

Oracle® Banking Payments

Configuring Weblogic Server



Release 14.8.1.0.0
G44842-01
October 2025

ORACLE®

Oracle Banking Payments Configuring Weblogic Server, Release 14.8.1.0.0

G44842-01

Copyright © 2018, 2025, Oracle and/or its affiliates.

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, software documentation, data (as defined in the Federal Acquisition Regulation), or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, then the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs (including any operating system, integrated software, any programs embedded, installed, or activated on delivered hardware, and modifications of such programs) and Oracle computer documentation or other Oracle data delivered to or accessed by U.S. Government end users are "commercial computer software," "commercial computer software documentation," or "limited rights data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, reproduction, duplication, release, display, disclosure, modification, preparation of derivative works, and/or adaptation of i) Oracle programs (including any operating system, integrated software, any programs embedded, installed, or activated on delivered hardware, and modifications of such programs), ii) Oracle computer documentation and/or iii) other Oracle data, is subject to the rights and limitations specified in the license contained in the applicable contract. The terms governing the U.S. Government's use of Oracle cloud services are defined by the applicable contract for such services. 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 fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

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

Intel and Intel Inside 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, Epyc, and the AMD 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

1	Preface	
1.1	Purpose	1
1.2	Audience	1
1.3	Documentation Accessibility	1
1.4	Critical Patches	1
1.5	Diversity and Inclusion	2
1.6	Conventions	2
2	Introduction	
2.1	Purpose of this Document	1
2.2	WebLogic Server Overview	1
2.3	Pre-Requisites	4
3	Domain Configuration	
3.1	Domain Creation	1
3.2	Pack and Unpack Domain	10
3.3	Domain and Resource Creation using WDT Tool	11
3.4	Start Admin server	11
3.5	Start Node Manager	11
4	Cluster Configuration	
4.1	Machines Configuration	1
4.2	Dynamic Cluster Creation	2
4.3	Managed Server Template configuration	2
4.3.1	Logging	2
4.3.2	HTTP Logging	3
4.3.3	Stuck Tread Max Time	4
5	Tuning	
5.1	General Parameters	1

6 Data Source creation and JDBC Configuration

6.1	Setup Required for OCI Driver	1
6.2	Data Source Creation	2
6.3	JDBC Parameters Tuning	2

7 JMS Resource Creation

8 Oracle WebLogic Load Balancing

9 Frequently Asked Questions

9.1	Machine status is Unreachable	1
9.2	How to restart node manager?	1
9.3	Scaling Up Dynamic Cluster	1
9.4	Session Timeout	1

1

Preface

- [Purpose](#)
- [Audience](#)
This manual is intended for the following User/User Roles:
- [Documentation Accessibility](#)
- [Critical Patches](#)
- [Diversity and Inclusion](#)
- [Conventions](#)

1.1 Purpose

This guide is designed to help acquaint you with the Oracle Banking Payments application. This guide provides answers to specific features and procedures that the user need to be aware of the module to function successfully.

1.2 Audience

This manual is intended for the following User/User Roles:

Table 1-1 User Roles

Role	Function
Implementation & IT Staff	Implementation & Maintenance of the Software

1.3 [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.

1.4 Critical Patches

Oracle advises customers to get all their security vulnerability information from the Oracle Critical Patch Update Advisory, which is available at [Critical Patches, Security Alerts and Bulletins](#). All critical patches should be applied in a timely manner to make sure effective security, as strongly recommended by [Oracle Software Security Assurance](#).

1.5 Diversity and Inclusion

Oracle is fully committed to diversity and inclusion. Oracle respects and values having a diverse workforce that increases thought leadership and innovation. As part of our initiative to build a more inclusive culture that positively impacts our employees, customers, and partners, we are working to remove insensitive terms from our products and documentation. We are also mindful of the necessity to maintain compatibility with our customers' existing technologies and the need to ensure continuity of service as Oracle's offerings and industry standards evolve. Because of these technical constraints, our effort to remove insensitive terms is ongoing and will take time and external cooperation.

1.6 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.
<i>italic</i>	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
<code>monospace</code>	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

2

Introduction

- [Purpose of this Document](#)
- [WebLogic Server Overview](#)
- [Pre-Requisites](#)

2.1 Purpose of this Document

The purpose of this document is to explain the steps required for Configuration and applying best practices in cluster mode for:

- Weblogic Version 14.1.2
- JDK 17

2.2 WebLogic Server Overview

This section of the document provides brief explanation on the main components involved in WebLogic server

Domain

A domain is the basic administration unit for WebLogic Server instances. A domain consists of one or more WebLogic Server instances (and their associated resources) that is managed with a single Administration Server. Multiple domains can be defined based on different system administrators' responsibilities, application boundaries, or geographical locations of servers. Conversely, a single domain can be used to centralize all WebLogic Server administration activities.

Each WebLogic Server domain must have one server instance that acts as the Administration Server. Administration Server can be used via the Administration Console or using the command line for configuring all other server instances and resources in the domain.

Figure 2-1 WebLogic Domain Structure

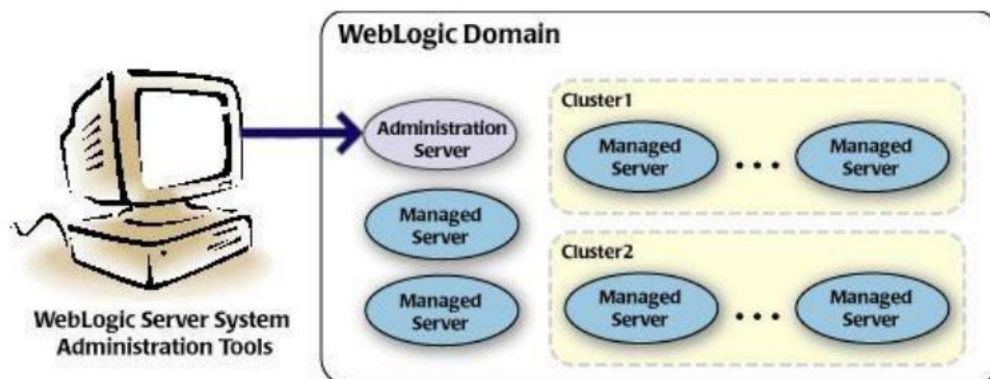
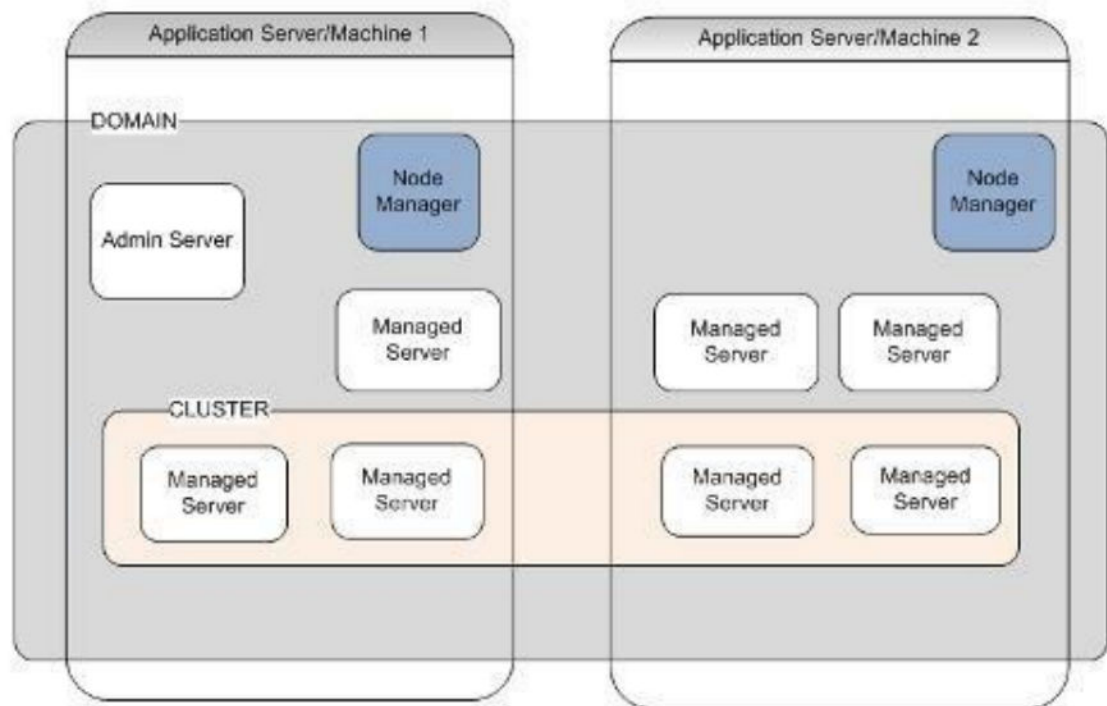


