

Oracle® Banking APIs

Installation Guide- Non-Linux Platforms



Patchset Release 22.2.4.0.0

F99655-01

June 2024

The Oracle logo, consisting of a red square with the word "ORACLE" in white, uppercase letters inside it.

ORACLE®

Copyright © 2006, 2022, 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

Preface

Purpose	v
Audience	v
Documentation Accessibility	v
Diversity and Inclusion	v
Conventions	vi
Related Resources	vi
Screenshot Disclaimer	vi
Acronyms and Abbreviations	vi

1 Manual OBAPI Installation

1.1	Create OBAPI Tablespace (file obapi_create_tablespace.sql)	1-1
1.2	Create Audit Tablespace (file obapi_audit_create_tablespace.sql)	1-2
1.3	Create User (file obapi_create_user.sql)	1-2
1.4	Create Role (file obapi_create_role.sql)	1-2
1.5	Grants Execution (file clip_user_grants.sql)	1-3
1.6	Files execution in sequences on above schema (ex. OBAPI_\${POST_FIX})	1-3
1.7	OBPM Database Installation (OBPM Favor)	1-3
1.8	Tablespace Creation (file obpm_create_tablespace.sql)	1-3
1.9	Create Bigfile Tablespace TBS_\${EHMS_SCHEMA_NAME }	1-3
1.10	User Creation (file obpm_create_user.sql)	1-3
1.11	Create Role (file obpm_create_role.sql)	1-4
1.12	Create Role ROLE_\${ EHMS_SCHEMA_NAME } NOT IDENTIFIED	1-4
1.13	Grants Execitions	1-4
1.14	Scripts Execution	1-4
1.15	Policy Seeding	1-5

2 WEBLOGIC Setup and Configuration

2.1	Setting Domain JTA Transaction Timeout	2-1
2.2	Creating DIGX Data Source	2-2
2.3	Creating NONXA Data Source	2-5
2.4	Creating BATCH Data Source	2-8

2.5	Creating SYSCONFIG Data Source	2-11
2.6	Creating B1A1 Data Source	2-14
2.7	Create JMS Server and JMS Module	2-17
2.8	Creating WLS_JMS_AUDIT_PS FileStore	2-29
2.9	Creating AuditJMSServer JMS Server	2-29
2.10	Creating WLS_JMS_REPORT_PS FileStore	2-29
2.11	Creating ReportsJMSServer JMS Server	2-39
2.12	Creating jpa-cache JMS Server	2-41
2.13	Creating WLS_JPA_PS FileStore	2-41
2.14	Creating ExtSystemReceiver JMS Server - WLS_JMS_EXTSYSRECEIVER_PS FileStore	2-42
2.15	Creating ExtSystemSender JMS Server Persistent Store FileStore as WLS_JMS_EXTSYSENDER_PS	2-43
2.16	Creating UBSForeignServer JMS Server	2-45
2.17	Creating OBPMForeignServer JMS Server	2-45

3 Deploying Applications

4 Configured jps-config.xml

5 List of Topics

Index

Preface

- [Purpose](#)
- [Audience](#)
- [Documentation Accessibility](#)
- [Diversity and Inclusion](#)
- [Conventions](#)
- [Related Resources](#)
- [Screenshot Disclaimer](#)
- [Acronyms and Abbreviations](#)

Purpose

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

Audience

This document is intended for the following audience:

- Customers
- Partners

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 customer access to and use of Oracle support services will be pursuant to the terms and conditions specified in their Oracle order for the applicable services.

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.

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.

Related Resources

For more information on any related features, refer to the following documents:

- Oracle Banking APIs Installation Manuals

Screenshot Disclaimer

Personal information used in the interface or documents is dummy and does not exist in the real world. It is only for reference purposes.

Acronyms and Abbreviations

The list of the acronyms and abbreviations used in this guide are as follows:

Table 1 Acronyms and Abbreviations

Abbreviation	Description
OBAPI	Oracle Banking APIs

1

Manual OBAPI Installation

OBAPI Database Installation with OBPM FLAVOR

Create required OBAPI tablespace and user in below sequence.

- Create OBAPI Tablespace (file obapi_create_tablespace.sql)
- Create Audit Tablespace (file obapi_audit_create_tablespace.sql)
- Create User (file obapi_create_user.sql)
- Create Role (file obapi_create_role.sql)
- Grants Execution (file clip_user_grants.sql)
- Files execution in sequences on above schema (ex. OBAPI_\${POST_FIX})
- OBPM Database Installation (OBPM Favor)
- Tablespace Creation (file obpm_create_tablespace.sql)
- Create Bigfile Tablespace TBS_\${EHMS_SCHEMA_NAME }
- User Creation (file obpm_create_user.sql)
- Create Role (file obpm_create_role.sql)
- Create Role ROLE_\${ EHMS_SCHEMA_NAME } NOT IDENTIFIED
- Grants Execitions
- Scripts Execution
- Policy Seeding

1.1 Create OBAPI Tablespace (file obapi_create_tablespace.sql)

Execute the file available @ \${OBAPI_INSTALLER}/installables/db/OBAPI/obapi_create_tablespace.sql.

Update the datafile path and tablespace name and execute the file.

Example:

Refer installer.properties file variable POST_FIX and replace in the below command.

```
CREATE BIGFILE TABLESPACE OBAPI_${POST_FIX }
  DATAFILE '${DATAFILE_PATH}/OBAPI_${ POST_FIX }.dbf'
  SIZE 500M
  AUTOEXTEND ON NEXT 100M
  LOGGING
  EXTENT MANAGEMENT LOCAL SEGMENT SPACE MANAGEMENT AUTO;
```

1.2 Create Audit Tablespace (file obapi_audit_create_tablespace.sql)

Execute the file available @ \${OBAPI_INSTALLER}/installables/db/OBAPI/obapi_audit_create_tablespace.sql.

Example :

Refer installer.properties file variable POST_FIX and replace in the below command.

```
CREATE BIGFILE TABLESPACE OBAPI_AUDIT_${ POST_FIX}
DATAFILE '${DATAFILE_PATH} /OBAPI_AUDIT_${ POST_FIX}.dbf'
SIZE 500M
AUTOEXTEND ON NEXT 100M
LOGGING EXTENT MANAGEMENT LOCAL SEGMENT SPACE MANAGEMENT AUTO;
```

1.3 Create User (file obapi_create_user.sql)

Execute the file available @ \${OBAPI_INSTALLER} /installables/db/OBAPI/obapi_create_user.sql.

Example:

Refer installer.properties file variable POST_FIX and replace in the below command

```
create user OBAPI_${ POST_FIX} identified by welcome1;
alter user OBAPI_${ POST_FIX} default tablespace OBAPI_${POST_FIX};
alter user OBAPI_${ POST_FIX} temporary tablespace temp;
alter user OBAPI_${ POST_FIX} quota unlimited on OBAPI_${ POST_FIX} ;
alter user OBAPI_${ POST_FIX} quota unlimited on OBAPI_AUDIT_${ POST_FIX};
```

1.4 Create Role (file obapi_create_role.sql)

Execute the file available @ \${OBAPI_INSTALLER} /installables/db/OBAPI/obapi_create_role.sql.

Example:

Refer installer.properties file variable POST_FIX and replace in the below command

```
CREATE ROLE OBAPI_ROLE_${POST_FIX} NOT IDENTIFIED;
grant CONNECT, CREATE SESSION, CREATE TABLE,
CREATE SEQUENCE,CREATE TRIGGER, CREATE DATABASE LINK,CREATE VIEW,
CREATE PROCEDURE, CREATE SYNONYM, CREATE TYPE,
CREATE JOB to OBAPI_ROLE_${POST_FIX};
grant OBAPI_ROLE_${POST_FIX}to OBAPI_${POST_FIX};
```


1.5 Grants Execution (file clip_user_grants.sql)

Execute the file available @ \${OBAPI_INSTALLER} /installables/db/OBAPI/
clip_user_grants.sql.

Example:

Refer installer.properties file variable POST_FIX and replace in the below command

```
grant create any sequence to OBAPI_${POST_FIX};  
grant execute on DBMS_LOCK to OBAPI_${POST_FIX};
```

1.6 Files execution in sequences on above schema (ex. OBAPI_\${POST_FIX})

- clip_master_script.sql
- clip_constraints.sql
- clip_seeds_executable.sql
- clip_master_generic_rest_script.sql

***** SUCCESSFULLY installed OBAPI database *****

1.7 OBPM Database Installation (OBPM Favor)

Create required OBAPI tablespace and user in below sequence.

1.8 Tablespace Creation (file obpm_create_tablespace.sql)

Execute the file available @ \${OBAPI_INSTALLER} /installables/db/OBPM/
obpm_create_tablespace.sql.

Example:

Refer installer.properties file variable EHMS_SCHEMA_NAME and replace in the below command.

1.9 Create Bigfile Tablespace TBS_\${EHMS_SCHEMA_NAME }

```
DATAFILE '${DATAFILE_PATH}/TBS_${EHMS_SCHEMA_NAME}.dbf'  
SIZE 500M  
AUTOEXTEND ON NEXT 100M  
LOGGING  
EXTENT MANAGEMENT LOCAL  
SEGMENT SPACE MANAGEMENT AUTO;
```

1.10 User Creation (file obpm_create_user.sql)

Execute the file available @ \${OBAPI_INSTALLER} /installables/db/OBPM/
obpm_create_user.sql.

Example:

Refer installer.properties file variable EHMS_SCHEMA_NAME and replace in the below command

```
create user ${ EHMS_SCHEMA_NAME } identified by welcome1;  
alter user ${ EHMS_SCHEMA_NAME } default tablespace TBS_${EHMS_SCHEMA_NAME };  
alter user ${ EHMS_SCHEMA_NAME } temporary tablespace temp;  
alter user ${ EHMS_SCHEMA_NAME } quota unlimited on TBS_${EHMS_SCHEMA_NAME };
```

1.11 Create Role (file obpm_create_role.sql)

Execute the file available @ \${OBAPI_INSTALLER} /installables/db/OBPM/
obpm_create_role.sql.

Example:

Refer installer.properties file variable EHMS_SCHEMA_NAME and replace in the below command

1.12 Create Role ROLE_\${ EHMS_SCHEMA_NAME } NOT IDENTIFIED

```
grant CONNECT, CREATE SESSION, CREATE TABLE, CREATE SEQUENCE,  
CREATE TRIGGER, CREATE DATABASE LINK,CREATE VIEW, CREATE PROCEDURE,  
CREATE SYNONYM, CREATE TYPE,CREATE JOB to ROLE_${ EHMS_SCHEMA_NAME };  
grant ROLE_${ EHMS_SCHEMA_NAME } to ${ EHMS_SCHEMA_NAME } ;
```

1.13 Grants Execitions

Replace \$\$schema with \${ EHMS_SCHEMA_NAME } in the below files

- Execute the file available @ \${OBAPI_INSTALLER} /installables/db/OBPM/
FCUBS_GR_PRIV.sql
- Execute the file available @ \${OBAPI_INSTALLER} /installables/db/OBPM/
FCOBPM_GR_PRIV.sql

1.14 Scripts Execution

- Execute the file available @ \${OBAPI_INSTALLER} /installables/db/OBPM/table-
scripts.sql
- Execute the file available @ \${OBAPI_INSTALLER} /installables/db/OBPM/
ubs_object_scripts.sql
- Execute the file available @ \${OBAPI_INSTALLER} /installables/db/OBPM/
obpm_object_scripts.sql
- Execute the file available @ \${OBAPI_INSTALLER} /installables/db/OBPM/execute-
seeds.sql
- Execute the file available @ \${OBAPI_INSTALLER} /installables/db/OBPM/obpm-
seeds.sql

- **Execute the file available** @ \${OBAPI_INSTALLER} /installables/db/OBPM/DIGX_FW_CONFIG_ALL_O.sql
- **Execute the file available** @ \${OBAPI_INSTALLER} /installables/db/OBPM/DIGX_FW_ABOUT_OBPM.sql
- **Execute the file available** @ \${OBAPI_INSTALLER} /installables/db/OBPM/DIGX_FW_CONFIG_VAR_B.sql
- **Execute the file available** @ \${OBAPI_INSTALLER} /installables/db/OBPM/DIGX_FW_CONFIG_UBS_ALL_O.sql

1.15 Policy Seeding

```
TEMP_PATH=Temporary Path
cp ${OBAPI_INSTALLER}/installables/policies/Entitlement_log4j.properties to
TEMP_PATH/db/Entitlement_log4j.properties

cp ${OBAPI_INSTALLER}/installables/policies/Task_log4j.properties to
TEMP_PATH/db/Task_log4j.properties

cp ${OBAPI_INSTALLER}/installables/policies/Dashboard_seed_log4j.properties to
TEMP_PATH/db/Dashboard_seed_log4j.properties

update <logs_path> in the above file (TEMP_PATH) to desired location.
```

Execute below command in sequence.

