).
- **Service listener:** Enter the SCAN address and port for the RAC database being used. You can identify this 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.mycompany.com:1521
```

Note: For Oracle Database 11g Release 1 (11.1), use the virtual IP and port of each database instance listener, for example:

```
custdbhost1-vip.mycompany.com (port 1521)
```

and

```
custdbhost2-vip.mycompany.com (1521)
```

For Oracle Database 10g, use multi data sources to connect to an Oracle RAC database.

- **ONS Host:** Enter the SCAN address for the Oracle RAC database and the ONS remote port as reported by the database:

```
[orcl@db-scan1 ~]$ srvctl config nodeapps -s
```

```
ONS exists: Local port 6100, remote port 6200, EM port 2016
```

Note: For Oracle Database 11g Release 1 (11.1), use the hostname and port of each database's ONS service, for example

```
custdbhost1.mycompany.com (port 6200)
```

and

```
custdbhost2.mycompany.com (6200)
```

11. In 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.
12. In the Select Optional Configuration screen, select the following:
 - JMS Distributed Destinations
 - Managed Servers, Clusters, and Machines
 - Deployments and Services
 - JMS File Store
 Click **Next**.
13. In the Select JMS Distributed Destination Type screen, ensure that ALL JMS system resources are configured to be UDD. This should be the default.
14. In the Configure Managed Servers screen, add the required managed servers.
 - a. Select the automatically created server and click **Rename** to change the name to **WLS_SOA1**.

- b. Click **Add** to add another new server and enter **WLS_SOA2** as the server name.
- c. Give servers **WLS_SOA1** and **WLS_SOA2** the attributes in [Table 9–2](#). Do not modify the other servers that are shown in this screen; leave them as they are.

Table 9–2 Managed Servers

Name	Listen Address	Listen Port	SSL Listen Port	SSL Enabled
WLS_SOA1	SOAHOST1-PRIV-V1	8001	n/a	No
WLS_SOA2	SOAHOST2-PRIV-V1	8001	n/a	No
WLS_WSM1	SOAHOST1-PRIV	7010	n/a	No
WLS_WSM2	SOAHOST2-PRIV	7010	n/a	No

Click **Next**.

15. In the Configure Clusters screen, add the following clusters:

Table 9–3 Clusters

Name	Cluster Messaging Mode	Multicast Address	Multicast Port	Cluster Address
SOA_Cluster	unicast	n/a	n/a	SOAHOST1-PRIV-V1:8001,SOAHOST2-PRIV-V1:8001 Note: The cluster address should be changed to EoIB (SOAHOST1VHN1 : 8001,SOAHOST2VHN1 : 8001) addresses if external clients will access the SOA servers.
WSM-PM_Cluster	unicast	n/a	n/a	Leave it empty.

Click **Next**.

Note: For asynch request/response interactions over direct binding, the SOA composites must provide their jndi provider URL for the invoked service to look up the beans for callback.

If soa-infra config properties are not specified, but the WebLogic Server Cluster address is specified, the cluster address from the JNDI provider URL is used. This cluster address can be a single DNS name which maps to the clustered servers' IP addresses or a comma separated list of server ip:port. Alternatively, the soa-infra config property `JndiProviderURL/SecureJndiProviderURL` can be used for the same purpose if explicitly set by users.

16. In the Assign Servers to Clusters screen, assign servers to clusters as follows:
 - **SOA_Cluster:**
 - WLS_SOA1
 - WLS_SOA2
 - **WSM-PM_Cluster:**
 - WLS_WSM1

- WLS_WSM2

Click **Next**.

17. In the Configure Machines screen, delete the **LocalMachine** that appears by default and click the **Unix Machine** tab.

The following entries appear (listed in [Table 9-4](#)):

Table 9-4 Machines

Name	Node Manager Listen Address
SOAHOST1	SOAHOST1-PRIV
SOAHOST2	SOAHOST2-PRIV
ADMINHOST	localhost

Leave all other fields to their default values.

Click **Next**.

18. In the Assign Servers to Machines screen, assign servers to machines as follows:

- **ADMINHOST:**
 - AdminServer
- **SOAHOST1:**
 - WLS_SOA1
 - WLS_WSM1
- **SOAHOST2:**
 - WLS_SOA2
 - WLS_WSM2

Click **Next**.

19. In the Target Deployments to Clusters or Servers screen, ensure the following targets:

- Target **usermessagingserver** and **usermessagingdriver-email** only to **SOA_Cluster**. (The **usermessaging-xmpp**, **usermessaging-smpp**, and **usermessaging-voicexml** applications are optional.)
- Target the **oracle.sdp.*** and **oracle.soa.*** libraries only to **SOA_Cluster**.
- Target the **oracle.rules.*** library only to **Admin Server** and **SOA_Cluster**.
- Target the **wsm-pm** application only to **WSM-PM_Cluster**.

Click **Next**.

20. In the Target Services to Clusters or Servers screen, verify the following targets:

- Target **mds-owsm** to both **WSM-PM_Cluster** and **AdminServer**.
- Target the **OPSS** to the **WSM-PM_Cluster**, **SOA_Cluster** and **AdminServer**

Click **Next**.

21. In the Configure JMS File Stores screen, enter the shared directory location specified for your JMS stores as recommended in [Section 4.2, "Shared Storage Recommendations for Exalogic Enterprise Deployments."](#) For example:

`ASERVER_HOME/jms`

Note: The `ASERVER_HOME/jms` directory must exist before you enter this location in the JMS File Stores screen.

22. Click **Next**.

23. In the Configuration Summary screen click **Extend**.

Note: Click **OK** to dismiss the warning dialog about the domain configuration ports conflicting with the host ports. This warning appears because of the existing WSM-PM installation.

24. In the Extending Domain screen, click **Done**.

25. Restart the Administration Server using the procedure in [Section 8.5.3, "Starting the Administration Server on SOAHOST1."](#)

9.4 Configuring Oracle Coherence for Deploying Composites

Although deploying composites uses multicast communication by default, Oracle recommends using unicast communication in SOA Exalotic enterprise deployments. Use unicast if you disable multicast communication for security reasons.

Unicast communication does not enable nodes to discover other cluster members in this way. Consequently, you must specify the nodes that belong to the cluster. You do not need to specify all of the nodes of a cluster, however. You need only specify enough nodes so that a new node added to the cluster can discover one of the existing nodes. As a result, when a new node has joined the cluster, it is able to discover all of the other nodes in the cluster. Additionally, in configurations such as SOA Exalotic enterprise deployments where multiple IPs are available in the same system, you must configure Oracle Coherence to use a specific host name to create the Oracle Coherence cluster.

Note: An incorrect configuration of the Oracle Coherence framework used for deployment may prevent the SOA system from starting. The deployment framework must be properly customized for the network environment on which the SOA system runs. Oracle recommends the configuration described in this section.

This section includes the following topics:

- [Section 9.4.1, "Enabling Communication for Deployment Using Unicast Communication"](#)
- [Section 9.4.2, "Specifying the Host Name Used by Oracle Coherence"](#)

9.4.1 Enabling Communication for Deployment Using Unicast Communication

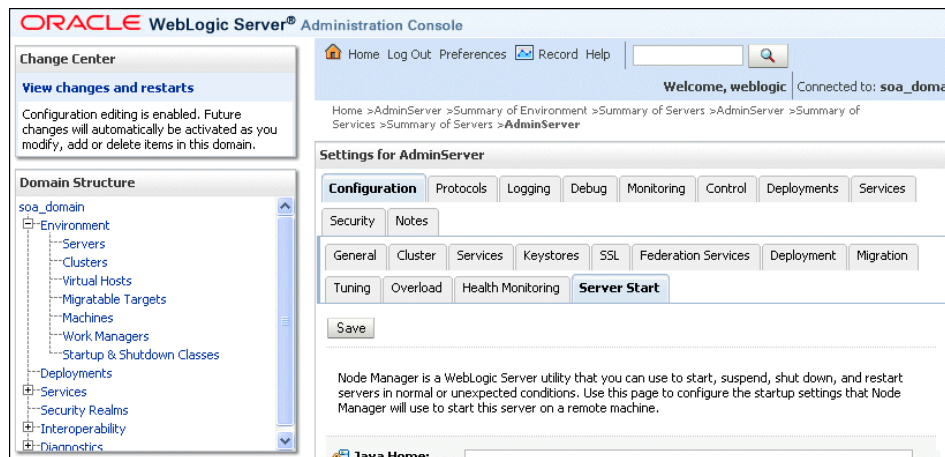
Specify the nodes using the `tangosol.coherence.wka<n>` system property, where `<n>` is a number between 1 and 9. You can specify up to nine nodes as Well Known Addresses, but you can have more than nine nodes in the cluster. Start the numbering at 1. This numbering must be sequential and must not contain gaps. In addition,

specify the host name used by Oracle Coherence to create a cluster through the `tangosol.coherence.localhost` system property. This local host name should be the virtual host name used by the SOA server as the listener addresses (SOAHOST1-PRIV-V1 and SOAHOST2-PRIV-V1). Set this property by adding the `-Dtangosol.coherence.localhost` parameters to the Arguments field of the Oracle WebLogic Server Administration Console's Server Start tab (Figure 9–4).

Tip: To guarantee high availability during deployments of SOA composites, specify enough nodes so that at least one of them is running at any given time.

Note: SOAHOST1-PRIV-V1 is the virtual host name that maps to the virtual IP where WLS_SOA1 is listening (in SOAHOST1). SOAHOST2-PRIV-V1 is the virtual host name that maps to the virtual IP where WLS_SOA2 is listening (in SOAHOST2).

Figure 9–2 Setting the Host Name Using the Start Server Tab of Oracle WebLogic Server Administration Console



9.4.2 Specifying the Host Name Used by Oracle Coherence

Use the Administration Console to specify a host name used by Oracle Coherence.

To add the host name used by Oracle Coherence:

1. Log into the Oracle WebLogic Server Administration Console using the following URL:
`http://ADMINVHN:7001/console`
2. In the Domain Structure window, expand the **Environment** node.
3. Click **Servers**. The Summary of Servers page appears.
4. Click the name of the server (**WLS_SOA1** or **WLS_SOA2**, which are represented as hyperlinks) in the **Name** column of the table. The settings page for the selected server appears.
5. Click **Lock & Edit**.
6. Click the **Server Start** tab (shown in Figure 9–2).
7. Enter the following for WLS_SOA1 and WLS_SOA2 into the Arguments field.

Note: There should be no breaks in lines between the different `-D` parameters. Do not copy or paste the text to your Administration Console's arguments text field. It may result in HTML tags being inserted in the Java arguments. The text should not contain other text characters than those included the example below.

Note: The Coherence cluster used for deployment uses port 8088 by default. You can change this port by specifying a different port (for example, 8089) with the `-Dtangosol.coherence.wkan.port` and `-Dtangosol.coherence.localport` startup parameters. For example:

WLS_SOA1 (enter the following into the Arguments field on a single line, without a carriage return):

```
-Dtangosol.coherence.wka1=SOAHOST1-PRIV-V1
-Dtangosol.coherence.wka2=SOAHOST2-PRIV-V1
-Dtangosol.coherence.localhost=SOAHOST1-PRIV-V1
-Dtangosol.coherence.localport=8089
-Dtangosol.coherence.wka1.port=8089
-Dtangosol.coherence.wka2.port=8089
```

WLS_SOA2 (enter the following into the Arguments field on a single line, without a carriage return):

```
-Dtangosol.coherence.wka1=SOAHOST1-PRIV-V1
-Dtangosol.coherence.wka2=SOAHOST2-PRIV-V1
-Dtangosol.coherence.localhost=SOAHOST2-PRIV-V1
-Dtangosol.coherence.localport=8089
-Dtangosol.coherence.wka1.port=8089
-Dtangosol.coherence.wka2.port=8089
```

For more information about Coherence Clusters see the *Oracle Coherence Developer's Guide*.

For WLS_SOA1, enter the following:

```
-Dtangosol.coherence.wka1=SOAHOST1-PRIV-V1
-Dtangosol.coherence.wka2=SOAHOST2-PRIV-V1
-Dtangosol.coherence.localhost=SOAHOST1-PRIV-V1
```

For WLS_SOA2, enter the following:

```
-Dtangosol.coherence.wka1=SOAHOST1-PRIV-V1
-Dtangosol.coherence.wka2=SOAHOST2-PRIV-V1
-Dtangosol.coherence.localhost=SOAHOST2-PRIV-V1
```

8. Click **Save** and **Activate Changes**.

Note: You must ensure that these variables are passed to the managed server correctly. (They should be reflected in the server's output log.) Failure of the Oracle Coherence framework can prevent the soa-infra application from starting.

Note: The multicast and unicast addresses are different from the ones used by the WebLogic Server cluster for cluster communication. SOA guarantees that composites are deployed to members of a single WebLogic Server cluster even though the communication protocol for the two entities (the WebLogic Server cluster and the groups to which composites are deployed) are different.

9.5 Post-Configuration and Verification Tasks

After extending the domain with the configuration Wizard and configuring Oracle Coherence, follow these instructions for post-configuration and validation.

This section includes the following topics:

- [Section 9.5.1, "Disabling Host Name Verification for the WLS_SOAn Managed Servers"](#)
- [Section 9.5.2, "Restarting the Node Manager on SOAHOST1"](#)
- [Section 9.5.3, "Propagating the Domain Changes to the Managed Server Domain Directory"](#)
- [Section 9.5.4, "Starting and Validating the WLS_SOA1 Managed Server"](#)
- [Section 9.5.5, "Propagating the Domain Configuration to SOAHOST2 Using the unpack Utility"](#)
- [Section 9.5.6, "Starting and Validating the WLS_SOA2 Managed Server"](#)
- [Section 9.5.7, "Validating GridLink Data Sources"](#)

9.5.1 Disabling Host Name Verification for the WLS_SOAn Managed Servers

For the Exalogic enterprise deployment described in this guide, you set up the appropriate certificates to authenticate the different nodes with the Administration Server after you have completed the procedures to extend the domain for Oracle SOA Suite. You must disable the host name verification for the WLS_SOA1 and WLS_SOA2 managed servers to avoid errors when managing the different WebLogic Server instances. For more information, see [Chapter 8.16.3, "Disabling Host Name Verification for the WLS_WSM2 Managed Server."](#)

You enable host name verification again once the Exalogic enterprise deployment topology configuration is complete. For more information, see [Chapter 12.3, "Enabling Host Name Verification Certificates for Node Manager."](#)

9.5.2 Restarting the Node Manager on SOAHOST1

Use the `startNodeManager.sh` script to restart Node Manager from the private directory for Node Manager in each compute node.

To restart the Node Manager on SOAHOST1:

1. Stop Node Manager by stopping the process associated with it:
 - If it is running in the foreground in a shell, simply use **CTRL+C**.
 - If it is running in the background in the shell, find the associate process and use the `kill` command to stop it. For example:

```
ps -ef | grep NodeManager
orcl      9139  9120  0 Mar03 pts/6    00:00:00 /bin/sh
```



```
./startNodeManager.sh
```

```
kill -9 9139
```

2. Start Node Manager:

```
./startNodeManager.sh
```

9.5.3 Propagating the Domain Changes to the Managed Server Domain Directory

Propagate the start scripts and classpath configuration from the Administration Server's domain directory to the managed server domain directory.

To propagate start scripts and classpath configuration:

1. Create a copy of the managed server domain directory and the managed server applications directory.
2. Run the `pack` command on SOAHOST1 to create a template pack using the following commands:

```
cd ORACLE_COMMON_HOME/common/bin
```

```
./pack.sh -managed=true -domain=ASERVER_HOME
```

```
-template=soadomaintemplateExtSOA.jar -template_name=soa_domain_templateExtSOA
```

3. Run the `unpack` command on SOAHOST1 to unpack the propagated template to the domain directory of the managed server using the following command:

```
./unpack.sh -domain=MSERVER_HOME
```

```
-overwrite_domain=true -template=soadomaintemplateExtSOA.jar
```

```
-app_dir=APP_DIR
```

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.

Note: The configuration steps provided in this Exalogic enterprise deployment topology are documented with the assumption that a private (per node) domain directory is used for each managed server.

9.5.4 Starting and Validating the WLS_SOA1 Managed Server

Before starting the WLS_SOA1 managed server please make sure the WLS__WSM1 managed server is up and running. Otherwise WLS_SOA1 will not start.

Start and validate the WLS_SOA1 managed server using the Administration Console.

To start the WLS_SOA1 managed server on SOAHOST1:

1. Start the WLS_SOA1 managed server using the Oracle WebLogic Server Administration Console as follows:
 - a. Access the Administration Console at the following URL:

```
http://ADMINVHN:7001/console
```

ADMINVHN is the virtual host name that maps to the virtual IP where the Administration Server is listening (in SOAHOST1).

- b. Expand the **Environment** node in the **Domain Structure** window.
- c. Click **Servers**.

The Summary of Servers screen appears.

- d. Click the **Control** tab.
 - e. Select **WLS_SOA1** and then click **Start**.
2. Verify that the server status is reported as **Running**. 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.
 3. Access the following URL to verify status of WLS_SOA1:

<http://SOAHOST1-PRIV-V1:8001/soa-infra/>

Note: Because SOAHOST1-PRIV-V1 is a private/infiniband address, you must use browser from within the Exalogic cells to open the <http://SOAHOST1-PRIV-V1:8001/soa-infra/> URL.

Access the following URL to verify the status of B2B:

<http://SOAHOST1-PRIV-V1:8001/b2bconsole/>

Access the following URL to verify status of the worklist application:

<http://SOAHOST1-PRIV-V1:8001/integration/worklistapp/>

Access the following URL to verify status of the composer application:

<http://SOAHOST1-PRIV-V1:8001/soa/composer/>

Before verifying access is granted, ensure that the WLS_WSM1 managed server is up and running.

9.5.5 Propagating the Domain Configuration to SOAHOST2 Using the unpack Utility

Propagate the domain you just configured to SOAHOST2 using the unpack utility.

To propagate the domain configuration:

1. Run the following command on SOAHOST1 to copy the template file created in the previous step to SOAHOST2.

```
cd ORACLE_COMMON_HOME/common/bin
```

```
scp soadomaintemplateExtSOA.jar oracle@SOAHOST2:ORACLE_COMMON_HOME/common/bin
```

2. Run the `unpack` command on SOAHOST2 to unpack the propagated template.

```
cd ORACLE_COMMON_HOME/common/bin
```

```
/unpack.sh
```

```
-domain=MSERVER_HOME
```

```
-template=soadomaintemplateExtSOA.jar -overwrite_domain=true
```

```
-app_dir=APP_DIR
```

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.

Note: The configuration steps provided in this Exalogic enterprise deployment topology are documented with the assumption that a private (per node) domain directory is used for each managed server.

9.5.6 Starting and Validating the WLS_SOA2 Managed Server

Use the Administration Console to start the WLS_SOA2 managed server. Validate it by accessing `soa-infra`, `b2bconsole`, and `worklistapp` URLs.

To start the WLS_SOA2 managed server and check that it is configured correctly:

1. Start the WLS_SOA2 managed server using the Administration Console.
2. 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.
3. Access the following URL for `soa-infra`:

```
http://SOAHOST2-PRIV-V1:8001/soa-infra
```

4. Access the following URL to verify status of B2B:

```
http://SOAHOST2-PRIV-V1:8001/b2bconsole
```

5. Access the following URL to verify status of the worklist application.

```
http://SOAHOST2-PRIV-V1:8001/integration/worklistapp/
```

Before verifying access is granted, ensure that at least one of the managed servers (WLS_WSM1 or WLS_WSM2) is up and running.

Note: Although the WLS_SOA1 server may be up, some applications may be in a failed state. Therefore, Oracle recommends verifying the URLs above and watch for errors pertaining each individual application in the server's output file.

6. Access the following URL to verify status of the composer application.

```
http://SOAHOST2-PRIV-V1:8001/soa/composer/
```

9.5.7 Validating GridLink Data Sources

When the servers are started, verify that the GridLink data sources are correctly configured and that the ONS setup is correct. Perform this procedure for every GridLink data source created.

To validate the GridLink data sources configuration:

1. Log on to the Oracle WebLogic Administration Console.

```
http://ADMINVHN:7001/console
```
2. In the **Domain Structure** tree, expand **Services**, and select **Data Sources**.
3. Click one of the new data sources.
4. Click the **Monitoring** tab and select one of the servers.
5. Click the **Statistics** tab and select one of the servers.
6. Click the **ONS** tab, and then click the **Testing** tab.
7. Select the server and click **Test ONS**.

If both tests are successful, the configuration is correct. If the ONS test fails, verify that the ONS service is running in the RAC database nodes:

```
orcl@db-scan1 ~]$ srvctl status scan_listener
SCAN Listener LISTENER_SCAN1 is enabled
SCAN listener LISTENER_SCAN1 is running on node db-scan1
SCAN Listener LISTENER_SCAN2 is enabled
SCAN listener LISTENER_SCAN2 is running on node db-scan2
SCAN Listener LISTENER_SCAN3 is enabled
SCAN listener LISTENER_SCAN3 is running on node db-scan2

[orcl@db-scan1 ~]$ srvctl config nodeapps -s
ONS exists: Local port 6100, remote port 6200, EM port 2016

[orcl@db-scan1 ~]$ srvctl status nodeapps | grep ONS
ONS is enabled
ONS daemon is running on node: db-scan1
ONS daemon is running on node: db-scan2
```

Run the ONS test from every WebLogic server that uses the data source.

9.6 Configuring Network Channels for HTTP and T3 Clients Through EoIB

If HTTP/RMI clients must access the SOA servers directly, for example, deployment of a composite from JDEV, Direct-Binding client directly invoking Direct-Binding service. For more information, see [Chapter 3, "Configuring the Network for an Exalogic Enterprise Deployment."](#)

9.6.1 Configuring Network Channels for SOA Servers on SOAHOST1 and SOAHOST2

For Managed Servers, you must create the following network channels using Ethernet over InfiniBand (EoIB):

- [Section 9.6.1.1, "Creating an HTTP Client Channel"](#)
- [Section 9.6.1.2, "Creating the T3 Client Channel"](#)

9.6.1.1 Creating an HTTP Client Channel

To create a HTTP network channel for a Managed Server such as WLS1, complete the following steps:

1. Go to the following URL:
`http://ADMINVHN1:7001/console`
2. Log in as the administrator.
3. Click **Lock & Edit** in the Change Center if you have not already done so.
4. In the left pane of the Administration Console, expand **Environment** and then **Servers** to open the Summary of Servers page.
5. In the Servers table, click **WLS_SOA1** to open the WLS_SOA1 page.
6. Select **Protocols** and then **Channels**. Click **New**.
7. Enter **SOA_HTTPChannel** as the name of the new network channel and select **http** as the protocol. Click **Next**.
8. Enter the following information in the Network Channel Addressing page:

Table 9–5 Network Channel Addressing Page

Field	Value	Notes
Listen address	SOAHOST1VHN1	This address is the virtual hostname assigned to the WLS_SOA1 Server using the BOND1 interface.
Listen port	8001	
External Listen Address	soa.mycompany.com	This is the DNS name to access application on the server.
External Listen Port	443	

9. Click **Next** and select the following in the Network Channel Properties page:
 - Enabled
 - HTTP Enabled for This Protocol
10. Click **Finish**.
11. Click **Activate Changes** in the Change Center of the Administration Console to activate these changes.

You must repeat the preceding steps to create a network channel for the WLS_SOA2 Managed Servers on SOAHOST2 and enter the required properties in [Table 9–6](#) describes.

Table 9–6 Network Channels Properties

Managed Server	Name	Protocol	Listen Address	Listen Port	External Listen Address	External Listen Port ¹
WLS_SOA1	SOA_HTTPChannel	HTTP	SOAHOST1VHN1	8001	soa.mycompany.com	443
WLS_SOA2	SOA_HTTPChannel	HTTP	SOAHOST2VHN1	8001	soa.mycompany.com	443

¹ In the Configuration tab and General subtab, this port is referred to as the External Listen Port. In the Protocols tab and Channels subtab, this port is listed in the Network Channels table as the Public Port.

Note: After these channels are created, the server is accessible from a browser outside the Exalogic rack using the appropriate EoIB addresses (SOAHOST1VHN1:8001/soa-infra and SOAHOST2VHN1:8001/soa-infra).

9.6.1.2 Creating the T3 Client Channel

To create the T3network channel for the SOA Managed Servers, complete the following steps:

1. Go to the following URL:
`http://ADMINVHN1:7001/console`
2. Log in as the administrator.
3. Click **Lock & Edit** in the Change Center if you have not already done so.
4. In the left pane of the Administration Console, expand **Environment** and then **Servers** to open the Summary of Servers page.
5. In the Servers table, click **WLS_SOA1** to open the WLS_SOA1 page.
6. Select **Protocols** and then **Channels**. Click **New**.
7. Enter **SOA_T3_Channel** as the name of the new network channel and select **t3** as the protocol. Click **Next**.
8. Enter the following information in the Network Channel Addressing page:

Table 9–7 Network Channel Addressing Page

Field	Value	Notes
Listen address	SOAHOST1VHN1	This address is the virtual hostname assigned to the WLS_SOA1 Server using the BOND1 interface.
Listen port	8003	Remove the default external listen port value.

9. Click **Next** and select the following in the Network Channel Properties page:
 - **Enabled**
 - **HTTP Enabled for This Protocol**
10. Click **Finish**.
11. Click **Activate Changes** in the Change Center of the Administration Console to activate these changes.

Note: You must repeat the preceding steps to create a network channel for the WLS_SOA2 Managed Servers on SOAHOST2 and enter the required properties that [Table 9–8](#) describes.

Table 9–8 Network Channels Properties

Managed Server	Name	Protocol	Listen Address	Listen Port	External Listen Address	External Listen Port ¹
WLS_SOA1	SOA_T3_Channel	T3 Channel	SOAHOST1VHN1	8003	SOAHOST1VHN1	8003
WLS_SOA2	SOA_T3_Channel	T3 Channel	SOAHOST2VHN1	8003	SOAHOST2VHN1	8003

¹ In the Configuration tab and General subtab, this port is referred to as the External Listen Port. In the Protocols tab and Channels subtab, this port is listed in the Network Channels table as the Public Port.

9.7 Configuring Oracle Traffic Director with the Extended Domain

After you create the appropriate channels, you must configure Oracle Traffic Director to route to the SOA servers for the appropriate URLs.

Note: You created a virtual server for `soa.mycompany.com` and `soainternal.mycompany.com` in [Chapter 7, "Installing and Configuring Oracle Traffic Director for an Exalogic Enterprise Deployment."](#)

This section includes the following topics:

- [Section 9.7.1, "Configuring Access Through Oracle Traffic Director for the WLS_SOA \$n\$ Managed Servers"](#)
- [Section 9.7.2, "Validating Access Through Oracle Traffic Director"](#)
- [Section 9.7.3, "Setting Server and HTTP URLs for SOA Servers"](#)

9.7.1 Configuring Access Through Oracle Traffic Director for the WLS_SOA n Managed Servers

To create the required virtual server routes:

1. Log into the Administration Console using the following URL:

`https://OTDADMINVHN.mycompany.com:8989`

2. Click **Configurations** at the upper left corner of the page to view a list of available configurations.
3. Select the configuration that you want to configure routes for.
4. In the Navigation pane, expand **Virtual Servers** and the **SOAEXA** virtual server. Select **Routes** to open a list of routes that are defined for the virtual server.

Continue to [Section 9.7.1.1, "Creating a New Route"](#)

9.7.1.1 Creating a New Route

To enable Oracle HTTP Server to route to the SOA_Cluster:

1. Click **New Route** to open the New Route dialog box.
2. In the Step 1: Route Properties screen, enter `soa-route` in the **Name** field.
3. In the Origin Server Pool drop-down menu, select `origin-server-pool-1` and click **Next**.

4. In the Step2: Condition Information screen, select the variable \$uri from the Variable/Function drop-down list. Select = ~ in the Operator drop-down menu, then enter /soa-infra in the Value field.

Note: You cannot use a joiner (and/or) for the first expression in the sequence.

5. Select **OK** then select **Plus** to add the next expression.

Note: You can now select the joiner **or**.

6. Select **uri** as the Variable /Function, = ~ as the Operator, and **inspection.wsil** as the Value. Select **OK**.
7. Add the rest of the conditions using the information in the preceding step.

Table 9–9 Routes and Conditions

Route	Origin: Server Pool	Conditions
soa-route	origin-server-pool-1	'/soa-infra' or \$uri =~ '/inspection.wsil' or \$uri =~ '/integration' or \$uri =~ '/b2bconsole' or \$uri =~ '/b2b/services/ws' or \$uri =~ '/sdpMessaging/userprefs-ui' or \$uri =~ '/DefaultToDoTaskFlow' or \$uri =~ '/workflow' or \$uri =~ '/ADFAttachmentHelper' or \$uri =~ '/soa/composer' or \$uri =~ '/frevvo'

8. Click **Next** and then **Create Route**.

Your route appears on the Routes and the Deployment Pending message appears in the main pane.

The new route appears on the Routes page and a Deployment Pending message appears in the main pane. You can deploy the updated configuration immediately by selecting **Deploy Changes** or wait until you make changes. See the topic [Section 7.8, "Deploying the Configuration and Testing the Virtual Server Addresses"](#) for more information.

To create a new route in the soainternal virtual server, repeat the preceding steps and conditions. You can copy the rules from the first route you created and use origin-server-pool-1.

9.7.2 Validating Access Through Oracle Traffic Director

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. See [Section 14.14, "Troubleshooting the Topology in an Enterprise Deployment."](#) for possible causes.

Verify that you can access these URLs, where 'webhostN' specifies the name of each Oracle Traffic Director host. Check these URLs for both WEBHOST1 and WEBHOST2):

- <http://webhostN-priv-v1:7777/soa-infra>

- `http://webhostN-priv-v1:7777/integration/worklistapp`
- `http://webhostN-priv-v1:7777/b2bconsole`
- `http://webhostN-priv-v1:7777/sdpmessaging/userprefs-ui`
- `http://webhostN-priv-v1:7777/soa/composer`

Validate SOA_Cluster through both Oracle Traffic Director instances.

For information on configuring system access through the load balancer, see [Section 3.7, "Configuring the Load Balancer."](#)

Note: The `webhostn-priv-v1` addresses are internal to the Exalogic rack; you must start a browser from a node within the rack itself.

In addition, verify that you can access SOA using the external/EoIB addresses exposed by the OTD servers. Use the URLs, where `webhostN` specifies the name of each Oracle Traffic Director hosts. Check the following URLs for both WEBHOST1 and WEBHOST2):

- `http://WEBHOSTnVHN1:7777/soa-infra`
- `http://WEBHOSTnVHN1:7777/integration/worklistapp`
- `http://WEBHOSTnVHN1:7777/b2bconsole`
- `http://WEBHOSTnVHN1:7777/sdpmessaging/userprefs-ui`
- `http://WEBHOSTnVHN1:7777/soa/composer`

9.7.3 Setting Server and HTTP URLs for SOA Servers

To make the system use the infiniband and web services local optimization, you must use the correct Server URL and Oracle Traffic Director URL properties in the SOA Infrastructure. To do this you must set both the SERVER and HTTP Server URLs to point to Oracle Traffic Director's virtual server (infiniband address). This section describes the procedures to set both URLs.

To set the Server URL to point to Oracle Traffic Director's virtual server (infiniband address):

1. Log in to Oracle Enterprise Manager Fusion Middleware Control with the username and password you specified in [Section 8.5.1, "Creating boot.properties for the Administration Server on SOAHOST1."](#)
2. On the navigation tree on the left, expand **Farm_domain_name, SOA**. Right click **soa-infra (WLS_SOA1), SOA Administration**, and then **Common Properties**.
3. In the Server URL field, enter the virtual hostname set up in Oracle Traffic Director as a load balancer entry point on IP over InfiniBand (IPoIB):

`http://webhost1-priv-v1:7777`

4. Select **Apply**.

Note: The changes are automatically propagated to the other managed server (WLS_SOA2) at the SOA cluster level, because these MBeans are shared for all managed servers on the cluster.

To set the HTTP Server URL property point to Oracle Traffic Director's virtual server (Infiniband address:)

1. Log into Fusion Middleware Control with the username and password you specified in [Section 8.5.1, "Creating boot.properties for the Administration Server on SOAHOST1."](#)
2. On the navigation tree on the left, expand `Farm_domain_name`, SOA. Right click on the `soa-infra (Server_name)` **SOA Administration**, then **Common Properties**.
3. Select the link **More SOA Infra Advanced Configuration Properties...** at the bottom of the page.
4. In the **HTTPServerURL** field, enter the same values that you entered for the **Server URL** field in the Server URL procedure.
5. Select **Apply**.

Callbacks now route through Infiniband using the Oracle Traffic Director instance instead of the external load balancer. Also, the SOA Service Engines use Web services local optimizations.

9.7.3.1 Webservice Local Optimization

For webservice local optimization, the basic requirement is to make sure that the two SOA composites are co-located on the same server/process. For that determination, SOA 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 of 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.

9.8 Configuring a Default Persistence Store for Transaction Recovery

Each server has a transaction log that stores information about committed transactions that are coordinated by the server that may not have been completed. The 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 servers within a cluster, store the transaction log in a location accessible to a server and its backup servers.

To set the location for the default persistence stores:

1. Log on to the Oracle WebLogic Administration Console.