Figure 2-2 Weblogic 12c Domain Overview**Administration Server**

A domain includes one WebLogic Server instance that is configured as an Administration Server. All changes to configuration and deployment of applications are done through the Administration Server. The Administration Server provides a central point for managing the domain and providing access to the WebLogic Server administration tools.

These tools include the following:

- WebLogic Server Administration Console: Graphical user interface to the Administration Server.
- WebLogic Server Node Manager: A Java program that lets you start and stop server instances - both Administration Servers and Managed Servers - remotely, and to monitor and automatically restart them after an unexpected failure.

Admin server start mode needs to be configured as Production Mode.

Managed Server

In a domain, server instances other than the Administration Server are referred to as Managed Servers. Managed servers host the components and associated resources that constitute your applications—for example, JSPs and EJBs.

When a Managed Server starts up, it connects to the domain's Administration Server to obtain configuration and deployment settings. In a domain with only a single WebLogic Server instance, that single server works as both the administration server and managed server.

Node Manager

The Managed Servers in a production WebLogic Server environment are often distributed across multiple machines and geographic locations.

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

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

Machine

A machine in the Weblogic Serve context is the logical representation of the computer that hosts one or more Weblogic Server instances(servers). The Admin Server uses the machine definitions to start remote servers through the Node Managers that run on those servers. A machine could be a physical or virtual server that hosts an Admin or Managed Server that belongs to a domain.

Managed Server Cluster

Two or more Managed Servers can be configured as a WebLogic Server cluster to increase application scalability and availability. In a WebLogic Server cluster, most resources and services are deployed to each Managed Server (as opposed to a single Managed Server,) enabling failover and load balancing.

The servers within a cluster can either run on the same machine or reside in different machines. To the client, a cluster appears as a single WebLogic Server instance.

Dynamic Cluster

A dynamic cluster is any cluster that contains one or more dynamic servers. Each server in the cluster will be based upon a single shared server template. The server template allows you to configure each server the same and ensures that servers do not need to be manually configured before being added to the cluster. This allows you to easily scale up or down the number of servers in your cluster without the need for setting up each server manually. Changes made to the server template are rolled out to all servers that use that template.

You cannot configure dynamic servers individually; there are no server instance definitions in the config.xml file when using a dynamic cluster. Therefore, you cannot override the server template with server-specific attributes or target applications to an individual dynamic server instance.

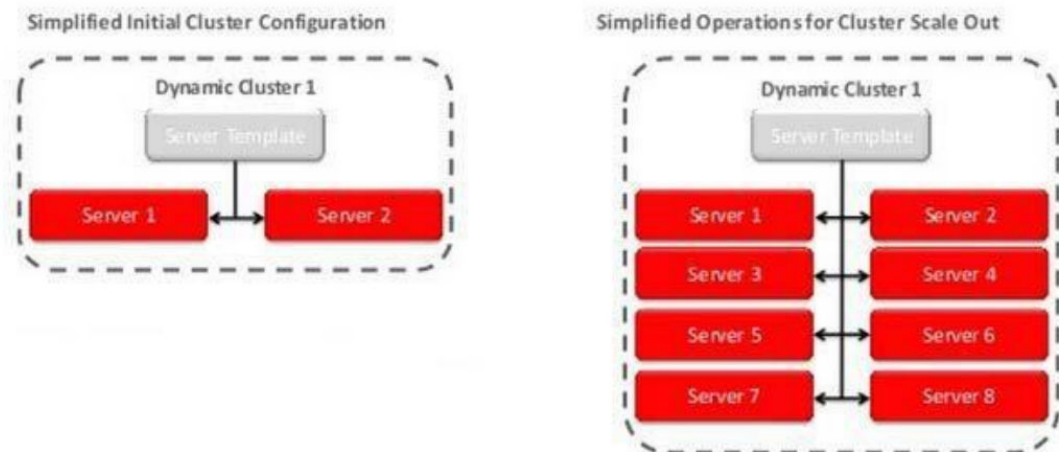
When configuring your cluster you specify the maximum number of servers you expect to need at peak times. The specified number of server instances is then created, each based upon your server template. You can then start up however many you need and scale up or down over time according to your needs. If you need additional server instances on top of the number you originally specified, you can increase the maximum number of servers instances (dynamic) in the dynamic cluster configuration.

Server Templates

A single server template provides the basis for the creation of the dynamic servers. Using this single template provides the possibility of every member being created with exactly the same attributes. Where some of the server-specific attributes like Servername, listen-ports, machines, etc. can be calculated based upon tokens.

You can pre-create server templates and let Weblogic clone one when a Dynamic Cluster is created.

When none is available a server template is created with the Dynamic Cluster. The name and the listen ports are the only server template attributes that you provide during Dynamic Cluster creation.

Figure 2-3 Simplified Configuration with Scalability and Elasticity

2.3 Pre-Requisites

In this document, we are going to create a domain with two managed servers. The managed servers are going to be created on two different physical servers (nodes). Note that, this document has been prepared based on a test conducted in Linux servers.

This requires Weblogic Server of same version to be installed on both the machines and services.

Environment

2 servers where linux is installed, 1 will be primary where admin console will be running along with managed servers and the other where only managed servers will be.

Softwares

1. Oracle Weblogic Server 14.1.2 installed on both the machines under same folder structure.
2. JDK 17 Latest available version installed on both the machines. In this document JDK 17 version is used.

Clock Synchronization

The clocks of both the servers participating in the cluster must be synchronized to within one second difference to enable proper functioning of jobs otherwise it will lead to session timeouts.