Were SCHEMA_NAME=OBAPI_\${POST_FIX} and SCHEMA_PASS= Password of OBAPI_\${POST_FIX} .

```
# $JAVA_HOME/bin/java -Djava.util.logging.config.file= TEMP_PATH/db/
Task_log4j.properties
-jar ${OBAPI_INSTALLER}/installables/policies/
com.ofss.digx.utils.feed.data.task.jar
/installables/policies/Task.csv oracle.jdbc.OracleDriver SCHEMA_NAME
SCHEMA_PASS
'jdbc:oracle:thin:@OBAPI_DATABASE_HOSTNAME:OBAPI_DATABASE_PORT/
OBAPI_DATABASE_SID'
```

```
# $JAVA_HOME/bin/java -Djava.util.logging.config.file= TEMP_PATH/db/
Dashboard_seed_log4j.properties
-jar ${OBAPI_INSTALLER}/installables/policies/
com.ofss.digx.utils.dashboard.jar
${OBAPI_INSTALLER}/installables/policies/dashboard_json/
oracle.jdbc.OracleDriver SCHEMA_NAME
SCHEMA_PASS 'jdbc:oracle:thin:@OBAPI_DATABASE_HOSTNAME:OBAPI_DATABASE_PORT/
OBAPI_DATABASE_SID'
```

```
# $JAVA_HOME/bin/java -Djava.util.logging.config.file= TEMP_PATH/db/
Entitlement_log4j.properties
-jar ${OBAPI_INSTALLER}/installables/policies/
com.ofss.digx.utils.entitlement.feed.data.jar
${OBAPI_INSTALLER}/installables/policies/Resources.csv
${OBAPI_INSTALLER}/installables/policies/Entitlement.csv ${OBAPI_INSTALLER}/
installables/policies/Day0Policy.csv
```

```
KERNEL oracle.jdbc.OracleDriver SCHEMA_NAME SCHEMA_PASS '  
jdbc:oracle:thin:@OBAPI_DATABASE_HOSTNAME:OBAPI_DATABASE_PORT/  
OBAPI_DATABASE_SID'
```

2

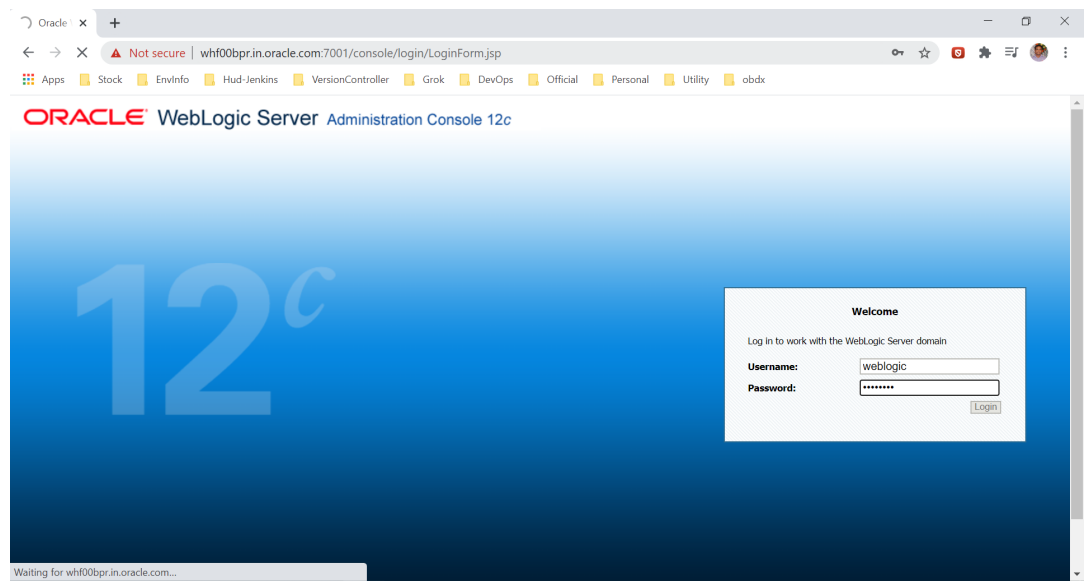
WEBLOGIC Setup and Configuration

Assuming that rcu , weblogic domain created , managed server, cluster created , node manager configured.

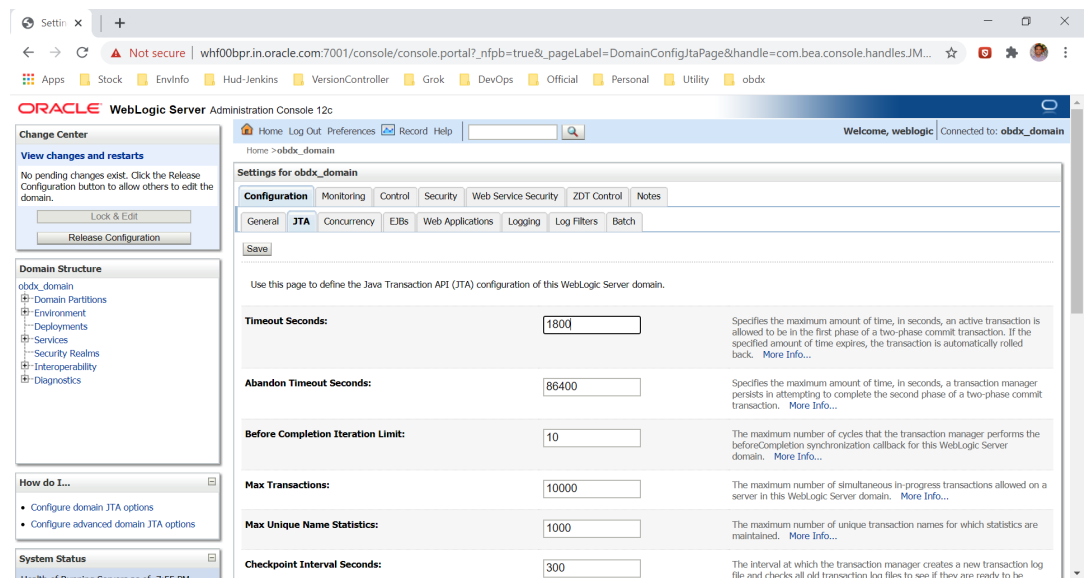
- [Setting Domain JTA Transaction Timeout](#)
- [Creating DIGX Data Source](#)
- [Creating NONXA Data Source](#)
- [Creating BATCH Data Source](#)
- [Creating SYSCONFIG Data Source](#)
- [Creating B1A1 Data Source](#)
- [Create JMS Server and JMS Module](#)
- [Creating WLS_JMS_AUDIT_PS FileStore](#)
- [Creating AuditJMSServer JMS Server](#)
- [Creating WLS_JMS_REPORT_PS FileStore](#)
- [Creating ReportsJMSServer JMS Server](#)
- [Creating jpa-cache JMS Server](#)
- [Creating WLS_JPA_PS FileStore](#)
- [Creating ExtSystemReceiver JMS Server - WLS_JMS_EXTSYSRECEIVER_PS FileStore](#)
- [Creating ExtSystemSender JMS Server Persistent Store FileStore as WLS_JMS_EXTSYSENDER_PS](#)
- [Creating UBSForeignServer JMS Server](#)
- [Creating OBPMForeignServer JMS Server](#)

2.1 Setting Domain JTA Transaction Timeout

1. Logging into weblogic domain with admin credentials (ex. weblogic).

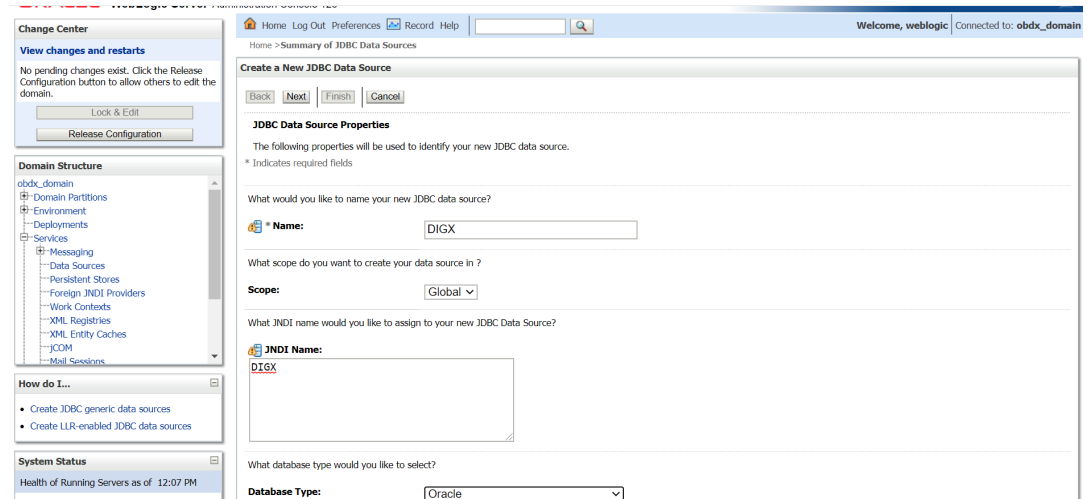


2. Click on DOMAIN_NAME → JTA → set Timeout Seconds to 1800 → click **Save** → Activate changes.

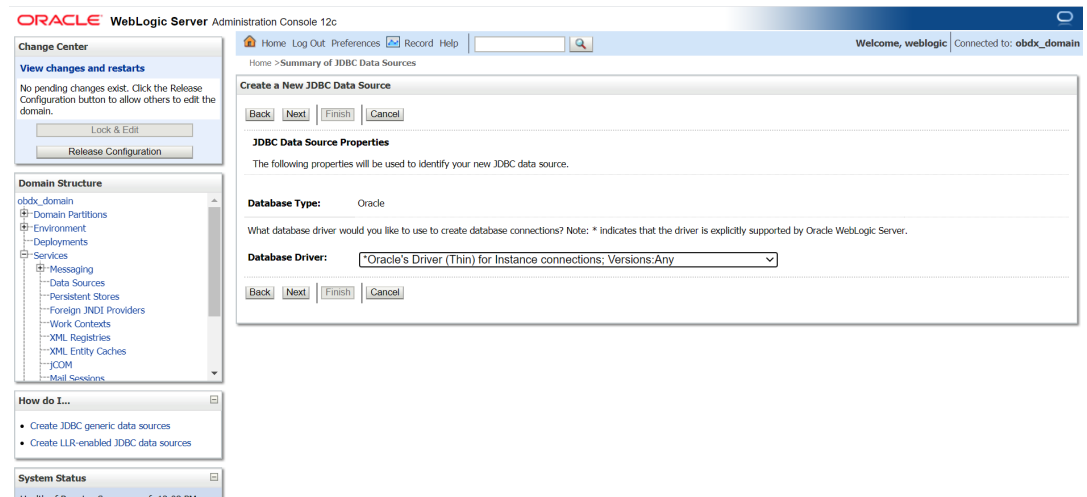


2.2 Creating DIGX Data Source

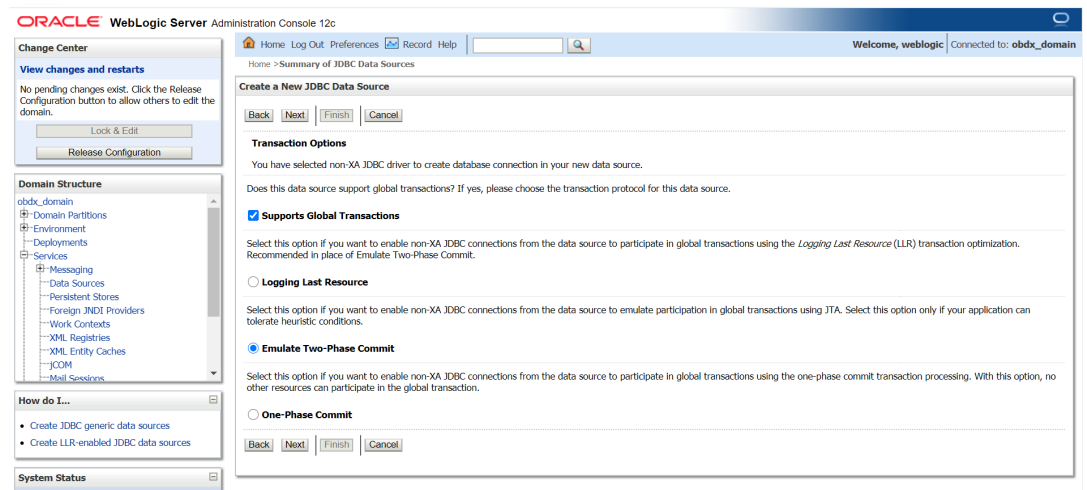
1. Navigate to Data Source → click **New** → Provide details and click **Finish**.



- Name:** DIGX
JNDI Name: DIGX



- Select Oracle's Driver (Thin) for Instance connections;



4. Select Emulate Two-Phase Commit.

Configuration button to allow others to edit the domain.

Lock & Edit
Release Configuration

Domain Structure

- obdx_domain
 - Domain Partitions
 - Environment
 - Deployments
 - Services
 - Data Sources
 - Persistent Stores
 - Foreign JNDI Providers
 - Work Contexts
 - XML Registries
 - XML Entity Caches
 - JCOM
 - Mail Sessions

How do I...