```
http://ADMINVHN:7001/console
```

2. In the Change Center section, click **Lock & Edit**.
3. In the Domain Structure window, expand the **Environment** node and then click the **Servers** node. The Summary of Servers page appears.
4. Click the **WLS_SOA1** (represented as a hyperlink) in **Name** column of the table. The settings page appears and defaults to the **Configuration** tab.
5. Click the **Configuration** tab, and then the **Services** tab.
6. In the Default Store section of the page, enter the path to the folder where the default persistent stores will store its data files. The directory structure of the path is as follows:

```
ASERVER_HOME/tlogs
```

7. Repeat steps 3, 4, 5, and 6 for the WLS_SOA2 server.
8. Click **Save** and **Activate Changes**.
9. Restart both SOA servers.
10. Verify that the following files are created in the following directory after WLS_SOA1 and WLS_SOA2 are restarted:

```
ASERVER_HOME/tlogs
```

- `_WLS_WLS_SOA1000000.DAT`
- `_WLS_WLS_SOA2000000.DAT`

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. Both WLS_SOA1 and WLS_SOA2 must be able to access this directory. This directory must also exist before you restart the server.

9.9 Configuring Coherence Caches for Dehydrations

This release of Oracle Fusion Middleware permits using Oracle Coherence Caches for optimized performance in BPEL object access. If you configure the BPEL engine with the `CacheEnabled` property, the engine runs many fewer reads against database. Because many reads require locks and version checks, eliminating them improves BPEL engine performance substantially. Configuring Oracle Coherence for dehydration requires the following steps:

- [Section 9.9.1, "Enabling the CacheEnabled property"](#)

- [Section 9.9.2, "Setting Server Properties for In-Process Coherence Cache for Dehydration"](#)

9.9.1 Enabling the CacheEnabled property

To enable the cache enabled property in SOA, follow these steps:

1. Log in to Oracle Enterprise Manager Fusion Middleware Control with the username and password you specified in [Section 8.5.1, "Creating boot.properties for the Administration Server on SOAHOST1."](#)
2. On the navigation tree on the left, expand **Farm_domain_name, SOA**, and then right click **soa-infra (server_name)** (either one of the servers).
3. Click **SOA Administration**, and then **BPEL Properties**.
4. Select **More BPEL Configuration properties**.
5. Enter `CacheEnabled` for the property **QualityOfService** property.
6. Select **Apply**.

9.9.2 Setting Server Properties for In-Process Coherence Cache for Dehydration

Set the required server properties for using in-process coherence cache for dehydration.

To enable the cache enabled property in SOA:

1. Log into the Oracle WebLogic Server Administration Console using the following URL:

```
http://ADMINVHN:7001/console
```

2. In the **Domain Structure** window, expand the **Environment** node.
3. Click **Servers**.

The Summary of Servers page appears.

4. Click the name of the server (**WLS_SOA1** or **WLS_SOA2**, which are represented as hyperlinks) in the **Name** column of the table.

The settings page for the selected server appears.

5. Click **Lock & Edit**.
6. Click the **Server Start** tab.
7. In the **Arguments** field, add the following for **WLS_SOA1** and **WLS_SOA2**

```
-Dbpel.cache.localStorage=true  
-Dbpel.cache.threadCount=20  
-Dbpel.cache.cubeInstance.sizeLimit=4g  
-Dbpel.cache.invokeMessage.sizeLimit=2g  
-Dbpel.cache.deliveryMessage.sizeLimit=2g  
-Dbpel.cache.deliverySubscription.sizeLimit=2g
```

8. Click **Save and Activate Changes**.

9.10 Updating SOA JVM settings

To optimize behavior with local storage caches and improve performance, update the SOA servers' start parameters to include the following flags:

```
-Djava.net.preferIPv4Stack=true
-Xlargepages:exitOnFailure=true
-Doracle.xdkjava.exalogic.optimization=true
-Xms16g -Xmx16g
```

To update the SOA servers' start parameters:

1. Log into the Oracle WebLogic Server Administration Console using the following URL:

```
http://ADMINVHN:7001/console
```

2. In the **Domain Structure** window, expand the **Environment** node.
3. Click **Servers**.
The Summary of Servers page appears.
4. Click the name of the server (**WLS_SOA1** or **WLS_SOA2**, which are represented as hyperlinks) in the **Name** column of the table.
The settings page for the selected server appears.
5. Click **Lock & Edit**.
6. Click the **Server Start** tab.
7. Add the following in the **Arguments** field for WLS_SOA1 and WLS_SOA2:

```
-Djava.net.preferIPv4Stack=true
-Xlargepages:exitOnFailure=true
-Doracle.xdkjava.exalogic.optimization=true
-Xms16g -Xmx16g
```

Note: There should be no breaks in lines between the different -D parameters. Do not copy or paste the text to your Administration Console's arguments text field. It may result in HTML tags being inserted in the Java arguments. The text should not contain other text characters than those included in the example.

8. Click **Save and Activate Changes**.
9. Apply these changes in both servers (WLS_SOA1 and WLS_SOA2).

9.11 Enabling Cluster-Level Session Replication Enhancements

You can enable session replication enhancements for the SOA servers if stateful applications, such as SOA's Composer, or B2B console are going to be used.

To enable session replication enhancements for the SOA_Cluster:

1. Shut down the servers in the SOA_Cluster.
2. To set replication ports for a managed server, such as WLS_SOA1:
 - a. Under **Domain Structure**, click **Environment and Servers**. The Summary of Servers page appears.
 - b. Click **WLS_SOA1** on the list of servers. The Settings for WLS_SOA1 appears.
 - c. Under the **Configuration** tab, click the **Cluster** tab.

- d. In the **Replication Ports** field, enter a range of ports for configuring multiple replication channels. For example, replication channels for managed servers in `SOA_Cluster` can listen on ports starting from 8006 to 8009. To specify this range of ports, enter **8006-8009**.
 - e. Repeat these steps for `WLS_SOA2`.
3. Create a custom network channel for each managed server in the cluster (for example, `WLS_SOA1`) as follows:
- a. Log on to the Oracle WebLogic Administration Console.
`http://ADMINVHN:7001/console`
 - b. If you have not already done so, click **Lock & Edit** in the **Change Center**.
 - c. In the left pane of the console, expand **Environment** and select **Servers**.
The Summary of Servers page appears.
 - d. In the **Servers** table, click `WLS_SOA1` managed server instance.
 - e. Select **Protocols**, and then **Channels**, and click **New**.
 - f. Enter the following information:
Listen Address: `SOAHOST1-PRIV-V1`
-
- Note:** This is the floating IP assigned to `WLS_SOA1`.
-
- Listen port: 8006
Remove the default external listen port.
- g. Click **Next**, and in the Network Channel Properties page, select **Enabled** and **Outbound Enabled**, then click **Finish**.
 - h. Under the **Network Channels** table, select **ReplicationChannel**, the network channel you created for the `WLS_SOA1` managed server.
 - i. Expand **Advanced**, and select **Enable SDP Protocol** and click **Save**.
 - j. To activate these changes, in the Change Center of the Administration Console, click **Activate Changes**.

You must repeat these steps to create a network channel each for the remaining managed servers in the `SOA_Cluster` cluster. Enter the required properties, as [Table 9–10](#) describes.

Table 9–10 Network Channel Properties

Managed Server in SOA_Cluster	Name	Protocol	Listen Address	Listen Port	Additional Channel Ports
WLS_SOA1	ReplicationChannel	t3	SOAHOST1-PR IV-V1	8006	8006 to 8009
WLS_SOA2	ReplicationChannel	t3	SOAHOST2-PR IV-V1	8006	8006 to 8009

4. After creating the network channel for each of the managed servers in your cluster, click **Environment** and then **Clusters**.
The Summary of Clusters page appears.

5. Click **SOA_Cluster**.
The Settings for SOA_Cluster page appears.
6. Click the **Replication** tab.
7. In the **Replication Channel** field, ensure that `ReplicationChannel` is set as the name of the channel to be used for replication traffic.
8. In the **Advanced** section, select the **Enable One Way RMI for Replication** option., and click **Save**.
9. Activate the changes and restart the managed servers.
10. Add the system property `-Djava.net.preferIPv4Stack=true` to the start parameters for the WebLogic servers
 - a. Log on to the Oracle WebLogic Administration Console.
`http://ADMINVHN:7001/console`
 - b. In the **Domain Structure** window, expand the **Environment** node.
 - c. Click **Servers** to open the Summary of Servers page.
 - d. Click the name of the server (**WLS_SOA1** or **WLS_SOA2**, which are represented as hyperlinks) in the **Name** column of the table.
The settings page for the selected server appears.
 - e. Click **Lock & Edit**.
 - f. Enter `ReplicationChannel` as the name of the new network channel and select **t3** as the protocol, then click **Next**.
 - g. Click the **Server Start** tab.
 - h. Add the following for **WLS_SOA1** and **WLS_SOA2** into the **Arguments** field:
`-Djava.net.preferIPv4Stack=true`
 - i. Click **Save** and **Activate Changes**.

Note: To verify that multiple listening ports were opened, you can either run the `netstat -na` command on the command line or check the managed server logs.

Note: These changes require restarting the SOA servers to be effective.

9.12 Configuring Oracle Adapters

Configure Oracle File, FTP, and database adapters for the extended SOA domain.

This section includes the following topics:

- [Section 9.12.1, "Enabling High Availability for Oracle File and FTP Adapters"](#)
- [Section 9.12.2, "Enabling High Availability for Oracle JMS Adapters"](#)
- [Section 9.12.3, "Scaling the Oracle Database Adapter"](#)

9.12.1 Enabling High Availability for Oracle File and FTP Adapters

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 FTP (File Transfer Protocol). 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. To make Oracle File and FTP Adapters highly available for outbound operations, use the database mutex locking operation as described in "High Availability in Outbound Operations" in the *Oracle Fusion Middleware User's Guide for Technology Adapters*. The database mutex locking operation enables these adapters to ensure that multiple references do not overwrite one another if they write to the same directory. For inbound operations, the database is used also to prevent duplication scenarios where multiple instances read the same input file.

Note: The operations described in this section are necessary only if your application requires these adapters.

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.

9.12.1.1 Using the Database Mutex Locking Operation

Make an outbound Oracle File or FTP Adapter service highly available using database table as a coordinator.

Note: The steps and configuration options for the FTP adapter are exactly the same as the options for the file adapter. The connection factory to be used for FTP HA configuration is `eis/Ftp/HAFtpAdapter` which appears under the Outbound Connection Pools for the FTPAdapter deployment.

Note: If you use database as a coordinator, increase global transaction timeouts.

To make outbound Oracle File or FTP Adapters highly available, modify Oracle File Adapter deployment descriptor for the connection-instance corresponding to `eis/HFileAdapter` from the Oracle WebLogic Server console:

1. Log on to the Oracle WebLogic Administration Console.
`http://ADMINVHN:7001/console`
2. Click **Deployments** in the left pane for Domain Structure.
3. Click **FileAdapter** under Summary of Deployments on the right pane.
4. Click the **Configuration** tab.
5. Click the **Outbound Connection Pools** tab and expand **javax.resource.cci.ConnectionFactory** to see the configured connection factories.

6. Click **eis/HAFileAdapter**. The Outbound Connection Properties for the connection factory corresponding to high availability appears.
7. Click on **Lock & Edit**. The property value column becomes editable; you can click on any of the rows under **Property Value** and modify its value.

[Table 9–11](#) describes new parameters in connection factory for Oracle File and FTP Adapters:

Table 9–11 Oracle File and FTP Adapters

Parameter	Description
controlDir	<p>Set it to the directory structure 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:</p> <pre>ASERVER_HOME/fadapter</pre> <p>The directory <code>ASERVER_HOME/fadppter</code> must already exist.</p>
inboundDataSource	<p>Set the value to <code>jdbc/SOADataSource</code>. This is the data source, where the schemas corresponding to high availability are pre-created. The pre-created schemas can be found in the following directory:</p> <pre>ORACLE_HOME/rcu/integration/soainfra/sql/createschema_soainfra_oracle.sql</pre> <p>If you want to create the schemas elsewhere, use this script. You must set the <code>inboundDataSource</code> property accordingly if you choose a different schema.</p>
outboundDataSource	<p>Set the value to <code>jdbc/SOADataSource</code>. This is the data source where the schemas corresponding to high availability are pre-created. The pre-created schemas are located in the directory:</p> <pre>ORACLE_HOME/rcu/integration/soainfra/sql/adapter/createschema_adapter_oracle.sql</pre> <p>To create the schemas elsewhere, use this script. You must set the <code>outboundDataSource</code> property if you choose to do so.</p>
outboundDataSourceLocal	<p>Set the value to <code>jdbc/SOALocalTxDataSource</code>. This is the data source where the schemas corresponding to high availability are pre-created.</p>

Table 9–11 (Cont.) Oracle File and FTP Adapters

Parameter	Description
outboundLockTypeForWrite	<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.
workingDirectory	Use "default" for working directory.

8. Click **Save** after you update the properties. The Save Deployment Plan page appears.
9. Enter a shared storage location for the deployment plan. The directory structure is as follows:

```
ORACLE_BASE/config/dp/soaedg_domain/FileAdapterPlan.xml
```

10. Click **OK**. Then click **Save**, and **Activate Changes**.
11. Once the new deployment plan is saved and activated, activate the FileAdapter deployment. The deployment remains in **Prepared** state if not started. To activate the FileAdapter deployment plan:

In the Administration Console, click **Deployments** in the left pane for **Domain Structure**.

Select the FileAdapter under **Summary of Deployments** on the right pane and Select **Start**, and then **Servicing All Requests**.

12. Configure BPEL Process or Mediator Scenario to use the connection factory as shown in the following example (in the `jca` file included in the composite for the binding component):

```
<adapter-config name="FlatStructureOut" adapter="File Adapter"
xmlns="http://platform.integration.oracle/blocks/adapter/fw/metadata">
  <connection-factory location="eis/HAFileAdapter" 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>
```

Note: The location attribute is set to `eis/HADFileAdapter` for the connection factory.

Note: Perform the same steps for updating the control dir for the FTPAdapter. Use the `eis/Ftp/HAFtpAdapter` connection factory instance for these modifications.

9.12.2 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 that the adapter's JCA connection factory:

1. Log on to the Oracle WebLogic Administration Console.

`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:

```
java.naming.factory.initial=weblogic.jndi.WLInitialContextFactory;
java.naming.provider.url=t3://SOAHOST1-PRIV-1:8001,SOAHOST2-PRIV-V1:8001;
security.principal=weblogic;java.naming.security.credentials=mypassword
```

9. Change the `java.naming.provider.url` property to `EoIB`, (`SOAHOST1VHN1:8001,SOAHOST2VHN1:8001`) addresses if external consumers and/or producers are to access the SOA servers
10. Click **Save** after you update the properties. Enter a shared storage location for the deployment plan. The directory structure is as follows:

`ORACLE_BASE/config/dp/soaedg_domain/JMSPlan.xml`

11. Click **OK**. Then click **Save**, and **Activate Changes**.

Update the deployment in the console:

1. Click **Deployments** and select the JMS Adapter.
2. Click **Lock & Edit** then **Update**.
3. 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).

4. Click **Finish**.
5. Activate the changes.

9.12.3 Scaling the Oracle Database Adapter

Previously, the best practice for multiple Oracle Database Adapter process instances deployed to multiple Oracle BPEL Process Manager, or Oracle Mediator nodes was using `LogicalDeletePollingStrategy` or `DeletePollingStrategy` with a unique `MarkReservedValue` on each polling node, and setting `MaxTransactionSize`. If you used this approach, you can remove (in `db.jca`) or clear (Logical Delete Page of the Configuration Wizard) the `MarkReservedValue` to 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, minimizing the risk of a non-recoverable situation in a high availability environment.
- You do not need to specify a unique `MarkReservedValue`, which requires configuring a complex variable such as `R${weblogic.Name-2}-${IP-2}-${instance}`.

If you use 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*.

9.13 Updating the Workflow Front End Address for Appropriate Task Display

You must configure Oracle Workflow with the appropriate URL so that Default-to-do tasks and custom tasks' details use the front end load balancer to create task-display URLs.

To configure the appropriate URLs:

1. Log in to Oracle Enterprise Manager Fusion Middleware Control with the username and password you specified in [Section 8.5.1, "Creating boot.properties for the Administration Server on SOAHOST1."](#)
2. In the left navigation tree, expand `Farm_domain_name`, **SOA**, and then right click **soa-infra** `server_name` (either one of the servers).
3. Click **SOA Administration**, and then **Workflow Properties**.
4. Select **More Workflow Notification Configuration Properties...**
5. On the Mbean tree on the left, expand **Workflow Config** and click **humanworkflow**.
6. In the `FusionAppsFrontendHostUrl` field, enter the following:

```
*=https://soa.mycompany.com:443
```
7. Click **Apply**.

9.14 Updating the B2B Instance Identifier for Transports

To set up File, FTP, or Email transports in a high availability environment, specify a unique name for each instance by using **b2b.HAInstanceName** `unique_instance_name`. If you use **ServerName** for the value, Oracle B2B retrieves the WebLogic Server name as the HAInstanceName.

To specify a unique name:

1. Log in to Oracle Enterprise Manager Fusion Middleware Control with the username and password specified in [Section 8.5.1, "Creating boot.properties for the Administration Server on SOAHOST1."](#)
2. On the navigation tree on the left, expand **Farm_<domain_name>**, **SOA**, and then right click on the **soa-infra<server_name>**, and select the **SOA Administration**, and then **B2B Server Properties**.
3. Click on **More B2B Configuration Properties...** on the right.
4. Click the **b2b** MBean.
5. Click the **Operations** tab.
6. Click **addProperty** in the list on the right.
7. In the **Key** field enter **b2b.HAInstanceName**.
8. In the value field enter **#ServerName#**.
Enter this value in only one of the two servers.
9. Click **Invoke**.

9.15 Backing Up the Oracle SOA Configuration

Back up the Oracle SOA domain configuration. For more information, see [Section 14.8, "Backing Up the Oracle SOA Enterprise Deployment."](#)

Extending the Domain to Include Oracle BPM

This chapter describes the procedures for extending the domain to include Oracle BPM.

This chapter contains the following section:

- [Section 10.1, "Overview of Extending the Domain to include Oracle BPM"](#)
- [Section 10.2, "Option 1: Extending a Domain to Include SOA and BPM"](#)
- [Section 10.3, "Option 2: Extending a SOA Domain to Include Oracle BPM"](#)
- [Section 10.4, "Backing Up the Oracle BPM Configuration"](#)

10.1 Overview of Extending the Domain to include Oracle BPM

You can install and configure Oracle BPM in a Fusion Middleware installation in the following two ways:

- Extend an existing domain that contains an Administration Server (and optionally other non-SOA servers) to include SOA and BPM (in one single Configuration Wizard session). For configuration steps, see [Section 10.2, "Option 1: Extending a Domain to Include SOA and BPM."](#)
- Extend a domain that already contains SOA (and optionally other non-SOA servers) to BPM. For configuration steps, see [Section 10.3, "Option 2: Extending a SOA Domain to Include Oracle BPM."](#)

Prerequisites for Extending the Domain to Include Oracle BPM

Before you extend 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. For more information, see [Section 14.8, "Backing Up the Oracle SOA Enterprise Deployment."](#)
- There is an existing WL_HOME and ORACLE_HOME installed in previous chapters on a shared storage.

10.2 Option 1: Extending a Domain to Include SOA and BPM

It is assumed that a SOA ORACLE_HOME (binaries) has already been installed, patched to latest patch set (if applicable), and is available from SOAHOST1 and

SOAHOST2. It is also assumed that a domain with an Administration Server has been created. This is the domain that is extended in this chapter to support SOA and BPM components.

To extend a domain with SOA and BPM components using the Configuration Wizard, follow the exact same steps in [Chapter 9, "Extending the Domain for SOA Components"](#) and make the following modifications:

- In the Select Extension Source screen, select **Oracle BPM Suite - 11.1.1.0 [soa]** beside the products indicated in [Chapter 9, "Extending the Domain for SOA Components."](#)
- In the Target Deployments to Clusters screen, besides the appropriate targets indicated in [Chapter 9, "Extending the Domain for SOA Components,"](#), ensure that `oracle.BPM.*` deployments target `SOA_Cluster` only.
- Add the `/bpm/composer` and `/bpm/workspace` URIs to the Oracle Traffic Director (OTD) route properties created for SOA.
- Update the appropriate persistent stores for the BPM JMS Servers

10.3 Option 2: Extending a SOA Domain to Include Oracle BPM

In this step, you extend the domain created to include Oracle BPM.

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:

```
tar -cvpf fmwhomeback.tar ORACLE_BASE/product/fmw
tar -cvpf domainhomeback.tar ORACLE_BASE/config/domain_name
```

These commands create a backup of the installation files for both Oracle WebLogic Server and Oracle Fusion Middleware, as well as the domain configuration.

- There is an existing WL_HOME and SOA ORACLE_HOME (binaries) are installed in previous chapters on a shared storage and are available from SOAHOST1 and SOAHOST2 (this is required before the WebLogic Configuration Wizard steps are performed to extend the domain).
- Node Manager, Admin Server, SOA Servers and WSM Servers exist and have been configured as described in previous chapters to run a SOA system. Server migration, transaction logs, coherence, and all other configuration steps for the SOA System have already been performed and will be used by BPM. BPM is added as a superset of the existing configuration.

This section contains the following topics:

- [Section 10.3.1, "Running the Configuration Wizard on SOAHOST1 to Extend a SOA Domain to Include BPM"](#)
- [Section 10.3.2, "Propagating the Domain Configuration to the managed server directory in SOAHOST1 and to SOAHOST2"](#)
- [Section 10.3.3, "Starting the BPM Suite Components"](#)

- [Section 10.3.4, "Configuring Oracle Traffic Director for the WLS_SOAn Managed Servers"](#)

10.3.1 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 and Oracle Web Services Manager to support SOA and BPM components.

1. Change the directory to the location of the Configuration Wizard. This is within the SOA home directory. Domain extensions are run from the node where the Administration Server resides.