Enable Graphical User Interface (GUI)

Establish a telnet or SSH connection to primary server. Start X- manager (or any similar tool) in windows desktop. Export DISPLAY environment variable to the machine IP where x-manager is running.

Syntax: export DISPLAY=<ip-address>:<port> Test using xclock

3

Domain Configuration

- [Domain Creation](#)
- [Pack and Unpack Domain](#)
- [Domain and Resource Creation using WDT Tool](#)
- [Start Admin server](#)
- [Start Node Manager](#)

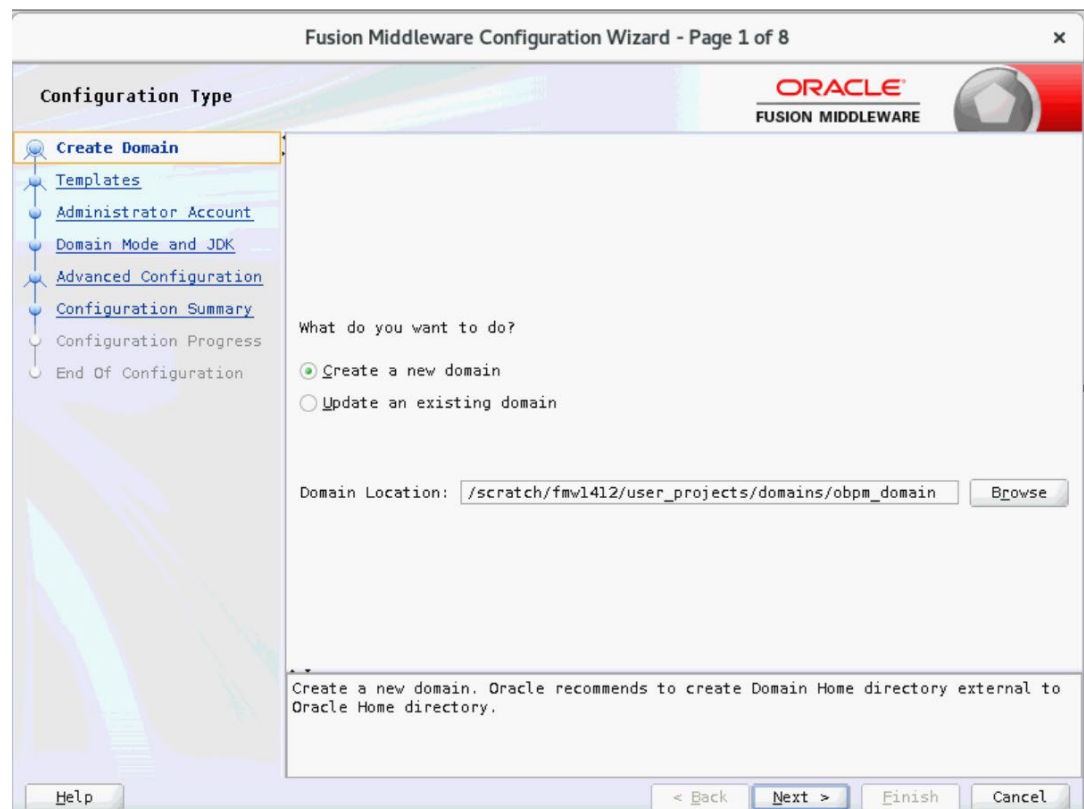
3.1 Domain Creation

Weblogic domain creation and configuration will be done from primary server.

From primary server, launch the fusion Middleware configuration wizard using the command `config.sh` available under `$WLS_HOME/common/bin` directory.

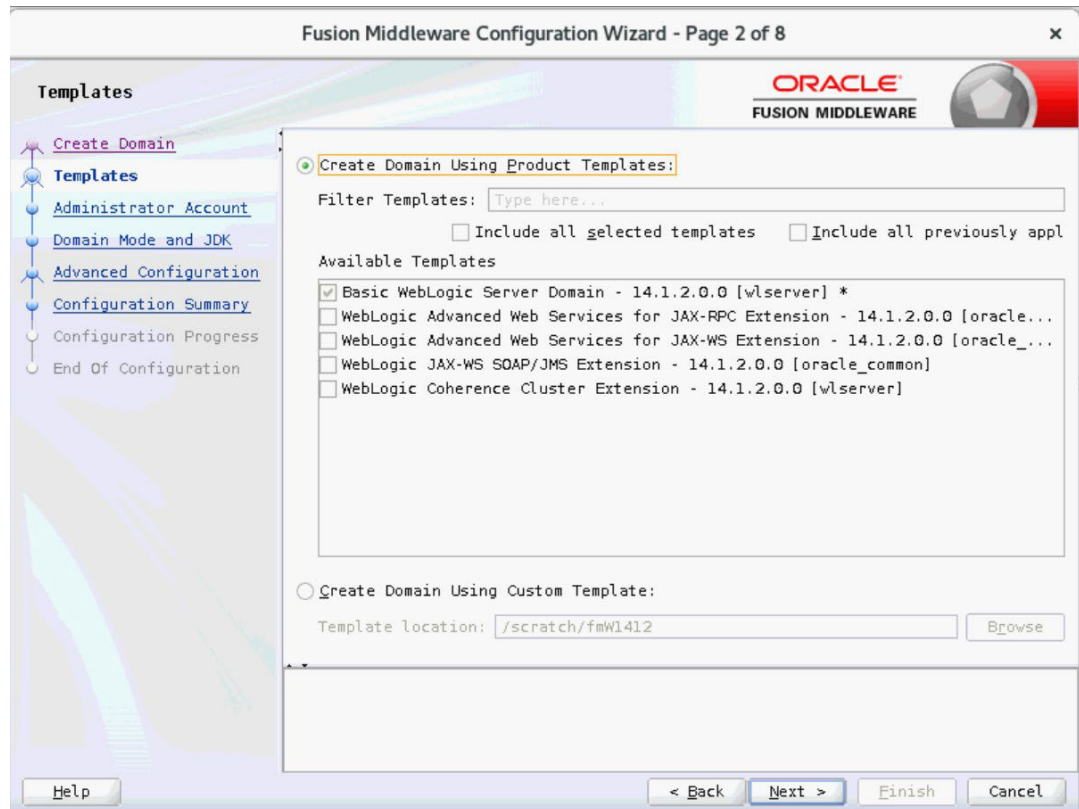
1. In the Welcome screen, select **Create a new domain** option. Enter the **domain name** and Click on **Next**.