- Create JDBC generic data sources
- Create LLR-enabled JDBC data sources

System Status

Health of Running Servers as of 12:12 PM

Failed (0)

Critical (0)

Overloaded (0)

Warning (0)

Back Next Finish Cancel

Connection Properties
Define Connection Properties.

What is the name of the database you would like to connect to?

Database Name: obdx

What is the name or IP address of the database server?

Host Name: whf00bop.in.oracle.com

What is the port on the database server used to connect to the database?

Port: 1521

What database account user name do you want to use to create database connections?

Database User Name: OBDX_OBDX201QTR2

What is the database account password to use to create database connections?

Password: *****

Confirm Password: *****

Additional Connection Properties:

oracle.jdbc.DRCPConnectionClass:

5. Provide
- Database Name:** Database SID
 - Host Name:** Database hostname
 - Port:** Database port Number
 - Database user Name:** OBAPI_\${POST_FIX}

ORACLE WebLogic Server Administration Console 12c

Home Log Out Preferences Record Help

Welcome, weblogic Connected to: obdx_domain

Home > Summary of JDBC Data Sources

Messages

Connection test succeeded.

Create a New JDBC Data Source

Test Configuration Back Next Finish Cancel

Test Database Connection

Test the database availability and the connection properties you provided.

What is the full package name of JDBC driver class used to create database connections in the connection pool?
(Note that this driver class must be in the classpath of any server to which it is deployed.)

Driver Class Name: oracle.jdbc.OracleDriver

What is the URL of the database to connect to? The format of the URL varies by JDBC driver.

URL: jdbc:oracle:thin:@whf00bop.in.oracle.com:1521:obdx

What database account user name do you want to use to create database connections?

Database User Name: OBDX_OBDX201QTR2

What is the database account password to use to create database connections?
(Note: for secure password management, enter the password in the Password field instead of the Properties field below)

Change Center

View changes and restarts

No pending changes exist. Click the Release Configuration button to allow others to edit the domain.

Lock & Edit
Release Configuration

Domain Structure

- obdx_domain
 - Domain Partitions
 - Environment
 - Deployments
 - Services
 - Data Sources
 - Persistent Stores
 - Foreign JNDI Providers
 - Work Contexts
 - XML Registries
 - XML Entity Caches
 - JCOM
 - Mail Sessions

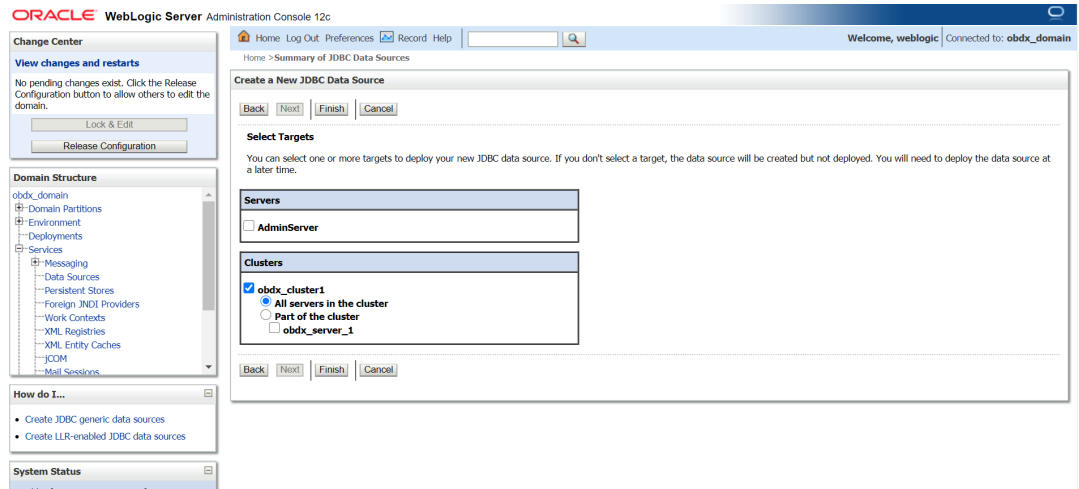
How do I...

- Create JDBC generic data sources
- Create LLR-enabled JDBC data sources

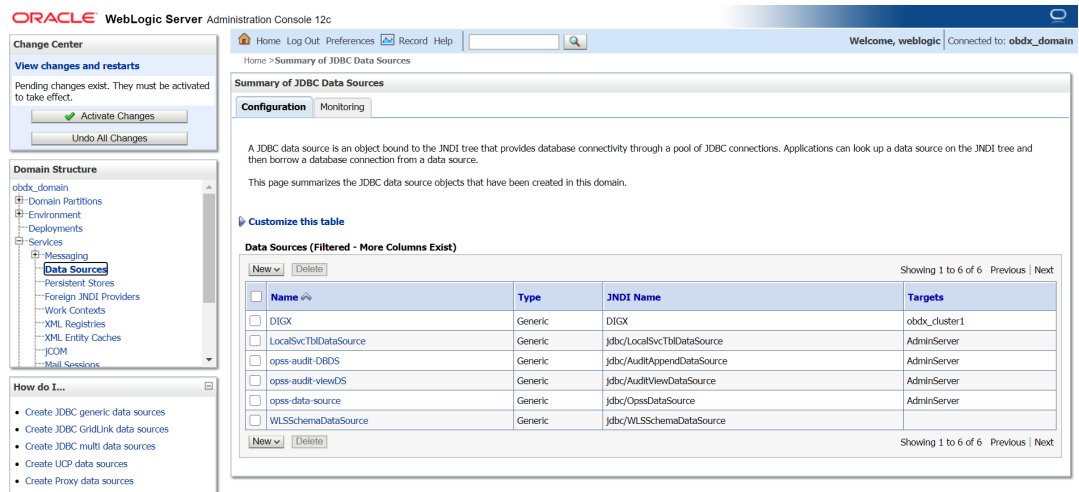
System Status

Health of Running Servers as of 12:15 PM

6. Test Configuration.

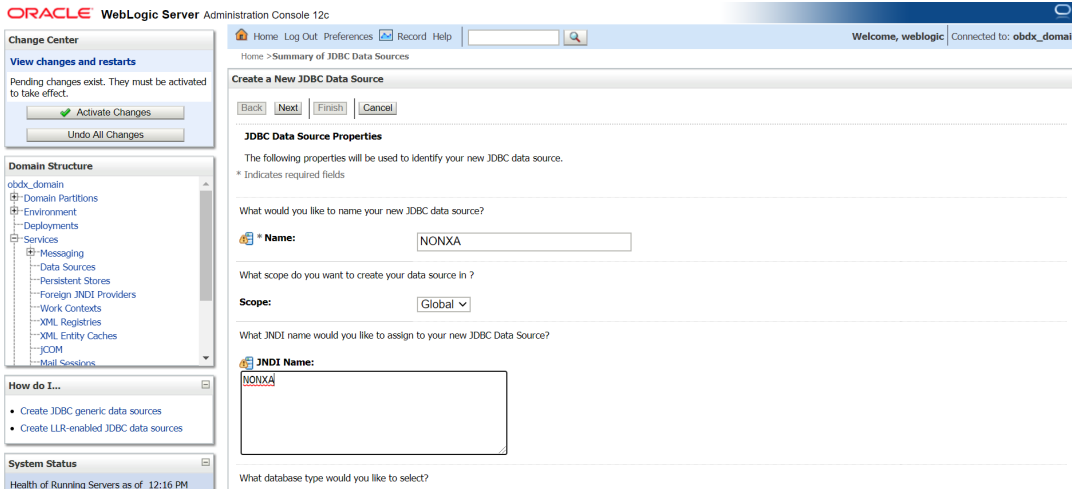


7. Target to cluster.

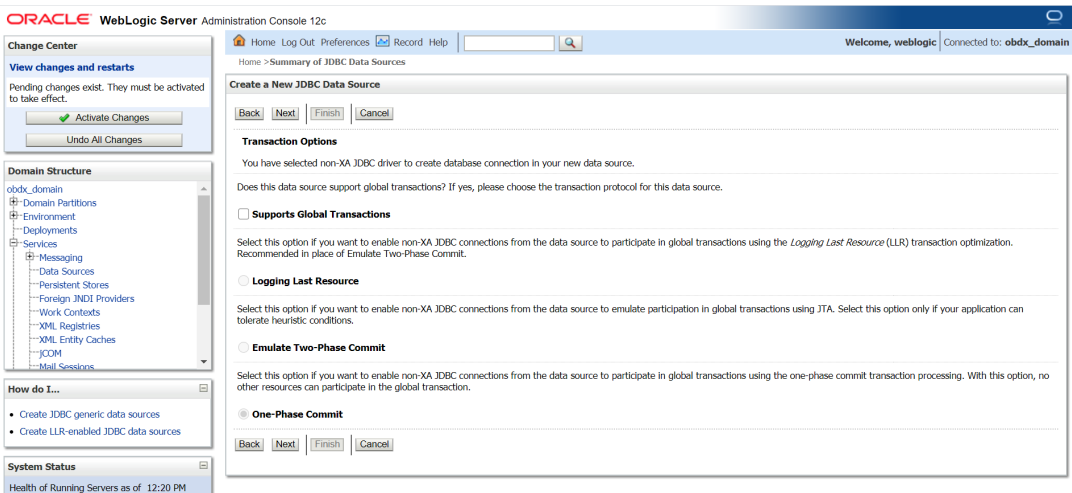
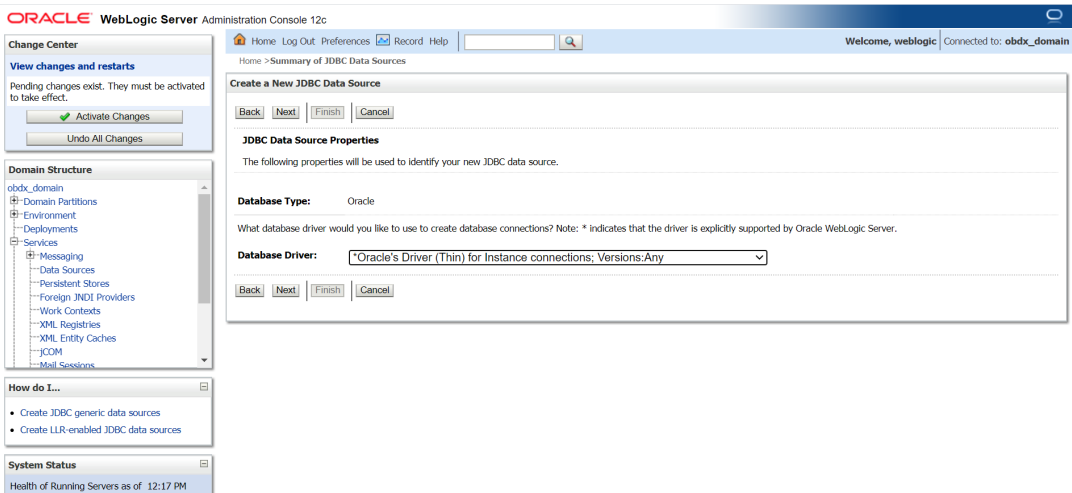