```
cd ORACLE_COMMON_HOME/common/bin
```

2. Start the Oracle Fusion Middleware Configuration Wizard:

```
./config.sh
```

3. In the Welcome screen, select **Extend an Existing WebLogic Domain**, and click **Next**.
4. In the WebLogic Domain Directory screen, select the WebLogic domain directory `ASERVER_HOME`, and click **Next**.
5. In the Select Extension Source screen, do the following:
 - Select **Extend my domain automatically to support the following added products**. Select the following products:
 - Select the following product:
 - Oracle BPM Suite - 11.1.1.0 [soa]
6. In the Configure JDBC Component Schema screen, accept existing values (schemas created in the existing SOA system) and click **Next**.

Oracle BPM uses the same Data Sources as the existing soa-infra system.

7. In the Optional Configuration screen, select the following:
 - JMS Distributed Destinations
 - Deployments and Services
 - JMS File Store

Click **Next**.

8. In the Select JMS Distributed Destination Type screen, select UDD from the drop down list for BPMJMSModule. Leave existing modules as they are.
9. In the Target Deployments to Clusters or Servers screen, ensure the following targets:
 - Target **WSM-PM** only to **WSM-PM_Cluster**.
 - Target **usermessagingserver** and **usermessagingdriver-email** only to **SOA_Cluster**. (The **usermessaging-xmpp**, **usermessaging-smpp**, and **usermessaging-voicexml** applications are optional.)
 - Target the **oracle.sdp.***, **oracle.bpm.***, and **oracle.soa.*** libraries only to **SOA_Cluster**.
 - Target the **oracle.rules.*** library to **SOA_Cluster** and **Admin Server**.

Click **Next**.

10. In the Target Services to Clusters or Servers screen, target the **mds-owsm** datasource to the **WSM-PM_Cluster** and the **AdminServer** and click **Next**.
11. In the Configure JMS File Stores screen, enter the shared directory location specified for your JMS stores as recommended in [Section 4.4, "Recommended Directory Locations for an Oracle Exalogic Enterprise Deployment."](#) For example:

```
ASERVER_HOME/jms
```

Select **Direct-write** policy for all stores.

Click **Next**.

12. In the Configuration Summary screen click **Extend**.
13. In the Creating Domain screen, click **Done**.

You must restart the Administration Server for this configuration to take effect; see [Section 8.5.3, "Starting the Administration Server on SOAHOST1."](#)

10.3.2 Propagating the Domain Configuration to the managed server directory in SOAHOST1 and to SOAHOST2

Oracle BPM Suite requires some updates to the WebLogic Server start scripts. Propagate these changes using the pack and unpack commands.

To propagate the start scripts and classpath configuration from the Administration Server's domain directory to the managed server domain directory:

1. Create a backup copy of the managed server domain directory and the managed server applications directory.
2. Run the pack command on SOAHOST1 to create a template pack:

```
cd ORACLE_COMMON_HOME/common/bin

./pack.sh -managed=true -domain=ASERVER_HOME
-template=soadomaintemplateExtSOABPM.jar
-template_name=soa_domain_templateExtSOABPM
```

3. Run the unpack command on SOAHOST1 to unpack the propagated template to the domain directory of the managed server:

```
./unpack.sh -domain=MSERVER_HOME
-overwrite_domain=true -template=soadomaintemplateExtSOABPM.jar
-app_dir=APP_DIR
```

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.

4. Run the unpack command on SOAHOST2 to unpack the propagated template:

```
cd ORACLE_COMMON_HOME/common/bin
```

```
./unpack.sh -domain=MSERVER_HOME/  
-overwrite_domain=true -template=soadomaintemplateExtBPM.jar  
-app_dir=APP_DIR
```

Note: The configuration steps provided in this Exalogic enterprise deployment topology are documented with the assumption that a private (per node) domain directory is used for each managed server.

10.3.3 Starting the BPM Suite Components

For configuration changes and start scripts to be effective, you must restart the WLS_SOA n server to which BPM has been added. Since BPM extends an already existing SOA system, the Administration Server and respective Node Managers are already running in SOAHOST1 and SOAHOST2.

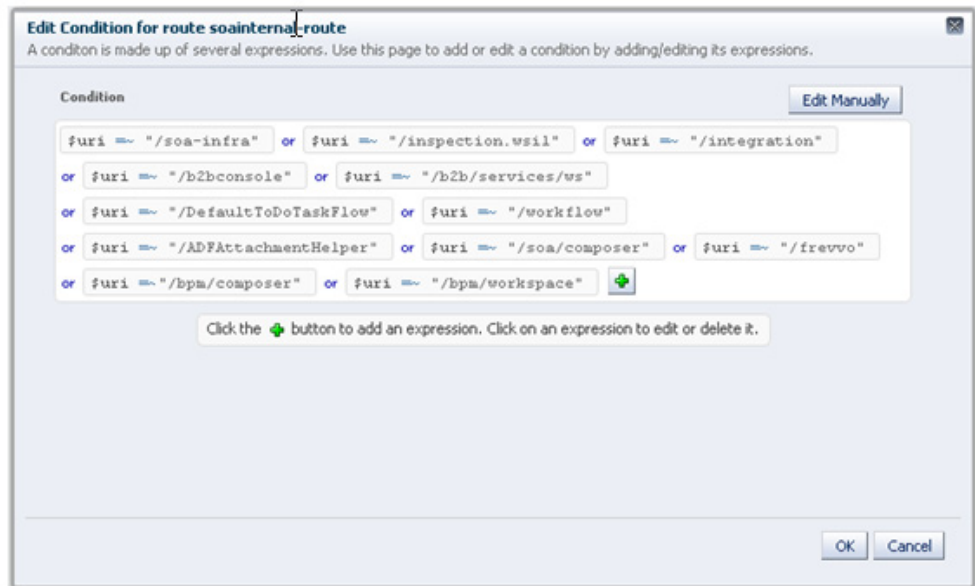
To start the added BPM components:

1. Restart the WLS_SOA1 managed server:
 - a. Log into the Oracle WebLogic Server Administration Console at:
`http://ADMINVHN:7001/console`.
 - b. In the Domain Structure window, expand the **Environment** node, then select **Servers**.
The Summary of Servers page appears.
 - c. Click the **Control** tab.
 - d. Select **WLS_SOA1** from the **Servers** column of the table.
 - e. Click **Shutdown**. Wait for the shutdown to complete (refresh the WebLogic Server Console page to verify shutdown status).
 - f. Click **Start**.
2. Repeat steps a-f for **WLS_SOA2**.

10.3.4 Configuring Oracle Traffic Director for the WLS_SOA n Managed Servers

To enable Oracle Traffic Director (OTD) to route to the appropriate BPM URIs, add the following to the route rules for both the `soainternal.mycompany.com` and the `soa.mycompany.com` virtual servers:

- `/bpm/composer`
- `/bpm/workspace`

Figure 10–1 Edit Condition for Route Screen

Verify URLs to ensure that appropriate routing and failover is working from the OTD Server to the BPM Suite Components.

For information on configuring system access through the load balancer, see [Section 3.7, "Configuring the Load Balancer."](#)

To verify the URLs:

1. While WLS_SOA is running, stop WLS_SOA1 using the Oracle WebLogic Server Administration Console.
2. Access <http://webhostN-priv-v1:7777/bpm/composer> and <http://webhostN-priv-v1:7777/bpm/workspace> to verify the appropriate functionality for BPM project Composer.
3. Start WLS_SOA1 from the Oracle WebLogic Server Administration Console.
4. Stop WLS_SOA2 from the Oracle WebLogic Server Administration Console.
5. Access <http://webhostN-priv-v1:7777/bpm/composer> and <http://webhostN-priv-v1:7777/bpm/workspace> to verify the appropriate functionality for BPM Workspace.

You can also verify these URLs using your load balancer address:

- <http://soa.mycompany.com:80/bpm/composer>
- <http://soa.mycompany.com:80/bpm/workspace>

10.4 Backing Up the Oracle BPM Configuration

Back up the Oracle BPM configuration. For more information, see [Section 14.8, "Backing Up the Oracle SOA Enterprise Deployment."](#)

Extending a SOA Domain to Oracle Service Bus

This chapter describes the procedures for extending the domain to include Oracle Service Bus.

This chapter contains the following sections:

- Section 11.1, "Overview of Adding Oracle Service Bus to a SOA Domain"
- Section 11.2, "Installing the Required Oracle Service Bus Binaries"
- Section 11.3, "Verifying Virtual IP Addresses for OSB Managed Servers"
- Section 11.4, "Running the Configuration Wizard on SOAHOST1 to Extend a SOA Domain to Include Oracle Service Bus"
- Section 11.5, "Disabling Host Name Verification for the WLS_OSBN Managed Servers"
- Section 11.6, "Configuring Oracle Coherence for the Oracle Service Bus Result Cache"
- Section 11.7, "Configuring a Default Persistence Store for Transaction Recovery"
- Section 11.8, "Propagating the Domain Configuration to the Managed Server Directory in SOAHOST1 and to SOAHOST2"
- Section 11.9, "Starting the Oracle Service Bus Servers"
- Section 11.10, "Configuring Network Channels for HTTP and T3 Clients via EoB"
- Section 11.11, "Validating the WLS_OSBN Managed Servers"
- Section 11.12, "Configuring Oracle Traffic Director with the Extended Domain"
- Section 11.13, "Setting the Front End HTTP Host and Port for OSB_Cluster"
- Section 11.14, "Validating Access Through Oracle Traffic Director and Load Balancer"
- Section 11.15, "High Availability for Oracle DB, File and FTP Adapters"
- Section 11.16, "Configuring Server Migration for the WLS_OSBN Servers"
- Section 11.17, "Backing Up the Oracle Service Bus Configuration"

11.1 Overview of Adding Oracle Service Bus to a SOA Domain

This section provides an overview of adding Oracle Service bus to an SOA domain. [Table 11–1](#) lists and describes to high-level steps for extending a SOA domain for Oracle Service Bus.

Table 11–1 Steps for Extending a SOA Domain to Include Oracle Service Bus

Step	Description	More Information
Install Oracle Service Bus Binaries.	Install Oracle Fusion Middleware Oracle Service Bus on SOAHOST1.	Section 11.2, "Installing the Required Oracle Service Bus Binaries"
Enable VIP5 on SOAHOST1 and VIP6 on SOAHOST2	Enable a virtual IP mapping for each of these hostnames on the two SOA Machines.	Section 11.3, "Verifying Virtual IP Addresses for OSB Managed Servers"
Run the Configuration Wizard to Extend the Domain	Extend the SOA domain to contain Oracle Service Bus components	Section 11.4, "Running the Configuration Wizard on SOAHOST1 to Extend a SOA Domain to Include Oracle Service Bus"
Disable Host Name Verification for the WLS_OSBN Managed Server	If you have not set up the appropriate certificates for hostname verification between the Administration Server, Managed Servers, and Node Manager, disable host name verification.	Section 11.5, "Disabling Host Name Verification for the WLS_OSBN Managed Servers"
Configure Oracle Coherence for the Oracle Service Bus Result Cache	Use unicast communication for the Oracle Service Bus result cache.	Section 11.6, "Configuring Oracle Coherence for the Oracle Service Bus Result Cache"
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.	Section 11.7, "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.	Section 11.8, "Propagating the Domain Configuration to the Managed Server Directory in SOAHOST1 and to SOAHOST2"
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.	Section 11.9, "Starting the Oracle Service Bus Servers"
Validate the WLS_OSBN Managed Servers	Verify that the server status is reported as Running in the Admin Console and access URLs to verify status of servers.	Section 11.11, "Validating the WLS_OSBN Managed Servers"
Configuring Oracle Traffic Director for the WLS_OSBN Managed Servers	To enable Oracle Traffic Director to route to Oracle Service Bus console and Oracle Service Bus service, set the WebLogicCluster parameter to the list of nodes in the cluster.	Section 11.12, "Configuring Oracle Traffic Director with the Extended Domain"
Set the Front End HTTP Host and Port for OSB_Cluster	Set the front end HTTP host and port for Oracle WebLogic Server cluster.	Section 11.13, "Setting the Front End HTTP Host and Port for OSB_Cluster"

Table 11–1 (Cont.) Steps for Extending a SOA Domain to Include Oracle Service Bus

Step	Description	More Information
Validating Access	Verify that the server status is reported as Running.	Section 11.14, "Validating Access Through Oracle Traffic Director and 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.	Section 11.15, "High Availability for Oracle DB, File and FTP Adapters"
Configure Server Migration for the WLS_OSB Servers	The high availability architecture for an Oracle Service Bus system uses server migration to protect some singleton services against failures.	Section 11.16, "Configuring Server Migration for the WLS_OSB Servers"
Backing Up the Configuration	Back up the domain configuration. This backup is for the purpose of an having an immediate restore available in the event of failures in future procedures.	Section 11.17, "Backing Up the Oracle Service Bus Configuration"

11.1.1 Prerequisites for Extending the SOA 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. For more information see [Section 14.8, "Backing Up the Oracle SOA Enterprise Deployment."](#)
- You have installed WL_HOME and MW_HOME (binaries) that contain Oracle Fusion Middleware SOA on a shared storage and they are available from SOAHOST1 and SOAHOST2.
- You have already configured Node Manager, Admin Server, SOA Servers and WSM Servers as described in previous chapters to run a SOA system. You have already configured Server migration, transaction logs, coherence, and all other configuration steps for the SOA System.

11.2 Installing the Required Oracle Service Bus Binaries

To install Oracle Fusion Middleware Oracle Service Bus on SOAHOST1. A single installation is used for all the nodes in the domain. You install on SOAHOST1, and SOAHOST2 mounts the same mount point.

1. On Linux platforms, if the `/etc/oraInst.loc` file exists, check that its contents are correct. Specifically, check that the inventory directory is correct and that you have write permissions for that directory. If the `/etc/oraInst.loc` file does not exist, skip this step.
2. Start the installer for Oracle Service Bus from the installation media:

```
./runInstaller
```

When the installer prompts you for a JRE/JDK location, enter the Oracle SDK location created in the Oracle WebLogic Server installation, for example, `ORACLE_BASE/product/fmw/jrockit-jdk1.6.0_version`. For more information, see [Section 8.2.2, "Installing WebLogic Server Using the Generic Installer."](#)

3. In the Welcome screen, click **Next**.

Note: Since SOAHOST1 already contains the SOA Suite Oracle Home, an Oracle Inventory should already be present and used by this installation. In this case, the Specify Inventory Directory screen should not appear

4. In the Install Software Updates screen, choose **Skip Software Updates** and click **Next**.
5. In the Installation Location screen, provide the installation location for Oracle Service Bus. Select the previously installed Oracle Middleware Home from the drop-down list. For the Oracle Home directory, enter the directory name (**osb**), and click **Next**.
6. In the Installation Type, select **Custom**, and click **Next**.
7. In the Components to Install screen, **DESELECT Oracle Service Bus IDE and Oracle Service Bus Examples**, and click **Next**.
8. In the Prerequisite Checks screen, verify that all checks complete successfully, and click **OK**.
9. In the Product Home Location specify the WebLogic Server installation directory previously installed and click **Next**.
10. In the Installation Summary screen, click **Install**.
11. In the Installation Complete screen, click **Finish**.
12. Validate the installation by verifying that the following directories appear in the *ORACLE_HOME_OSB* directory (under *osb*) after installing Oracle Service Bus:
 - 3rdparty
 - bin
 - cfgtoollogs
 - clone
 - common
 - config
 - dbscripts
 - diagnostics
 - financial
 - harvester
 - install
 - inventory
 - L10N
 - lib
 - modules
 - OPatch
 - osb

- oui
- soa
- tools

11.3 Verifying Virtual IP Addresses for OSB Managed Servers

The SOA domain uses virtual hostnames as the listen addresses for the Oracle Service Bus managed servers: SOAHOST1-PRIV-V2, SOAHOST2-PRIV-V2, SOAHOST1VHN2, SOAHOST2VHN2. Enable a virtual IP mapping for each of these hostnames on the two SOA Machines, VIP5 on SOAHOST1 and VIP6 on SOAHOST2, and correctly resolve the virtual hostnames in the network system that the topology uses (either by DNS Server, hosts resolution).

These virtual IPs and VHNs are required to enable server migration for the Oracle Service Bus Servers. Server migration must be configured for the Oracle Service Bus Cluster for high availability purposes. See [Chapter 13, "Configure Server Migration for an Exalogic Enterprise Deployment"](#) for more details on configuring server migration for the Oracle Service Bus servers.

Note: Verify that you can ping the virtual hostnames from both WEBHOST1 and WEBHOST2 and from each one of the compute nodes in the topology as the chapter [Chapter 3, "Configuring the Network for an Exalogic Enterprise Deployment"](#) describes.

11.4 Running the Configuration Wizard on SOAHOST1 to Extend a SOA Domain to Include Oracle Service Bus

In this step, you extend the domain created in Chapter 9, "Extending the Domain for SOA Components" to contain Oracle Service Bus components. The steps reflected in this section would be very similar if Oracle Service Bus was extending a domain containing only an Admin Server and a WSM-PM Cluster, but some of the options, libraries and components shown in the screens could vary.

To extend the domain for Oracle Service Bus:

1. Change directory to the location of the Configuration Wizard. This is within the Oracle Service Bus directory. (All database instances should be up.)

```
cd ORACLE_COMMON_HOME/common/bin
```

2. Start the Configuration Wizard.

```
./config.sh
```

3. In the Welcome screen, select **Extend an existing WebLogic domain**, and click **Next**.

4. In the WebLogic Domain Directory screen, select the WebLogic domain directory:

```
ASERVER_HOME
```

Click **Next**.

5. In the Select Extension Source screen, select **Extend my domain automatically to support the following added products** and select the following products (the components required by Oracle SOA and Oracle WSM Policy Manager should already be selected and grayed out):

- Oracle Service Bus OWSM Extension - 11.1.1.7 [osb]
 - Oracle Service Bus - 11.1.1.7 [osb]
 - WebLogic Advance Web Services JAX-RPC Extension
6. In the Configure JDBC Components Schema screen, do the following:
- Select the **OSB JMS reporting Provider** schema.
 - For the Oracle RAC configuration for component schemas, select **Convert to GridLink**

Note: If additional data sources, such as SOAINFRA, the server migration/leasing data source or OPSS data source, were configured for the domain, they appear here. You can select **Next** without modifying and testing them because the expectation is that they have already been verified and are working data sources.

Click **Next**. The Configure Gridlink RAC Component Schema screen appears.

7. In the Configure Gridlink RAC Component Schema screen accept the values for the data sources that are already present in the domain and click **Next**.
8. In the Test JDBC Component Schema screen, verify that the Oracle Service Bus JMS reporting datasources are correctly verified and click **Next**.
9. In the Select Optional Configuration screen, select the following:
- **JMS Distributed Destinations**
 - **Managed Servers, Clusters, and Machines**
 - **Deployments and Services**
 - **JMS File Store**

Click **Next**.

10. In the Select JMS Distributed Destination Type screen leave the pre-existing JMS System Resources as they are and Select **UDD** from the drop down list for **WseeJMSModule** and **JmsResources**.

Click **Next**.

11. In the Configure Managed Servers screen, add the required managed servers for Oracle Service Bus.
- a. Select the automatically created server and click **Rename** to change the name to WLS_OSB1.
 - b. Click **Add** to add another new server and enter WLS_OSB2 as the server name.
 - c. Give servers WLS_OSB1 and WLS_OSB2 the attributes listed in [Table 11–2](#).

In the end, the list of managed servers should match [Table 11–2](#).

Click **Next**.

Table 11–2 Managed Servers

Name	Listen Address	Listen Port	SSL Listen Port	SSL Enabled
WLS_SOA1(*)	SOAHOST1-PRIV_V1	8001	n/a	No

Table 11–2 (Cont.) Managed Servers

Name	Listen Address	Listen Port	SSL Listen Port	SSL Enabled
WLS_SOA2(*)	SOAHOST2-PRIV_V1	8001	n/a	No
WLS_WSM1	SOAHOST1-PRIV	7010	n/a	No
WLS_WSM2	SOAHOST2-PRIV	7010	n/a	No
WLS_OSB1	SOAHOST1-PRIV-V2	8011	n/a	No
WLS_OSB2	SOAHOST2-PRIV-V2	8011	n/a	No

12. In the Configure Clusters screen, add the Oracle Service Bus cluster (leave the present cluster as they are):

Table 11–3 Clusters

Name	Cluster Messaging Mode	Multicast Address	Multicast Port	Cluster Address
SOA_Cluster(*)	unicast	n/a	n/a	SOAHOST1-PRIV-V1:8001, SOAHOST2-PRIV-V1:8001 Note: The cluster address should be changed to the appropriate EoIB addresses if external clients are going to access the SOA servers.
WSM-PM_Cluster	unicast	n/a	n/a	Leave it empty.
OSB_Cluster	unicast	n/a	n/a	SOAHOST1-PRIV-V2:8011, SOAHOST2-PRIV-V2:8011

(*) - if you are extending a SOA domain

Click **Next**.

Note: For asynch request/response interactions over direct binding, the SOA composites must provide their jndi provider URL for the invoked service to look up the beans for callback.

If soa-infra configuration properties are not specified, but the WebLogic Server Cluster address is specified, the cluster address from the JNDI provider URL is used. This cluster address can be a single DNS name which maps to the clustered servers' IP addresses or a comma separated list of server ip:port. Alternatively, the soa-infra configuration property `JndiProviderURL/SecureJndiProviderURL` can be used for the same purpose if explicitly set by users.

13. In 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:
 - WLS_OSB1
 - WLS_OSB2

Click **Next**.

14. Confirm that the following entries appear:

Table 11-4 Machines

Name	Node Manager Listen Address
SOAHOST1	SOAHOST1-PRIV
SOAHOST2	SOAHOST2-PRIV
ADMINHOST	localhost

Leave all other fields to their default values.

Click **Next**.

15. In 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**.

16. In the Target Deployments to Clusters or Servers screen, ensure the following targets:

- Target **usermessagingserver** and **usermessagingdriver-email** only to **SOA_Cluster**. (The **usermessaging-xmpp**, **usermessaging-smpp**, and **usermessaging-voicexml** applications are optional.)
- Target the **oracle.sdp.*** and **oracle.soa.*** libraries only to **SOA_Cluster**.
- Target the **oracle.rules.*** library only to **AdminServer** and **SOA_Cluster**.
- Target the **wsm-pm** application only to **WSM-PM_Cluster**.
- Target all Transport Provider Deployments to both the **OSB_Cluster** and the **AdminServer**.
- Target the **oracle.bpm.*** library only to the **SOA_Cluster**.

Click **Next**.

17. In the Target Services to Clusters or Servers screen:

- Target **mds-owsm** only to **WSM-PM_Cluster** and **AdminServer**.
- Target **mds-soa** to **SOA_Cluster**.

Click **Next**.

18. In the Configure JMS File Stores screen, enter the shared directory location specified for your JMS stores as recommended in [Section 4.2, "Shared Storage Recommendations for Exalogic Enterprise Deployments."](#) For example:

```
ASERVER_HOME/jms
```

Select **Direct-write** policy for all stores.

Click **Next**.

19. In the Configuration Summary screen click **Extend**.
20. In the Extending Domain screen, click **Done**.
21. Restart the Administration Server for this configuration to take effect.

11.5 Disabling Host Name Verification for the WLS_OSB_n Managed Servers

For the Exalogic enterprise deployment described in this guide, you set up the appropriate certificates to authenticate the different nodes with the Administration Server after you have completed the procedures to extend the domain for Oracle SOA Suite. You must disable the host name verification for the WLS_OSB1 and WLS_OSB2 managed servers to avoid errors when managing the different WebLogic Server instances. For more information, see [Section 8.4.8, "Disabling Host Name Verification."](#)

You enable host name verification again once the Exalogic enterprise deployment topology configuration is complete. For more information, see [Section 13.3, "Enabling Host Name Verification Certificates for Node Manager in SOAHOST1."](#)

11.6 Configuring Oracle Coherence for the Oracle Service Bus Result Cache

By default, result caching uses multicast communication. Oracle recommends using unicast communication for the Oracle Service Bus result cache. Additionally, Oracle recommends separating port ranges for Coherence clusters used by different products. The ports for the Oracle Service Bus result cache Coherence cluster should be different from the Coherence cluster used for SOA.

To enable unicast for the Oracle Service Bus result cache Coherence infrastructure:

1. Log into Oracle WebLogic Server Administration Console. In the Change Center, click **Lock & Edit**.
2. In the Domain Structure window, expand the Environment node.
3. Click **Servers**.
4. Click the name of the server (represented as a hyperlink) in the Name column of the table. The settings page for the selected server appears.
5. Click the **Server Start** tab.
6. Enter the following for WLS_OSB1 on a single line, no carriage returns:

```
-DOSB.coherence.localhost=SOAHOST1-PRIV-V2 -DOSB.coherence.localport=7890  
-DOSB.coherence.wka1=SOAHOST1-PRIV-V2 -OSB.coherence.wka1.port=7890
```

```
-DOSB.coherence.wka2=SOAHOST2-PRIV-V2 -DOSB.coherence.wka2.port=7890
```

For WLS_OSB2, enter the following on a single line, no carriage returns:

```
-DOSB.coherence.localhost=SOAHOST2-PRIV-V2 -DOSB.coherence.localport=7890  
-DOSB.coherence.wka1=SOAHOST1-PRIV-V2 -OSB.coherence.wka1.port=7890  
-DOSB.coherence.wka2=SOAHOST2-PRIV-V2 -DOSB.coherence.wka2.port=7890
```

Note: There should be no breaks in lines between the different -D parameters. Do not copy or paste the text from above to your Administration Console's arguments text field. This may result in HTML tags being inserted in the Java arguments. The text should not contain other text characters than those included the example above.

7. Save and activate the changes. You must restart Oracle Service Bus servers for these changes take effect.

Note: The Coherence cluster used for Oracle Service Bus' result cache is configured above using port 7890. This port can be changed by specifying a different port (for example, 8089) with the following startup parameters:

```
-Dtangosol.coherence.wkan.port  
-Dtangosol.coherence.localport
```

For more information about Coherence Clusters see the *Oracle Coherence Developer's Guide*.

8. Ensure that these variables are passed to the managed server correctly by checking the server's output log.

Failure of the Oracle Coherence framework can prevent the result caching from working.

11.7 Configuring a Default Persistence Store for Transaction Recovery

Each server has a transaction log that stores information about committed transactions that are coordinated by the server that may not have been completed. The 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 servers within a cluster, store the transaction log in a location accessible to a server and its backup servers.

To set the location for the default persistence stores for:

1. Log into the Oracle WebLogic Server Administration Console.
2. In the Domain Structure window, expand the **Environment** node and then click the **Servers** node.

The Summary of Servers page appears.

3. Click the **WLS_OSB1** 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.

4. Click the **Services** tab.
5. In the Default Store section of the page, enter the path to the folder where the default persistent stores will store its data files.

The directory structure of the path is as follows:

```
ASERVER_HOME/tlogs
```

6. Click **Save** and **Active Changes**.
7. Repeat steps 3 through 6 for **WLS_OSB2**.

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. Both WLS_OSB1 and WLS_OSB2 must be able to access this directory. This directory must also exist before you restart the servers.

11.8 Propagating 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.

Prerequisite

Create a backup copy of the managed server domain directory and the managed server applications directory.

To propagate the start scripts and classpath configuration from the Administration Server's domain directory to the managed server domain directory:

1. Run the pack command on SOAHOST1 to create a template pack:

```
cd ORACLE_COMMON_HOME/common/bin

./pack.sh -managed=true -domain=ASERVER_HOME
-template=soadomaintemplateExtOSB.jar
-template_name=soa_domain_templateExtOSB
```

2. Run the unpack command on SOAHOST1 to unpack the propagated template to the domain directory of the managed server:

```
./unpack.sh -domain=MSERVER_HOME
-overwrite_domain=true -template=soadomaintemplateExtOSB.jar
-app_dir=APP_DIR
```

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.

3. Run the unpack command on SOAHOST2 to unpack the propagated template:

```
cd ORACLE_COMMON_HOME/common/bin

./unpack.sh -domain=MSERVER_HOME/ -overwrite_domain=true
-template=soadomaintemplateExtOSB.jar -app_dir=APP_DIR
```

Note: The configuration steps provided in this Exalogic enterprise deployment topology are documented with the assumption that a private (per node) domain directory is used for each managed server.

11.9 Starting the Oracle Service Bus Servers

Since Oracle Service Bus servers extend an already existing domain it is assumed that the Administration Server and respective Node Managers are already running in SOAHOST1 and SOAHOST2.

To start the added the WLS_OSB servers:

1. Log into the Oracle WebLogic Server Administration Console at:
`http://ADMINVHN:7001/console`
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_OSB1** from the **Servers** column of the table.
5. Click **Start**. Wait for the server to come up and check that its status is reported as **RUNNING** in the Administration Console.
6. Repeat steps 2 through 5 for WLS_OSB2.

11.10 Configuring Network Channels for HTTP and T3 Clients via EoIB

If your HTTP clients and T3 clients use the 10 Gb Ethernet network, you must create additional network channels for the OSB Servers on SOAHOST1 and SOAHOST2. For more information, for more details

- [Section 11.10.1, "Creating HTTP Client Channels"](#)
- [Section 11.10.2, "T3 Client Channel"](#)

11.10.1 Creating HTTP Client Channels

To create a HTTP network channel for a Managed Server, such as WLS_OSB1, complete the following steps:

1. In your browser, go to `http://ADMINVHN1:7001/console` and log in as the administrator.
2. If you have not already done so, click **Lock & Edit** in the Change Center.
3. In the left pane of the Console, expand Environment, and then Servers to open the Summary of Servers page.
4. In the Servers table, click **WLS_OSB1** to open the Settings for WLS_OSB1 page.

5. Select **Protocols, Channels**, then **New**.
6. Enter **OSB_HTTPChannel** as the name of the new network channel and select **http** as the protocol, then click **Next**.
7. Enter the following information in the Network Channel Addressing page:

- Listen address: **SOAHOST1VHN2**

Note: This address is the virtual host name assigned to the WLS_OSB1 Server using the BOND1 interface.

- Listen port: 8011
- External Listen Address: `osb.mycompany.com`

Note: This address is the DNS name to access the application on the server.

- External Listen Port: 80
8. Click **Next**. Select **Enabled** then **HTTP Enabled for This Protocol** in the Network Channel Properties page.
 9. Select **Finish**.
 10. To activate these changes, click **Activate Changes** in the Change Center of the Administration Console,.

You must repeat the preceding steps to create a network channel for the WLS_OSB2 Managed Servers on SOAHOST2 and enter the required properties that describes.

Note: In this example, IP addresses are used as listen addresses. However, you can specify host names if they resolve to their corresponding floating IP addresses.

Managed Server	Name	Protocol	Listen Address	Listen Port	External Listen Address
WLS_OSB1	OSB_HTTPChannel	HTTP	SOAHOST1VHN2	8011	osb.mycompany.com
WLS_OSB2	OSB_HTTPChannel	HTTP	SOAHOST2VHN2	8011	osb.mycompany.com

11.10.2 T3 Client Channel

To create a T3 network channel for the SOA Managed Servers, complete the following steps:

1. In your browser, go to `http://ADMINVHN1:7001/console` and log in as the administrator.
2. If you have not already done so, click **Lock & Edit** in the Change Center.
3. In the left pane of the Console, expand **Environment** and then **Servers** to open the **Summary of Servers** page.
4. In the **Servers** table, click **WLS_OSB1** to open the **Settings for WLS_OSB1** page.
5. Select **Protocols, Channels**, then **New**.

6. Enter **OSB_T3Channel** as the name of the new network channel and select **T3** as the protocol, then click **Next**.
7. Enter the following information in the Network Channel Addressing page:

- Listen address: **SOAHOST1VHN2**

Note: This address is the virtual host name assigned to the WLS_OSB1 Server using the BOND1 interface.

- Listen port: **8013**

Note: Remove the default external Listen port value.

8. Click **Next**. Select **Enabled** then **HTTP Enabled for This Protocol** in the Network Channel Properties page.
9. Select **Finish**.
10. To activate these changes, click **Activate Changes** in the Change Center of the Administration Console.

You must repeat the preceding steps to create a network channel for the WLS_OSB2 Managed Servers on SOAHOST2 and enter the required properties that [Table 11–5](#) describes.

Table 11–5 Managed Server Properties

Managed Server	Name	Protocol	Listen Address	Listen Port	External Listen Address	External Listener Port
WLS_OSB1	OSB_T3Channel	T3	SOAHOST1VHN2	8013	osb.mycompany.com	80
WLS_OSB2	OSB_T3Channel	T3	SOAHOST2VHN2	8013	osb.mycompany.com	80

11.11 Validating the WLS_OSB Managed Servers

Validate the WLS_OSB managed servers using the Oracle WebLogic Server Administration Console and by accessing URLs.

To validate the WLS_OSB managed server:

1. Verify that the server status is reported as **Running** in the Admin 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. See [Section 14.14, "Troubleshooting the Topology in an Enterprise Deployment."](#) for possible causes.
2. Access the following URL to verify status of WLS_OSB1:

`http://SOAHOST1-PRIV-V2:8011/sbinspection.wsil`

With the default installation, this should be the HTTP response:

3. Access the following URL:

`http://SOAHOST1-PRIV-V2:8011/alsb/ws/_async/AsyncResponseServiceJms?WSDL`

With the default installation, this should be the HTTP response:

4. Access the equivalent URLs for:

`http://SOAHOST2-PRIV-V2:8011/`

5. Verify also the correct deployment of the Oracle Service Bus console to the Administration Server by accessing the following URL:

`http://ADMINHOSTVHN:7001/sbconsole/`

The Oracle Service Bus console should appear with no errors.

11.12 Configuring Oracle Traffic Director with the Extended Domain

After you create the appropriate channels, configure Oracle Traffic Director to route to the OSB servers for the appropriate URIs.

Note: (You created a virtual server for `osb.mycompany.com` and `osbinternal.mycompany.com` in [Chapter 7, "Installing and Configuring Oracle Traffic Director for an Exalogic Enterprise Deployment."](#))

11.12.1 Configuring Access Through Oracle Traffic Director for the WLS_OSBn Managed Servers

To create the required virtual server routes:

1. Log into the Administration Console using the following URL:

`https://OTDADMINVHN.mycompany.com:8989`

2. Click the **Configurations** in the upper left corner of the page to view a list of available configurations.
3. Select the configuration that you want to configure routes for.
4. In the Navigation pane, expand **Virtual Servers** and the **osb.mycompany.com** virtual server. Select **Routes** to open a list of routes that are defined for the virtual server.

11.12.1.1 Creating a New Route

To enable Oracle HTTP Server to route to the SOA_Cluster:

1. Click **New Route** to open the New Route dialog box.
2. In the Step 1: Route Properties screen, enter `osb-route` in the **Name** field.
3. In the Origin Server Pool drop-down menu, select `osb-pool` and click **Next**.
4. In the Step2: Condition Information screen, select the variable `$uri` from the Variable/Function drop-down list. Select `= ~` in the Operator drop-down menu, then enter `/sbinspection.wsil` in the **Value** field.

Note: You cannot use a joiner (and/or) for the first expression in the sequence.

5. Select **OK** then select **Plus** to add the next expression.

Note: You can now select the joiner 'or'.

6. Select **uri** as the Variable /Function, = ~ as the Operator, and **inspection.wsil** as the Value. Select **OK**.
7. Add the rest of the conditions using the information in the preceding step.

Table 11–6 Routes and Conditions

Route	Origin: Server Pool	Conditions
osb-route	osb-pool	/sbinspection.wsil" or \$uri =~ "/sbresource" or \$uri =~ "/osb" or \$uri =~ "/alsb"

8. Click **Next** and then **Create Route**.

Your route appears on the Routes and a Deployment Pending message appears in the main pane.

You can deploy the updated configuration immediately by selecting **Deploy Changes**, or wait until you make changes. See the topic [Section 7.8, "Deploying the Configuration and Testing the Virtual Server Addresses"](#) for more information.

To create a new route in the `osbinternal.mycompany.com` virtual server, repeat the preceding steps and conditions. You can copy the rules from the first route that you created. `osb-pool`

11.13 Setting the Front End HTTP Host and Port for OSB_Cluster

Set the front end HTTP host and port for Oracle WebLogic Server cluster using the WebLogic Server Administration Console.

To set the front end host and port:

1. In the WebLogic Server Administration Console, in the Change Center section, click **Lock & Edit**.
2. In the left pane, select **Environment** and then **Clusters**.
3. Select the **OSB_Cluster**.
4. Select **HTTP**.
5. Set the values for the following:
 - Frontend Host: **osb.mycompany.com**
 - Frontend HTTP Port: **80**
 - Frontend HTTPS Port: **443**

Note: Verify that the preceding address is correct and available, that is, that the load balancing router is up. An incorrect value, such as `http://` in the address or a trailing `/` in the host name, may prevent SOA from being accessible, even if you use virtual IPs to access it.

Click **Save**.

6. To activate the changes, click **Activate Changes** in the Change Center section of the Administration Console.
7. Restart managed servers WLS_OSB1 and WLS_OSB2.

11.14 Validating Access Through Oracle Traffic Director and Load Balancer

Since you have already set the cluster address for the OSB_Cluster, the Oracle Service Bus URLs can only be verified once Oracle HTTP Server has been configured to route the Oracle Service Bus context URLs to the WebLogic Servers. Verify the URLs to ensure that appropriate routing and failover is working from the HTTP Server to the Oracle Service Bus 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_OSB1 is running, stop WLS_OSB2 using the Oracle WebLogic Server Administration Console.
2. Access `webhost2-priv-v1:7777/sbinspection.wsil` and verify the HTTP response as indicated in [Section 11.11, "Validating the WLS_OSB Managed Servers."](#)
3. Start WLS_OSB2 from the Oracle WebLogic Server Administration Console.
4. Stop WLS_OSB1 from the Oracle WebLogic Server Administration Console.
5. Access `osb.mycompany.com:7777/sbinspection.wsil` and verify the HTTP response as indicated in section [Section 11.11, "Validating the WLS_OSB Managed Servers."](#)

Note: Because the `webhostn-priv-v1` addresses are internal to the Exalogic rack, you must start a browser from a node within the rack itself.

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 load balancer, but in all cases it should suffice to verify the appropriate mount points and correct failover in Oracle Traffic Director.

6. Verify this URLs using your load balancer address:

`http://osb.mycompany.com:80/sbinspection.wsil`

11.15 High Availability for Oracle DB, File and FTP Adapters

Oracle SOA Suite and Oracle Service Bus use the same database and File and FTP JCA adapters. You create the required database schemas for these adapters when you use the Oracle Repository Creation Utility for SOA. The required configuration for the adapters is described in section [Section 9.12.1, "Enabling High Availability for Oracle File and FTP Adapters."](#) The DB adapter does not require any configuration at the WebLogic Server resource level. 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 a WSM-PM and Admin Server domain, do the following:

- Run RCU to seed the Oracle Service Bus database with the required adapter schemas (Select **SOA Infrastructure**, and **SOA and BAM Infrastructure** in RCU).
- Perform the steps in, and the steps reflected in [Section 9.12.1, "Enabling High Availability for Oracle File and FTP Adapters."](#)

11.16 Configuring Server Migration for the WLS_OSB Servers

For more information on configuring server migration, [Chapter 13, "Configure Server Migration for an Exalogic Enterprise Deployment."](#)

11.17 Backing Up the Oracle Service Bus Configuration

Back up the Oracle Service Bus configuration. For more information, see [Section 14.8, "Backing Up the Oracle SOA Enterprise Deployment."](#)

Setting Up Node Manager for an Exalogic Enterprise Deployment

This chapter describes how to configure Node Manager in accordance with Oracle best practice recommendations.

This chapter contains the following sections:

- [Section 12.1, "Overview of the Node Manager"](#)
- [Section 12.2, "Setting Up Node Manager"](#)
- [Section 12.3, "Enabling Host Name Verification Certificates for Node Manager"](#)
- [Section 12.4, "Starting Node Manager"](#)

12.1 Overview of the Node Manager

Node Manager enables you to start and stop the Administration Server and the Managed Servers.

Process

The procedures described in this chapter must be performed on SOAHOST1 and SOAHOST2 for various components of the Exalogic enterprise deployment topology outlined in [Section 2.2, "Viewing the Oracle SOA Deployment Topology on Exalogic."](#)

Note that the procedures in this chapter must be performed multiple times for each VIP-and-IP pair using the information provided in the component-specific chapters.

Recommendations

Oracle provides two main recommendations for Node Manager configuration in Exalogic enterprise deployment topologies:

1. Oracle recommends placing the Node Manager log file in a location different from the default one (which is inside the Middleware Home where Node Manager resides). See [Section 12.2, "Setting Up Node Manager"](#) for further details.
2. Oracle also recommends using host name verification for the communication between Node Manager and the servers in the domain. This requires the use of certificates for the different addresses used in the domain. This chapter explains the steps for configuring certificates in the hosts for host name verification. See [Section 12.3, "Enabling Host Name Verification Certificates for Node Manager"](#) for further details.

Note: The passwords used in this guide are used only as examples. Use secure passwords in a production environment. For example, use passwords that consist of random sequences of both uppercase and lowercase characters as well as numbers.

12.2 Setting Up Node Manager

This section describes how to set up Node Manager for an Exalogic enterprise deployment.

This section contains the following topics:

- [Section 12.2.1, "Changing the Location of Node Manager Configuration Files"](#)
- [Section 12.2.2, "Editing the Node Manager Property File"](#)
- [Section 12.2.3, "Starting Node Manager"](#)

12.2.1 Changing the Location of Node Manager Configuration Files

Create a new directory for Node Manager configuration and log files outside the *MW_HOME* directory, and perform all Node Manager configuration tasks from this directory.

To create the new directory:

1. Run the following commands on SOAHOST1 and SOAHOST2:

```
mkdir -p /u02/private/oracle/config/nodemanager
```

2. Copy the `nodemanager.properties` file in the following directory:

```
/u01/oracle/products/access/wlserver_10.3/common/nodemanager
```

To the new `nodemanager` folders you created on SOAHOST1 and SOAHOST2.

3. Copy the `startNodeManager.sh` file in the following directory:

```
/u01/oracle/products access/wlserver_10.3/server/bin
```

And the `nodemanager.domains` files located in the following folder:

```
/u01/oracle/products/access/wlserver_10.3/common/nodemanager
```

To the new `nodemanager` folders you created on SOAHOST1 and SOAHOST2.

4. Open `startNodeManager.sh` for SOAHOST1 and SOAHOST2 located in the new `nodemanager` folder in SOAHOST1 and SOAHOST2) using a text editor, and make the following change:

On SOAHOST1 and SOAHOST2:

```
NODEMGR_HOME="/u02/private/oracle/config/nodemanager
```

12.2.2 Editing the Node Manager Property File

Update the `nodemanager.properties` file located in the following directory on SOAHOST1 and SOAHOST2:

```
/u02/private/oracle/config/nodemanager
```


On SOAHOST1 edit the file as follows:

```
NodeManagerHome: /u02/private/oracle/config/nodemanager
ListenAddress= 192.168.10.1
LogFile= /u02/private/oracle/config/nodemanager/nodemanager.log
Properties Value
SecureListener Set the value to "false".
StartScriptEnabled Set the value to "true",
StopScriptEnabled Set the value to "true",
StopScriptName Specify a name for the stop script, for example stopWebLogic.sh.
DomainsFile /u02/private/oracle/config/nodemanager/nodemanager.domains
```

On SOAHOST2:

```
NodeManagerHome: /u02/private/oracle/config/nodemanager
ListenAddress= 192.168.10.2
LogFile= /u02/private/oracle/config/nodemanager/nodemanager.log
Properties Value
SecureListener Set the value to "false".
StartScriptEnabled Set the value to "true",
StopScriptEnabled Set the value to "true",
StopScriptName Specify a name for the stop script, for example stopWebLogic.sh.
DomainsFile /u02/private/oracle/config/nodemanager/nodemanager.domains
```

12.2.3 Starting Node Manager

Start Node Manager on SOAHOST1 and SOAHOST 2 using `startNodeManager.sh` located in the following directory:

```
/u02/private/oracle/config/nodemanager
```

For example run the following command on SOAHOST1 and SOAHOST2:

```
./startNodeManager.sh
```

12.3 Enabling Host Name Verification Certificates for Node Manager

This section describes how to set up host name verification certificates for communication between Node Manager and the Administration Server. It consists of the following steps:

- [Section 12.3.1, "Generating Self-Signed Certificates Using the `utils.CertGen` Utility"](#)
- [Section 12.3.2, "Creating an Identity Keystore Using the `utils.ImportPrivateKey` Utility"](#)
- [Section 12.3.3, "Creating a Trust Keystore Using the `Keytool` Utility"](#)
- [Section 12.3.4, "Configuring Node Manager to Use the Custom Keystores"](#)
- [Section 12.3.5, "Using a Common or Shared Storage Installation"](#)
- [Section 12.3.6, "Configuring Managed WebLogic Servers to Use the Custom Keystores"](#)
- [Section 12.3.7, "Changing the Host Name Verification Setting for the Managed Servers"](#)

12.3.1 Generating Self-Signed Certificates Using the `utils.CertGen` Utility

The certificates added in this chapter (as an example) address a configuration where Node Manager listens on a physical host name (*HOST.mycompany.com*) and a WebLogic Managed Server listens on a virtual host name (*VIP.mycompany.com*). Whenever a server is using a virtual host name, it is implied that the server can be migrated from one node to another. Consequently, the directory where keystores and trust keystores are maintained ideally must reside on a shared storage that is accessible from the failover. If additional host names are used in the same or different nodes, the steps in this example must be extended to:

1. Add the required host names to the certificate stores (if they are different from *HOST.mycompany.com* and *VIP.mycompany.com*).
2. Change the identity and trust store location information for Node Manager (if the additional host names are used by Node Manager) or for the servers (if the additional host names are used by Managed Servers).

Follow these steps to create self-signed certificates on *HOST*. These certificates should be created using the network name or alias. For information on using trust CA certificates instead, see "Configuring Identity and Trust" in *Oracle Fusion Middleware Securing Oracle WebLogic Server*. The following examples configure certificates for *HOST.mycompany.com* and *VIP.mycompany.com*; that is, it is assumed that both a physical host name (*HOST*) and a virtual host name (*VIP*) are used in *HOST*. It is also assumed that *HOST.mycompany.com* is the address used by Node Manager and *VIP.mycompany.com* is the address used by a Managed Server or the Administration Server. This is the common situation for nodes hosting an Administration Server and a Fusion Middleware component, or for nodes where two Managed Servers coexist with one server listening on the physical host name and one server using a virtual host name (which is the case for servers that use migration servers).

1. Set up your environment by running the `WL_HOME/server/bin/setWLSEnv.sh` script. In the Bourne shell, run the following commands:

```
cd WL_HOME/server/bin
. ./setWLSEnv.sh
```

Verify that the `CLASSPATH` environment variable is set:

```
echo $CLASSPATH
```

2. Create a user-defined directory for the certificates. For example, create a directory called 'certs' under the `ASERVER_HOME` directory. Note that certificates can be shared across WebLogic domains.

```
cd ASERVER_HOME
mkdir certs
```

Note: 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 (like SSL set up for HTTP invocations, for example).

3. Change directory to the directory that you just created:

```
cd certs
```

4. Run the `utils.CertGen` tool from the user-defined directory to create the certificates for both `HOST.mycompany.com` and `VIP.mycompany.com`.

Syntax (all on a single line):

```
java utils.CertGen Key_Passphrase Cert_File_Name Key_File_Name
[export | domestic] [Host_Name]
```

Examples:

```
java utils.CertGen Key_Passphrase SOAHOST1.mycompany.com_cert
SOAHOST1.mycompany.com_key domestic SOAHOST1.mycompany.com
```

```
java utils.CertGen Key_Passphrase SOAHOST2.mycompany.com_cert
SOAHOST2.mycompany.com_key domestic SOAHOST2.mycompany.com
```

```
java utils.CertGen Key_Passphrase ADMINVHN.mycompany.com_cert
ADMINVHN.mycompany.com_key domestic ADMINVHN.mycompany.com
```

```
java utils.CertGen Key_Passphrase OUDADMINVHN.mycompany.com_cert
OUDADMINVHN.mycompany.com_key domestic OUDADMINVHN.mycompany.com
```

12.3.2 Creating an Identity Keystore Using the `utils.ImportPrivateKey` Utility

Follow these steps to create an identity keystore on `SOAHOST1`:

1. Create a new identity keystore called `appIdentityKeyStore` using the `utils.ImportPrivateKey` utility. Create this keystore under the same directory as the certificates (that is, `ASERVER_HOME/certs`).

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.

2. Import the certificate and private key for `SOAHOST1.mycompany.com`, `SOAHOST2.mycompany.com` and `ADMINVNH.mycompany.com` into the Identity Store. Ensure that you use a different alias for each of the certificate/key pairs imported.

Syntax (all on a single line):

```
java utils.ImportPrivateKey Keystore_File Keystore_Password
Certificate_Alias_to_Use Private_Key_Passphrase
Certificate_File
Private_Key_File
[Keystore_Type]
```

Examples:

```
java utils.ImportPrivateKey appIdentityKeyStore.jks Key_Passphrase
appIdentitySOAHOST1 Key_Passphrase ASERVER_HOME/certs/SOAHOST1.mycompany.com_
cert.pem ASERVER_HOME/certs/SOAHOST1.mycompany.com_key.pem
```

```
java utils.ImportPrivateKey appIdentityKeyStore.jks Key_Passphrase
appIdentitySOAHOST2 Key_Passphrase ASERVER_HOME/certs/SOAHOST2.mycompany.com_
cert.pem ASERVER_HOME/certs/SOAHOST2.mycompany.com_key.pem
```

```
java utils.ImportPrivateKey appIdentityKeyStore.jks Key_Passphrase
appIdentityADMVHN Key_Passphrase ASERVER_HOME/certs/ADMINVNH.mycompany.com_
```

```
cert.pem ASERVER_HOME/certs/ADMINVNH.mycompany.com_key.pem
```

12.3.3 Creating a Trust Keystore Using the Keytool Utility

Follow these steps to create the trust keystore on each host, SOAHOST1 and SOAHOST2:

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 keystore 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 ASERVER_HOME/certs/appTrustKeyStoreSOAHOST1.jks
```

2. The default password for the standard Java keystore is `changeit`. Oracle recommends always changing the default password. Use the `keytool` utility to do this. The syntax is:

```
keytool -storepasswd -new New_Password -keystore Trust_Keystore -storepass Original_Password
```

For example:

```
keytool -storepasswd -new Key_Passphrase -keystore appTrustKeyStoreSOAHOST1.jks -storepass changeit
```

3. The CA certificate `CertGenCA.der` is used to sign all certificates generated by the `utils.CertGen` tool. It is located in the `WL_HOME/server/lib` directory. This CA certificate must be imported into the `appTrustKeyStore` using the `keytool` utility. The syntax is:

```
keytool -import -v -noprompt -trustcacerts -alias Alias_Name -file CA_File_Location -keystore Keystore_Location -storepass Keystore_Password
```

For example:

```
keytool -import -v -noprompt -trustcacerts -alias clientCACert -file WL_HOME/server/lib/CertGenCA.der -keystore appTrustKeyStoreSOAHOST1.jks -storepass Key_Passphrase
```

12.3.4 Configuring Node Manager to Use the Custom Keystores

Configure Node Manager to use the custom keystores by editing the `nodemanager.properties` file located in the following directory on SOAHOST1 and SOAHOST2:

```
/u02/private/oracle/config/nodemanager_directory
```

Add the following lines to the `nodemanager.properties` file:

```
KeyStores=CustomIdentityAndCustomTrust
CustomIdentityKeyStoreFileName=Identity_Keystore
CustomIdentityKeyStorePassPhrase=Identity_Keystore_Password
CustomIdentityAlias=Identity_Keystore_Alias
CustomIdentityPrivateKeyPassPhrase=Private_Key_Used_When_Creating_Certificate
```

For example:

```
KeyStores=CustomIdentityAndCustomTrust
```

```

CustomIdentityKeyStoreFileName=ASERVER_HOME/certs/appIdentityKeyStore.jks
CustomIdentityKeyStorePassPhrase=Key_Passphrase
CustomIdentityAlias=appIdentitySOAHOST1
CustomIdentityPrivateKeyPassPhrase=Key_Passphrase

```

The passphrase entries in the `nodemanager.properties` file get encrypted when you start Node Manager as described in [Section 12.4, "Starting Node Manager."](#) For security reasons, minimize the time the entries in the `nodemanager.properties` file are left unencrypted. After you edit the file, start Node Manager as soon as possible so that the entries get encrypted.

12.3.5 Using a Common or Shared Storage Installation

When using a common or shared storage installation for `MW_HOME`, Node Manager is started from different nodes using the same base configuration (`nodemanager.properties`). Add the certificate for all the nodes that share the binaries to the `appIdentityKeyStore.jks` identity store by creating the certificate for the new node and import it to `appIdentityKeyStore.jks`, as described in [Section 12.3.1, "Generating Self-Signed Certificates Using the `utils.CertGen` Utility."](#) Once the certificates are available in the store, each node manager must point to a different identity alias to send the correct certificate to the Administration Server.

To set different environment variables before starting Node Manager in the different nodes:

```

cd WL_HOME/server/bin
export JAVA_OPTIONS=-DCustomIdentityAlias=appIdentitySOAHOST1

cd WL_HOME/server/bin
export JAVA_OPTIONS=-DCustomIdentityAlias=appIdentitySOAHOST2

```

Note: Make sure to specify the custom identity alias specifically assigned to each host, for example `appIdentity1` for `...HOST1` and `appIdentity2` for `...HOST2`.

12.3.6 Configuring Managed WebLogic Servers to Use the Custom Keystores

Follow these steps to configure the identity and trust keystores for `WLS_SERVER`:

1. Log in to Oracle WebLogic Server Administration Console at the URL listed in [Section 8.18.2, "Validating Access through Oracle Traffic Director."](#)
2. Click **Lock and Edit**.
3. Expand the **Environment** node in the Domain Structure window.
4. Click **Servers**. The Summary of Servers page is displayed.
5. Click the name of the server for which you want to configure the identity and trust keystores (`WLS_SERVER`). The settings page for the selected server is displayed.
6. Select **Configuration**, then **Keystores**.
7. In the Keystores field, select the **Custom Identity and Custom Trust** method for storing and managing private keys/digital certificate pairs and trusted CA certificates.
8. In the Identity section, define attributes for the identity keystore:
 - **Custom Identity Keystore:** The fully qualified path to the identity keystore:

`ASERVER_HOME/certs/appIdentityKeyStore.jks`

- **Custom Identity Keystore Type:** Leave blank; it defaults to JKS.
- **Custom Identity Keystore Passphrase:** The password (*Keystore_Password*) you provided in [Section 12.3.3, "Creating a Trust Keystore Using the Keytool Utility."](#) This attribute is optional or required depending on the type of keystore. All keystores require the passphrase to write to the keystore. However, some keystores do not require the passphrase to read from the keystore. WebLogic Server only reads from the keystore, so whether you define this property depends on the requirements of the keystore.

9. In the Trust section, define properties for the trust keystore:

- **Custom Trust Keystore:** The fully qualified path to the trust keystore:

`ASERVER_HOME/certs/appTrustKeyStoreSOAHOST1.jks`

- **Custom Trust Keystore Type:** Leave blank; it defaults to JKS.
- **Custom Trust Keystore Passphrase:** The password you provided as *New_Password* in [Section 12.3.3, "Creating a Trust Keystore Using the Keytool Utility."](#) This attribute is optional or required depending on the type of keystore. All keystores require the passphrase to write to the keystore. However, some keystores do not require the passphrase to read from the keystore. WebLogic Server only reads from the keystore, so whether you define this property depends on the requirements of the keystore.

10. Click **Save**.

11. Click **Activate Changes** in the Administration Console's Change Center to make the changes take effect.

12. Select **Configuration**, then **SSL**.

13. Click **Lock and Edit**.

14. In the **Private Key Alias** field, enter the alias you used for the host name the Managed Server listens on, for example:

- For `wls_oam1`, use `appIdentitySOAHOST1`.
- For `wls_oam2` use `appIdentitySOAHOST2`.
- For ADMIN SERVER user `appIdentityADMVHN`.

In the **Private Key Passphrase** and the **Confirm Private Key Passphrase** fields, enter the password for the keystore that you created in [Section 12.3.2, "Creating an Identity Keystore Using the `utils.ImportPrivateKey` Utility."](#)

15. Click **Save**.

16. Click **Activate Changes** in the Administration Console's Change Center to make the changes take effect.

17. Restart the server for which the changes have been applied, as described in [Section 8.5.3, "Starting the Administration Server on SOAHOST1."](#)

12.3.7 Changing the Host Name Verification Setting for the Managed Servers

Once the previous steps have been performed, set host name verification for the affected Managed Servers to `Bea Hostname Verifier`. To do this, perform the following steps:

1. Log in to Oracle WebLogic Server Administration Console.

2. Select **Lock and Edit** from the change center.
3. Expand the **Environment** node in the Domain Structure window.
4. Click **Servers**. The Summary of Servers page is displayed.
5. Select the Managed Server in the Names column of the table. The settings page for the server is displayed.
6. Open the SSL tab.
7. Expand the **Advanced** section of the page.
8. Set host name verification to `Bea Hostname Verifier`.
9. Click **Save**.
10. Click **Activate Changes**.

12.4 Starting Node Manager

Start Node Manager on SOAHOST1 and SOAHOST2 by running `startNodeManager.sh` located in the following directory:

```
/u02/private/oracle/config/nodemanager
```

To start Node manager, run the following command on SOAHOST1 and SOAHOST2:

```
./startNodeManager.sh
```

Note: If you have not configured and started Node Manager for the first time yet, run the `setNMProps.sh` script. This enables the use of the start script that is required for Fusion Middleware SOA Components.

Note: Verify that Node Manager is using the appropriate stores and alias from the Node Manager output. You should see the following when Node Manager starts.:

```
<Loading identity key store:  
  FileName=ASERVER_HOME/certs/appIdentityKeyStore.jks, Type=jks,  
  PassPhraseUsed=true>
```

Host name verification works if you apply a test configuration change to the servers and it succeeds without Node Manager reporting any SSL errors.

Configure Server Migration for an Exalogic Enterprise Deployment

Configuring server migration allows SOA managed servers to be migrated from one host to another, so that if a node hosting one of the servers fails, the service can continue on another node. This chapter describes how to configure server migration for an Fusion Middleware SOA Exalogic enterprise deployment.

This chapter contains the following steps:

- [Section 13.1, "Overview of Server Migration for an Exalogic Enterprise Deployment"](#)
- [Section 13.2, "Setting Up a User and Tablespace for the Server Migration Leasing Table"](#)
- [Section 13.3, "Creating a GridLink Data Source for Leasing Using the Oracle WebLogic Administration Console"](#)
- [Section 13.4, "Editing Node Manager's Properties File"](#)
- [Section 13.5, "Setting Environment and Superuser Privileges for the wlsifconfig.sh Script"](#)
- [Section 13.6, "Configuring Server Migration Targets"](#)
- [Section 13.7, "Testing the Server Migration"](#)
- [Section 13.8, "Backing Up the Server Migration Configuration"](#)

13.1 Overview of Server Migration for an Exalogic Enterprise Deployment

Configure server migration for the WLS_OSB1, WLS_SOA1, WLS_OSB2, and WLS_SOA2 Managed Servers. The WLS_OSB1 and WLS_SOA1 Managed Server are configured to restart on SOAHOST2 should a failure occur. The WLS_OSB2 and WLS_SOA2 Managed Servers are configured to restart on SOAHOST1 should a failure occur. The WLS_OSB1, WLS_SOA1, WLS_OSB2 and WLS_SOA2 servers listen on specific floating IPs that are failed over by WebLogic Server Migration.

Perform the steps in the following sections configure server migration for the WLS_OSB1, WLS_SOA1, WLS_OSB2, and WLS_SOA2 Managed Servers.

13.2 Setting Up a User and Tablespace for the Server Migration Leasing Table

In this section, you set up a user and tablespace for the server migration leasing table:

Note: If other servers in the same domain have already been configured with server migration, the same tablespace and data sources can be used. In that case, the data sources and multi data source for database leasing do not need to be re-created, but they must be retargeted to the clusters being configured with server migration.

1. Create a tablespace called `leasing`. For example, log on to SQL*Plus as the `sysdba` user and run the following command:

```
create tablespace leasing
logging datafile 'DB_HOME/oradata/orcl/leasing.dbf' size 32m autoextend on next
32m maxsize 2048m extent management local;
```

2. Create a user named `leasing` and assign to it the `leasing` tablespace:

```
create user leasing identified by password;
grant create table to leasing;
grant create session to leasing;
alter user leasing default tablespace leasing;
alter user leasing quota unlimited on leasing;
```

3. Create the `leasing` table using the `leasing.ddl` script:

- a. Copy the `leasing.ddl` file located in either of the following directories to your database node:

```
WL_HOME/server/db/oracle/817
WL_HOME/server/db/oracle/920
```

- b. Connect to the database as the `leasing` user.
- c. Run the `leasing.ddl` script in SQL*Plus:

```
@Copy_Location/leasing.ddl;
```

- d. After the tool completes, enter the following at the SQL*Plus prompt:

```
commit;
```

13.3 Creating a GridLink Data Source for Leasing Using the Oracle WebLogic Administration Console

Use [Appendix D, "Creating a GridLink Data Source."](#) to create a GridLink data source for the Leasing table using the Oracle WebLogic Administration Console.

For the Leasing table data source, use the following names:

- Datasource Name: **leasing**
- JNDI Name: **jdbc/leasing**
- Database User Name: **Leasing**

13.4 Editing Node Manager's Properties File

In this section, you edit Node Manager's properties file. This must be done for the Node Managers on the nodes where the servers are running, `SOAHOST1` and `SOAHOST2`.

The `nodemanager.properties` file is located in the following directory:

`WL_HOME/common/nodemanager`

- As interface, enter the EoIB and IPoIB interfaces that the SOA and OSB servers use for sever migration. Then enter the pertaining IP address ranges that will be controlled by server migration. Notice that this entry does not allow using hostnames but only IP addresses. Follow the IP addresses by the pertaining netmasks. For example:

```
bond0=soahost1-priv-v1-ip, soahost1-priv-v2-ip,NetMask=255.255.248.0
```

```
bond1=soahost1vhn1vip, soahost1vhn2vip,NetMask=255.255.248.0
```

Note: Do not specify the sub-interface, such as `bond0:1` or `bond0:2`. This interface is to be used without `:0` or `:1`. 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 `bond0`, `bond1`, `bond2`, `bond3`, `bondn`, depending on the number of interfaces configured.

- NetMask:

```
NetMask=255.255.248.0
```

This property specifies the net mask for the interface for the floating IP. The net mask should be the same as the net mask on the interface.

- UseMACBroadcast:

```
UseMACBroadcast=true
```

This property specifies whether to use a node's MAC address when sending ARP packets, that is, whether to use the `-b` flag in the `arping` command.

Verify in Node Manager's output (shell where Node Manager is started) that these properties are being used, or problems may arise during migration. You should see something like this in Node Manager's output:

```
StateCheckInterval=500
bond0=*,NetMask=255.255.248.0
UseMACBroadcast=true
```

Note: The following steps are not required if the server properties (start properties) have been properly set and Node Manager can start the servers remotely.