Figure 3-1 Configuration Type



2. Select the required templates from **Available Templates** and click **Next**.

Figure 3-2 Templates



3. Specify Administrator **User Name** and **Password**.

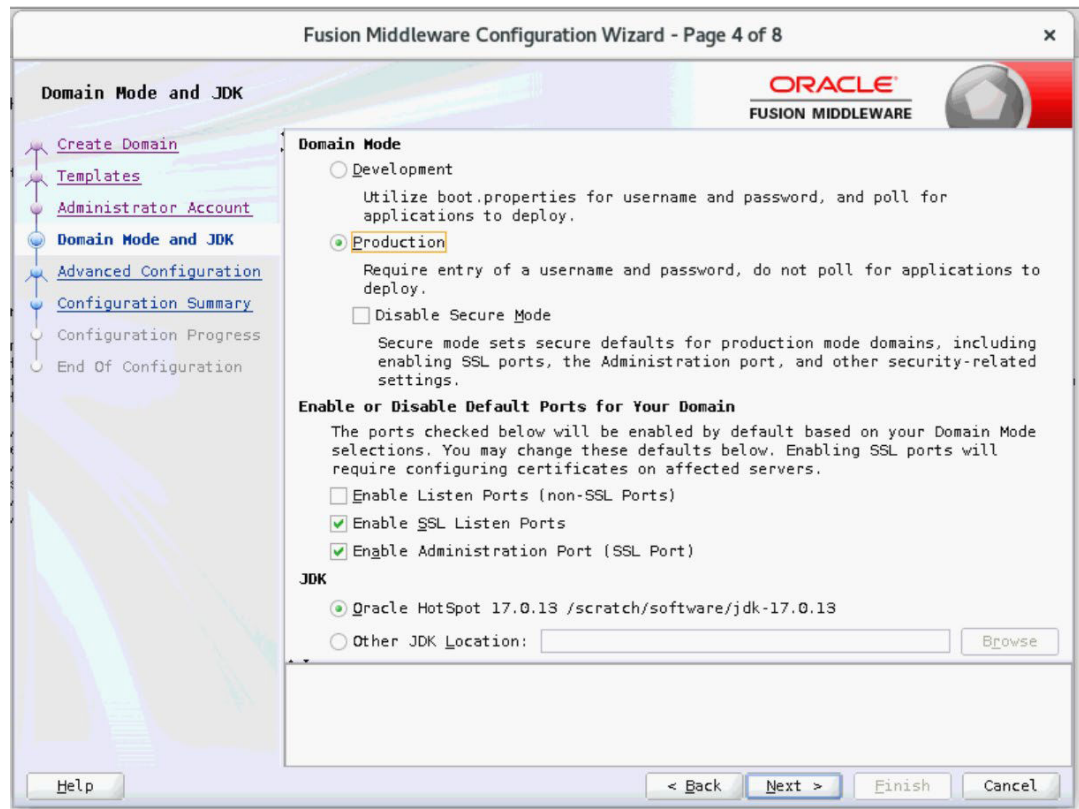
- The specified credentials are used to access Administration console.
- You can use this screen to define the default WebLogic Administrator account for the domain. This account is used to boot and connect to the domain's Administration Server. Click **Next**.

Figure 3-3 Administrator Account

The screenshot shows the 'Fusion Middleware Configuration Wizard - Page 3 of 8' window. The title bar includes the Oracle logo and 'FUSION MIDDLEWARE'. The left sidebar contains a navigation tree with the following items: 'Create Domain' (highlighted), 'Templates', 'Administrator Account' (selected), 'Domain Mode and JDK', 'Advanced Configuration', 'Configuration Summary', 'Configuration Progress', and 'End Of Configuration'. The main content area is titled 'Administrator Account' and contains three input fields: 'Name' (with the value 'weblogic'), 'Password' (masked with dots), and 'Confirm Password' (masked with dots). Below these fields is a red dot and a message: 'Must be the same as the password. Password must contain at least 8 alphanumeric characters with at least one number or special character.' At the bottom, there are four buttons: 'Help', '< Back', 'Next >', and 'Finish'. The 'Next >' button is highlighted.

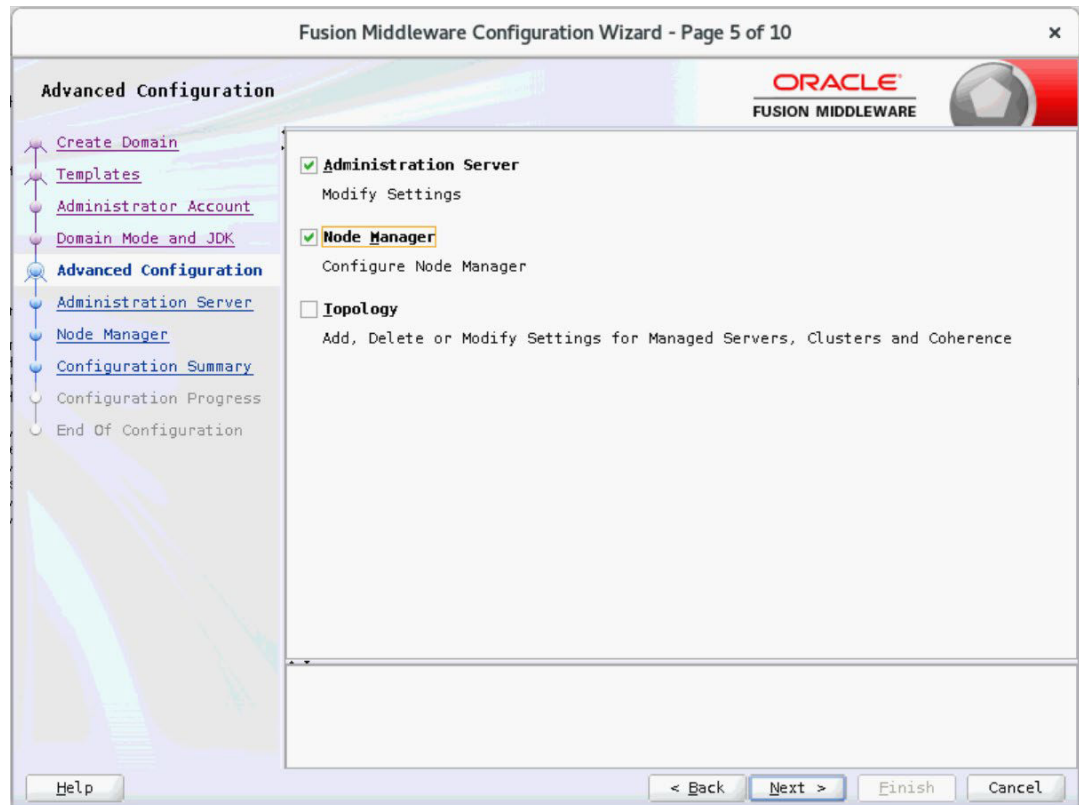
4. Select Server Startup as **Production Mode** and the available JDKs. Click **Next**.

Figure 3-4 Domain Mode



5. Select the check box adjacent to **Administration Server** and **Node Manager** Click **Next**.

Figure 3-5 Advanced Configuration



6. Specify the Administration server Listen address and Listen port.

Figure 3-6 Administration Server

The screenshot shows the 'Administration Server' configuration window. On the left is a navigation pane with the following items: Create Domain, Templates, Administrator Account, Domain Mode and JDK, Advanced Configuration, **Administration Server** (selected), Node Manager, Configuration Summary, Configuration Progress, and End Of Configuration. The main area is titled 'Administration Server' and contains the following fields and options:

- Server Name: AdminServer
- Listen Address: All Local Addresses
- Configure Administration Server Ports**
 - ☐ Enable Listen Port
 - ☒ Enable SSL Listen Port
 - Listen Port: 7001
 - SSL Listen Port: 7002
 - Administration Port: 9002

At the bottom are buttons for Help, < Back, Next >, Finish, and Cancel.