2.3 Creating NONXA Data Source

1. Navigate to Data Source → click **New** → Provide details and click **Finish**.



2. Name : NONXA
JNDI Name : NONXA



3. Click Next.

4. Provide

Database Name: Database SID

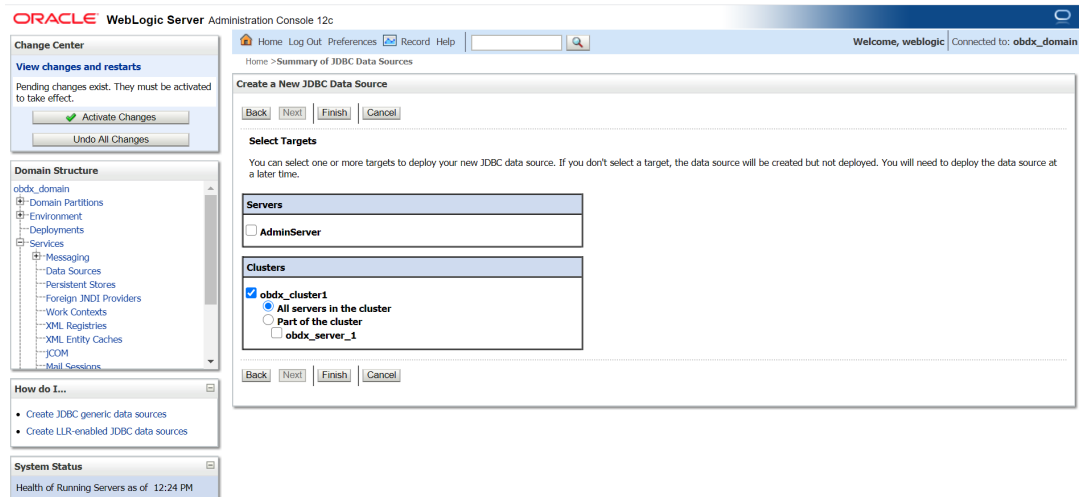
Host Name: Database hostname

Port: Database port Number

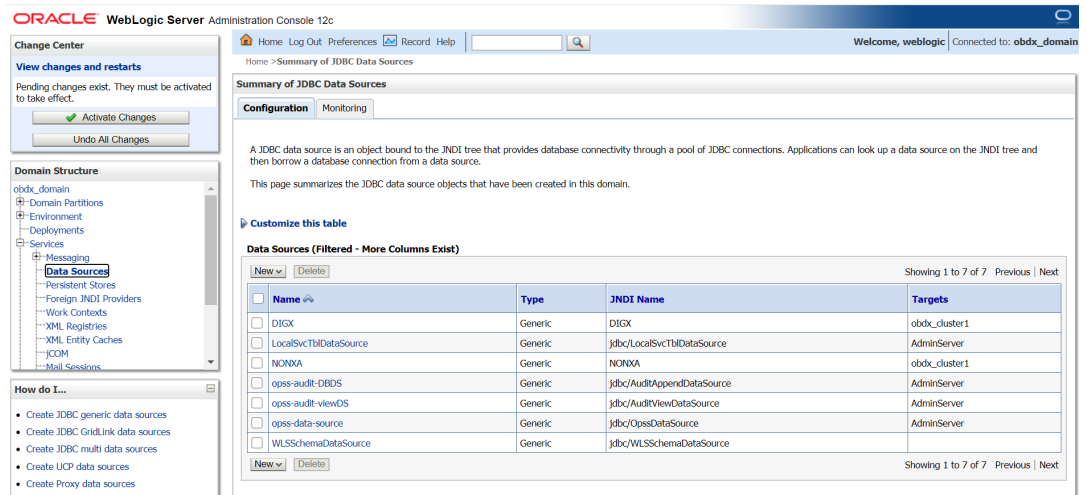
Database user Name: OBAPI_\${POST_FIX}

Password: Database user password

5. Test Configuration.

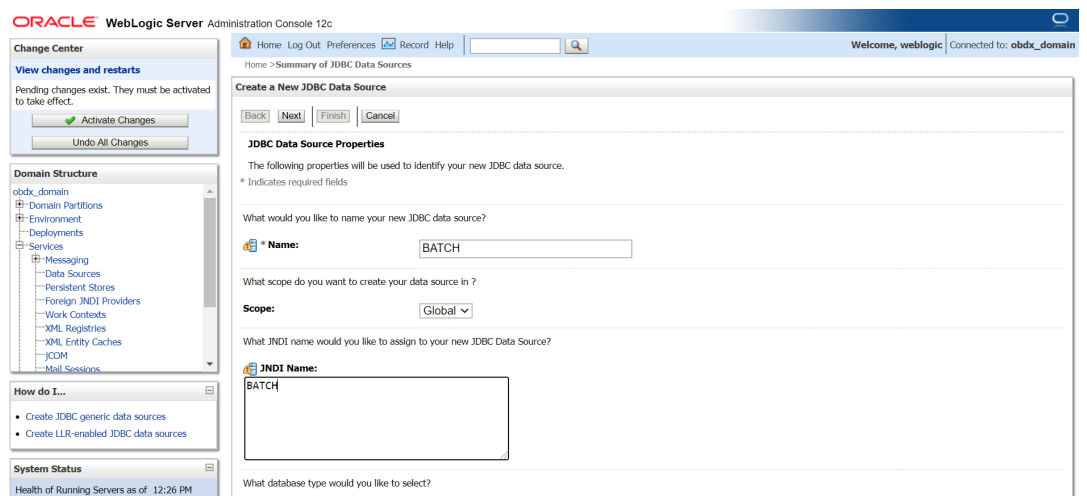


6. Select target as cluster → **Finish**.

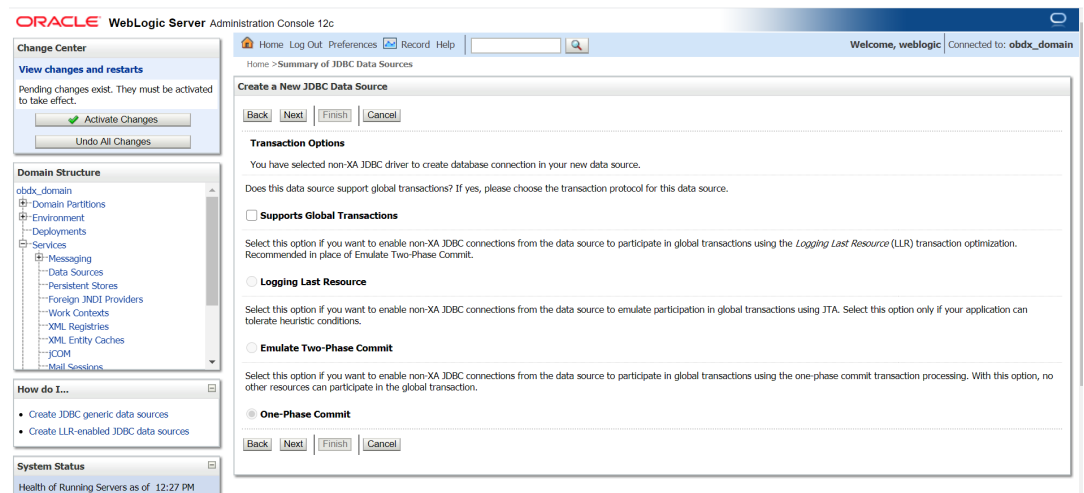
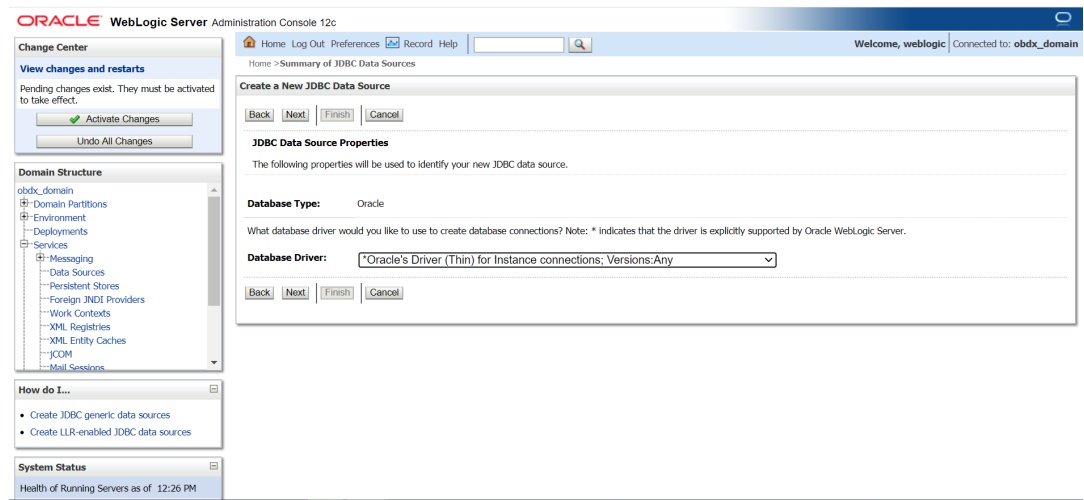


2.4 Creating BATCH Data Source

1.



- Name : BATCH**
JNDI Name : BATCH



- Click Next.**

to take effect.

Domain Structure

- obdx_domain
 - Domain Partitions
 - Environment
 - Deployments
 - Services
 - Data Sources
 - Persistent Stores
 - Foreign JNDI Providers
 - Work Contexts
 - XML Registries
 - XML Entity Caches
 - JCOM
 - Mail Sessions

How do I...

- Create JDBC generic data sources
- Create LLR-enabled JDBC data sources

System Status

Health of Running Servers as of 12:30 PM

- Failed (0)
- Critical (0)
- Overloaded (0)
- Warning (0)
- OK (1)

Connection Properties

Define Connection Properties.

What is the name of the database you would like to connect to?

Database Name:

What is the name or IP address of the database server?

Host Name:

What is the port on the database server used to connect to the database?

Port:

What database account user name do you want to use to create database connections?

Database User Name:

What is the database account password to use to create database connections?

Password:

Confirm Password:

Additional Connection Properties:

oracle.jdbc.DRCPConnectionClass:

- 4. Provide**

Database Name: Database SID

Host Name: Database hostname

Port: Database port Number

Database user Name: OBDX_\${POST_FIX}

Password: Database user password

ORACLE WebLogic Server Administration Console 12c

Home Log Out Preferences Record Help

Welcome, weblogic | connected to: obdx_domain

Home > Summary of JDBC Data Sources

Messages

Test Database Connection

Test the database availability and the connection properties you provided.

What is the full package name of JDBC driver class used to create database connections in the connection pool?
(Note that this driver class must be in the classpath of any server to which it is deployed.)

Driver Class Name:

What is the URL of the database to connect to? The format of the URL varies by JDBC driver.

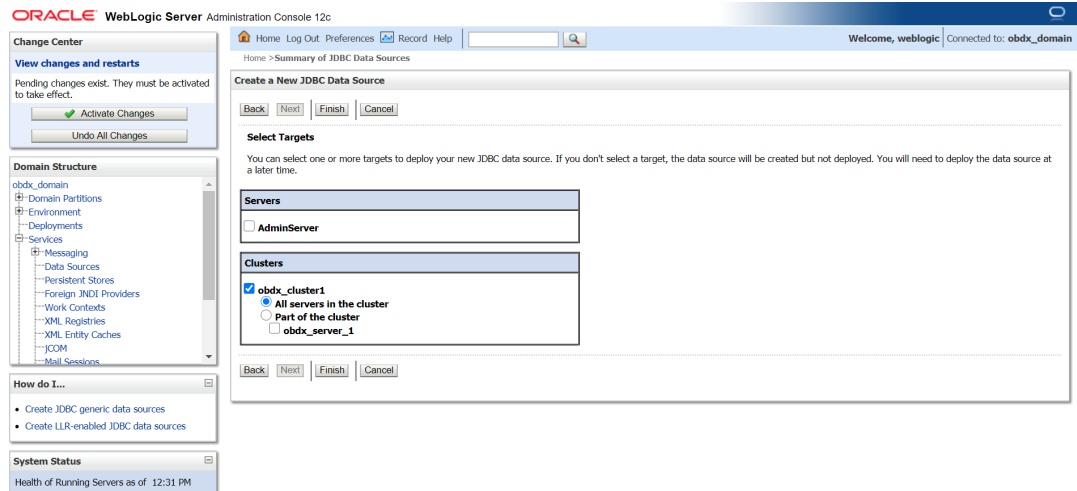
URL:

What database account user name do you want to use to create database connections?

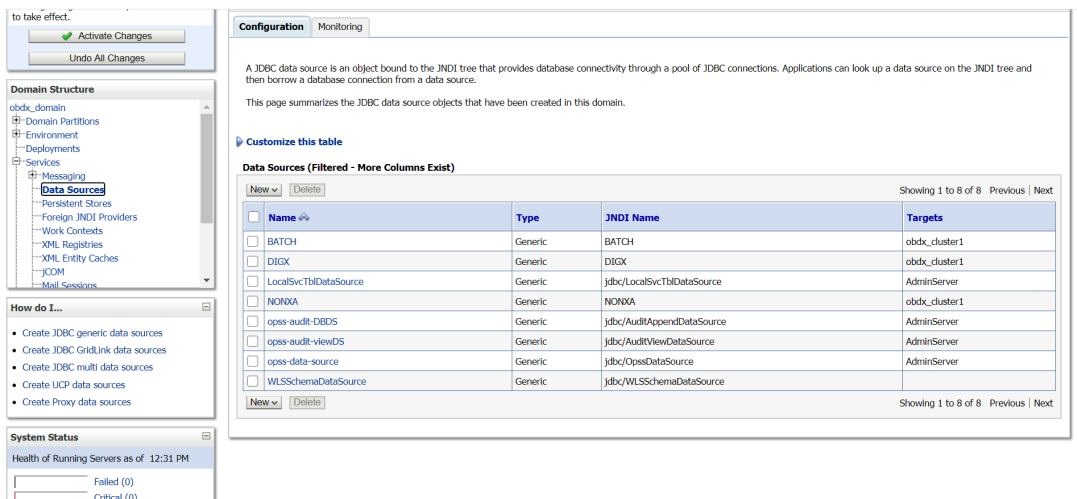
Database User Name:

What is the database account password to use to create database connections?
(Note: for secure password management, enter the password in the Password field instead of the Properties field below)