1. 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.
2. Start Node Manager on SOAHOST1 and SOAHOST2 by running the `startNodeManager.sh` script, which is located in the following directory:

```
/u02/private/oracle/config/nodemanager
```

13.5 Setting Environment and Superuser Privileges for the wlsifconfig.sh Script

Set environment and superuser privileges for the `wlsifconfig.sh` script:

Ensure that your `PATH` environment variable includes the files listed in [Table 13-1](#).

Table 13-1 Files Required for the `PATH` Environment Variable

File	Located in this directory
<code>wlsifconfig.sh</code>	<code>MSERVER_HOME/bin/server_migration</code>
<code>wlscontrol.sh</code>	<code>WL_HOME/common/bin</code>
<code>nodemanager.domains</code>	<code>WL_HOME/common/nodemanager</code>

Grant `sudo` privilege to the WebLogic user ('oracle') with no password restriction, and grant execute privilege on the `/sbin/ifconfig` and `/sbin/arping` binaries.

For security reasons, `sudo` should be restricted to the subset of commands required to run the `wlsifconfig.sh` script. For example, perform the following steps to set the environment and superuser privileges for the `wlsifconfig.sh` script.

Note: Ask the system administrator for the appropriate `sudo` and system rights to perform this step.

Grant `sudo` privilege to the WebLogic user `oracle` with no password restriction, and grant execute privilege on the `/sbin/ifconfig` and `/sbin/arping` binaries.

Make sure the script is executable by the WebLogic user ('oracle'). The following is an example of an entry inside `/etc/sudoers` granting `sudo` execution privilege for `oracle` and also over `ifconfig` and `arping`.

To grant `sudo` privilege to the WebLogic user ('oracle') with no password restriction, and grant execute privilege on the `/sbin/ifconfig` and `/sbin/arping` binaries:

```
Defaults:oracle !requiretty
oracle ALL=NOPASSWD: /sbin/ifconfig,/sbin/arping
```

13.6 Configuring Server Migration Targets

In this section, you configure server migration targets for `soa_cluster` and `osb_cluster`. Configuring Cluster Migration sets the `DataSourceForAutomaticMigration` property to `true`.

To configure migration in a cluster:

1. Log in to the Oracle WebLogic Server Administration Console at the URL listed in [Section 8.18.2, "Validating Access through Oracle Traffic Director."](#)
2. In the Domain Structure window, expand **Environment** and select **Clusters**. The Summary of Clusters page is displayed.
3. Click the cluster name for which you want to configure migration in the **Name** column of the table.
4. Click the **Migration** tab.
5. Click **Lock and Edit**.

6. In the **Available** field, select the machines to which to allow migration, **SOAHOST1** and **SOAHOST2**, and click the right arrow.
7. Select the data source to be used for automatic migration. In this case, select the leasing data source.
8. Click **Save**.
9. Click **Activate Changes**.
10. In the Domain Structure window of the Oracle WebLogic Server Administration Console, expand **Environment** and select **Servers**.
11. Select the server for which you want to configure migration.
12. Click the **Migration** tab.
13. Select **Automatic Server Migration Enabled** and click **Save**.
14. Click **Activate Changes**.
15. In the Available field, select the machine to which to allow migration and click the right arrow. In this case, select SOAHOST1 and SOAHOST2.
16. Restart the managed servers for which server migration has been configured as described in [Section 8.5.3, "Starting the Administration Server on SOAHOST1."](#)

Note: If migration is only going to be allowed to specific machines, do not specify candidates for the cluster, but rather specify candidates only on a server per server basis.

13.7 Testing the Server Migration

In this section, you test the server migration. For example, to test migration for OSB servers:

To test from SOAHOST1:

1. Stop the WLS_OSB1 Managed Server. To do this, run this command:

```
kill -9 pid
```

where *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_OSB1
```

2. Watch the Node Manager console. You should see a message indicating that WLS_OSB1's floating IP has been disabled.
3. Wait for Node Manager to try a second restart of WLS_OSB1. It waits for a fence period of 30 seconds before trying this restart.
4. Once Node Manager restarts the server, stop it again. Node Manager should now log a message indicating that the server will not be restarted again locally.

To test from SOAHOST2:

1. Watch the local Node Manager console. After 30 seconds since the last try to restart WLS_OSB1 on SOAHOST1, Node Manager on SOAHOST2 should prompt that the floating IP for WLS_OSB1 is being brought up and that the server is being restarted in this node.
2. Access the OSB Console using the Virtual Host Name, for example:

soahost1vhn1.mycompany.com/soa-infra/

Follow the previous steps to test server migration for the WLS_OSB2, WLS_SOA1, and WLS_SOA2 Managed Servers.

Table 13–2 shows the Managed Servers and the hosts they migrate to in case of a failure.

Table 13–2 Managed Server Migration

Managed Server	Migrated From	Migrated To
WLS_OSB1	SOAHOST1	SOAHOST2
WLS_OSB2	SOAHOST2	SOAHOST1
WLS_SOA1	SOAHOST1	SOAHOST2
WLS_SOA2	SOAHOST2	SOAHOST1

Verification From the Administration Console

Migration can also be verified in the 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** sub tab.

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 node/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.

13.8 Backing Up the Server Migration Configuration

Back up the server migration configuration. For more information, see [Section 14.8, "Backing Up the Oracle SOA Enterprise Deployment."](#)

Managing the Topology for an Exalogic Enterprise Deployment

This chapter describes some operations that you can perform after you have set up the Fusion Middleware SOA topology. These operations include monitoring, scaling, backing up your topology, and troubleshooting.

This chapter includes the following topics:

- [Section 14.1, "Overview of Managing the Topology"](#)
- [Section 14.2, "Tips for Deploying Composites and Artifacts in a SOA Enterprise Deployment Topology"](#)
- [Section 14.3, "Managing Space in the SOA Infrastructure Database"](#)
- [Section 14.4, "Configuring UMS Drivers"](#)
- [Section 14.5, "Scaling Up the Topology \(Adding Managed Servers to Existing Nodes\)"](#)
- [Section 14.6, "Scaling Out the Topology \(Adding Managed Servers to New Nodes\)"](#)
- [Section 14.7, "Verifying Manual Failover of the Administration Server"](#)
- [Section 14.8, "Backing Up the Oracle SOA Enterprise Deployment"](#)
- [Section 14.9, "Preventing Timeouts for SQLNet Connections"](#)
- [Section 14.10, "Recovering Failed BPEL and Mediator Instances"](#)
- [Section 14.11, "Configuring Web Services to Prevent Denial of Service and Recursive Node Attacks"](#)
- [Section 14.12, "Using Shared Storage for Deployment Plans and SOA Infrastructure Applications Updates"](#)
- [Section 14.13, "Using External BPEL Caches for Improved HAS and Performance Isolation"](#)
- [Section 14.14, "Troubleshooting the Topology in an Enterprise Deployment"](#)

14.1 Overview of Managing the Topology

After configuring the SOA enterprise deployment, use the information in this chapter to manage the topology.

SOA applications are deployed as composites, consisting of different kinds of components. SOA composite applications include the following:

- Service components such as Oracle Mediator for routing, BPEL processes for orchestration, 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. This chapter offers tips for managing and troubleshooting SOA composite applications in an Enterprise Deployment Topology on Oracle Exalogic.

For information on monitoring SOA composite applications, see *Monitoring SOA Composite Applications* in the *Oracle Fusion Middleware Administrator's Guide for Oracle SOA Suite and Oracle Business Process Management Suit*.

For information on managing SOA composite applications, see *Managing SOA Composite Applications* in the *Oracle Fusion Middleware Administrator's Guide for Oracle SOA Suite and Oracle Business Process Management Suit*.

At some point you may need to expand the topology by scaling it up, or out. See [Section 14.5, "Scaling Up the Topology \(Adding Managed Servers to Existing Nodes\)"](#), and [Section 14.6, "Scaling Out the Topology \(Adding Managed Servers to New Nodes\)"](#) for information about the difference between scaling up and scaling out, and instructions for performing these tasks.

Back up the topology before and after any configuration changes. [Section 14.8, "Backing Up the Oracle SOA Enterprise Deployment"](#) provides information about the directories and files that should be back up to protect against failure as a result of configuration changes.

This chapter also documents solutions for possible known issues that may occur after you have configured the topology.

14.2 Tips for Deploying Composites and Artifacts in a SOA Enterprise Deployment Topology

This section describes tips for deploying composites and artifacts for a SOA enterprise deployment. See the "Deploying SOA Composite Applications" chapter in the *Oracle Fusion Middleware Administrator's Guide for Oracle SOA Suite and Oracle Business Process Management Suit* for instructions on deploying composites.

Deploy composites to a specific server address

When deploying SOA composites to a SOA enterprise deployment topology, deploy to a specific server's address and not to the load balancer address (soa.mycompany.com). Deploying to the load balancer address may require direct connection from the deployer nodes to the external load balancer address which may require additional ports to be opened in the firewalls used by the system.

Use B2B Console for deployment agreements and purge/import metadata

For B2B, deploy agreements and purge/import metadata ONLY from the GUI available in B2B console. Do not use the command line utility. Using the command line utility for these operations may cause inconsistencies and errors in the B2B system.

Additional instructions for FOD deployment

If you are deploying the SOA Fusion Order Demo, complete the deployment steps provided in the FOD's README file, and then complete the following additional steps:

1. Change the nostage property to **false** in the build.xml file of the Web applications so that ear files are copied to each node. Edit the CreditCardAuthorization and OrderApprvalHumanTask build.xml files, located at FOD_dir\CreditCardAuthorization\bin and FOD_dir\OrderApprovalHumanTask\bin directories, and change the following field:

```
<target name="deploy-application">
  <wldploy action="deploy" name="${war.name}"
    source="${deploy.ear.source}" library="false"
    nostage="false"
    user="${wls.user}" password="${wls.password}"
    verbose="false" adminurl="${wls.url}"
    remote="true" upload="true"
    targets="${server.targets}" />
</target>
```

To:

```
<target name="deploy-application">
  <wldploy action="deploy" name="${war.name}"
    source="${deploy.ear.source}" library="false"
    nostage="true"
    user="${wls.user}" password="${wls.password}"
    verbose="false" adminurl="${wls.url}"
    remote="true" upload="true"
    targets="${server.targets}" />
</target>
```

2. Change the target for the Web applications so that deployments are targeted to the SOA Cluster and not to an individual server. Edit the build.properties file for FOD, located in the FOD_Dir/bin directory, and change the following field:

```
# wls target server (for shiphome set to server_soa, for ADRS use AdminServer)
server.targets=SOA_Cluster (the SOA cluster name in your SOA EDG)
```

3. Change the JMS seed templates so that instead of regular Destinations, Uniform Distributed Destinations are used and the JMS artifacts are targeted to the Enterprise Deployment JMS Modules. Edit the createJMSResources.seed file, located in the FOD_DIR\bin\templates directory, and change:

```
# lookup the SOAJMSModule - it's a system resource
jmsSOASystemResource = lookup("SOAJMSModule", "JMSSystemResource")

jmsResource = jmsSOASystemResource.getJMSResource()

cfbean = jmsResource.lookupConnectionFactory('DemoSupplierTopicCF')
if cfbean is None:
  print "Creating DemoSupplierTopicCF connection factory"
  demoConnectionFactory =
jmsResource.createConnectionFactory('DemoSupplierTopicCF')
  demoConnectionFactory.setJNDIName('jms/DemoSupplierTopicCF')
  demoConnectionFactory.setSubDeploymentName('SOASubDeployment')
.

topicbean = jmsResource.lookupTopic('DemoSupplierTopic')
if topicbean is None:
  print "Creating DemoSupplierTopic jms topic"
```

```
demoJMSTopic = jmsResource.createTopic("DemoSupplierTopic")
demoJMSTopic.setJNDIName('jms/DemoSupplierTopic')
demoJMSTopic.setSubDeploymentName('SOASubDeployment')
```

To:

```
jmsSOASystemResource = lookup("SOAJMSModule", "JMSSystemResource")

jmsResource = jmsSOASystemResource.getJMSResource()

topicbean=jmsResource.lookupTopic('DemoSupplierTopic_UDD')

if topicbean is None:
    print "Creating DemoSupplierTopicC jms topic"
    #create a udd - so clustering is automatically working and done
    demoJMSTopic =
jmsResource.createUniformDistributedTopic("DemoSupplierTopic_UDD")

    demoJMSTopic.setJNDIName('@jms.topic.jndi@')
    #Replace the subdeployment name with the one that appears in the WLS
AdminConsole as listed for the SOAJMSModule

    demoJMSTopic.setSubDeploymentName()

else: print "Found DemoSupplierTopic_UDD topic - noop"
```

Notice that ideally you should use a separate deployment module for the FOD JMS resources.

4. Update the `managed.server.host` entry in the `build.properties` file to the EoIB listen address of one of the two SOA servers.
5. Update the `admin.server.host` entry in `build.properties` to the EoIB listen address of the Administration Server.

14.3 Managing Space in the SOA Infrastructure Database

Although not all composites may use the database frequently, the service engines generate a considerable amount of data in the `CUBE_INSTANCE` and `MEDIATOR_INSTANCE` schemas. Lack of space in the database may prevent SOA composites from functioning.

To manage space in the SOA infrastructure database:

- Watch for generic errors, such as "oracle.fabric.common.FabricInvocationException" in the Oracle Enterprise Manager Fusion Middleware Control console (dashboard for instances).
- Search in the SOA server's logs for errors, such as:

```
Error Code: 1691
...
ORA-01691: unable to extend lob segment
SOAINFRA.SYS_LOB0000108469C00017$$ by 128 in tablespace SOAINFRA
```

These messages are typically indicators of space issues in the database that may likely require adding more data files or more space to the existing files. The SOA Database Administrator should determine the extension policy and parameters to be used when adding space.

- Purge old composite instances to reduce the SOA Infrastructure database's size. Oracle does not recommend using the Oracle Enterprise Manager Fusion Middleware Control for this type of operation. In most cases the operations cause a transaction time out. There are specific packages provided with the Repository Creation Utility to purge instances. For example:

```

DECLARE
  FILTER INSTANCE_FILTER := INSTANCE_FILTER();

  MAX_INSTANCES NUMBER;
  DELETED_INSTANCES NUMBER;
  PURGE_PARTITIONED_DATA BOOLEAN := TRUE;
BEGIN
  .
  FILTER.COMPOSITE_PARTITION_NAME:='default';
  FILTER.COMPOSITE_NAME := 'FlatStructure';
  FILTER.COMPOSITE_REVISION := '10.0';
  FILTER.STATE := fabric.STATE_UNKNOWN;
  FILTER.MIN_CREATED_DATE := to_timestamp('2010-09-07','YYYY-MM-DD');
  FILTER.MAX_CREATED_DATE := to_timestamp('2010-09-08','YYYY-MM-DD');
  MAX_INSTANCES := 1000;
  .
  DELETED_INSTANCES := FABRIC.DELETE_COMPOSITE_INSTANCES(
    FILTER => FILTER,
    MAX_INSTANCES => MAX_INSTANCES,
    PURGE_PARTITIONED_DATA => PURGE_PARTITIONED_DATA
  );

```

This deletes the first 1000 instances of the FlatStructure composite (version 10) created between '2010-09-07' and '2010-09-08' that are in "UNKNOWN" state. For more information on the possible operations included in the SQL packages provided, see "Managing SOA Composite Applications" in the *Oracle Fusion Middleware Administrator's Guide for Oracle SOA Suite*. Always use the scripts provided for a correct purge. Deleting rows in just the composite_dn table may leave dangling references in other tables used by the Oracle Fusion Middleware SOA Infrastructure. For a more detailed explanation, refer to the SOA 11g Database Growth Management Strategy paper in the Oracle FMW MAA site.

14.4 Configuring UMS Drivers

UMS driver configuration is not automatically propagated in a SOA cluster. To propagate UMS driver configuration in a cluster:

- Apply the UMS driver configuration in each server in the Enterprise Deployment topology that is using the driver.
- If you are using server migration, servers are moved to a different node's domain directory. Pre-create the UMS driver configuration in the failover node. The UMS driver configuration file is located in the following directory:

```

MSERVER_HOME//servers/server_name/ tmp/_WL_user/ums_driver_
name/*/configuration/driverconfig.xml

```

Where '*' represents a directory name that is randomly generated by Oracle WebLogic Server during deployment. For example, 3682yq.

Create the UMS driver configuration file in preparation for possible failovers by forcing a server migration, and copy the file from the source node.

For example, to create the file for BAM:

1. Configure the driver for WLS_BAM1 in BAMHOST1.
2. Force a failover of WLS_BAM1 to BAMHOST2. Verify the following directory structure for the UMS driver configuration in the failover node:

```
cd MSERVER_HOME/servers/server_name/tmp/_WL_user/ums_driver_
name/*/configuration/
```

(where '*' represents a directory whose name is randomly generated by WLS during deployment, for example, "3682yq").

3. Do a remote copy of the driver configuration file from BAMHOST1 to BAMHOST2:

```
BAMHOST1> scp MSERVER_HOME/servers/server_name/tmp/_WL_user/ums_driver_
name/*/configuration/driverconfig.xml
oracle@BAMHOST2:MSERVER_HOME/servers/server_name/tmp/_WL_user/ums_driver_
name/*/configuration/
```

4. Restart the driver for these changes to take effect.

To restart the driver:

- a. Log on to the Oracle WebLogic Administration Console.
- b. Expand the environment node on the navigation tree.
- c. Click on **Deployments**.
- d. Select the driver.
- e. Click **Stop->When work completes** and confirm the operation.
- f. Wait for the driver to transition to the "Prepared" state (refresh the administration console page, if required).
- g. Select the driver again, and click **Start->Servicing all requests** and confirm the operation.

Verify in Oracle Enterprise Manager Fusion Middleware Control that the properties for the driver have been preserved.

14.5 Scaling Up the Topology (Adding Managed Servers to Existing Nodes)

When you scale up the topology, you already have a node that runs a managed server that is configured with Fusion Middleware components, or a managed server with WSM-PM. The node contains a WebLogic Server home and an Oracle Fusion Middleware SOA home in shared storage. Use these existing installations (such as WebLogic Server home, Oracle Fusion Middleware home, and domain directories), when you create the new managed servers called WLS_SOA and WLS_WSM. You do not need to install WLS or SOA binaries at a new location or to run pack and unpack.

This section contains the following topics:

- [Section 14.5.1, "Planning for Scale Up"](#)
- [Section 14.5.2, "Scale-up Procedure for Oracle SOA"](#)
- [Section 14.5.3, "Scale-up Procedure for Oracle Service Bus"](#)

14.5.1 Planning for Scale Up

When you scale up a server that uses server migration, plan for your appropriate capacity and resource allocation needs. Take the following scenario for example:

- Server1 exists in node1 and uses server migration in its cluster with server2 on node2.
- Server3 is added to the cluster in node1 in a scale up operation. It also uses server migration.

In this scenario, a situation may occur where all servers (server1, server2, server3 and admin server) end up running in a node1 or node2. This means each node needs to be designed with enough resources to sustain the worst case scenario where all servers using server migration end in one single node (as defined in the server migration candidate machine configuration).

14.5.2 Scale-up Procedure for Oracle SOA

To scale up the SOA topology:

1. Configure a TX persistent store for the new server in a location visible from the other nodes and according the shared storage recommendations provided in this guide.

- a. From the Administration Console, select *Server_name* and then the **Services** tab.
- b. Under **Default Store**, in **Directory**, enter the path to the directory where the data files are stored:

```
ASERVER_HOME/tlogs
```

2. Using the Oracle WebLogic Server Administration Console, clone WLS_SOA1 or WLS_WSM1 into a new managed server. The source managed server to clone should be one that already exists on the node where you want to run the new managed server.

To clone a managed server:

- a. From the Domain Structure window of the Oracle WebLogic Server Administration Console, expand the **Environment** node and then **Servers**. The Summary of Servers page appears.
- b. Click **Lock & Edit** and select the managed server that you want to clone (for example, WLS_SOA1).
- c. Click **Clone**.
- d. Name the new managed server WLS_SOA n , where n is a number that identifies the new managed server. In this case, you are adding a new server to Node 1, where WLS_SOA1 was running.

For the remainder of the steps, you are adding a new server to SOAHOST1, which is already running WLS_SOA1.

3. For the listen address, assign the host name or IP to use for this new managed server. For the SOA Servers' listen address, assign a new floating IPoIB host name to use for this new managed server. This is the default listen address for the new server and is an Exalogic-rack internal floating address. The virtual IP should be different from the one used by the managed server that is already running.

Note: For WLS_WSM servers, you can use a different port and the same listen address used for the existing WSM server, since WSM servers do not use server migration. Run the Java Object Cache configuration utility again to include the new server in the JOC distributed cache as described in [Section 8.17, "Configuring the Java Object Cache for Oracle WSM."](#) You can use the same discover port for multiple WLS_WSM servers in the same node. Repeat the steps provided in [Section 8.17](#) for each WLS_WSM server and the server list is updated.

4. Create JMS servers for SOA and UMS on the new managed server.

Note: You do not have to create JMS servers for SOA and UMS on the new managed server if you are scaling up the WLS_WSM managed server. This procedure is required only if you are scaling up the WLS_SOA managed servers.

To create the JMS servers for SOA and UMS:

- a. Use the Oracle WebLogic Server Administration Console to create two new persistent stores named **SOAJMSFileStore_n** and **PS6SOAJMSFileStore_auto_N** for the new SOAJMSServer and PS6SOAJMSServer_auto_n JMS servers (which will be created in a later step). Specify the path for the store as recommended in [Chapter 4, "Configuring Storage for an Exalogic Enterprise Deployment"](#) as the directory for the JMS persistent stores:

ASERVER_HOME/jms

- b. Create a two new JMS servers for SOA named **SOAJMSServer_n** and **PS6SOAJMSServer_auto_n**. Use the SOAJMSFileStore_n and PS6SOAJMSFileStore_auto_N for these JMS servers. Target the JMS servers to the recently created managed server (WLS_SOA*n*).
- c. Create a new persistence store for the new UMS JMS server (which will be created in a later step) and name it, for example, **UMSJMSFileStore_N**. Specify the path for the store as recommended in [Chapter 4, "Configuring Storage for an Exalogic Enterprise Deployment"](#) as the directory for the JMS persistent stores:

ASERVER_HOME/jms

Note: It is also possible to assign SOAJMSFileStore_N as the store for the new UMS JMS servers. For the purpose of clarity and isolation, individual persistent stores are used in the following steps.

- d. Create a new JMS Server for UMS: for example, **UMSJMSServer_N**. Use the UMSJMSFileStore_N for this JMS server. Target the UMSJMSServer_N server to the recently created managed server (WLS_SOA*n*).
- e. **For BPM Systems only:** Create two new persistent stores named **BPMJMSFileStore_n** and **AGJMSFileStore_auto_n** for the new BPMJMSServer_N and AGJMSServer_auto_n JMS servers (which will be created in a later step). Specify the path for the store as recommended in

Chapter 4, "Configuring Storage for an Exalogic Enterprise Deployment" as the directory for the JMS persistent stores:

```
ASERVER_HOME/jms
```

Note: This directory must exist before the managed server is started, or the start operation fails.

You can also assign `SOAJMSFileStore_N` as store for the new BPM JMS Servers. For the purpose of clarity and isolation, individual persistent stores are used in the following steps.

- f. **For BPM systems only:** Create two new JMS Servers for BPM named **BPMJMSServer_N** and **AGJMSServer_auto_n**. Use the `BPMJMSSFileStore_N` and `AGJMSSFileStore_auto_n` for these JMS Servers. Target these servers to the recently created managed server (`WLS_SOAn`).
- g. Target the `UMSJMSSystemResource` to the `SOA_Cluster` as it may have changed during extend operations. To do this, expand the **Services** node and then expand the **Messaging** node. Choose JMS Modules from the Domain Structure window of the Oracle WebLogic Server Administration Console. The JMS Modules page appears. Click `UMSJMSSystemResource` and open the Targets tab. Make sure all of the servers in the `SOA_Cluster` appear selected (including the recently cloned `WLS_SOAn`).
- h. Update the SubDeployment Targets for SOA, UMS, and BPM JMS Modules (if applicable) to include the recently created JMS servers.

To do this, expand the **Services** node and then expand the **Messaging** node. Choose **JMS Modules** from the **Domain Structure** window of the Oracle WebLogic Server Administration Console. The JMS Modules page appears. Click on the JMS module (for SOA: `SOAJMSModule`, for BPM: `BPMJMSSModule` and for UMS: `UMSSystemResource` or `SOA: SOAJMSModule` and for UMS: `UMSSystemResource`) represented as a hyperlink in the **Names** column of the table. The Settings page for module appears. Open the **SubDeployments** tab. The subdeployment for the deployment module appears.

Note: This subdeployment module name is a random name in the form of `SOAJMSServerXXXXXX`, or `UMSJMSServerXXXXXX`, or `BPMJMSServerXXXXXX`, resulting from the Configuration Wizard JMS configuration for the first two servers (`WLS_SOA1` and `WLS_SOA2`).

Click on it. Add the new JMS Server (for UMS add `UMSJMSServer_N`, for SOA add `SOAJMSServer_N`, for BPM add `BPMJMSServer_N`).

5. Configuring Oracle Coherence for deploying composites for the new server as described in [Section 9.4, "Configuring Oracle Coherence for Deploying Composites."](#)

Note: Only the **localhost** field must be changed for the server.
Replace the localhost with the listen address of the new server added:

```
Dtangosol.coherence.localhost=SOAHOST1-PRIV-Vn
```

6. Update the cluster address to include the new server:
 - a. In the Administration Console, select **Environment**, and then **Cluster**.
 - b. Click the **SOA_Cluster** server.
 - c. Click **Lock & Edit**.
 - d. Add the new server's address and port to the **Cluster address** field. For example:

```
SOAHOST1-PRIV-V1:8011,SOAHOST2-PRIV-V1:8001,SOAHOST1-PRIV-Vn
```

- e. Save and activate the changes.
7. Disable host name verification for the new managed server.

Before starting and verifying the WLS_SOAn managed server, you must disable host name verification. You can re-enable it after you have configured server certificates for the communication between the Oracle WebLogic Administration Server and the Node Manager in SOAHOSTn.

If the source server from which the new one has been cloned had already disabled hostname verification, these steps are not required (the hostname verification settings is propagated to the cloned server). To disable host name verification:

- a. In the Oracle Fusion Middleware Enterprise Manager Console, select **Oracle WebLogic Server Administration Console**.
- b. In the Domain Structure window, expand the **Environment** node and then click **Servers**.
The Summary of Servers page appears.
- c. Click the name of the server (represented as a hyperlink) in the **Name** column of the table for which you want to configure migration.
The settings page for the selected server appears.
- d. Click the **SSL** tab, and click **Advanced**.
- e. Set Hostname Verification to **None**.
- f. Click **Save**.

Note: Add new virtual IP addresses to key stores and change server identity (private key alias) when you are using host verification.

8. Create the appropriate HTTP, T3 and Replication channels for the new server. See sections [Section 9.6, "Configuring Network Channels for HTTP and T3 Clients Through EoIB,"](#) and [Section 9.11, "Enabling Cluster-Level Session Replication Enhancements."](#)

After cloning, the managed server channels' listed server listeners will be incorrect. Enter the appropriate server listeners.

9. Add the new server's listen address to the `origin-server-pool-1` in Oracle Traffic Director. See section [Section 7.7, "Defining Oracle Traffic Director Virtual Servers for an Exalogic Enterprise Deployment"](#) for details.

10. Reconfigure the JMS Adapter with the new server using the `FactoryProperties` field in the Administration Console. Click on the corresponding cell under the **Property** value and enter the following:

```
java.naming.factory.initial=weblogic.jndi.WLInitialContextFactory;java.naming.provider.url=t3://SOAHOST1-PRIV-V1:8001,SOAHOST2-PRIV-V1:8001,SOAHOST1-PRIV-Vn:8001;java.naming.security.principal=weblogic;java.naming.security.credentials=weblogic
```

Click **Save and Activate**.

For more information about changing the `FactoryProperties` value, see section [Section 9.12.2, "Enabling High Availability for Oracle JMS Adapters."](#)

11. Configure server migration for the new managed server. To configure server migration using the Oracle WebLogic Server Administration Console:

Note: Because this is a scale-up operation, the node should already contain a Node Manager and environment configured for server migration that includes netmask, interface, wlsifconfig script superuser privileges, and so on. The floating IP for the new SOA managed server should also be already present.

- a. In the Domain Structure window, expand the **Environment** node and then click **Servers**. The Summary of Servers page appears.
- b. Click the name of the server (represented as a hyperlink) in **Name** column of the table for which you want to configure migration.

The settings page for the selected server appears.

- c. Click the **Migration** subtab.
- d. In the **Migration Configuration** section, select the servers that participate in migration in the **Available** window by clicking the right arrow. Select the same migration targets as for the servers that already exist on the node.

For example, for new managed servers on SOAHOST1, which is already running WLS_SOA1, select SOAHOST2. For new managed servers on SOAHOST2, which is already running WLS_SOA2, select SOAHOST1.

Note: The appropriate resources must be available to run the managed servers concurrently during migration.

- e. Choose the **Automatic Server Migration Enabled** option and click **Save**. This enables the Node Manager to start a failed server on the target node automatically.

- f. If the new server's listen address does not fall into the range defined in `nodemanager.properties`, update it accordingly:

```
bond0=SOAHOST1-PRIV-V1-ip- soahost1-priv-vn-ip,NetMask=255.255.248.0
```

For more information see [Section 12.2.2, "Editing the Node Manager Property File."](#)

- g. Restart the Administration Server, managed servers, and Node Manager.

To restart the Administration Server, use the procedure in [Section 8.5.3, "Starting the Administration Server on SOAHOST1."](#)

12. Test server migration for this new server. To test migration, perform the following from the node where you added the new server:

- a. Stop the WLS_SOAn managed server using the following command:

```
kill -9 pid
```

You can identify the PID (process ID) of the node using the following command:

```
ps -ef | grep WLS_SOAn
```

- b. Monitor the Node Manager Console for a message indicating that WLS_SOAn's floating IP has been disabled.

- c. Wait for the Node Manager to attempt a second restart of WLS_SOAn.

Node Manager waits for a fence period of 30 seconds before trying this restart.

- d. Once Node Manager restarts the server, stop it again.

Node Manager logs a message indicating that the server will not be restarted again locally.

14.5.3 Scale-up Procedure for Oracle Service Bus

You can scale up the Oracle Service Bus servers by adding new managed servers to nodes that are already running one or more managed servers.

Prerequisites

Before scaling up your Oracle Service Bus servers, review the following prerequisites:

- You already have a cluster that runs managed servers configured with Oracle Service Bus components.
- The nodes contain Middleware home, an Oracle HOME (SOA and Oracle Service Bus) and a domain directory for existing managed servers.
- The source managed server you clone already exists on the node where you want to run the new managed server.

You can use the existing installations (the Middleware home, and domain directories) for creating new WLS_OSAn servers. You do not need to install SOA or Oracle Service Bus binaries in a new location, or run pack and unpack.

To scale up the Oracle Service Bus servers:

1. Configure a TX persistent store for the new server in a location visible from the other nodes and according the shared storage recommendations provided in this guide.
 - a. From the Administration Console, select *Server_name* and then the **Services** tab.
 - b. Under **Default Store**, in **Directory**, enter the path to the directory where the data files are stored:

ASERVER_HOME/tlogs

2. Using the Administration Console, clone WLS_OSBn into a new managed server:
 - a. Select **Environment** and then **Servers**.
 - b. Select the managed server that you want to clone (for example, **WLS_OSB1**).
 - c. Select **Clone**.

Name the new managed server **WLS_OSBn**, where n is a number to identify the new managed server.

For these steps you are adding a new server to SOAHOST1, which is already running WLS_OSB1.

3. For the servers' listen address, assign a new floating IPoIB host name to use for this new managed server. This is the default listen address for the new OSB server and is an Exalogic-rack internal floating address. This virtual hostname should be different from the one used by the managed server that is already running.

To set the managed server listen address:

- a. Log into the Oracle WebLogic Server Administration Console.
- b. In the **Change Center**, click **Lock & Edit**.
- c. In the Domain Structure window expand the **Environment** node.
- d. Click **Servers**.

The Summary of Servers page appears.

- e. In the **Names** column of the table, select the managed server with the listen address you want to update.

The Settings page for that managed server appears.

- f. Set the Listen Address to **SOAHOST1-PRIV-Vn** and click **Save**.

Restart the managed server for the change to take effect.

4. Update the cluster address to include the new server:
 - a. In the Administration console, select **Environment**, and then **Cluster**.
 - b. Click the **OSB_Cluster** server.

The Settings Screen for the OSB_Cluster appears.

- c. In the **Change Center**, click **Lock & Edit**.
- d. Add the new server's address and port to the **Cluster Address** field. For example:

```
SOAHOST1-PRIV-V2:8011, SOAHOST2-PRIV-V2:8011, SOAHOST1-PRIV-VN:8011
```

5. Create the appropriate HTTP and T3 channels for the new server.

After cloning, the managed server channels' listed server listeners will be incorrect. Enter the appropriate server listeners.

For more information, see [Section 9.6, "Configuring Network Channels for HTTP and T3 Clients Through EoIB."](#)

6. Add the new server's listen address to the `osb-pool` Oracle Traffic Director.

For more information, see [Section 7.7, "Defining Oracle Traffic Director Virtual Servers for an Exalogic Enterprise Deployment."](#)

7. If your Oracle Service Bus configuration includes one or more business services that use JMS request/response functionality, perform the following procedure using the Oracle Service Bus Console after adding the new managed server to the cluster:
 - a. In the **Change Center**, click **Create** to create a session.
 - b. Using the Project Explorer, locate and select a business service that uses JMS request/response.

Business services of this type display Messaging Service as their Service Type.
 - c. At the bottom of the View Details page, click **Edit**.
 - d. If there is a cluster address in the endpoint URI, add the new server to the cluster address.
 - e. In the Edit a Business Service - Summary page, click **Save**.
 - f. Repeat the previous steps for each remaining business service that uses JMS request/response.
 - g. In the Change Center, click **Activate**.
 - h. Restart the managed server.
 - i. Restart the Administration Server.

The business services are now configured for operation in the extended domain.

Note: For business services that use a JMS MessageID correlation scheme, edit the connection factory settings to add an entry to the table mapping managed servers to queues. For information about configuring queues and topic destinations, see "JMS Server Targeting" in *Oracle Fusion Middleware Configuring and Managing JMS for Oracle WebLogic Server*.

8. If your Oracle Service Bus configuration includes one or more proxy services that use JMS endpoints with cluster addresses, perform the following procedure using the Oracle Service Bus Console after adding the new managed server to the cluster:
 - a. In the **Change Center**, click **Create** to create a session.
 - b. Using the Project Explorer, locate and select a proxy service that uses JMS endpoints with cluster addresses.
 - c. At the bottom of the View Details page, click **Edit**.
 - d. If there is a cluster address in the endpoint URI, add the new server to the cluster address.
 - e. On the Edit a Proxy Service - Summary page, click **Save**.
 - f. Repeat the previous steps for each remaining proxy service that uses JMS endpoints with cluster addresses.
 - g. In the **Change Center**, click **Activate**.
 - h. Restart the managed server.

The proxy services are now configured for operation in the extended domain.

9. Update the Oracle Service Bus result cache Coherence configuration for the new server:

- a. Log into Oracle WebLogic Server Administration Console.
- b. In the **Change Center**, click **Lock & Edit**.
- c. In the **Domain Structure** window, expand the **Environment** node.
- d. Click **Servers**.

The Summary of Servers page appears.

- e. Click the name of the server (a hyperlink) in the **Name** column of the table.
The settings page for the selected server appears.
- f. Click the **Server Start** tab.

Enter the following for WLS_OSBn (on a single line, without a carriage returns):

```
-DOSB.coherence.localhost=soahost1-priv-vn -DOSB.coherence.localport=7890
-DOSB.coherence.wka1=SOAHOST1-PRIV-V2 -DOSB.coherence.wka1.port=7890
-DOSB.coherence.wka2=SOAHOST2-PRIV-V2 -DOSB.coherence.wka1.port=7890
```

Note: For this configuration servers WLS_OSB1 and WLS_OSB2 must be running (listening on Virtual Host Names SOAHOST1VHN and SOAHOST2VHN as used in the rest of the guide) when WLS_OSBn is started. This allows WLS_OSBn to join the coherence cluster started by either WLS_OSB1 or WLS_OSB2 using the WKA addresses specified. In addition, make sure WLS_OSB1 and WLS_OSB2 are started before WLS_OSBn is started when all three servers are restarted. This ensures WLS_OSBn joins the cluster started by one of WLS_OSB1 or WLS_OSB2. If the order in which the servers start is not important, add the host and port for WLS_OSBn as WKA for WLS_OSB1 and WLS_OSB2, and also add WLS_OSBn as WKA for WLS_OSBn.

- g. Save and activate the changes.

Restart the Oracle Service Bus servers for the changes to take effect.

10. Reconfigure the JMS Adapter with the new server using the **FactoryProperties** field in the Administration Console. Click on the corresponding cell under the **Property** value and enter the following:

```
java.naming.factory.initial=weblogic.jndi.WLInitialContextFactory;java.naming.p
rovider.url=t3://SOAHOST1-PRIV-V2:8011,SOAHOST2-PRIV-V2:8011,SOAHOSTn-PRIV-V1:8
011;java.naming.security.principal=weblogic;java.naming.security.credentials=we
blogic1
```

Click **Save and Activate**.

11. Create JMS Servers and persistent stores for Oracle Service Bus reporting/internal destinations on the new managed server.
 - a. Use the Oracle WebLogic Server Administration Console to create a new persistent store for the new WseeJMSServer and name it, for example, **OSB_rep_JMSFileStore_N**. Specify the path for the store. This should be a directory on shared storage, as recommended in [Chapter 4, "Configuring Storage for an](#)

[Exalogic Enterprise Deployment.](#)" For example:

`ASERVER_HOME/jms/`

Target the store the new cloned server (WLS_OSBn).

- b. Create a new JMS Server for Oracle Service Bus, for example, OSB_rep_JMSServer_N. Use the OSB_rep_JMSFileStore_N for this JMSServer. Target the OSB_rep_JMSServer_N Server to the recently created Managed Server (WLS_OSBn).
- c. Update the SubDeployment targets for the "jmsResources" Oracle Service Bus JMS Module to include the recently created OSB JMS Server:

Expand the **Services** node and then expand the **Messaging** node. Choose **JMS Modules** from the **Domain Structure** window of the Oracle WebLogic Server Administration Console. The JMS Modules page appears.

Click **jmsResources** (a hyperlink in the **Names** column of the table). The Settings page for jmsResources appears.

Click the **SubDeployments** tab. The subdeployment module for jmsresources appears.

Note: This subdeployment module name for destinations is a random name in the form of wlsbJMSServerXXXXXX resulting from the Configuration Wizard JMS configuration for the first two servers (WLS_OSB1 and WLS_OSB2).

Click the **wlsbJMSServerXXXXXX** subdeployment and update the targets to include the new OSB_rep_JMSServer_n server.

12. Create JMS Servers, persistent stores and destinations for OSB JAX-RPC on the new managed server.

Note: WebLogic Advanced Web Services for JAX-RPC Extension uses regular (non-distributed) destinations to ensure that a locally processed request on a service gets enqueued only to a local member.

- a. Use the Oracle WebLogic Server Administration Console to create a new persistent store for the new WseeJMSServer and name it, for example, **Wsee_rpc_JMSFileStore_N**. Specify the path for the store. This should be a directory on shared storage, as recommended in [Chapter 4, "Configuring Storage for an Exalogic Enterprise Deployment"](#)
- b. Create a new JMS Server for OSB JAX-RPC, for example, OSB_rpc_JMSServer_N. Use the Wsee_rpc_JMSFileStore_N for this JMSServer. Target the OSB_rpc_JMSServer_N Server to the recently created Managed Server (WLS_OSBn).
- c. Update the WseeJMSModule OSB JMS Module with destinations and the recently created OSB JMS Server by expanding the **Services** node and then expand the **Messaging** node. Choose **JMS Modules** from the Domain Structure window of the Oracle WebLogic Server Administration Console. The JMS Modules page appears. Click **WseeJmsModule** (a hyperlink in the **Names** column of the table). The Settings page for WseeJmsModule appears. Follow steps d through v to complete this step.
- d. In the **Change Center**, click **Lock & Edit** and click **New**.

- e. Select **Queue** and click **Save**.
 - f. Click **Create a New Subdeployment**.
 - g. Accept the default name and click **OK**.
 - h. Select **OSB_rpc_JMSserver_n** as the target and click **Finish**.
 - i. Update the local JNDI name for the destination:
 - In the **Change Center**, click **Lock & Edit**.
 - In the **Settings** for the **WseeJmsModule** page, click the **DefaultCallbackQueue-WseeJmsServer_auto_n** destination.
 - In the general **Configuration** tab, click **Advanced**.
 - Update the local JNDI name to **weblogic.wsee.DefaultCallbackQueue**.
 - j. Repeat steps d through h for the **DefaultQueue-WseeJmsServer_auto_n queue**, using **weblogic.wsee.DefaultQueue-WseeJmsServer_auto_n** as the JNDI name and **weblogic.wsee.DefaultQueue** as the local JNDI name.
13. Create a new SAF agent and target it to the newly added managed server:
 - a. In the Oracle WebLogic Server Administration Console, expand **Services, Messaging**, and then **Store-and-Forward Agents**
 - b. Add a new SAF agent **ReliableWseeSAFAgent_auto_N**.
 - c. Select persistent store **Wsee_rpc_JMSFileStore_N** (persistent store created for OSB JAX-RPC).
 - d. Target the SAF Agent to the new managed server and activate changes.
 14. Disable host name verification for the new managed server. Before starting and verifying the **WLS_OSbn** managed server, disable host name verification. You can re-enable it after you have configured server certificates for the communication between the Oracle WebLogic Administration Server and the Node Manager in **SOAHOSTn**. You can ignore these steps if you have already disabled hostname verification for the source server from which the new server has been cloned (the hostname verification settings is propagated to the cloned server).

To disable host name verification:

 - a. In the Oracle Enterprise Manager Console, select **Oracle WebLogic Server Administration Console**.
 - b. Expand the **Environment** node in the **Domain Structure** window and click **Servers**.

The Summary of Servers page appears.
 - c. Select **WLS_OSbn** in the **Names** column of the table.

The Settings page for the server appears.
 - d. Click the **SSL** tab and click **Advanced**.
 - e. Set **Hostname Verification** to **None** and click **Save**.
 15. If it is not already started, start the Node Manager on the node. To start the Node Manager, use the installation in shared storage from the existing nodes as follows:


```
SOAHOSTN> WL_HOME/server/bin/startNodeManager
```
 16. Start and test the new managed server from the Administration Console.

- a. Shut down the existing managed servers in the cluster.
- b. Ensure that the newly created managed server, **WLS_OSBn**, is up.
- c. Access the application on the newly created managed server using the following URL:

```
http://vip:port/sbinspection.wsil
```

17. Configure Server Migration for the new managed server.

Note: Since this is a scale-up operation, the node should already contain a Node Manager and environment configured for server migration. The floating IP for the new Oracle Service Bus managed server should already be present.

To configure server migration:

- a. Log into the Administration Console.
- b. In the left pane, expand **Environment** and select **Servers**.
- c. Select the name of the new managed server for which you want to configure migration.
- d. Click the **Migration** tab.
- e. In the **Available** field, in the **Migration Configuration** section, select the machines to which migration is allowed and click the right arrow.
- f. Select the same migration targets used for the servers that already exist on the node.

For example, for new managed servers on SOAHOST1, which is already running WLS_OSB1, select SOAHOST2. For new managed servers on SOAHOST2, which is already running WLS_OSB2, select SOAHOST1.

Make sure the appropriate resources are available to run the managed servers concurrently during migration.

- g. Select the **Automatic Server Migration Enabled** option and click **Save**.
This enables the Node Manager to start a failed server on the target node automatically.
- h. If the new server's listen address does not fall into the range defined in `nodemanager.properties`, update it accordingly

```
bond0=SOAHOST1-PRIV-V1-ip- soahost1-priv-vn-ip,NetMask=255.255.248.0
```

For more information, see [Section 13.4, "Editing Node Manager's Properties File."](#)

- i. Restart the Administration Server, managed servers, and Node Manager.
18. Test server migration for this new server from the node where you added the new server:
- a. Stop the **WLS_OSBn** managed server by running the following command on the PID (process ID) of the managed server:

```
kill -9 pid
```

You can identify the PID of the node using the following command:


```
ps -ef | grep WLS_OSBn
```

Note: For Windows, you can terminate the Managed Server using the `taskkill` command. For example:

```
taskkill /f /pid pid
```

Where *pid* is the process ID of the Managed Server.

To determine the process ID of the WLS_OSBn Managed Server, run the following command:

```
MW_HOME\jrockit_160_20_D1.0.1-2124\bin\jps -l -v
```

-
- b. In the Node Manager Console you can see a message appears indicating that WLS_OSBn's floating IP has been disabled.
 - c. Wait for the Node Manager to try a second restart of WLS_OSBn.
Node Manager waits for a fence period of 30 seconds before trying this restart.
 - d. Once Node Manager restarts the server, stop it again.
Node Manager logs a message indicating that the server will not be restarted again locally.

Note: After a server is migrated, to fail it back to its original node, 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.

14.6 Scaling Out the Topology (Adding Managed Servers to New Nodes)

When you scale out the topology, you add new managed servers configured with SOA, OSB, and or WSM-PM to new nodes.

This section contains the following topics:

- [Section 14.6.1, "Prerequisites for Scaling Out the Topology"](#)
- [Section 14.6.2, "Scale-out Procedure for the Oracle SOA"](#)
- [Section 14.6.3, "Scale-out Procedure for Oracle Service Bus"](#)

14.6.1 Prerequisites for Scaling Out the Topology

Before performing the steps in this section, check that you meet these requirements:

- There must be existing nodes running managed servers configured with SOA and WSM-PM within the topology
- The new node can access the existing home directories for WebLogic Server and SOA. (Use the existing installations in shared storage for creating a new WLS_SOA or WLS_WSM managed server. You do not need to install WebLogic Server or SOA binaries in a new location but you do need to run `pack` and `unpack` to bootstrap the domain configuration in the new node.)

- When an ORACLE_HOME or WL_HOME is shared by multiple servers in different nodes, keep the Oracle Inventory and Middleware home list in those nodes updated for consistency in the installations and application of patches. To update the oraInventory in a node and "attach" an installation in a shared storage to it, use the attachHome.sh script in the following location:

```
ORACLE_HOME/oui/bin/
```

To update the Middleware home list to add or remove a WL_HOME, edit the beahomelist file located in the following directory:

```
MW_HOME/bea
```

14.6.2 Scale-out Procedure for the Oracle SOA

To scale out the topology:

1. Configure a TX persistent store for the new server in a location visible from the other nodes and according to the shared storage recommendations provided in this guide.
 - a. From the Administration Console, select *Server_name* and then the **Services** tab.
 - b. Under **Default Store**, in **Directory**, enter the path to the directory where the data files are stored:

```
ASERVER_HOME/tlogs
```

2. On the new node, mount the existing Fusion Middleware Home, and the rest of the private and shared mounts indicated in [Chapter 4, "Configuring Storage for an Exalogic Enterprise Deployment."](#)
3. To attach ORACLE_HOME in shared storage to the local Oracle Inventory, execute the following command:

```
SOAHOSTn>cd ORACLE_COMMON_HOME/oui/bin/attachHome.sh
SOAHOSTn>./attachHome.sh -jreLoc MSERVER_HOME/jrockit_160_<version>
```

To update the Middleware home list, create (or edit, if another WebLogic installation exists in the node) the beahomelist file and add MW_HOME to it. The beahomelist file is located in the following directory:

```
MW_HOME/bea
```

4. Log in to the Oracle WebLogic Administration Console.
5. Create a new machine for the new node that will be used, and add the machine to the domain.
6. Update the machine's Node Manager's address to map the private IPoIB of the node that is being used for scale out.
7. Use the Oracle WebLogic Server Administration Console to clone WLS_SOA1/WLS_WSM1 into a new managed server. Name it WLS_SOAn/WLS_WSMn, where *n* is a number. Assign it to the new machine created above.

Note: These steps assume that you are adding a new server to node *n*, where no managed server was running previously.

8. Assign the host name or IP to use for the new managed server for the listen address of the managed server.

If you are planning to use server migration for this server (which Oracle recommends) this should be the virtual IP (also called a floating IP) for the server. This virtual IP should be different from the one used for the existing managed server. For example SOAHOSTn-PRIV-V1.

9. For WLS_WSM servers, run the Java Object Cache configuration utility again to include the new server in the JOC distributed cache as described in [Section 8.17, "Configuring the Java Object Cache for Oracle WSM."](#)
10. Create JMS Servers for SOA and UMS on the new managed server.

Note: You do not have to create JMS servers for SOA and UMS on the new managed server if you are scaling up the WSM_WSM managed server or the BAM Web Applications system. This procedure is required only if you are scaling up the WLS_SOA managed servers

Create the JMS servers for SOA and UMS as follows:

- a. Use the Oracle WebLogic Server Administration Console to create two new persistent stores named **SOAJMSFileStore_n** and **PS6SOAJMSFileStore_auto_N** for the new SOAJMSServer and PS6SOAJMSServer_auto_n JMS servers (which will be created in a later step). Specify the path for the store as recommended in [Chapter 4, "Configuring Storage for an Exalogic Enterprise Deployment"](#) as the directory for the JMS persistent stores:


```
ASERVER_HOME/jms/
```
- b. Create a two new JMS servers for SOA named **SOAJMSServer_n** and **PS6SOAJMSServer_auto_n**. Use the SOAJMSFileStore_n and PS6SOAJMSFileStore_auto_N for these JMS servers. Target the JMS servers to the recently created managed server (WLS_SOA*n*).
- c. Create a new persistence store for the new UMSJMSServer, and name it, for example, **UMSJMSFileStore_N**. As the directory for the persistent store, specify the path recommended in [Chapter 4, "Configuring Storage for an Exalogic Enterprise Deployment"](#) as the directory for the JMS persistent stores:


```
ASERVER_HOME/jms
```

Note: It is also possible to assign SOAJMSFileStore_N as the store for the new UMS JMS servers. For the purpose of clarity and isolation, individual persistent stores are used in the following steps.

- d. Create a new JMS server for UMS: for example, **UMSJMSServer_N**. Use the UMSJMSFileStore_N for this JMS server. Target the UMSJMSServer_N Server to the recently created managed server (WLS_SOA*n*).
- e. **For BPM Systems only:** Create two new persistent stores named **BPMJMSFileStore_n** and **AGJMSFileStore_auto_n** for the new BPMJMSServer_N and AGJMSServer_auto_n JMS servers (which will be created in a later step). Specify the path for the store as recommended in [Chapter 4, "Configuring Storage for an Exalogic Enterprise Deployment"](#) as the directory for the JMS persistent stores:

- f. **For BPM systems only:** Create two new JMS Servers for BPM named **BPMJMSServer_N** and **AGJMSServer_auto_n**. Use the **BPMJMSFileStore_N** and **AGJMSFileStore_auto_n** for these JMS Servers. Target these servers to the recently created managed server (**WLS_SOAn**).
- g. Update the SubDeployment Targets for SOA, UMS, and BPM JMS Modules (if applicable) to include the recently created JMS servers.
- h. To do this, expand the **Services** node and then expand the **Messaging** node. Choose **JMS Modules** from the **Domain Structure** window of the Oracle WebLogic Server Administration Console. The **JMS Modules** page appears. Click on the JMS module (for SOA: **SOAJMSModule**, for BPM: **BPMJMSSModule** and for UMS: **UMSSYtemResource**) represented as a hyperlink in the **Names** column of the table. The **Settings** page for the module appears. Open the **SubDeployments** tab. The subdeployment for the deployment module appears.

Note: This subdeployment module name is a random name in the form of **SOAJMSServerXXXXXX**, **UMSJMSServerXXXXXX**, or **BPMJMSServerXXXXXX**, resulting from the Configuration Wizard JMS configuration for the first two servers (**WLS_SOA1** and **WLS_SOA2**).

Click on it. Add the new JMS Server (for UMS add **UMSJMSServer_N**, for SOA add **SOAJMSServer_N**, for BPM add **BPMJMSServer_N**). Click **Save and Activate**.

- i. Target the **UMSJMSSystemResource** to the **SOA_Cluster** as it may have changed during extend operations. To do this, expand the **Services** node and then expand the **Messaging** node. Choose **JMS Modules** from the **Domain Structure** window of the Oracle WebLogic Server Administration Console. The **JMS Modules** page appears. Click **UMSJMSSystemResource** and open the **Targets** tab. Make sure all of the servers in the **SOA_Cluster** appear selected (including the recently cloned **WLS_SOAn**).
- j. Update the SubDeployment Targets for SOA, UMS, and BPM JMS Modules (if applicable) to include the recently created JMS servers.

To do this, expand the **Services** node and then expand the **Messaging** node. Choose **JMS Modules** from the **Domain Structure** window of the Oracle WebLogic Server Administration Console. The **JMS Modules** page appears. Click on the JMS module (for SOA: **SOAJMSModule** and for UMS: **UMSSYtemResource**, for BRM: **BPMJMSSModule**) represented as a hyperlink in the **Names** column of the table. The **Settings** page for module appears. Open the **SubDeployments** tab. The subdeployment for the deployment module appears.

Note: This subdeployment module name is a random name in the form of **SOAJMSServerXXXXXX**, **UMSJMSServerXXXXXX**, or **BPMJMSServerXXXXXX** resulting from the Configuration Wizard JMS configuration for the first two servers (**WLS_SOA1** and **WLS_SOA2**).

Click on it. Add the new JMS Server (for UMS add **UMSJMSServer_N**, for SOA add **SOAJMSServer_N**, for BPM add **BPMJMSServer_N**).

11. Configure Node Manager directory and properties for the new node as indicated in [Section 8.5.2, "Configuring and Starting Node Manager on SOAHOST1 and SOAHOST2."](#)

12. Run the pack command on SOAHOST1 to create a template pack as follows:

```
cd ORACLE_COMMON_HOME/common/bin

./pack.sh -managed=true -domain=ASERVER_HOME
-template=soadomaintemplateScale.jar -template_name=soa_domain_templateScale
```

Run the unpack command on SOAHOST n to unpack the template in the managed server domain directory as follows:

```
SOAHOSTN> cd ORACLE_COMMON_HOME/common/bin

SOAHOSTN> ./unpack.sh -domain=MSERVER_HOME
-template=soadomaintemplateScale.jar
-app_dir=APP_DIR
```

Note: The configuration steps provided in this enterprise deployment topology are documented with the assumption that a private (per node) domain directory is used for each managed server.

13. Configure Oracle Coherence for deploying composites for the new server as described in [Section 9.4, "Configuring Oracle Coherence for Deploying Composites."](#)

Note: Only the **localhost** field needs to be changed for the server. Replace the localhost with the listen address of the new server added:

```
Dtangosol.coherence.localhost=SOAHOSTnVHN1-PRIV-V1
```

14. Reconfigure the JMS Adapter with the new server using the **FactoryProperties** field in the Administration Console. Click on the corresponding cell under the **Property** value and enter the following:

```
java.naming.factory.initial=weblogic.jndi.WLInitialContextFactory;java.naming.provider.url=t3://SOAHOST1-PRIV-V1:8001,SOAHOST2-PRIV-V1:8001,SOAHOSTn-PRIV-V1;java.naming.security.principal=weblogic;java.naming.security.credentials=weblogic1
```

Click **Save and Activate**.

15. Configure the persistent store for the new server. This should be a location visible from other nodes as recommended in [Chapter 4, "Configuring Storage for an Exalogic Enterprise Deployment."](#)

From the Administration Console, select the **Server_name**, and then the **Services** tab. Under **Default Store**, in **Directory**, enter the path to the folder where you want the default persistent store to store its data files.

16. Update the cluster address to include the new server:
 - a. In the Administration Console, select Environment, and then Cluster.
 - b. Click the SOA_Cluster server.

- c. Click **Lock & Edit**.
 - d. Add the new server's address and port to the **Cluster address** field. For example:


```
SOAHOST1-PRIV-V1:8011, SOAHOST2-PRIV-V1:8001, SOAHOSTn-PRIV-V1
```
 - e. Save and Activate the changes.
17. Disable host name verification for the new managed server. Before starting and verifying the `WLS_SOAn` managed server, you must disable host name verification. You can re-enable it after you have configured server certificates for the communication between the Oracle WebLogic Administration Server and the Node Manager in `SOAHOSTn`.

If the source server from which the new one has been cloned had already disabled hostname verification, these steps are not required (the hostname verification settings is propagated to the cloned server).

To disable host name verification:

- a. In the Oracle Fusion Middleware Enterprise Manager Console, select **Oracle WebLogic Server Administration Console**.
 - b. Expand the **Environment** node in the **Domain Structure** window.
 - c. Click **Servers**.

The Summary of Servers page appears.
 - d. Select `WLS_SOAn` in the **Names** column of the table.

The Settings page for server appears.
 - e. Click the **SSL** tab.
 - f. Click **Advanced**.
 - g. Set Hostname Verification to **None**.
 - h. Click **Save**.
18. Create the appropriate HTTP, T3 and Replication channels for the new server. For details, see [Section 9.6, "Configuring Network Channels for HTTP and T3 Clients Through EoIB,"](#) and [Section 9.11, "Enabling Cluster-Level Session Replication Enhancements."](#)
19. Add the new server's listen address to the `origin-server-pool-1` in Oracle Traffic Director. For details, see [Section 7.7, "Defining Oracle Traffic Director Virtual Servers for an Exalogic Enterprise Deployment."](#)

Reconfigure the JMS Adapter with the new server using the **FactoryProperties** field in the Administration Console. Click on the corresponding cell under the **Property** value and enter the following:

```
java.naming.factory.initial=weblogic.jndi.WLInitialContextFactory;java.naming.provider.url=t3://SOAHOST1-PRIV-V1:8001, SOAHOST2-PRIV-V1:8001, SOAHOSTN-PRIV-V1:8001;java.naming.security.principal=weblogic;java.naming.security.credentials=weblogic
```

- 20. Click **Save and Activate**.
- 21. Start Node Manager on the new node. To start Node Manager, use `startNodeManager.sh` located in the following directory:

```
SOAHOSTn> /u02/private/oracle/config/nodemanager/startNodeManager.sh
```

22. Start and test the new managed server from the Oracle WebLogic Server Administration Console.
 - a. Ensure that the newly created managed server, WLS_SOAn, is running.
 - b. Access the application from within the Exalogic rack using the following URL:
`http://SOAHOSTn-PRIV-V1:8001/soa-infra/`

The application should be functional.

23. Configure server migration for the new managed server.

Log into the Oracle WebLogic Server Administration Console and configure server migration.

To configure server migration:

- a. Expand the **Environment** node in the Domain Structure windows and then choose Servers. The Summary of Servers page appears.
- b. Select the server (represented as hyperlink) for which you want to configure migration from the Names column of the table. The Setting page for that server appears.
- c. Click the **Migration** tab.
- d. In the Available field of the Migration Configuration section, click the right arrow to select the machines to which to allow migration.

Note: Specify the least-loaded machine as the migration target for the new server. The required capacity planning must be completed so that this node has enough available resources to sustain an additional managed server.

- e. Select **Automatic Server Migration Enabled**. This enables Node Manager to start a failed server on the target node automatically.
- f. Click **Save**.
- g. Restart the Administration Server, managed servers, and the Node Manager.
 To restart the Administration Server, use the procedure in [Section 8.5.3, "Starting the Administration Server on SOAHOST1."](#)

24. Update the cluster address to include the new server:

- a. In the Administration Console, select **Environment**, and then **Cluster**.
- b. Click the **SOA_Cluster** server.
 The Settings screen for the SOA_Cluster appears.
- c. Click **Lock & Edit**.
- d. Add the new server's address and port to the **Cluster address** field. For example:

`SOAHOST1-PRIV-V1:8001, SOAHOST2-PRIV-V1:8001, SOAHOSTn-PRIV1:8001`

- e. Save and activate the changes.
25. Test server migration for this new server from the node where you added the new server:

- a. Abruptly stop the `WLS_SOA n` managed server by running the following command;

```
kill -9 pid
```

You can identify the PID (process ID) of the node using the following command:

```
ps -ef | grep WLS_SOA $n$ 
```

- b. In the Node Manager Console you should see a message indicating that `WLS_SOA1`'s floating IP has been disabled.
- c. Wait for the Node Manager to try a second restart of `WLS_SOA n` . Node Manager waits for a fence period of 30 seconds before trying this restart.
- d. Once Node Manager restarts the server, stop it again. Now Node Manager should log a message indicating that the server will not be restarted again locally.

14.6.3 Scale-out Procedure for Oracle Service Bus

When you scale out the topology, you add new managed servers configured with Oracle Service Bus to the new nodes.

Prerequisites

Before scaling out the Oracle Service Bus topology, make sure you meet these prerequisites:

- There must be existing nodes running managed servers configured with Oracle Service Bus within the topology.
- The new node optionally can access the existing home directories for WebLogic Server and Oracle Service Bus installation. Use the existing installations in shared storage for creating a new `WLS_OSB` managed server. You do not need to install WebLogic Server or Oracle Service Bus binaries in every new location in this case, but you do need to run the `pack` and `unpack` commands to bootstrap the domain configuration in the new node, unless you are scaling the Oracle Service Bus server to machines containing other servers of the same domain (the SOA servers).
- When multiple servers in different nodes share an `ORACLE_HOME` or `WL_HOME`, keep the Oracle Inventory and Middleware home list in those nodes updated for consistency in the installations and application of patches. To update the `oraInventory` in a node and attach an installation in a shared storage to it, use the `attachHome.sh` file located in the following directory:

```
ORACLE_HOME/oui/bin/
```

To update the Middleware home list to add or remove a `WL_HOME`, edit the `beahomelist` file located in the following directory:

```
MW_HOME/bea
```

To scale out the topology:

1. Configure a TX persistent store for the new server in a location visible from the other nodes and according the shared storage recommendations provided in this guide.
 - a. From the Administration Console, select *Server_name* and then the **Services** tab.

- b. Under **Default Store**, in **Directory**, enter the path to the directory where the data files are stored:

```
ASERVER_HOME/tlogs
```

2. On the new node, mount the existing Fusion Middleware Home, and the rest of the private and shared mounts as described in [Chapter 4, "Configuring Storage for an Exalogic Enterprise Deployment."](#)
3. Attach ORACLE_HOME in shared storage to the private Oracle Inventory using the following command:

```
SOAHOSTn>cd MW_HOME/soa/
SOAHOSTn>./attachHome.sh -jreLoc MW_HOME/jrockit_160_<version>
```

To update the Middleware home list, create (or edit, if another WebLogic installation exists in the node) the `beahomelist` file located in the following directory:

```
MW_HOME/bea/
```

Add `MW_HOME` to the list.

4. Log in to the Oracle WebLogic Administration Console.
5. Create a new machine for the new node that will be used, and add the machine to the domain.
6. Update the machine's Node Manager's address to map the private IPoIB of the node that is being used for scale out.
7. Use the Oracle WebLogic Server Administration Console to clone `WLS_OSB1` into a new managed server. Name it **WLS_OSBn**, where *n* is a number, and assign it to the new machine.

Note: For these steps, you are adding a new server to node *n*, where no managed server was running previously.

8. For the listen address, assign the virtual host name to use for this new managed server. If you are planning to use server migration as recommended for this server, this virtual host name allows it to move to another node. The virtual host name should be different from those used by other managed servers (may be in the same or different domain) that are running in the nodes used by the OSB/SOA domain.
 - a. Log into the Oracle WebLogic Server Administration Console.
 - b. In the **Change Center**, click **Lock & Edit**.
 - c. Expand the **Environment** node in the Domain Structure window.
 - d. Click **Servers**.
The Summary of Servers page appears.
 - e. Select the managed server with listen addresses you want to update in the **Names** column of the table.
The Setting page for that managed server appears.
 - f. Set the Listen Address to **SOAHOSTn-PRIV-V1** and click **Save**.
 - g. Save and activate the changes.
 - h. Restart the managed server.

9. Update the cluster address to include the new server:
 - a. Select **Environment**, and then **Cluster** from the Administration Console.
 - b. Click the **OSB_Cluster** server.
The Settings Screen for the OSB_Cluster appears.
 - c. In the **Change Center**, click **Lock & Edit**.
 - d. Add the new server's address and port to the **Cluster Address** field. For example:

```
SOAHOST1-PRIV-1:8011, SOAHOST2-PRIV-1:8011, SOAHOSTn-PRIV-1:8011
```
10. Create the appropriate HTTP and T3 channels for the new server.
For more information, see [Section 9.6, "Configuring Network Channels for HTTP and T3 Clients Through EoIB."](#)
11. Add the new server's listen address to the Oracle Traffic Director `osb-pool`.
For more information, see [Section 7.7, "Defining Oracle Traffic Director Virtual Servers for an Exalogic Enterprise Deployment."](#)
12. Create JMS servers and persistent stores for Oracle Service Bus reporting/internal destinations on the new managed server.
 - a. Use the Oracle WebLogic Server Administration Console to create a new persistent store for the new **WseeJMSServer** and name it, for example, **OSB_rep_JMSFileStore_N**. Specify the path for the store. This should be a directory on shared storage as recommended in [Chapter 4, "Configuring Storage for an Exalogic Enterprise Deployment."](#)

Note: This directory must exist before the managed server is started or the start operation fails.

```
ASERVER_HOME/jms/OSB_rep_JMSFileStore _N
```

- b. Create a new JMS Server for Oracle Service Bus, for example, **OSB_rep_JMSServer_N**. Use the **OSB_rep_JMSFileStore_N** for this JMSServer. Target the **OSB_rep_JMSServer_N** Server to the recently created managed server (**WLS_OSBn**).
- c. Update the **SubDeployment** targets for the **jmsresources** Oracle Service Bus JMS Module to include the recently created Oracle Service Bus JMS Server:
Expand the **Services** node and then expand the **Messaging** node.
Choose JMS Modules from the **Domain Structure** window of the Oracle WebLogic Server Administration Console. The JMS Modules page appears.
Click **jmsresources** (a hyperlink in the **Names** column of the table). The Settings page for `jmsResources` appears.
Open the **SubDeployments** tab. The subdeployment module for `jmsresources` appears.

Note: This subdeployment module name is a random name in the form of `wlsbjMSServerXXXXXX` resulting from the Configuration Wizard JMS configuration for the first two servers (WLS_OSB1 and WLS_OSB2).

Click the `wlsbjMSServerXXXXXX` subdeployment and update the targets to include the new `OSB_rep_JMSServer_N` server.

13. Create JMS Servers, persistent stores and destinations for OSB JAX-RPC on the new managed server.

Note: WebLogic Advanced Web Services for JAX-RPC Extension uses regular (non-distributed) destinations to ensure that a locally processed request on a service gets enqueued only to a local member.

- a. Use the Oracle WebLogic Server Administration Console to create a new persistent store for the new `WseeJMSServer` and name it, for example, `Wsee_rpc_JMSFileStore_N`. Specify the path for the store. This should be a directory on shared storage as recommended in [Chapter 4, "Configuring Storage for an Exalogic Enterprise Deployment."](#)

Note: This directory must exist before the managed server is started or the start operation fails.

`ASERVER_HOME/jms/Wsee_rpc_JMSFileStore_N`

- b. Create a new JMS Server for Oracle Service Bus JAX-RPC, for example, `OSB_rpc_JMSServer_N`. Use the `Wsee_rpc_JMSFileStore_N` for this JMSServer. Target the `OSB_rpc_JMSServer_N` Server to the recently created Managed Server (`WLS_OSBn`).
- c. Update the `WseeJMSModule` Oracle Service Bus JMS Module with destinations and the recently created Oracle Service Bus JMS Server:
- Expand the **Services** node and then expand the **Messaging** node. Choose **JMS Modules** from the **Domain Structure** window of the Oracle WebLogic Server Administration Console. The JMS Modules page appears.
- Click `WseeJmsModule` (a hyperlink in the **Names** column of the table). The Settings page for `WseeJmsModule` appears.
- Follow steps d through j to complete this step.
- d. In the **Change Center**, click **Lock & Edit** and click **New**.
- e. Select **Queue** and click **Next**.
- f. Enter `DefaultCallbackQueue-WseeJmsServer_auto_n` as name for the queue.
- g. Enter `weblogic.wsee.DefaultCallbackQueue-WseeJmsServer_auto_n` as the JNDI name and click **Next**.
- h. Click **Create a New Subdeployment**.
- i. Accept the default name and click **OK**.
- j. Select `OSB_rpc_JMSServer_n` as the target and click **Finish**.

Note: For business services that use a JMS MessageID correlation scheme, edit the connection factory settings to add an entry to the table mapping managed servers to queues. For information on how to configure queues and topic destinations, see "JMS Server Targeting" in *Oracle Fusion Middleware Configuring and Managing JMS for Oracle WebLogic Server*.

18. If your Oracle Service Bus configuration includes one or more proxy services that use JMS endpoints with cluster addresses, perform the following procedure using the Oracle Service Bus Console after adding the new managed server to the cluster:
- a. In the **Change Center**, click **Create** to create a session.
 - b. Using the Project Explorer, locate and select a proxy service that uses JMS endpoints with cluster addresses.
 - c. At the bottom of the View Details page, click **Edit**.
 - d. If there is a cluster address in the endpoint URI, add the new server to the cluster address.
 - e. On the Edit a Proxy Service - Summary page, click **Save**.
 - f. Repeat the previous steps for each remaining proxy service that uses JMS endpoints with cluster addresses.
 - g. In the **Change Center**, click **Activate**.
 - h. Restart the managed server.

The proxy services are now configured for operation in the extended domain.

19. Update the Oracle Service Bus result cache Coherence configuration for the new server:
- a. Log into Oracle WebLogic Server Administration Console. In the **Change Center**, click **Lock & Edit**.
 - b. In the **Domain Structure** window, expand the **Environment** node.
 - c. Click **Servers**.
- The Summary of Servers page appears.
- d. Click the name of the server (a hyperlink) in the **Name** column of the table.
- The settings page for the selected server appears.
- e. Click the **Server Start** tab.
 - f. Click **Advanced**.
 - g. Enter the following for WLS_OSBn (on a single line, without a carriage returns):

```
-DOSB.coherence.localhost=SOAHOSTn-PRIV-1 -DOSB.coherence.localport=7890
-DOSB.coherence.wka1=SOAHOST1-PRIV-1 -DOSB.coherence.wka1.port=7890
-DOSB.coherence.wka2=SOAHOST2-PRIV-1 -DOSB.coherence.wka1.port=7890
```

Note: For the previous configuration, servers WLS_OSB1 and WLS_OSB2 are running when WLS_OSBn starts. This allows WLS_OSBn to join the coherence cluster started by either WLS_OSB1 or WLS_OSB2 using the WKA addresses specified. In addition, make sure WLS_OSB1 and WLS_OSB2 are started before WLS_OSBn is started when starting all three servers. This ensures WLS_OSBn joins the cluster started by either WLS_OSB1 or WLS_OSB2. For a configuration where the order in which the servers are started does not matter, add the host and port for WLS_OSBn as WKA for WLS_OSB1 and WLS_OSB2, and also add WLS_OSBn as WKA for WLS_OSBn.

h. Save and activate the changes

Restart the Oracle Service Bus servers.

- 20.** Reconfigure the JMS Adapter with the new server using the **FactoryProperties** field in the Administration Console. Click on the corresponding cell under the **Property** value and enter the following:

```
java.naming.factory.initial=weblogic.jndi.WLInitialContextFactory;java.naming.provider.url=t3://soahost1-priv-1:8001,soahost2-priv-1:8001;java.naming.security.principal=weblogic;java.naming.security.credentials=weblogic1
```

Click **Save and Activate**.

- 21.** Configure NodeManager directory and properties for the new node as indicated in [Section 8.5.2, "Configuring and Starting Node Manager on SOAHOST1 and SOAHOST2."](#)
- 22.** Run the pack command on SOAHOST1 to create a template pack as follows:

```
cd $ORACLE_COMMON_HOME/common/bin

./pack.sh -managed=true -domain=$MW_HOME/user_projects/domains/soadomain/
-template=soadomaintemplateScale.jar -template_name=soa_domain_templateScale
```

Run the unpack command on SOAHOSTN to unpack the template in the managed server domain directory as follows:

```
cd $MW_HOME/soa/common/bin

./unpack.sh -domain=$MSERVER_HOME/ -template=soadomaintemplateScale.jar
```

- 23.** Configure a TX persistent store for the new server in a location visible from other nodes as indicated in the recommendations about shared storage
- From the Administration Console, select the server name, and then the **Services** tab.
 - Under **Default Store**, in **Directory**, enter the path to the folder where you want the default persistent store to store its data files.
- 24.** Disable host name verification for the new managed server.

Before starting and verifying the WLS_OSBn managed server, disable host name verification. You can re-enable it after you have configured server certificates for the communication between the Oracle WebLogic Administration Server and the Node Manager in SOAHOSTn. If you have already disabled host name verification for the source server from which the new server has been cloned, you

can skip this procedure (the hostname verification setting is propagated to the cloned server).

To disable host name verification:

- a. In the Oracle Enterprise Manager Console, select Oracle WebLogic Server Administration Console.
- b. Expand the **Environment** node in the Domain Structure window.
- c. Click **Servers**.

The Summary of Servers page appears.

- d. Select **WLS_OSBn** in the **Names** column of the table.

The Settings page for server appears.

- e. Click the **SSL** tab.
- f. Click **Advanced**.
- g. Set **Hostname Verification** to **None** and click **Save**.

25. Start the Node Manager on the new node using the installation in shared storage from the existing nodes. Pass the host name of the new node as a parameter:

```
SOAHOSTN> WL_HOME/server/bin/startNodeManager new_node_ip
```

26. Start and test the new managed server from the Oracle WebLogic Server Administration Console:

- a. Shut down all the existing managed servers in the cluster.
- b. Ensure that the newly created managed server, **WLS_OSBn**, is running. Access the application on the newly created managed server:

```
http://vip:port/sbinspection.wsil
```

The application should be functional.

27. Configure server migration for the new managed server.

Note: In the previous steps you already created a private directory for Node Manager in this node. Update Node Manager properties as indicated in [Chapter 13, "Configure Server Migration for an Exalogic Enterprise Deployment,"](#) considering the new server's listen address and channels.

To configure server migration:

- a. Log into the Administration Console.
- b. In the left pane, expand **Environment** and select **Servers**.
- c. Select the server (represented as hyperlink) for which you want to configure migration from the **Names** column of the table.

The Settings page for that server appears.

- d. Click the **Migration** tab.
- e. In the **Available** field, in the **Migration Configuration** section, select the machines to which the server is to be migrated migration and click the right arrow.

For example, for new managed servers on SOAHOST1, which is already running WLS_OSB1, select **SOAHOST2**. For new managed servers on SOAHOST2, which is already running WLS_OSB2, select **SOAHOST1**.

Note: Specify the least-loaded machine as the migration target for the new server. Complete the required capacity planning so that this node has enough available resources to sustain an additional managed server.

- f.** Select the **Automatic Server Migration Enabled** option and click **Save**.
This enables the Node Manager to start a failed server on the target node automatically.
 - g.** Restart the Administration Server, managed servers, and Node Manager.
- 28.** Test server migration for this new server from the node where you added the new server:
- a.** Abruptly stop the WLS_OSBn managed server by running the following command on the PID (process ID) of the managed server:

```
kill -9 pid
```

You can identify the PID of the node using the following command:

```
ps -ef | grep WLS_OSBn
```

Note: For Windows, you can terminate the managed server using the `taskkill` command. For example:

```
taskkill /f /pid pid
```

Where *pid* is the process Id of the managed server.

You can determine the process ID of the WLS_OSBn managed server using the following command:

```
MW_HOME\jrocket_160_20_D1.0.1-2124\bin\jps -l -v
```

- b.** In the Node Manager Console you can view a message indicating that WLS_OSBn's floating IP has been disabled.
- c.** Wait for the Node Manager to try a second restart of WLS_OSBn. Node Manager waits for a fence period of 30 seconds before trying this restart.
- d.** Once Node Manager restarts the server, stop it again.

Now Node Manager logs a message indicating that the server will not be restarted again locally.

Note: After a server is migrated, to fail it back to its original node/machine, stop the managed server from the Oracle WebLogic Administration Console and then start it again. The appropriate Node Manager will start the managed server on the machine to which it was originally assigned.

14.7 Verifying Manual Failover of the Administration Server

In case a node fails, you can fail over the Administration Server to another node. 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 ANY address. See step 14 in [Section 8.4, "Running the Configuration Wizard on SOAHOST1 to Create a Domain."](#)
- These procedures assume that the two nodes use two individual domain directories, and that the directories reside in private storage or in shared storage in different volumes.
- 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.
- The domain directory where the Administration Server is running in SOAHOST1 is on a shared storage and is mounted also from SOAHOST2.
- Oracle WebLogic Server and Oracle Fusion Middleware components have been installed in SOAHOST2 as described in [Section 8.2, "Installing Oracle Fusion Middleware."](#) (that is, the same paths for ORACLE_HOME and MW_HOME that exist on SOAHOST1 are also available on SOAHOST2).

This section contains the following topics:

- [Section 14.7.1, "Failing Over the Administration Server to a Different Node"](#)
- [Section 14.7.2, "Validating Access to SOAHOST2"](#)
- [Section 14.7.3, "Failing the Administration Server Back to SOAHOST1"](#)

14.7.1 Failing Over the Administration Server to a Different Node

The following procedure shows how to fail over the Administration Server to a different node (SOAHOST2), but the Administration Server will still use the same WebLogic Server machine (which is a logical machine, not a physical machine).

To fail over the Administration Server to a different node:

1. Stop the Administration Server.
2. Migrate IP to the second node.
 - a. Run the following command as root on SOAHOST1 (where X:Y is the current interface used by ADMINVHN):

```
/sbin/ifconfig bond1:Y down
```

- b. Run the following command on SOAHOST2:

```
/sbin/ifconfig <interface:index> IP_Address netmask <netmask>
```

For example:

```
/sbin/ifconfig bond1:1 10.0.0.1 netmask 255.255.255.0
```

Note: Ensure that the netmask and interface to be used to match the available network configuration in SOAHOST2.

3. Update routing tables through arping, for example:

```
/sbin/arping -q -U -c 3 -I bond1 10.0.0.1
```

4. Start the Administration Server on SOAHOST2 using the procedure in [Section 8.5.3, "Starting the Administration Server on SOAHOST1."](#)
5. 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 the Oracle Enterprise Manager using the following URL:

```
http://ADMINVHN:7001/em
```

Note: The Administration Server does not use Node Manager for failing over. After a manual failover, the machine name that appears in the **Current Machine** field in the Administration Console for the server is SOAHOST1, and not the failover machine, SOAHOST2. Since Node Manager does not monitor the Administration Server, the machine name that appears in the **Current Machine** field, is not relevant and you can ignore it.

14.7.2 Validating Access to SOAHOST2

Perform the same steps as in [Section 8.11, "Validating the Administration Server Configuration."](#) This is to check that you can access the Administration Server when it is running on SOAHOST2.

14.7.3 Failing the Administration Server Back to SOAHOST1

This step checks that you can fail back the Administration Server, that is, stop it on SOAHOST2 and run it on SOAHOST1 by migrating ADMINVHN back to SOAHOST1 node.

To migrate ADMINVHN back to SOAHOST1:

1. Make sure the Administration Server is not running.
2. Run the following command on SOAHOST2.

```
/sbin/ifconfig bond1:N down
```

3. Run the following command on SOAHOST1:

```
/sbin/ifconfig bond1: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

4. Update routing tables through arping. Run the following command from SOAHOST1.

```
/sbin/arping -q -U -c 3 -I bond1 100.200.140.206
```

5. Start the Administration Server again on SOAHOST1 using the procedure in [Section 8.5.3, "Starting the Administration Server on SOAHOST1."](#)

```
cd ASERVER_HOME/bin
./startWebLogic.sh
```

6. Test that you can access the Oracle WebLogic Server Administration Console using the following URL:

```
http://ADMINVHN:7001/console
```

7. Check that you can access and verify the status of components in the Oracle Enterprise Manager using the following URL:

```
http://ADMINVHN:7001/em
```

14.8 Backing Up the Oracle SOA Enterprise Deployment

Back up the topology before and after any configuration changes.

14.8.1 Backing Up the Database

Back up the database. This is a full database backup (either hot or cold) using Oracle Recovery Manager (recommended) or OS tools, such as tar for cold backups if possible.

14.8.2 Backing Up the Administration Server Domain Directory

Back up the Administration Server domain directory to save your domain configuration. The configuration files are located in the following directory:

```
ASERVER_HOME
```

To back up the Administration Server run the following command on SOAHOST1:

```
tar -cvpf edgdomainback.tar ASERVER_HOME
```

14.8.3 Backing Up the Web Tier

Backup the Web tier. The configuration files are located in the following directories:

```
WEB_ORACLE_ADMININSTANCE
```

To back up the Oracle Traffic Director Administration Server, run the following command on WEBHOST1:

```
tar -cvpf webasback.tar WEB_ORACLE_ADMININSTANCE
```

14.8.4 Backing up the Middleware Home

If a new install has modified the *MW_HOME*, back it up using the following command:

```
tar -cvpf mw_home.tar MW_HOME
```

14.9 Preventing Timeouts for SQLNet Connections

Much of the Enterprise Deployment production deployment involves firewalls. Because database connections are made across firewalls, Oracle recommends that the firewall be configured so that the database connection is not timed out. For Oracle Real Application Clusters (Oracle RAC), the database connections are made on Oracle RAC virtual IPs and the database listener port. You must configure the firewall to not time out such connections. If such a configuration is not possible, set the `*SQLNET.EXPIRE_TIME=n*` parameter in the `sqlnet.ora` file, located in the following directory:

```
ORACLE_HOME/network/admin
```

The `n` indicates the time in minutes. Set this value to less than the known value of the timeout for the network device (that is, a firewall). For Oracle RAC, set this parameter in all of the Oracle home directories.

14.10 Recovering Failed BPEL and Mediator Instances

This section describes how to check and recover failed instances in BPEL, Mediator and other service engines.

Note: For the steps that require you to run SQL statements, you connect to the database as the `soainfra` schema.

- To check for recoverable instances, run the following SQL statements in the database:


```
// Find recoverable activities
SQL> select * from work_item where state = 1 and execution_type != 1;

// Find recoverable invoke messages
SQL> select * from dlv_message where dlv_type = 1 and state = 0;

// Find recoverable callback messages
SQL> select * from dlv_message where dlv_type = 2 and (state = 0 or state = 1);
```
- To recover failed BPEL instances:

In Enterprise Manager, select **Farm** `<domain_name>`, then expand **SOA**, then right click on **soa-infra (server_soa)**, then **Service Engine**, then **BPEL**, and then **Recovery**.
- To recover a failed Mediator composite:

In Enterprise Manager, select **Farm** `<domain_name>`, then expand **SOA**, then right-click on **soa-infra (server_soa)**, then **Service Engine**, then select **Mediator**, and then **Fault**.
- To check for rejected messages:


```
SQL> select * from rejected_message
```

- To check data in the instance tracking table, run the following SQL query:

```
SQL> select ID, STATE from COMPOSITE_INSTANCE where CREATED_TIME > datetime
```

where *datetime* specifies the date and time to narrow the query. For example:

```
'04-NOV-09 03.20.52.902000000 PM'
```

The adapter enters data into the COMPOSITE_INSTANCE table before anywhere else.

When the adapter publishes data to the Adapter BC, the BC inserts an entry into the COMPOSITE_INSTANCE table with STATE as 0. After the message has been processed, the STATE becomes 1. In case of errors, STATE >= 2.

14.11 Configuring Web Services to Prevent Denial of Service and Recursive Node Attacks

Configure SCABindingProperties.xml and oracle-webservices.xml to configure Web services against denial of service attack and recursive node attack.

Configuring SCABindingProperties.xml

To prevent denial of service attacks and recursive node attacks, set the envelope size and nesting limits in SCABindingProperties.xml as illustrated in [Example 14-1](#).

Example 14-1 Configuring Envelope Size and Nesting Limits in SCABindingProperties.xml

```
<bindingType type="ws">
  <serviceBinding>
    <bindingProperty>
      <name>request-envelope-max-kilobytes</name>
      <type>xs:integer</type>
      <defaultValue>-1</defaultValue>
    </bindingProperty>
    <bindingProperty>
      <name>request-envelope-nest-level</name>
      <type>xs:integer</type>
      <defaultValue>-1</defaultValue>
    </bindingProperty>
  </serviceBinding>
```

Configuring oracle-webservices.xml

For standalone Web services, configure the envelope size and nesting limits in oracle-webservices.xml. For example:

```
<request-envelope-limits kilobytes="4" nest-level="6" />
```

Note: Setting the envelope and nesting limits to extremely high values, or setting no values at all, can lead to denial of service attacks.

14.12 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. Use the following location as for the deployment plan and applications:

```
u01/oracle/config/dp/soaedg_domain
```

This directory must be accessible from all nodes in the Enterprise Deployment topology, as recommended in [Chapter 4, "Configuring Storage for an Exalogic Enterprise Deployment."](#)

14.13 Using External BPEL Caches for Improved HAS and Performance Isolation

This section describes how to use external PBEL caches improving HAS and Performance isolation.

This section contains the following topics:

- [Section 14.13.1, "Setting the Server's bpel.cache.localStorage Property"](#)
- [Section 14.13.2, "Creating Cache Configuration Files and Start Scripts"](#)
- [Section 14.13.3, "Starting BPEL Cache Instances"](#)

14.13.1 Setting the Server's bpel.cache.localStorage Property

Set the server's `bpel.cache.localStorage` property to **false**.

To set the `bpel.cache.localStorage` property:

1. Log into the Oracle WebLogic Server Administration Console using the following URL:

```
http://ADMINVHN:7001/console
```

2. In the **Domain Structure** window, expand the **Environment** node.
3. Click **Servers**.

The Summary of Servers page appears.

4. Click the name of the server (**WLS_SOA1** or **WLS_SOA2**, which are represented as hyperlinks) in the **Name** column of the table.

The settings page for the selected server appears.

5. Click **Lock & Edit**.
6. Click the **Server Start** tab.
7. Replace the **Arguments** field for **WLS_SOA1** and **WLS_SOA2**, replace the following:

```
-Dbpel.cache.localStorage=true
```

With the following:

```
-Dbpel.cache.localStorage=false
```

8. Click **Save and Activate Changes**.

14.13.2 Creating Cache Configuration Files and Start Scripts

Create the appropriate cache configuration files and start scripts.

To create cache configuration files and start scripts:

1. Create a caches directory on each node (SOAHOST1 and SOAHOST2).

```
mkdir /u02/private/oracle/config/soaexa_domain/caches
```

2. Create a bpelCacheEnv.sh file on each node in

/u02/private/oracle/config/soaexa_domain/caches/. See the example in the notes section of the /u01/oracle/products/fmw/soa/bin/start-bpel-cache.sh script.

The following example is for a cell that runs SOA server only. The servers can be customized with different resources if, for example, other servers or components are competing in the same box for memory.

```
BPEL_UNICAST_WKA="SOAHOST1-PRIV-V1:8089;SOAHOST2-PRIV-V1:8089"
MEMORY="3gb"
INSTANCE_CACHE_SIZE="1024"
INVOKE_MESSAGE_CACHE_SIZE="512"
DELIVERY_MESSAGE_CACHE_SIZE="512"
DELIVERY_SUBSCRIPTION_CACHE_SIZE="256"
JAVA_HOME=/u01/oracle/products/fmw/jrocket_160_29_D1.2.0-10
MW_HOME=/u01/oracle/products/fmw
COHERENCE_LIB=$MW_HOME/oracle_common/modules/oracle.coherence/coherence.jar
MW_ORA_HOME=MW_HOME/soa
```

3. Make a copy of start-bpel-cache.sh from MW_HOME/soa/bin to the caches directory on both nodes:

```
cp MW_HOME/soa/bin/start-bpel-cache.sh MSERVER_HOME/caches/
```

14.13.3 Starting BPEL Cache Instances

Start two cache instances by running the start-bpel-cache.sh script twice in the background.

To start the cache instances:

```
MSERVER_HOME/caches/start-bpel-cache.sh &
MSERVER_HOME/caches/start-bpel-cache.sh &
```

To identify the JVMs as running processes in the operating system, note the PIDs reported.