Note

The default Listen port is 7001 and SSL port is 7101. This could be changed to any other available port. Ensure to make a note, of this port since the same is required for launching the Admin console, post domain creation.

Note

Check for the port availability using the command - `netstat -anp |grep <Port no>`
The next screen navigates to NodeManager configuration.

7. Configure Node Manager

Select Per Domain Default Location option from Node Manager Type. And in the Node Manager Credentials, provide the username and password for the node manager. Click **Next**.

Figure 3-7 Administration Server

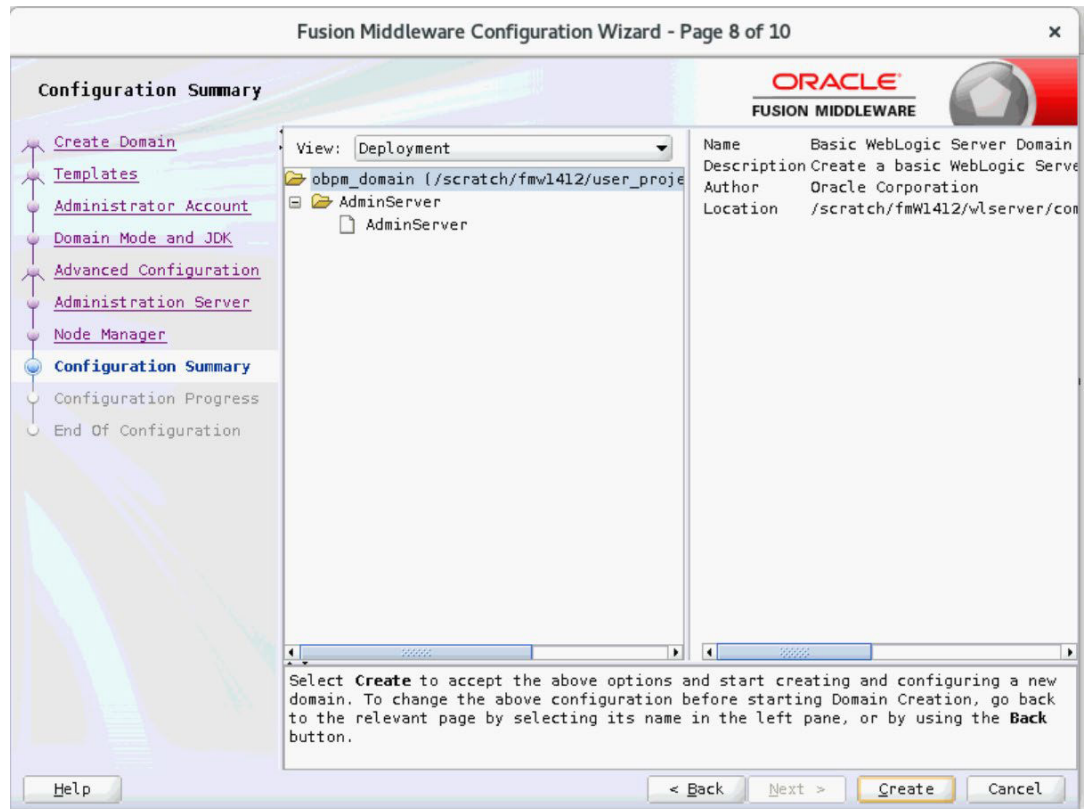
The screenshot displays the 'Fusion Middleware Configuration Wizard - Page 7 of 10'. The left sidebar shows a navigation tree with the following items: Create Domain, Templates, Administrator Account, Domain Mode and JDK, Advanced Configuration, Administration Server, Node Manager (selected), Configuration Summary, Configuration Progress, and End Of Configuration. The main content area is titled 'Node Manager' and contains the following sections:

- Node Manager Topology**
 - ☒ Per Domain Default Location
 - ☐ Per Domain Custom Location
 - Node Manager Home:
 - ☐ Manual Node Manager Setup
- Node Manager Credentials**
 - Username:
 - Password:
 - Confirm Password:

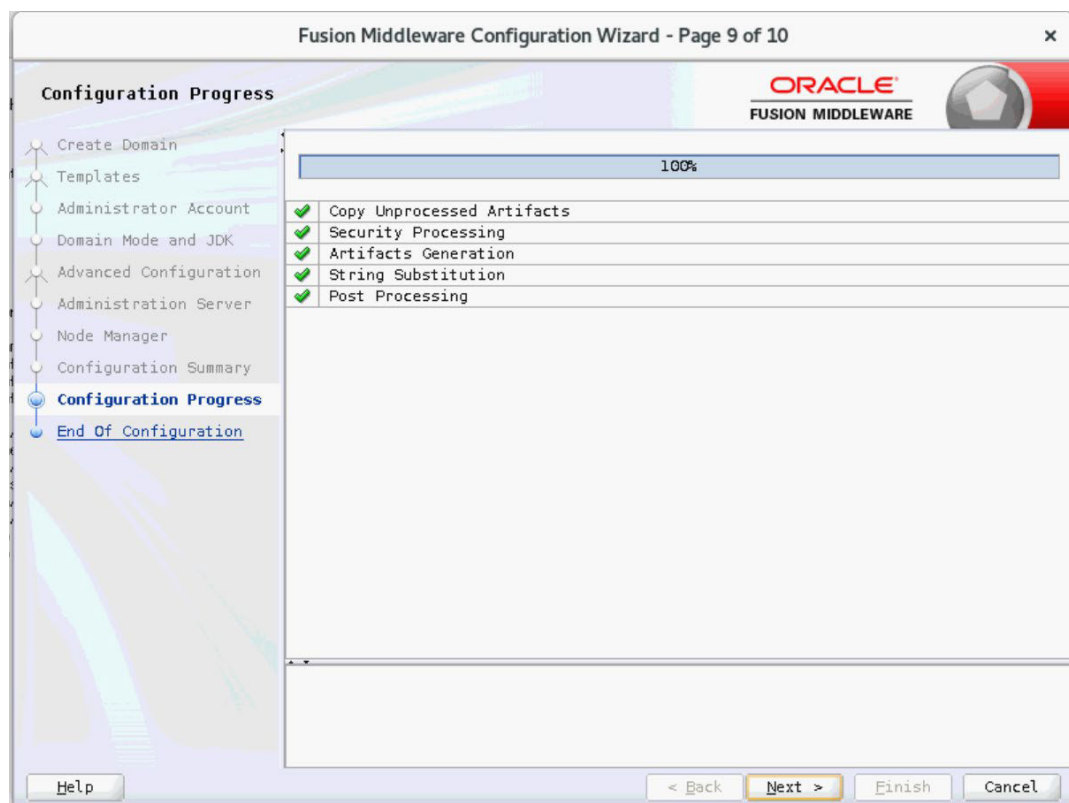
A note at the bottom states: 'Password must contain at least 8 alphanumeric characters with at least one number or special character.' The bottom of the window features a 'Help' button on the left and '< Back', 'Next >', 'Finish', and 'Cancel' buttons on the right.