- 5. Test Configuration.**

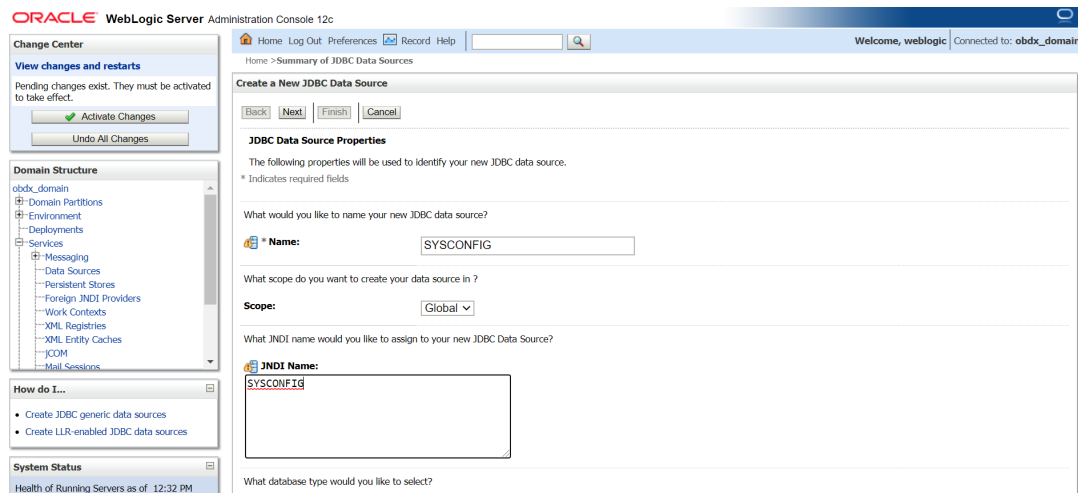


6. Target Cluster and click **Finish**.

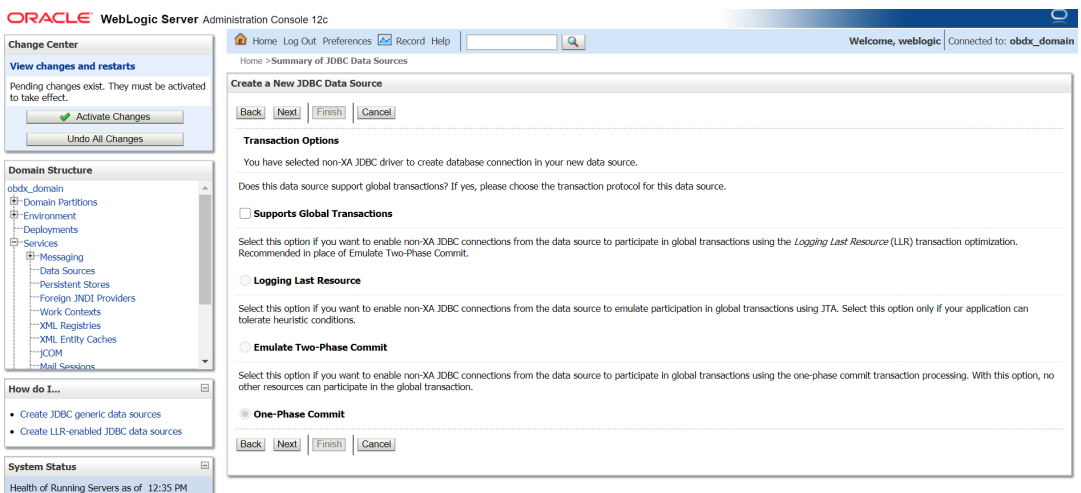
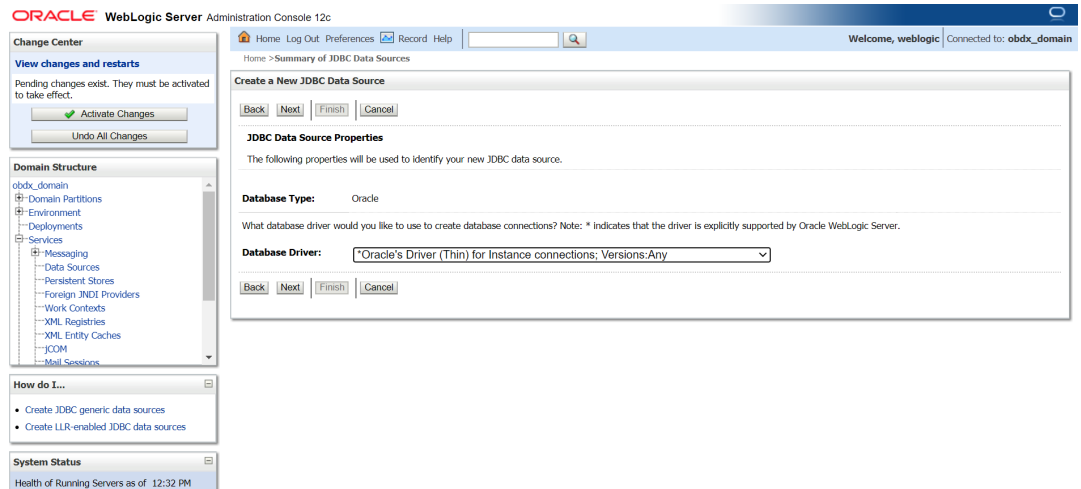


2.5 Creating SYSCONFIG Data Source

1.



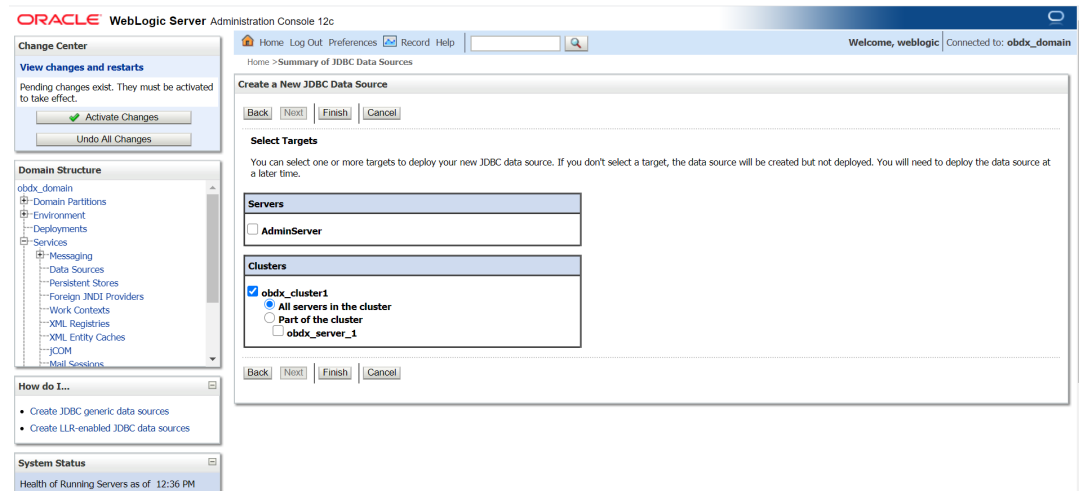
2. Name : SYSCONFIG
JNDI Name : SYSCONFIG



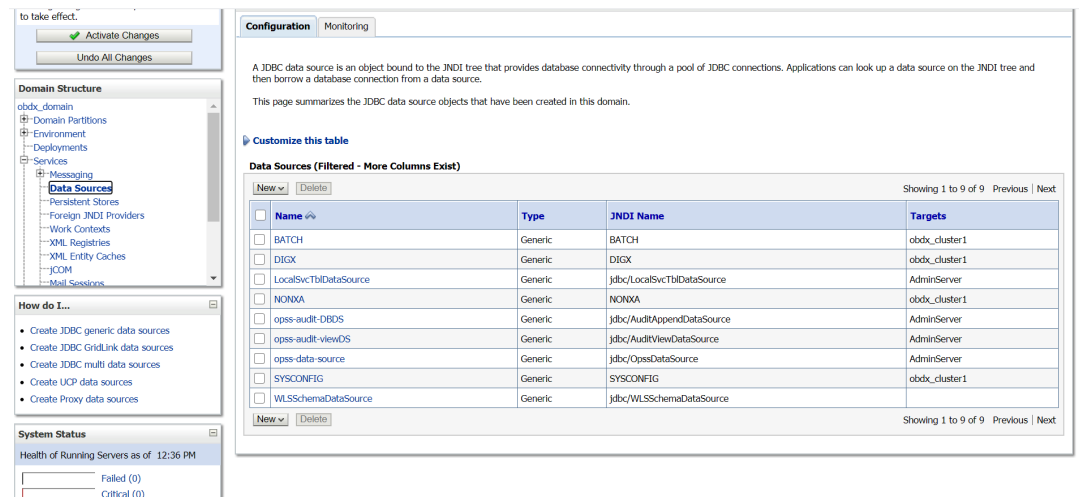
3. Click Next.

4. Provide
 - Database Name:** Database SID
 - Host Name:** Database hostname
 - Port:** Database port Number
 - Database user Name:** OBDX_\${POST_FIX}
 - Password:** Database user password

5. Test Configuration.

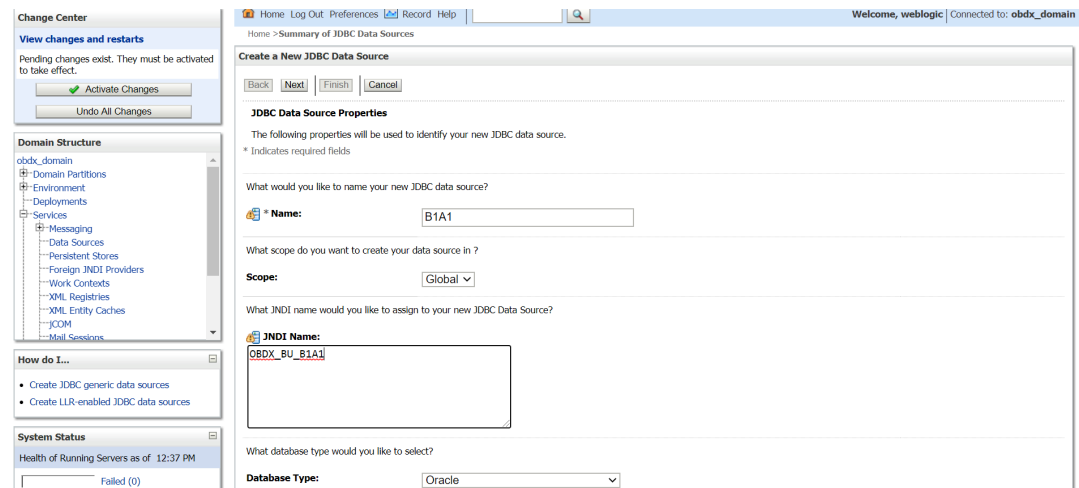


6. Select target as cluster and click **Finish**.

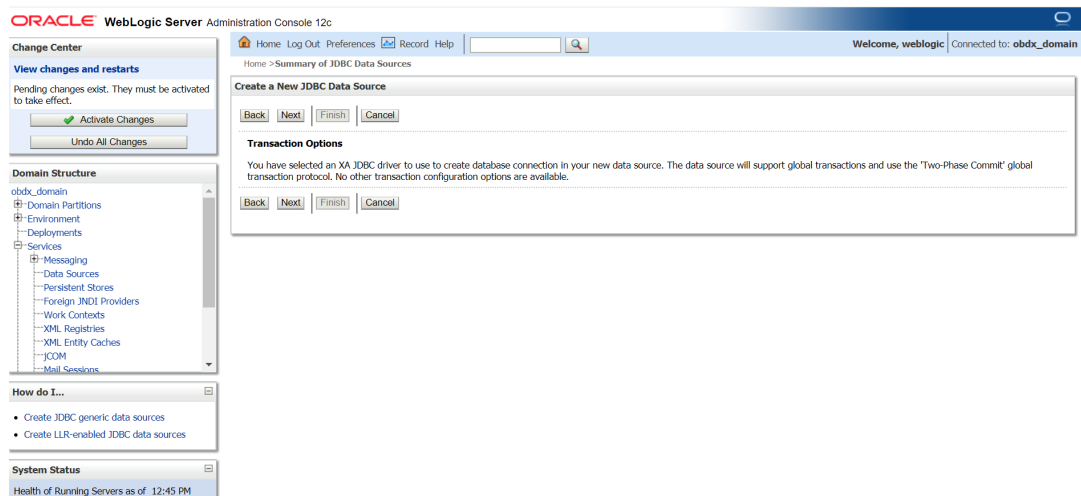
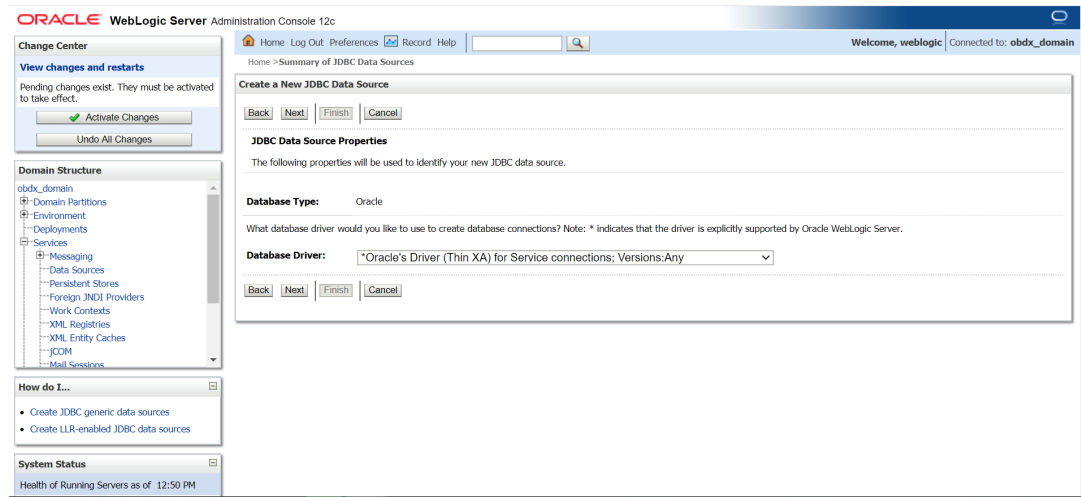


2.6 Creating B1A1 Data Source

1.