14.14 Troubleshooting the Topology in an Enterprise Deployment

This section describes possible issues with the SOA enterprise deployment and suggested solutions.

This section covers the following topics:

- [Section 14.14.1, "Page Not Found When Accessing soa-infra Application Through Load Balancer"](#)
- [Section 14.14.2, "Soa-infra Application Fails to Start Due to Deployment Framework Issues \(Coherence\)"](#)

- [Section 14.14.3, "SOA, OSB, or WMS Servers Fail to Start Due to Maximum Number of Processes Available in Database"](#)
- [Section 14.14.4, "Administration Server Fails to Start After a Manual Failover"](#)
- [Section 14.14.5, "Error While Activating Changes in Administration Console"](#)
- [Section 14.14.6, "SOA/OSB Server Not Failed Over After Server Migration"](#)
- [Section 14.14.7, "SOA/OSB Server Not Reachable From Browser After Server Migration"](#)
- [Section 14.14.8, "SOA Server Stops Responding after Being Active and Stressed for a Period of Time."](#)
- [Section 14.14.9, "Configured JOC Port Already in Use"](#)
- [Section 14.14.10, "SOA or OSB Server Fails to Start"](#)
- [Section 14.14.11, "SOA Coherence Cluster Conflicts when Multiple Clusters Reside in the Same Node"](#)
- [Section 14.14.12, "Sudo Error Occurs During Server Migration"](#)
- [Section 14.14.13, "Transaction Timeout Error"](#)
- [Section 14.14.14, "Exceeded Maximum Size Error Messages"](#)

14.14.1 Page Not Found When Accessing soa-infra Application Through Load Balancer

Problem: You receive a 404 "page not found" message in the Web browser when you try to access the soa-infra application using the load balancer address. The error is intermittent and SOA Servers appear as **Running** in the WLS Administration Console.

Solution: Even when the SOA managed servers may be up and running, some of the applications contained in them may be in **Admin**, **Prepared** or other states different from **Active**. The soa-infra application may be unavailable while the SOA server is running. Check the deployments page in the Administration Console to verify the status of the soa-infra application. It should be in **Active** state. Check the SOA Server's output log for errors pertaining to the soa-infra application and try to start it from the Deployments page in the Administration Console.

14.14.2 Soa-infra Application Fails to Start Due to Deployment Framework Issues (Coherence)

Problem: The soa-infra application fails to start after changes to the Coherence configuration for deployment have been applied. The SOA server output log reports the following:

```
Cluster communication initialization failed. If you are using multicast, Please
make sure multicast is enabled on your network and that there is no interference
on the address in use. Please see the documentation for more details.
```

Solutions:

1. When using multicast instead of unicast for cluster deployments of SOA composites, a message similar to the above may appear if a multicast conflict arises when starting the soa-infra application (that is, starting the managed server on which SOA runs). These messages, which occur when Oracle Coherence throws a runtime exception, also include the details of the exception itself. If such a message appears, check the multicast configuration in your network. Verify that you can ping multicast addresses. In addition, check for other clusters that may

have the same multicast address but have a different cluster name in your network, as this may cause a conflict that prevents soa-infra from starting. If multicast is not enabled in your network, you can change the deployment framework to use unicast as described in *Oracle Coherence Developer's Guide for Oracle Coherence*.

2. When entering well-known address list for unicast (in server start parameters), make sure that the node's addresses entered for the localhost and clustered servers are correct. Error messages like:

```
oracle.integration.platform.blocks.deploy.CompositeDeploymentCoordinatorMessage
s errorUnableToStartCoherence
```

are reported in the server's output log if any of the addresses is not resolved correctly.

14.14.3 SOA, OSB, or WMS Servers Fail to Start Due to Maximum Number of Processes Available in Database

Problem: SOA, WSM or OSB Server fails to start. The domain has been extended for new types of managed server (for example, SOA extended for OSB) or the system has been scaled up (added new servers of the same type). The SOA/OSB or WSM Server output log reports the following:

```
<Warning> <JDBC> <BEA-001129> <Received exception while creating connection for
pool "SOADatasource-rac0": Listener refused the connection with the following
error:
```

```
ORA-12516, TNS:listener could not find available handler with matching protocol
stack >
```

Solution: Verify the number of processes in the database and adjust accordingly. As the SYS user, issue the SHOW PARAMETER command:

```
SQL> SHOW PARAMETER processes
```

Set the initialization parameter using the following command:

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

Restart the database.

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

14.14.4 Administration Server Fails to Start After a Manual Failover

Problem: the Administration Server fails to start after it fails and you performed a manual failover to another node. The Administration Server output log reports the following:

```
<Feb 19, 2009 3:43:05 AM PST> <Warning> <EmbeddedLDAP> <BEA-171520> <Could not
obtain an exclusive lock for directory: ASERVER_
HOME/servers/AdminServer/data/ldap/ldapfiles. Waiting for 10 seconds and then
retrying in case existing WebLogic Server is still shutting down.>
```

Solution: Remove the file `EmbeddedLDAP.lok` file from the following directory:

```
ASERVER_HOME/servers/AdminServer/data/ldap/ldapfiles/
```

.

14.14.5 Error While Activating Changes in Administration Console

Problem: Activation of changes in Administration Console fails after you have made changes to a server's start configuration. The Administration Console reports the following when clicking **Activate Changes**:

An error occurred during activation of changes, please see the log for details.

```
[Management:141190]The commit phase of the configuration update failed with an exception:
```

```
In production mode, it's not allowed to set a clear text value to the property: PasswordEncrypted of ServerStartMBean
```

Solution: Either provide username/password information in the server start configuration in the Administration Console for the specific server whose configuration was being changed, or remove the `<password-encrypted></password-encrypted>` entry in the `config.xml` file (this requires a restart of the Administration Server).

14.14.6 SOA/OSB Server Not Failed Over After Server Migration

Problem: After reaching the maximum restart attempts by local Node Manager, Node Manager in the failover node tries to restart it, but the server does not come up. The server seems to be failed over as reported by Node Manager's output information. The virtual IP used by the SOA Server is not enabled in the failover node after Node Manager tries to migrate it (if config in the failover node does not report the virtual IP in any interface). Executing the command "sudo ifconfig \$INTERFACE \$ADDRESS \$NETMASK" does not enable the IP in the failover node.

Solution: The rights and configuration for `sudo` execution should not prompt for a password. Verify the configuration of `sudo` with your system administrator so that `sudo` works without a password prompt.

14.14.7 SOA/OSB Server Not Reachable From Browser After Server Migration

Problem: Server migration is working (SOA/OSB Server is restarted in the failed over node) but the `<Virtual Hostname>:8001/soa-infra` URL is not reachable in the Web browser. The server has been "killed" in its original host and Node Manager in the failover node reports that the virtual IP has been migrated and the server started. The virtual IP used by the SOA Server cannot be pinged from the client's node (that is, the node where the browser is being used).

Solution: Update the `nodemanager.properties` file to include the `MACBroadcast` or execute a manual arping:

```
/sbin/arping -b -q -c 3 -A -I $INTERFACE $ADDRESS > $NullDevice 2>&1
```

Where `$INTERFACE` is the network interface where the Virtual IP is enabled and `$ADDRESS` is the virtual IP address.

14.14.8 SOA Server Stops Responding after Being Active and Stressed for a Period of Time

Problem: WLS_SOA starts properly and functions for a period of time, but becomes unresponsive after running an application that uses the Oracle File Adapter or Oracle FTP Adapter. The log file for the server reports the following:

```
<Error> <Server> <BEA-002606> <Unable to create
a server socket for listening on channel "Default". The address
X.X.X.X might be incorrect or another process is using port 8001:
@ java.net.SocketException: Too many open files.>
```

Solution: For composites with Oracle File and FTP Adapters, which are designed to consume a very large number of concurrent messages, set the number of open files parameter for your operating system to a greater value. For example, to set the number of open files parameter to 8192 for Linux, use the `ulimit -n 8192` command. The value must be adjusted based on the expected system's load.

14.14.9 Configured JOC Port Already in Use

Problem: Attempts to start a Managed Server that uses the Java Object Cache, such as OWSM or WebCenter Spaces Managed Servers, fail. The following errors appear in the logs:

```
J2EE JOC-058 distributed cache initialization failure
J2EE JOC-043 base exception:
J2EE JOC-803 unexpected EOF during read.
```

Solution: Another process is using the same port that JOC is attempting to obtain. Either stop that process, or reconfigure JOC for this cluster to use another port in the recommended port range.

14.14.10 SOA or OSB Server Fails to Start

The SOA or OSB server fails to start for the first time and reports parsing failure in `config.xml`.

Problem: A server that is being started for the first time using Node Manager fails to start. A message such as the following appears in the server's output log:

```
<Critical> <WebLogicServer> <eicfdcn35> <wls_server1> <main> <<WLS Kernel>> <> <>
<1263329692528> <BEA-000386> <Server subsystem failed. Reason:
weblogic.security.SecurityInitializationException: Authentication denied: Boot
identity not valid; The user name and/or password from the boot identity file
(boot.properties) is not valid. The boot identity may have been changed since the
boot identity file was created. Please edit and update the boot identity file with
the proper values of username and password. The first time the updated boot
identity file is used to start the server, these new values are encrypted.
```

The Managed Server is trying to start for the first time, in MSI (managed server independence) mode. The Server has not been able to retrieve the appropriate configuration for the first start. The Managed Server must be able to communicate with the Administration Server on its first startup.

Solution: Make sure communication between the Administration Server's listen address and the Managed Server's listen address is possible (ping the Administration Server's listen address from the Managed Server's node, and telnet to the Administration Server's listen address and port). Once communication is enabled, pack and unpack the domain again to the new node or (if other servers are already

running correctly in the same domain directory), delete the following directory and restart the server:

```
OARCLE_BASE/admin/domain_name/mserver/domain_name/servers/server_
name/data/nodemanager
```

14.14.11 SOA Coherence Cluster Conflicts when Multiple Clusters Reside in the Same Node

Problem: soa-infra fails to come up when multiple soa clusters reside in the same nodes. Messages such as the following appear in the server's .out file:

```
<Error> <Coherence> <BEA-000000> <Oracle Coherence GE 3.6.0.4 <Error>
(thread=Cluster, member=1): This senior Member(...) appears to have been
disconnected from another senior Member...stopping cluster service.>
```

Solution: When a Coherence member restarts, it attempts to bind to the port configured in its localport setting. If this port is not available, it increments the port number (by two) and attempts to connect to that port. If multiple SOA clusters use similar range ports for coherence it is possible for a member to join a cluster with a different WKA, causing conflicts and preventing soa-infra application from starting. There are several ways to resolve this issue:

- Set up a port range for each of the various clusters instead of incrementing the cluster port by 2. For example, 8000-8090 for cluster 1, 8091-8180 for cluster 2. This is implicit in the model recommended in this guide specified in [Table 3-4](#) where different ranges should be used for each coherence cluster.
- Disable port auto adjust to force the members to use their configured localhost address. This can be done via system property "tangosol.coherence.localport.adjust" for example
-Dtangosol.coherence.localport.adjust=false.
- Configure a unique cluster name for each cluster. This can be done using the system property tangosol.coherence.cluster. For example:
-Dtangosol.coherence.cluster=SOA_Cluster1

For more information on these different options, refer to the coherence cluster configuration documentation at the following URL:

```
http://download.oracle.com/docs/cd/E24290_01/coh.371/e22837/cluster_
setup.htm
```

14.14.12 Sudo Error Occurs During Server Migration

Problem: When running wlsifconfig for server migration, the following warning displays:

```
sudo: sorry, you must have a tty to run sudo
```

Solution: The WebLogic user ('oracle') is not allowed to run sudo in the background. To solve this, add the following line into /etc/sudoers:

```
Defaults:oracle !requiretty
```

14.14.13 Transaction Timeout Error

Problem: The following transaction timeout error appears in the log:

```
Internal Exception: java.sql.SQLException: Unexpected exception while enlisting
XAConnection java.sql.SQLException: XA error: XAResource.XAER_NOTA start()
failed on resource 'SOADDataSource_soaedg_domain': XAER_NOTA : The XID
is not valid
```

Solution: Check your transaction timeout settings, and be sure that the JTA transaction time out is less than the DataSource XA Transaction Timeout, which is less than the distributed_lock_timeout (at the database).

With the out of the box configuration, the SOA datasources do not set XA timeout to any value. The Set XA Transaction Timeout configuration parameter is unchecked in the WebLogic Server Administration Console. In this case, the datasources use the domain level JTA timeout which is set to 30. Also, the default distributed_lock_timeout value for the database is 60. As a result, the SOA configuration works correctly for any system where transactions are expected to have lower life expectancy than such values. Adjust these values according to the transaction times your specific operations are expected to take.

14.14.14 Exceeded Maximum Size Error Messages

Problem: When complex rules are edited and saved in a SOA cluster, error messages reporting exceeded maximum size in replication messages may show up in the server's out file. For example:

```
<rws3211539-v2.company.com> <WLS_SOA1> <ExecuteThread: '2' for queue:
'weblogic.socket.Muxer'> <<WLS Kernel>> <> <> <1326464549135> <BEA-000403>
<IOException occurred on socket:
Socket[addr=/10.10.10.10,port=48290,localport=8001]
weblogic.socket.MaxMessageSizeExceededException: Incoming message of size:
'10000080' bytes exceeds the configured maximum of: '10000000' bytes for
protocol: 't3'.
weblogic.socket.MaxMessageSizeExceededException: Incoming message of size:
'10000080' bytes exceeds the configured maximum of: '10000000' bytes for
protocol: 't3'
```

Solution: This error is due to the large size of serialized rules placed in the HTTP Sessions and replicated in the cluster. Increase the maximum message size according to the size of the rules being used.

To increase the maximum message size:

1. Log in to the WebLogic Administration Console.
2. Select **Servers**, **Server_name**, **Protocols**, and then **General**.
3. Modify the **Maximum Message Size** field as needed.

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 the following topics:

- [Section A.1, "About Multi Data Sources and Oracle RAC"](#)
- [Section A.2, "Typical Procedure for Configuring Multi Data Sources for an EDG Topology"](#)

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 EDG Topology

You configure data sources when you configure a domain. For example, when you are configuring the initial Administration domain for an Exalogic 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 Exalogic 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.

B

Worksheet for Oracle SOA Enterprise Deployment on Exalogic Topology

This appendix contains worksheets to help you keep track of machine names, IP addresses, directories, and other important data.

We recommend that you open the PDF version of this Guide in a PDF reader and print out this appendix. Update these worksheets as you set up your Exalogic enterprise deployment.

This chapter contains the following worksheets:

- [Section B.1, "Hosts, Virtual Hosts, and Virtual IP Addresses for Oracle SOA"](#)
- [Section B.2, "Directory Mapping"](#)
- [Section B.3, "Port Mapping"](#)
- [Section B.4, "Database Details"](#)
- [Section B.5, "Web Tier Details"](#)
- [Section B.6, "Application Tier Details"](#)

B.1 Hosts, Virtual Hosts, and Virtual IP Addresses for Oracle SOA

Use this worksheet to record information about hosts and IP addresses.

Table B-1 *Hosts, Virtual Hosts, and Virtual IP Addresses for topologyName Worksheet Table*

Documented Alias	Type	Your Host Name	IP Address	Operating System and Version
WEBHOST1	Host			
WEBHOST2	Host			
SOAHOST1	Host			
SOAHOST2	Host			
SOADBHOST1	Database Host			
SOADBHOST2	Database Host			
ADMINVHN	Virtual Host			
SOAHOST1VHN	Virtual Host			
SOAHOST2VHN	Virtual Host			
soa.mycompany.com	Load Balancer Virtual Host Name			

Table B-1 (Cont.) Hosts, Virtual Hosts, and Virtual IP Addresses for topologyName Worksheet Table

Documented Alias	Type	Your Host Name	IP Address	Operating System and Version
admin.mycopamny.com	Load Balancer Virtual Host Name			
osb.mycompany.com	Load Balancer Virtual Host Name			
soainternal.mycompany.com	OTD Virtual Name			
SOAEXADOMAIN	Domain Name			

B.2 Directory Mapping

Use this worksheet to keep track of directories.

Table B-2 Directory Mapping Table

Documented Variable	Sample Directory Path	Your Directory Path
WEB_MW_HOME	/u02/private/oracle/products/web	
SOA_ORACLE_HOME	/u01/oracle/products/access/soa	
WEB_ORACLE_HOME	/u02/private/oracle/products/web/ web1	
ORACLE_COMMON_HOME	/u01/oracle/products/access/oracle_ common	
WL_HOME	/u01/oracle/products/access/wlserv er_10.3	
JAVA_HOME	/u01/oracle/products/access/jrocket_ version	
WEB_ORACLE_INSTANCE	/u02/private/oracle/config/instances /webn	
ASERVER_HOME (IDMDomain)	/u01/oracle/config/domains/soaexa_ _domain	
MSERVER_HOME (IDMDomain)	/u02/private/oracle/config/domains /soaexa_domain	

B.3 Port Mapping

Use this worksheet to keep track of ports.

Table B-3 Port Mapping Table

Documented Variable	Documented Port	Description	Your Port
HTTP_SSL_PORT	443	SSL Port for accessing the site externally	
HTTP_PORT	80	Non SSL Port used for accessing admin functions internally	
LDAP_DIR_PORT	1389	OUD/OID Access Port	

Table B-3 (Cont.) Port Mapping Table

Documented Variable	Documented Port	Description	Your Port
LDAP_DIR_SSL_PORT	1636	OUO/OID Access Port	
ONS_PORT	6200	ONS Port	
DB_LSNR_PORT	1521	Listener Port	
WLS_ADMIN_PORT	7001	WLS Administration Port	
WLS_ADMIN_SSL_PORT	7002	WLS Administration SSL Port	
NMGR_PORT	5556	Node Manager Listen Port	
SOA_PORT	8001	SOA Port	
OTD_PORT	8989	Oracle Traffic Director Port	
OSB_PORT	8010	Oracle Service Bus Port	
WSM_PORT	7010	WSM PM port	
OTDAS_Port	8989	Administration port for Oracle Traffic Director	
OTDAN	8900	Node port for the second instance of Oracle Traffic Director	

B.4 Database Details

Use this worksheet to keep track of database information.

Table B-4 Database Details Table

Description	Documented Value	Customer Value
Database Hosts	SOADBHOST1 SOADBHOST2	
Scan Address Name	DB-SCAN.mycompany.com	
Database Name	SOADB.mycompany.com	
Database Service Names defined	soaedg.mycompany.com osbedg.mycompany.com	
System Account Name and Password	system/xxxxx	
RCU Schema Prefix	EDG	
ONS Port	6200	
Listener Port	1521	

B.5 Web Tier Details

Use this worksheet to keep track of Web Tier information.

Table B-5 Web Tier Details Table

Description	Documented Value	Customer Value
Web Tier Hosts	WEBHOST1 WEBHOST2	
WEB_ORACLE_HOME	/u02/private/oracle/products/web/web	
WEB_ORACLE_INSTANCE	/u02/private/oracle/config/instances/web1 /u02/private/oracle/config/instances/web2	
Virtual Hosts	admin.mycompany.com soainternal.mycompany.com osbinternal.mycompany.com	

B.6 Application Tier Details

Use this worksheet to keep track of Application Tier information

Table B-6 Application Tier Details Table

Description	Documented Value	Customer Value
Host (Virtual Hosts)	ADMINVHN (SOAHOST1) SOAHOST1-PRIV-V1 SOAHOST2-PRIV-V1 SOAHOST1-PRIV-V2 SOAHOST2-PRIV-V2 SOAHOST1VHN1 SOAHOST2VHN1 SOAHOST1VHN2 SOSHOST2VHN2	
Domain Name	SOAEXADomain	
ASERVER_HOME	/u01/oracle/config/domains/soaexa_domain	
MSERVER_HOME	/u02/private/oracle/config/domains/soaexa_domain	
ASERVER_HOME	/u01/oracle/config/domains/soaexa_domain	
MSERVER_HOME	/u02/private/oracle/config/domains/soaexa_domain	

Table B-6 (Cont.) Application Tier Details Table

Description	Documented Value	Customer Value
SOA Managed Server Names	WLS_SOA1	
	WLS_SOA2	
WSM-PM Managed Server Names	WLS_WSM1	
	WLS_WSM2	
OSB Managed Server Port	WLS_OSB1	
	WLS_OSB2	

SOA Exalogic Enterprise Topology with Oracle HTTP Server

This chapter describes an Oracle SOA Exalogic enterprise deployment on Exalogic with an external Oracle HTTP Server Web tier. It is one of the alternative topologies, discussed in [Section 2.1.2, "Alternative Deployment Topologies."](#)

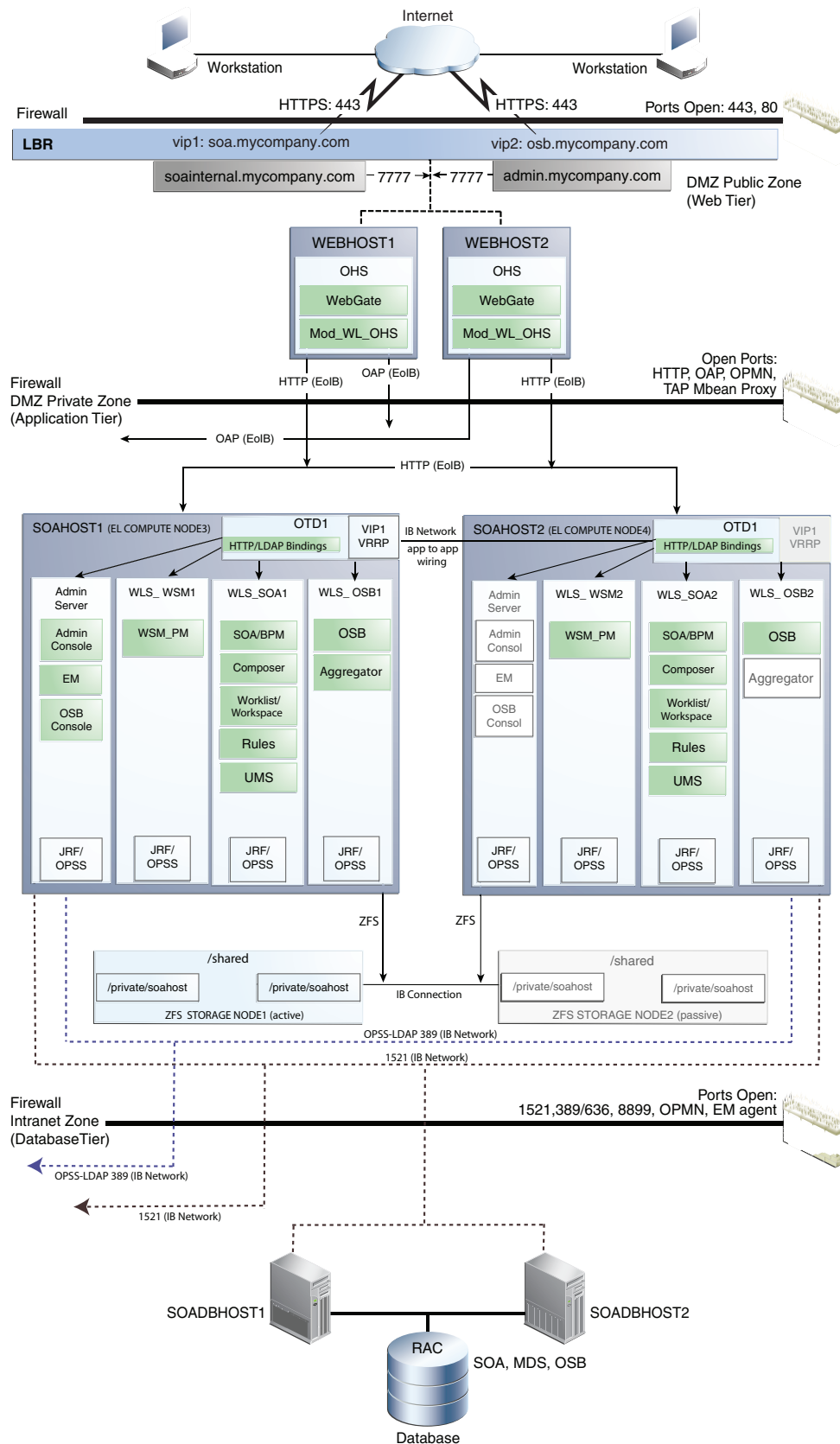
This appendix contains the following topics:

- [Viewing the Oracle SOA Deployment Topology with Oracle HTTP Server on Exalogic](#)
- [Understanding the Oracle SOA with Oracle HTTP Server Topology Components](#)

C.1 Viewing the Oracle SOA Deployment Topology with Oracle HTTP Server on Exalogic

In this alternative Oracle SOA topology on Exalogic topology, user requests are being routed by an Oracle HTTP Server Web tier, rather than the Oracle Traffic Director Web listeners. Compare this topology with the one shown in [Chapter 2, "Introduction and Planning."](#)

Figure C-1 Oracle SOA with Oracle HTTP Server and an Oracle RAC Database



C.2 Understanding the Oracle SOA with Oracle HTTP Server Topology Components

The components of the alternative Oracle SOA with Oracle HTTP Server topology are identical to those described in [Chapter 2](#), except for the following:

- [Section C.2.1, "About the Oracle HTTP Server Instances in the Web Tier"](#)
- [Section C.2.2, "About the Oracle Traffic Director Instances on the Application Tier"](#)

C.2.1 About the Oracle HTTP Server Instances in the Web Tier

The Web tier in the Oracle HTTP Server topology consists of two Oracle HTTP Server instances on separate WEHOST1 and WEBHOST2 host computers. These computers are outside of the Exalogic machine, and a firewall separates them from the application tier.

Most of the Fusion Middleware SOA components can function without the Web tier, but for most Exalogic enterprise deployments, the Web tier is desirable.

In the Web tier:

- WEBHOST1 and WEBHOST2 have Oracle HTTP Server, WebGate (an Access Manager component), and the `mod_wl_ohs` plug-in module installed. The `mod_wl_ohs` plug-in module enables requests to be proxied from Oracle HTTP Server to a WebLogic Server running in the application tier.
- WebGate (an Oracle Access Management component) in Oracle HTTP Server uses Oracle Access Protocol (OAP) to communicate with Access Manager running on SOAHOST1 and SOAHOST2, in the Fusion Middleware SOA DMZ. WebGate and Access Manager are used to perform operations such as user authentication.

On the firewall protecting the Web tier, the HTTP ports are 443 (`HTTP_SSL_PORT`) for HTTPS and 80 (`HTTP_PORT`) for HTTP. Port 443 is open.

C.2.2 About the Oracle Traffic Director Instances on the Application Tier

Similar to the topology in [Section 2.3, "Understanding the Topology Components."](#) Oracle Traffic Director is used as a load balancer for internal communications within the Exalogic rack. By using Oracle Traffic Director rather than routing requests through the load balancer, you can utilize the internal IPoIB network which is both more secure and faster.

In this topology, the Oracle Traffic Director instances are in an active-passive configuration and the required virtual IP addresses used for internal communication (such as `oudinternal.mycompany.com`) are defined in the Oracle Traffic Director configuration.

For more information on configuring Oracle Traffic Director failover groups for active-passive mode, see "Creating Failover Groups" in the *Oracle Traffic Director Administrator's Guide*.

Creating a GridLink Data Source

This appendix describes the procedure for creating a GridLink data source without support for global transactions using the Oracle WebLogic Server Administration Console.

This appendix contains the following topics:

- [Section D.1, "Creating a GridLink Data Source Using the Oracle WebLogic Server Administration Console"](#)

D.1 Creating a GridLink Data Source Using the Oracle WebLogic Server Administration Console

To create a GridLink data source:

1. Log in to the Oracle WebLogic Server Administration Console at the URL listed in [Section 8.18.2, "Validating Access through Oracle Traffic Director."](#)
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:
 - **Name:** Enter a logical name for the data source. (For example: DatasourceName)
 - **JNDI:** Enter a name for JNDI. (For example: jdbc/DatasourceName)
 - **Database Driver:** Select **For the Database Driver, select Oracle's Driver (Thin) for GridLink Connections Versions: 11 and later**.
 - Click **Next**.
5. In the Transaction Options page, de-select **Supports Global Transactions**, and click **Next**.
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.mycompany.com

- **Host Name and Port:** Enter the SCAN address and port for the RAC database being used. You can identify this address by querying the appropriate parameter in the database using the TCP Protocol:

```
show parameter remote_listener;
```

NAME	TYPE	VALUE

remote_listener	string	DB-SCAN.mycompany.com

Note: For Oracle Database 11g Release 1 (11.1), use the virtual IP and port of each database instance listener, for example:

CUSTDBHOST1-VIP.mycompany.com (port 1521) and
 CUSTDBHOST2-VIP.mycompany.com (port 1521), where 1521 is *DB_LSNR_PORT*

For Oracle Database 10g, use multi data sources to connect to an Oracle RAC database. For information about configuring multi data sources see [Appendix A, "Using Multi Data Sources with Oracle RAC."](#)

- **Database User Name:** DatasourceUser
 - **Password:** For example: welcome1
 - **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.mycompany.com)
(PORT=1521))) (CONNECT_DATA=(SERVICE_NAME=SOAEDG.mycompany.com))) succeeded.
```

where port 1521 is *DB_LSNR_PORT*.

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 for the RAC database and the ONS remote port as reported by the database (example below) and click **ADD**:

```
[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.mycompany.com (port 6200)

and

CUSTDBHOST2.mycompany.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.mycompany.com:6200 succeeded.

Click **Next**.

11. In the Select Targets page, select **osb_cluster** and **soa_cluster** as the targets, and **All Servers in the cluster**.
12. Click **Finish**.
13. Click **Activate Changes**.