8. Verify the details and click **Create**. The domain creation process is initiated and the progress of completion is indicated.

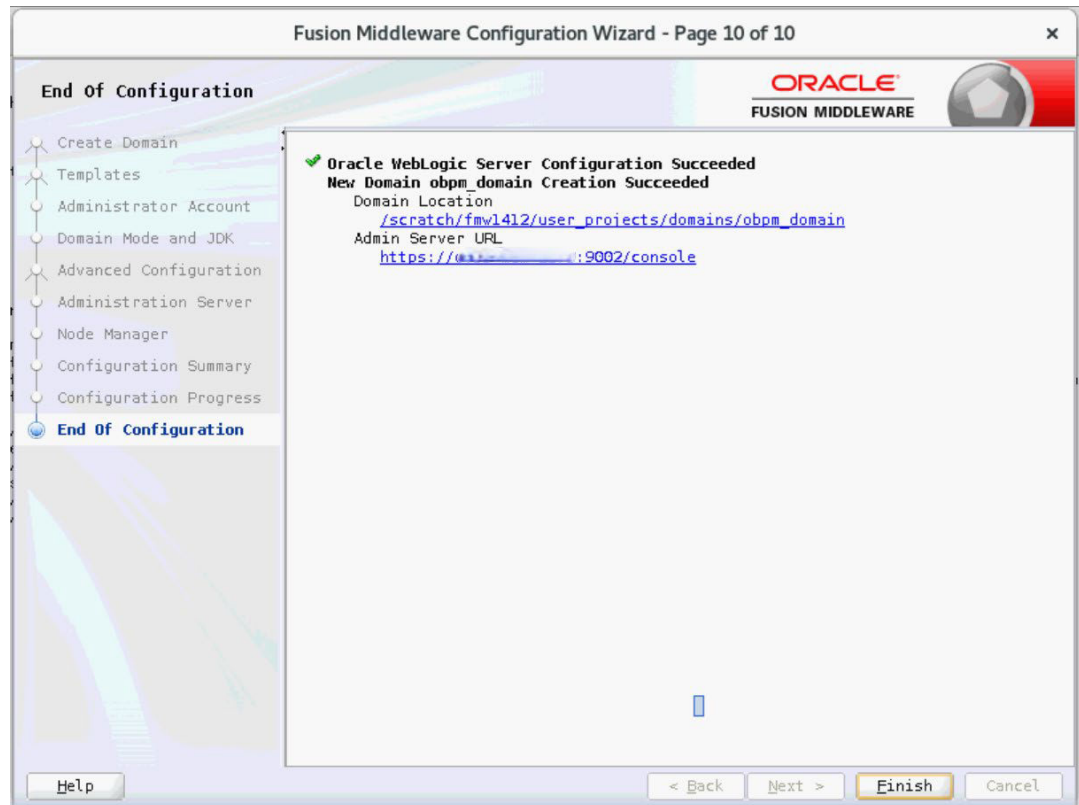
Figure 3-8 Configuration Summary



9. Click **Next**.

Figure 3-9 Configuration Progress

10. The Configuration Success message will be displayed as follows:

Figure 3-10 Configuration Success

The Admin Server console URL is as indicated below:

`http://<IP address>:<admin console port>/console`

- <IP address >: Host on which domain was created.
- <admin console port> : Port specified in Administration Server configuration page. In this case the Admin Console URL is: `https://<server1hostname>:7101/console`

3.2 Pack and Unpack Domain

The domain structure is expected to be copied to the second server during domain creation. To copy the same, you can use pack and unpack utility provided under `$WLSHOME/common/bin`.

Pack

Pack domain in primary server:

```
./pack.sh -managed=true -domain=/scratch/app/wl14c/user_projects/domains/FCUBSDomain -
template=/tmp/FCUBSDomain.jar -template_name="FCUBSDomain"
```

Unpack

Unpack FTP FCBUSDomain.jar in binary mode to secondary server under /tmp area and unpack the domain using unpack utility provided under `$WLSHOME/common/bin`

```
./unpack.sh -domain=/scratch/app/wl14c/user_projects/domains/FCUBSDomain -
template=/tmp/FCUBSDomain.jar
```

3.3 Domain and Resource Creation using WDT Tool

Using WDT tool, install and configure the weblogic domain:

1. Use `createDomain` cmd and provide the `FMW_home` path for further installation.
2. Download [WLmodel.text](#).
3. Change the file extension of **WLmodel.txt** to `.yaml` as **WLmodel.yaml**.
4. Update the values for **Domain Name**, **Cluster Name**, **Machine Name**, **IP Addresses**, **Ports**, and other relevant fields in `WLmodel` as required.
5. Run the command `sh createDomain.sh -oracle_home Oracle_Home Path -domain_type WLS -domain_parent Domain_Path -model_file Path to .yaml file`.

Note

This script creates domain with all the resources including JMS Queue.

3.4 Start Admin server

Admin server is started on the primary server. Login to primary server and navigate to folder `$DOMAIN_HOME/bin` and execute `startWeblogic.sh`.

3.5 Start Node Manager

Node Manager needs to be started on both the servers. Before starting the node manager update `ListenAddress` to the Hostname/IP Address of the machine in `nodemanager.properties` located in folder `$DOMAIN_HOME/nodemanager`

To start the node manager login to the servers and navigate to folder `$DOMAIN_HOME/bin` and execute `NodeManager.sh`

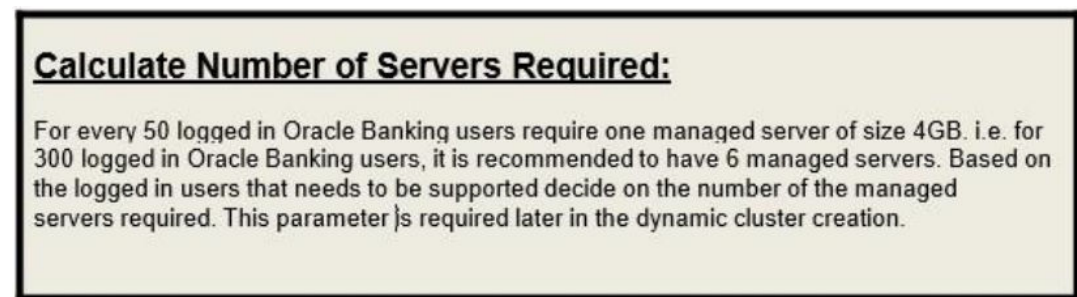
4

Cluster Configuration

Dynamic Cluster configuration involves below steps

1. Machine Configuration
2. Dynamic Cluster Creation: In a normal WebLogic Cluster you define Managed Server and add them to Cluster. In Dynamic Cluster, you select number of Servers you want in Cluster and Server Template you wish to assign to Servers in this WebLogic Dynamic Cluster.
- 3) Server template modification: Servers (or Managed Server) that are part of WebLogic Dynamic Cluster will have properties taken from Server Template.
3. Modify server template for best practices parameters for Dynamic Servers (part of Dynamic Cluster), you modify Server Template that is applicable to Dynamic Cluster. These settings are applicable to all the managed servers.
4. Activate Changes which would automatically create the managed servers (as mentioned in the number of servers required parameter).