- Name:** B1A1
JNDI Name : OBDX_BU_B1A1



- Click Next.**

to take effect.

Domain Structure

obdx_domain

- Domain Partitions
- Environment
- Deployments
- Services
 - Messaging
 - Data Sources
 - Persistent Stores
 - Foreign JNDI Providers
 - Work Contexts
 - XML Registries
 - XML Entity Caches
 - JCOM
 - Mail Sessions

How do I...?

- Create JDBC generic data sources
- Create LLR-enabled JDBC data sources

System Status

Health of Running Servers as of 12:46 PM

- Failed (0)
- Critical (0)
- Overloaded (0)
- Warning (0)
- OK (1)

Connection Properties

Define Connection Properties.

What is the name of the database you would like to connect to?

Database Name: ora19c.in.oracle.com

What is the name or IP address of the database server?

Host Name: whf00jml.in.oracle.com

What is the port on the database server used to connect to the database?

Port: 1522

What database account user name do you want to use to create database connections?

Database User Name: B1A1_201DEVQTR2

What is the database account password to use to create database connections?

Password: *****

Confirm Password: *****

Additional Connection Properties:

oracle.jdbc.DRCPConnectionClass:

4. Provide

Database Name: Database SID (\$EHMS_DATABASE_SID)

Host Name: Database hostname (\$EHMS_DATABASE_HOSTNAME)

Port: Database port Number (\$EHMS_DATABASE_PORT)

Database user Name: \${ EHMS_SCHEMA_NAME }

Password: Database user \${ EHMS_SCHEMA_NAME } password

ORACLE WebLogic Server Administration Console 12c

Home Log Out Preferences Record Help

Welcome, weblogic Connected to: obdx_domain

Home -> Summary of JDBC Data Sources

Messages

Connection test succeeded.

Create a New JDBC Data Source

Test Database Connection

Test the database availability and the connection properties you provided.

What is the full package name of JDBC driver class used to create database connections in the connection pool?
(Note that this driver class must be in the classpath of any server to which it is deployed.)

Driver Class Name: oracle.jdbc.xa.client.OracleXADataSource

What is the URL of the database to connect to? The format of the URL varies by JDBC driver.

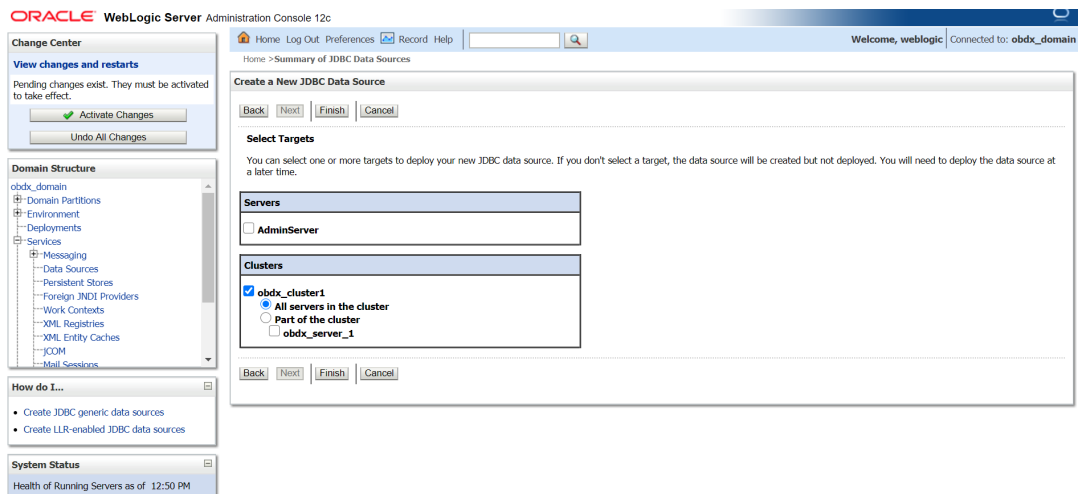
URL: jdbc:oracle:thin:@/whf00jml.in.oracle.com:1522/ora19c.in.ora

What database account user name do you want to use to create database connections?

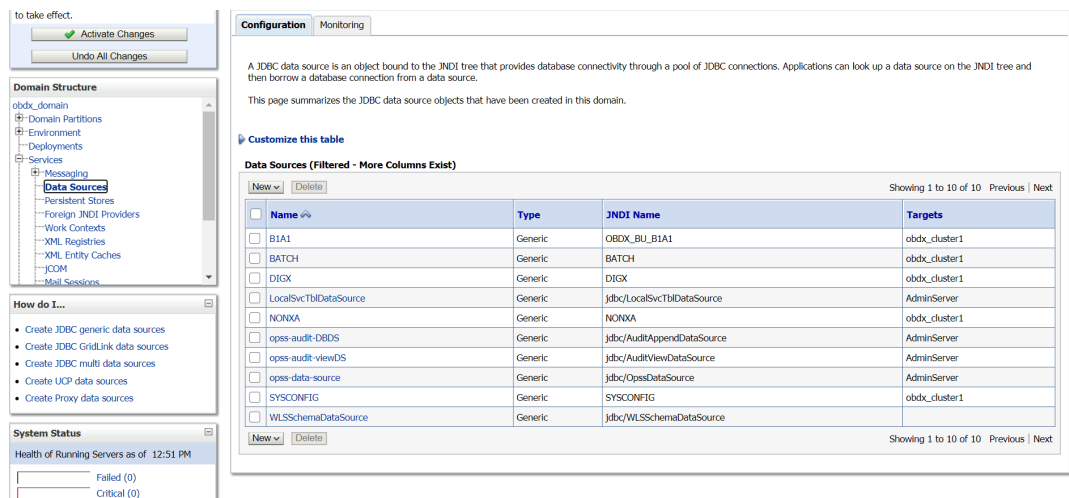
Database User Name: B1A1_201DEVQTR2

What is the database account password to use to create database connections?
(Note: for secure password management, enter the password in the Password field instead of the Properties field below)

5. Test Configuration.

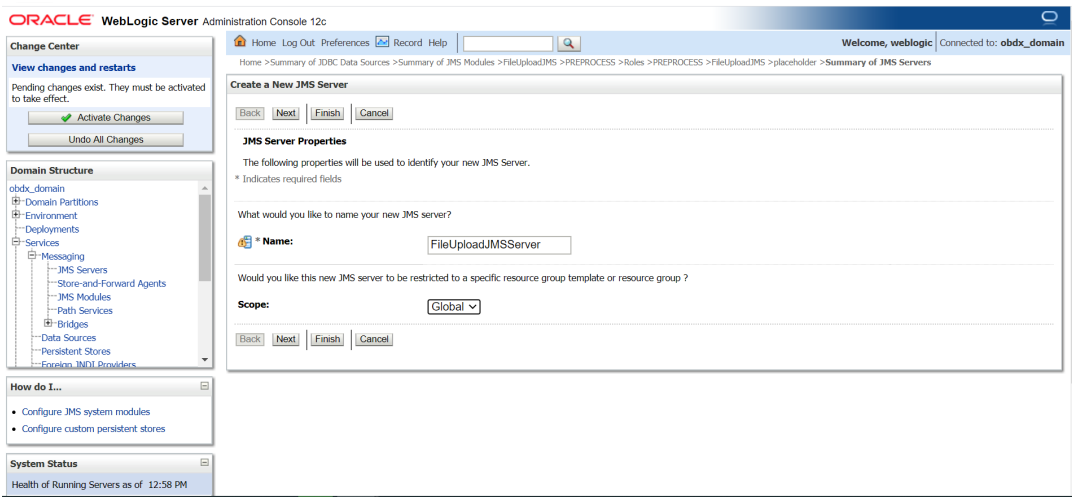
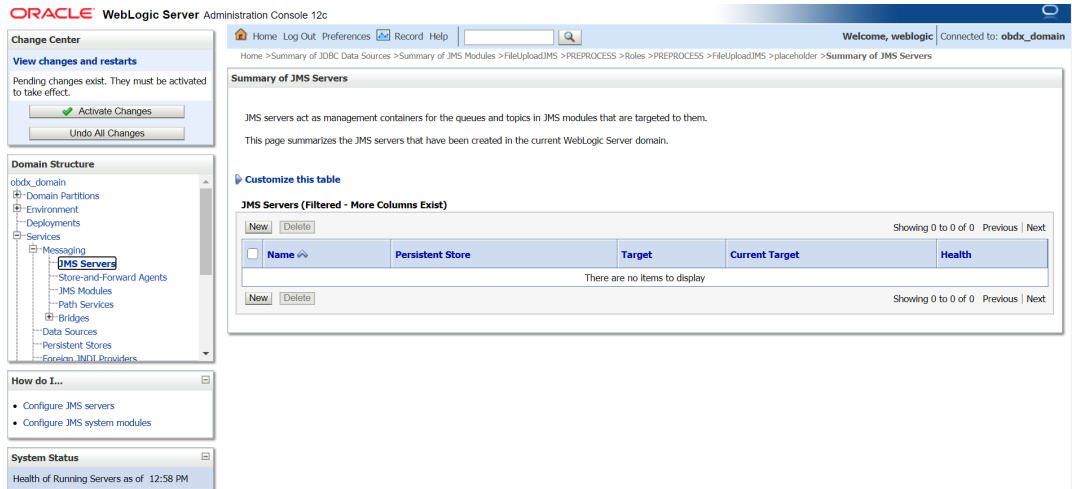


6. Set target as cluster and click **Finish**.

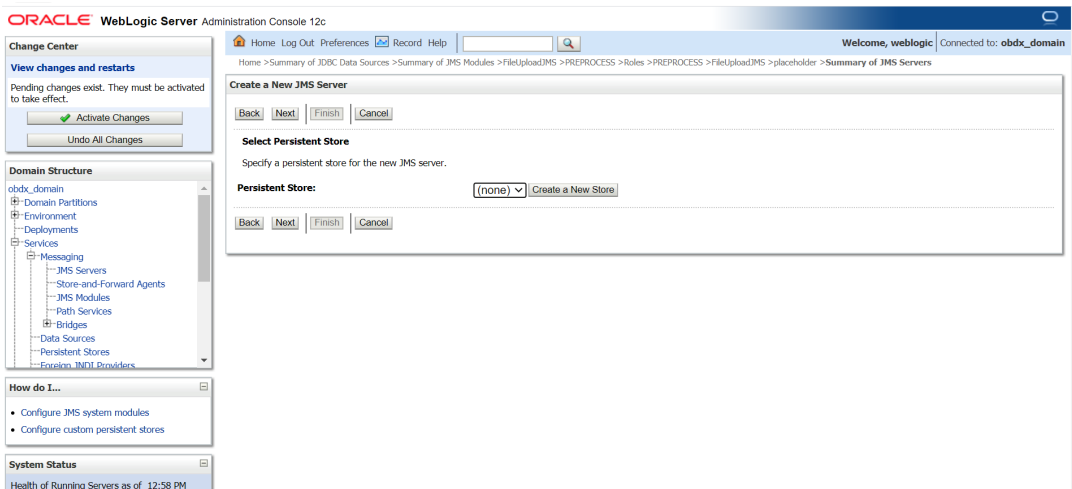


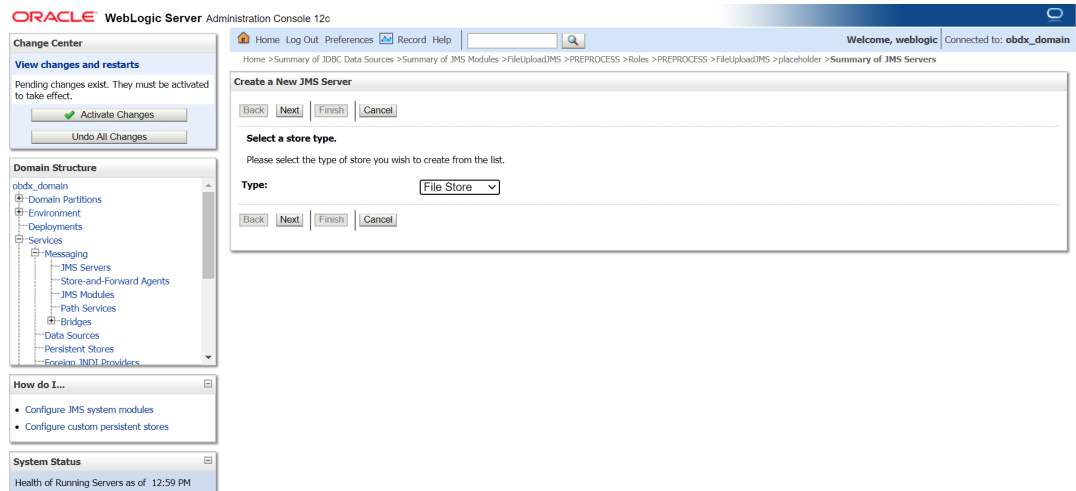
2.7 Create JMS Server and JMS Module

- Creating FileUploadJMS JSM Module
 - Creating WLS_JMS_FILEUPLOAD_PS FileStore
 - Creating FileUploadJMSServer JMS Server
- 1.