Figure 4-1 Calculate Number



- [Machines Configuration](#)
- [Dynamic Cluster Creation](#)
- [Managed Server Template configuration](#)

4.1 Machines Configuration

1. Login into WebLogic Remote Console.
2. In the **Edit Tree**, navigate to **Environment**, then **Machines**.
3. Click **New**. Specify a name for the new machine. Click **Create**.
4. Configure the settings of your new machine under Environment, then Machines, then myMachine.
5. On the **Node Manager** tab, configure the following properties:
 - a. in the **Type** drop-down list, select the Node Manager type.
 - b. In the **Listen Address** field, enter the DNS name or IP address upon which Node Manager listens for incoming requests.

- c. In the **Listen Port** field, enter the port value where Node Manager listens for incoming requests(this is the port mentioned in nodemanager.properties file).
6. Click Save
7. Similarly create a **new machine** entry for the other server.

Verifying machine status

Before starting the managed servers, ensure that the Node manager Status of all the machines are "Reachable".

4.2 Dynamic Cluster Creation

Create a cluster that dynamically scales depending on the resource needs of your applications.

1. In the **Edit Tree**, navigate to **Environment**, then **Clusters**. Click **New**.
2. Specify the **Cluster Name** and click **Create**.
3. For more details, refer [Create a Cluster](#) and [Create a Server Template](#).

4.3 Managed Server Template configuration

The server template created is modified to apply the below parameters

- [Logging](#)
- [HTTP Logging](#)
- [Stuck Tread Max Time](#)

4.3.1 Logging

The process of log file writing in a Weblogic server can impact the performance. Hence, you need to keep the logging to minimum in a production environment.

Update below parameters by in Logging Screen

Minimum Severity to log	Warning
Log file Severity level	Warning
Standard Out Severity level	Critical
Domain broadcaster Severity level	Critical

1. For this setting, In the **Edit Tree**, Navigate to **Environment -> Server Templates (Select template) -> Logging -> General** tab.

Figure 4-2 Settings For FCUBSMST Template

Server Templates / FCUBSMSTemplate

Save

Use this page to define the general logging settings for this server.

General Protocols Cluster Security Migration Web Services Health **Logging** Debug Services Advanced Referenced By

General HTTP Data Source Severity Properties Platform Logger Levels

☒ Show Advanced Fields

Files to retain	100	Log file rotation directory	
Rotate log file on startup	<input type="checkbox"/>	Date Format Pattern	MMM d, y, h:mm:ss,SSS a zz
Minimum severity to log	Warning	Redirect stdout logging enabled	<input type="checkbox"/>
Redirect stderr logging enabled	<input type="checkbox"/>	Log Monitoring Enabled	<input checked="" type="checkbox"/>
Log monitoring interval seconds	30	Log Monitoring Throttle Threshold	1500
Log Monitoring Throttle Message Length	50	Log monitoring maximum throttle message signature count	1000
Log File Severity Level	Warning	Log File Filter	None
Log File Buffer	8	Stdout Severity Level	Critical
Stdout Filter	None	Domain Log Broadcast Severity Level	Critical

4.3.2 HTTP Logging

1. In **Edit Tree**, Navigate to **Environment -> Server Templates (Select template) -> Logging -> HTTP** tab – Uncheck HTTP access logs enabled.

Figure 4-3 Settings for FCUBSMST Template

Server Templates / FCUBSMSTemplate

Save

Use this page to configure HTTP logging for the server. By default, HTTP logging is enabled and the server saves HTTP requests in a separate log file; it does not store HTTP requests in the server log file or the domain log file.

General Protocols Cluster Security Migration Web Services Health **Logging** Debug Services Advanced Referenced By

General **HTTP** Data Source Severity Properties Platform Logger Levels

☒ Show Advanced Fields

HTTP access log file enabled	<input type="checkbox"/>	Log file name	logs/access.log
Log Rotation Type	bySize	Rotation file size	5000
Begin rotation time	00:00	Rotation interval	24
Limit number of retained files	<input checked="" type="checkbox"/>	Files to retain	100
Log file rotation directory		Rotate log file on startup	<input type="checkbox"/>
Format	common	Extended Logging Format Fields	date time cs-method cs-uri
Log File Buffer	8	Log Time In GMT	<input type="checkbox"/>

4.3.3 Stuck Tread Max Time

1. In **Edit Tree**, Navigate to **Environment -> Server Templates (Select template) -> Advanced -> Tuning**, update the stuck thread max time to 900.

Figure 4-4 Settings for FCUBSMST Template

The screenshot shows the Oracle Weblogic Server configuration interface for the FCUBSMST template. The 'Advanced' tab is selected, and the 'Tuning' sub-tab is active. The 'Stuck Thread Timer Interval' is highlighted with a red box and set to 900. Other settings include 'Enable Native IO' (checked), 'Enable Gathered Writes' (unchecked), 'Maximum Open Sockets' (-1), 'Accept Backlog' (300), 'Reverse DNS Allowed' (unchecked), 'JavaSocketMuxer Socket Readers' (33), 'Enable Scattered Reads' (unchecked), 'Login Timeout' (5000), and 'Self Tuning Thread Minimum Pool Size' (1).

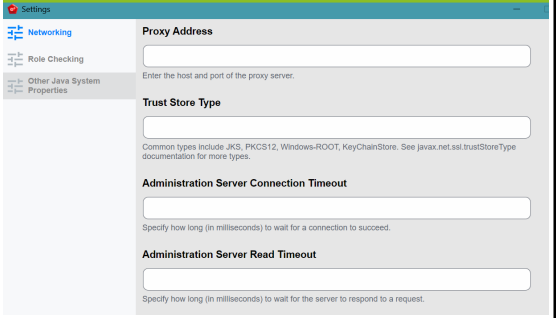
Setting	Value
Enable Native IO	<input checked="" type="checkbox"/>
Enable Gathered Writes	<input type="checkbox"/>
Maximum Open Sockets	-1
Accept Backlog	300
Reverse DNS Allowed	<input type="checkbox"/>
JavaSocketMuxer Socket Readers	33
Enable Scattered Reads	<input type="checkbox"/>
Stuck Thread Timer Interval	900
Login Timeout	5000
Self Tuning Thread Minimum Pool Size	1

5

Tuning

- [General Parameters](#)
- [JVM Tuning](#)

5.1 General Parameters

PARAMETER	VALUE	Navigate To
JTA Time out seconds	900 (seconds)	Login to Weblogic Server console. In the Edit Tree , go to Services , then JTA . Timeout Seconds
Session Timeout	900 (seconds)	WebLogic Remote Console -> File menu option - > Settings sub menu Administration Sever Connection Timeout 

5.2 JVM Tuning

This section of the document provides JVM optimization for Oracle Banking Payments Solution.

Figure 5-1 JVM

It is strictly recommended to use 64 bit JVM for OBPM installation, as 32 bit JVM is obsolete for enterprise application and imposes a restriction on heap size to 4GB.

The JAVA minimum and maximum heap size needs to be reset for 32 and 64 bit environments. Both the minimum and maximum heap size is set to 1.5GB and 4GB in case of 32 bit and 64 bit environments respectively.

How to find whether the JVM is 32bit or 64bit?