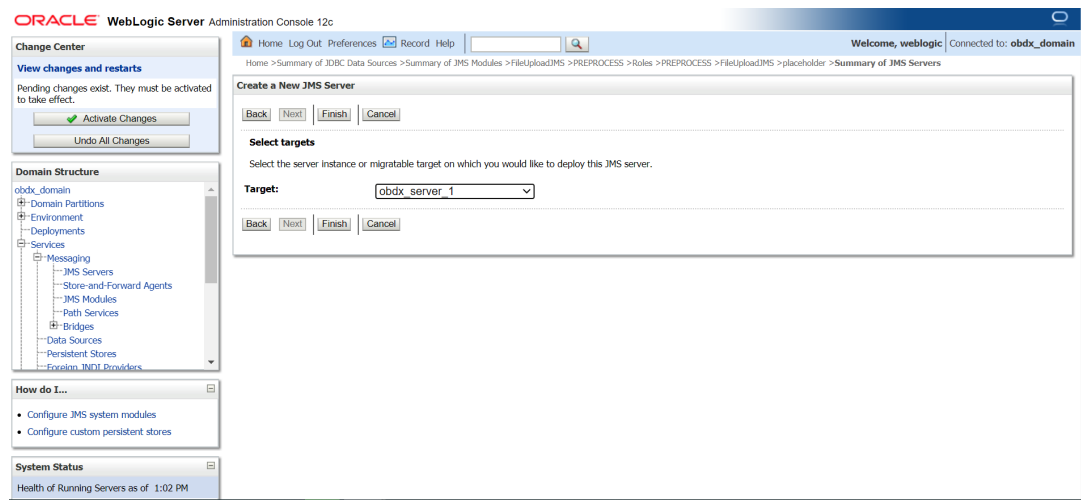


2. Click on JMS Servers → Name – FileUploadJMSServer → Click Next.

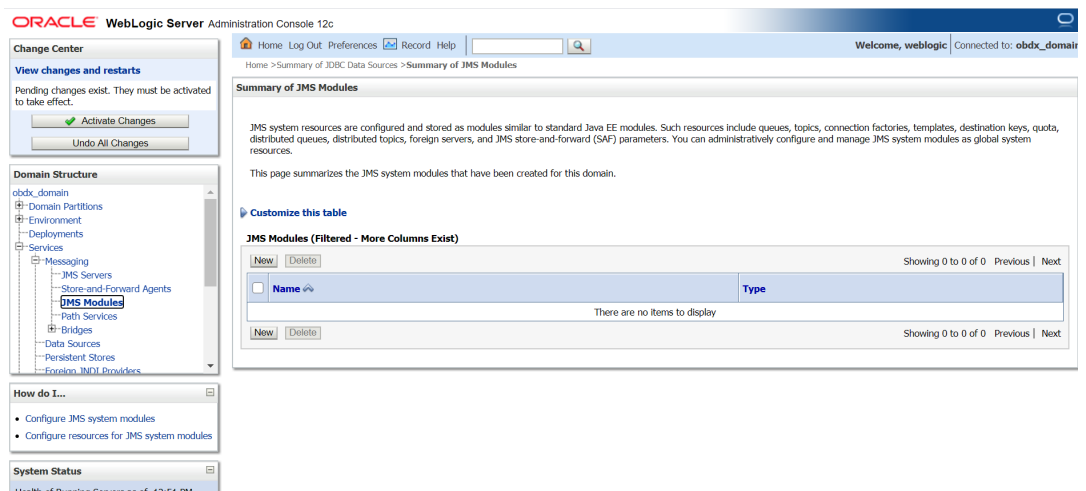




3. Select Type as File Store and click Next.



4. Select target as managed server and click Finish.



5. Left hand side click on JMS Module → click New.

to take effect.

Activate Changes
Undo All Changes

Domain Structure

obdx_domain

- Domain Partitions
- Environment
- Deployments
- Services
 - Messaging
 - JMS Servers
 - Store-and-Forward Agents
 - JMS Modules
 - Path Services
 - Bridges
 - Data Sources
 - Persistent Stores
 - Foreign JNDI Providers

How do I...

- Configure JMS system modules
- Configure JMS servers

System Status

Health of Running Servers as of 12:52 PM

- Failed (0)
- Critical (0)
- Overloaded (0)
- Warning (0)
- OK (1)

Back Next Finish Cancel

The following properties will be used to identify your new module.

JMS system resources are configured and stored as modules similar to standard Java EE modules. Such resources include queues, topics, connection factories, templates, destination keys, quota, distributed queues, distributed topics, foreign servers, and JMS store-and-forward (SAF) parameters. You can administratively configure and manage JMS system modules as global system resources.

* Indicates required fields

What would you like to name your System Module?

Name: FileUploadJMS

Would you like this new JMS System Module to be restricted to a specific resource group template or resource group?

Scope: Global

What would you like to name the descriptor file name? If you do not provide a name, a default will be assigned.

Descriptor File Name: jms/fileuploadjms-jms.xml

Where would you like to place the descriptor for this System Module, relative to the jms configuration sub-directory of your domain?

Location In Domain:

Back Next Finish Cancel

- Name :** FileUploadJMS
Scope: Global
Descriptor File Name: jms/fileuploadjms-jms.xml
- Click **Next**.

ORACLE WebLogic Server Administration Console 12c

Home Log Out Preferences Record Help

Welcome, weblogic Connected to: obdx_domain

Home > Summary of JDBC Data Sources > Summary of JMS Modules

Create JMS System Module

Back Next Finish Cancel

The following properties will be used to target your new JMS system module.

Use this page to select the server or cluster on which you would like to deploy this JMS system module. You can reconfigure targets later if you wish.

Targets :

Servers

- AdminServer

Clusters

- obdx_cluster1
 - All servers in the cluster
 - Part of the cluster
 - obdx_server_1

Back Next Finish Cancel

Change Center

View changes and restarts

Pending changes exist. They must be activated to take effect.

Activate Changes
Undo All Changes

Domain Structure

obdx_domain

- Domain Partitions
- Environment
- Deployments
- Services
 - Messaging
 - JMS Servers
 - Store-and-Forward Agents
 - JMS Modules
 - Path Services
 - Bridges
 - Data Sources
 - Persistent Stores
 - Foreign JNDI Providers

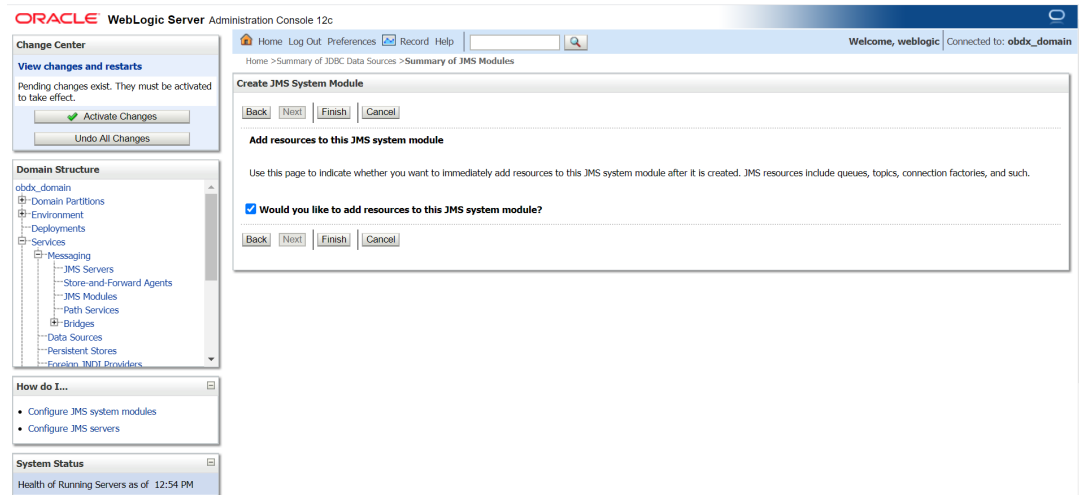
How do I...

- Configure JMS system modules
- Configure JMS servers

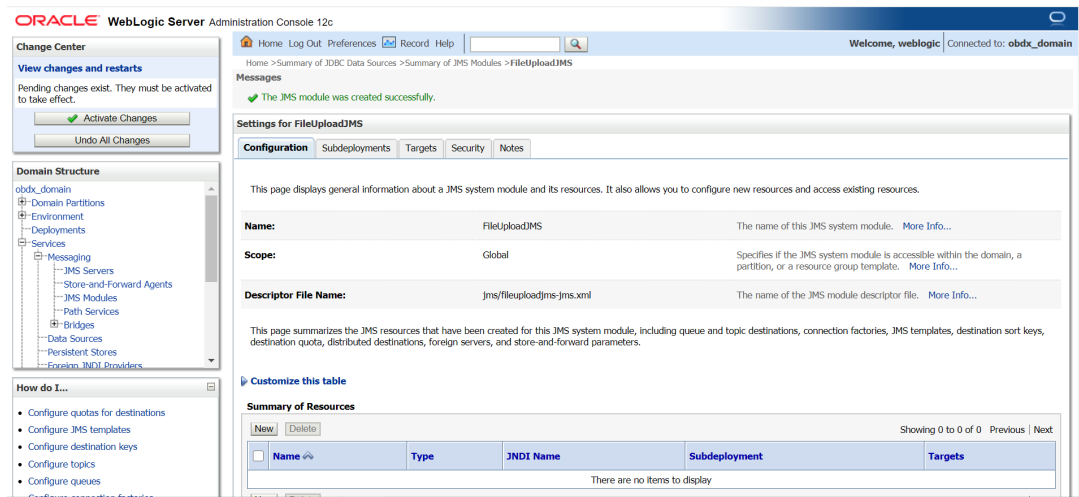
System Status

Health of Running Servers as of 12:53 PM

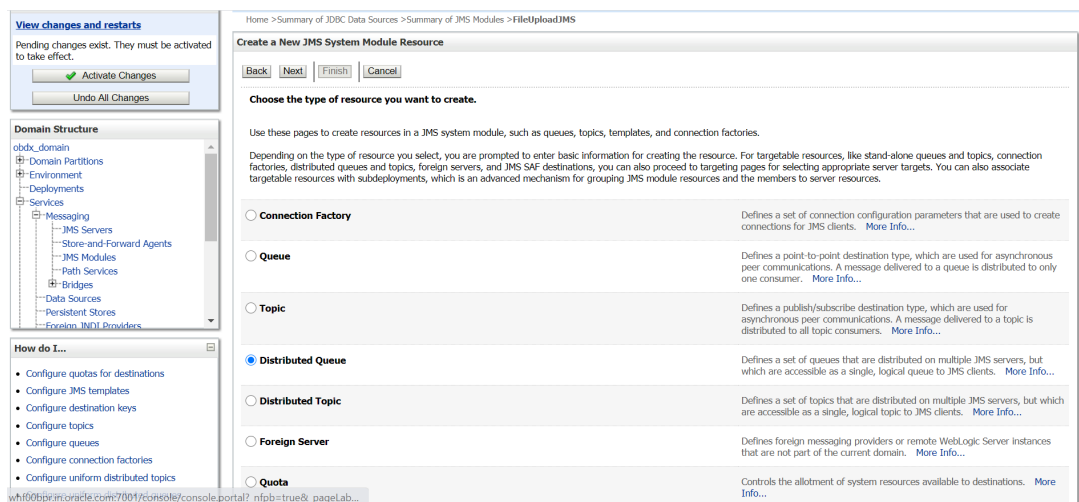
- Set target as cluster → click **Next**.