Go to \$JAVA_HOME/bin directory. Check java version using command `./java -d64` – version 64 bit JVM shows the version details where as 32bit throws an error.

How to modify the JVM heap parameters?

To change the JVM heap parameters modify `setDomainEnv.sh` under domain `FCUBSCL` in both servers. This file is located at

`"$WL_HOME/user_projects/domains/$WLS_DOMAIN/bin"` in both the servers.

Use below `USER_MEM_ARGS` variable to override the standard memory arguments passed to java for SUN JDK.

32 bit JDK

It is strictly not recommended to use a 32 bit JDK version because this have a limitation of maximum heap size setting to 4 GB only. This restricts the application scalability.

64 bit JDK

`USER_MEM_ARGS="-`

`Dorg.apache.xml.dtm.DTMManager=org.apache.xml.dtm.ref.DTMManagerDefault - Dorg.apache.xerces.xni.parser.XMLParserConfiguration=org.apache.xerces.parser.s.XML11Configuration -Dweblogic.threadpool.MinPoolSize=100 -Dweblogic.threadpool.MaxPoolSize=100 -Xms8g -Xmx8g -server -d64 -XX:+UseParallelOldGC -XX:ParallelGCThreads=4"` export `USER_MEM_ARGS`

Note

Take a backup of the files before modifying the same.

6

Data Source creation and JDBC Configuration

Following are the JNDI names of those data sources used by Oracle Banking application.

- jdbc/fcjCoreDS - This datasource is used by Oracle Banking Core online screen excluding branch screens.
- jdbc/fcjdevDSBranch - This datasource is used by Branch screens.
- jdbc/fcjSchedulerDS - This datasource is used by Quartz scheduler.
- fcjPayDS_GTXN – Datasource for Oracle Banking Payment OLTP with Global Transaction supported to enable the transaction management by the container and also to.
- fcjPayDS – Datasource used by Payment Batch operations where transaction is handled programmatically.

Note

- jdbc/fcjdevDS should be NonXA and make use of OCI driver.
- jdbc/fcjdevDSBranch and jdbc/fcjSchedulerDS should be XA.

- [Setup Required for OCI Driver](#)
- [Data Source Creation](#)
- [JDBC Parameters Tuning](#)

6.1 Setup Required for OCI Driver

Data sources are created with OCI enabled. For this, Oracle Instant Client is required, below steps needs to be followed

- Download Oracle Instant Client corresponding to the used Oracle DB and java (x64 or x32):

<https://www.oracle.com/database/technologies/instant-client/downloads.html>
- Set {ORACLE_HOME} in the environment variable.
- Update the environment variable LD_LIBRARY_PATH as {ORACLE_HOME}/lib. This is to load all the .so files.
- Ensure that the ojdbc*.jar file in {WL_HOME}/server/lib/ojdbc*.jar is the same as the file {ORACLE_HOME}/jdbc/lib/ojdbc*.jar. This is to ensure compatibility.
- Update LD_LIBRARY_PATH in StartWebLogic.sh or in setDomainEnv.sh. This must be the path of directory where Oracle Instant Client is installed.
- If you are still not able to load the .so files, then you need to update the EXTRA_JAVA_PROPERTIES by setting Djava.library.path as {ORACLE_HOME}/lib in StartWebLogic.sh or in setDomainEnv.sh.

6.2 Data Source Creation

Refer to the section "Create Data Source" in [Weblogic Configuration](#) installation guide.

OBPM related datasource parameter recommendation are provided in Annexure A

6.3 JDBC Parameters Tuning

Below JDBC parameters needs to updated for all the Datasources

PARAMETER	VALUE	Navigate To
Connection Reserve time out	30	Connection Pool->Advance
Test Frequency	60	Connection Pool->Advance
Inactive connection time out	30	Connection Pool->Advance
Initial Capacity	1	Connection Pool
Max capacity	Based on Site Requirement	Connection Pool
Capacity Increment	5	Connection Pool
Shrink Frequency	900	Connection Pool->Advance
Test Connection on Reserve	Checked	Connection Pool->Advance

7

JMS Resource Creation

JMS Resource Creation involves various steps

- Persistence Store Creation
- JMS Server Creation
- JMS Module Creation
- Resource Creation: Connection Factory and Queue's

Refer to the [JMS Cluster Configuration](#) document for further details on JMS setup.

8

Oracle WebLogic Load Balancing

There are four major components for load balancing:

- HTTPS Requests (HTTPS)
- Web Service Requests (HTTPS)
- Rest API Requests (HTTPS)
- JMS Requests (t3)

External Load Balancer:

First three modes can be load balanced by fronting a simple web server to virtualize the IP address/host name to one and route the requests to the managed servers in a round robin or weight based or any other supported algorithm. Oracle HTTP server is an example for such web server.

JMS uses a t3 protocol and it requires load balancer which can intercept the request at level 7 (TCP level) and route the request. Example is Oracle Traffic Director (OTD). Such load balancers can be used for first three type of requests as well.

Internal Load Balancer:

For http/s protocols, there is no production level internal load balancing support from weblogic. This has to be handled using an external load balancer.

Weblogic supports JMS load balancing internally. For this, the client have to know the host name/ip address and port of all the managed servers in a cluster. In all the places where the Initial Context Provider URL is configured in OBPM application, specify the URL as:

t3://[HOST NAME 1]:[PORT 1],[HOST NAME 2]:[PORT 2]

HOST NAME – This is a DNS name of individual manager servers

PORT – Port number of individual managed servers.

9

Frequently Asked Questions

- [Machine status is Unreachable](#)
- [How to restart node manager?](#)
- [Scaling Up Dynamic Cluster](#)
- [Session Timeout](#)

9.1 Machine status is Unreachable

If the machine status is unreachable, means that machine is not reachable and from console you cannot start/stop the managed servers.

In the console, navigate through Domain structure → Machines → machine1 → Monitoring → Node Manager Status will be Unreachable

To change the status, you need to start the nodemanager on that server. Refer to start nodemanager section on steps to start the nodemanager.

9.2 How to restart node manager?

1. Locate node manager pid using `ps -ef|grep weblogic.nodemanager.javaHome`
2. Change directory to `$DOMAIN_HOME/bin`
3. Kill the unix process using `kill -9 <pid>`
4. Verify that the node manager is killed by `tail -f nohup.out`
5. Start node manager using `nohup ./startNodeManager.sh`
6. Verify nodemanager is started using `tail -f nohup.out`

9.3 Scaling Up Dynamic Cluster

When the capacity is insufficient and you need to scale-up, you can add dynamic servers on demand.

1. Navigate to the **Environment**, then **Clusters**, then **myCluster** and select the **Dynamic** tab.
2. Change the **Max Dynamic Cluster Size** to 8 and Click **Save**.

9.4 Session Timeout

Session timeouts occur intermittently during load condition. Verify the following:

1. Clock Synchronization: Time across the nodes/machines is same.
2. Session Stickiness in load balancer: Persistence Type in load balancer should be set to SOURCE IP and should not be cookie.

Annexure A – Datasource Parameter Recommendations

For the recommended parameter values for data sources, refer OBPM-WeblogicRecommendations