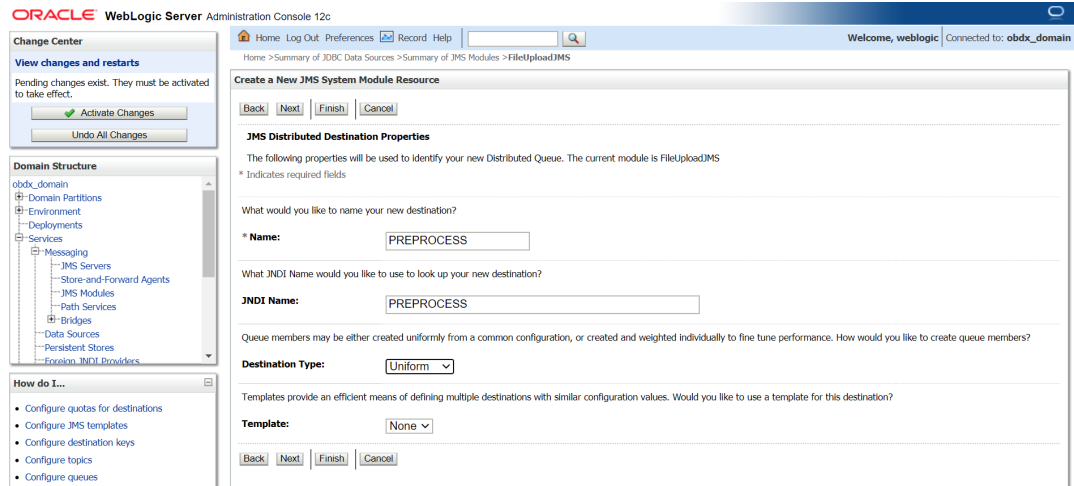
9. Select **Would you like to add resources to this JMS system module** and click **Finish**.



10. Select **New**.



11. Select **Distributed Queue** and click **Next**.



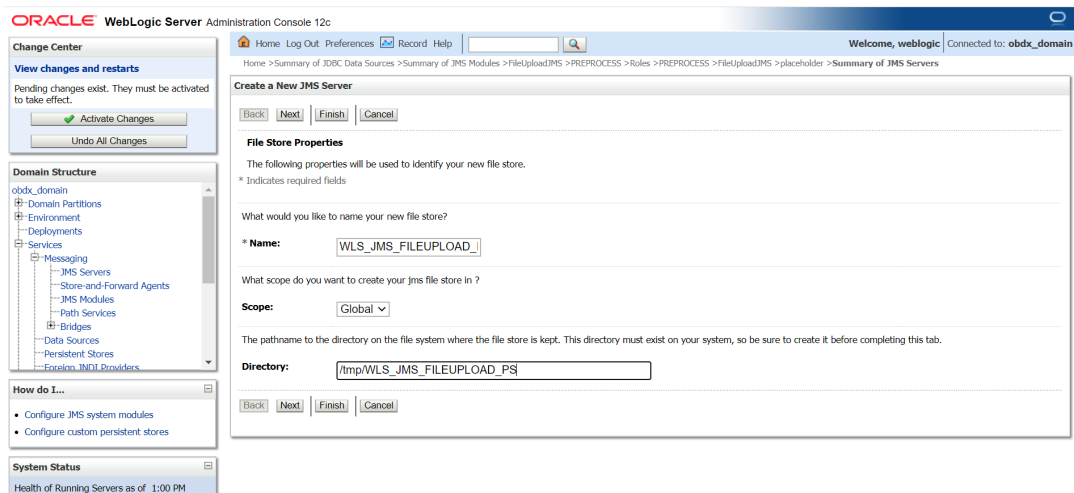
12. Provide

Name: PREPROCESS

JNDI Name: PREPROCESS

Destination Type: Uniform

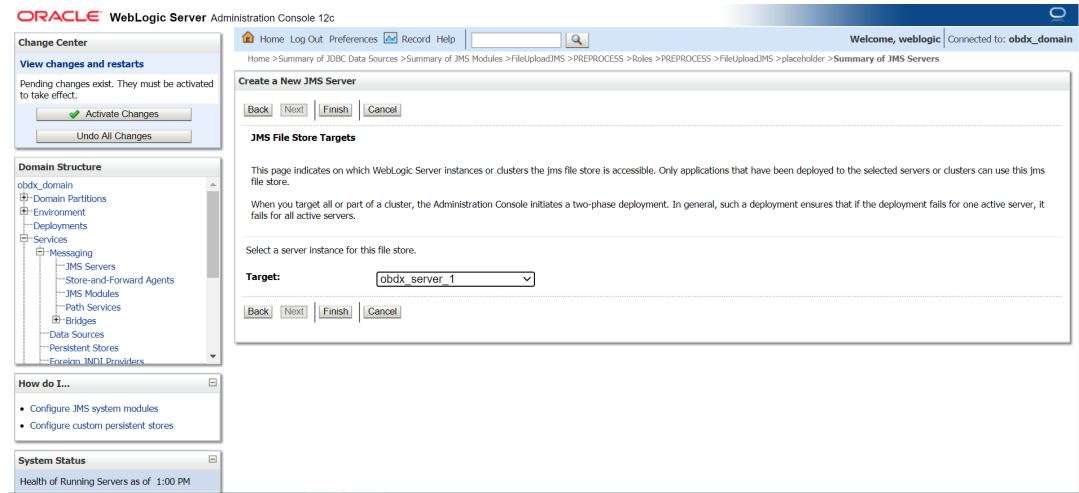
Template: None



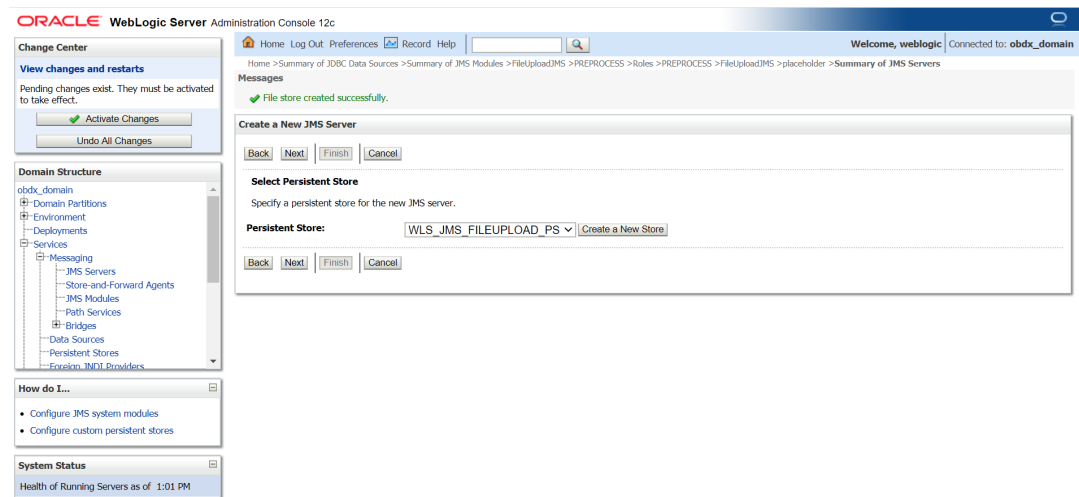
13. **Name :** WLS_JMS_FILEUPLOAD_PS

Scope : Global

Directory : /tmp/WLS_JMS_FILEUPLOAD_PS

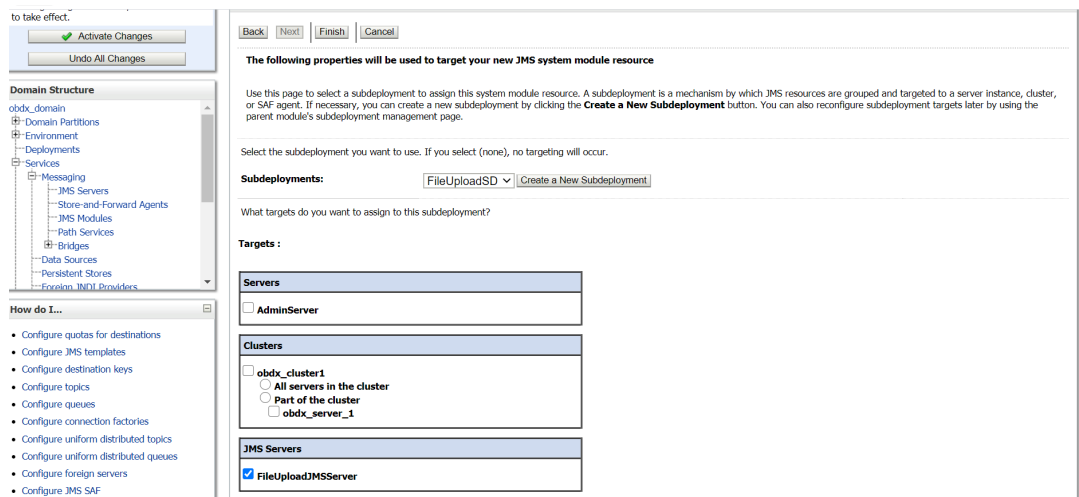


14. Select target as managed server.



15. Select WLS_JMS_FILEUPLOAD_PS and click Next.

16. Select Create a New Subdeployment and create FileUploadSD.



17. Select **FileUploadJMSServer** and click **Finish**.

The screenshot shows the 'Settings for FileUploadJMS' configuration page. On the left, there is a 'Domain Structure' tree with 'FileUploadJMS' selected under 'Services'. Below it is a 'How do I...' section with various configuration links. The main content area shows a success message: 'The JMS distributed queue was created successfully.' Below this, there are tabs for 'Configuration', 'Subdeployments', 'Targets', 'Security', and 'Notes'. The 'Configuration' tab is active, displaying general information about the JMS system module. A table titled 'Summary of Resources' is shown, listing one resource: 'PREPROCESS' of type 'Uniform Distributed Queue' with JNDI Name 'PREPROCESS', Subdeployment 'FileUploadJMS', and Target 'FileUploadJMSServer'.

18. Similarly Go into **FileuploadJMS** module and click **Next**.

The screenshot shows the 'Create a New JMS System Module Resource' page in the Oracle WebLogic Server Administration Console. The page title is 'Create a New JMS System Module Resource'. There are navigation buttons: 'Back', 'Next', 'Finish', and 'Cancel'. Below the navigation buttons, there is a section titled 'Choose the type of resource you want to create.' with a sub-header: 'Use these pages to create resources in a JMS system module, such as queues, topics, templates, and connection factories.' Below this, there are five radio button options: 'Connection Factory' (selected), 'Queue', 'Topic', 'Distributed Queue', and 'Distributed Topic'. Each option has a brief description and a 'More Info...' link. The 'Connection Factory' option is selected.

19. Select **Connection factory** → Click **Next**.

to take effect.

Domain Structure

- obdx_domain
 - Domain Partitions
 - Environment
 - Deployments
 - Services
 - JMS Servers
 - Store-and-Forward Agents
 - JMS Modules
 - Path Services
 - Bridges
 - Data Sources
 - Persistent Stores
 - External JNDI Providers

How do I...

- Configure quotas for destinations
- Configure JMS templates
- Configure destination keys
- Configure topics
- Configure queues
- Configure connection factories
- Configure uniform distributed topics
- Configure uniform distributed queues
- Configure foreign servers
- Configure JMS SAF

Connection Factory Properties

The following properties will be used to identify your new connection factory. The current module is FileUploadJMS.
* Indicates required fields

What would you like to name your new connection factory?

Name:

What JNDI Name would you like to use to look up your new connection factory?

JNDI Name:

The Connection Factory Subscription Sharing Policy Subscribers can be used to control which subscribers can access new subscriptions. Should subscriptions created using this factory be sharable?

Subscription Sharing Policy:

The Client ID Policy indicates whether more than one JMS connection can use the same Client ID. Oracle recommends setting the Client ID policy to Unrestricted if sharing durable subscribers. Subscriptions created with different Client ID policies are always treated as independent subscriptions. What Client ID Policy would you like to use?

Client ID Policy:

A connection factory can limit the number of messages that can be queued for an asynchronous session. Should this connection factory impose a limit?

Maximum Messages per Session:

Should this connection factory create sessions that are JTA aware, and create XA queues and XA topics?

XA Connection Factory Enabled

20. Provide

Name : OCF

JNDI Name : OCF

Subscription Sharing Policy : Exclusive

Client ID Policy : Restricted

ORACLE WebLogic Server Administration Console 12c

Home Log Out Preferences Record Help Welcome, weblogic Connected to: obdx_domain

Home > Summary of JMS Modules > FileUploadJMS > PREPROCESS > Roles > PREPROCESS > FileUploadJMS > placeholder > Summary of JMS Servers > Summary of JMS Modules > FileUploadJMS

Change Center

View changes and restarts

Pending changes exist. They must be activated to take effect.

Domain Structure

- obdx_domain
 - Domain Partitions
 - Environment
 - Deployments
 - Services
 - JMS Servers
 - Store-and-Forward Agents
 - JMS Modules
 - Path Services
 - Bridges
 - Data Sources
 - Persistent Stores
 - Foreign JNDI Providers

How do I...

- Configure quotas for destinations
- Configure JMS templates
- Configure destination keys
- Configure topics
- Configure queues

Create a New JMS System Module Resource

The following properties will be used to target your new JMS system module resource

Use this page to view and accept the default targets where this JMS resource will be targeted. The default targets are based on the parent JMS system module targets. If you do not want to accept the default targets, then click **Advanced Targeting** to use the subdeployment mechanism for targeting this resource.

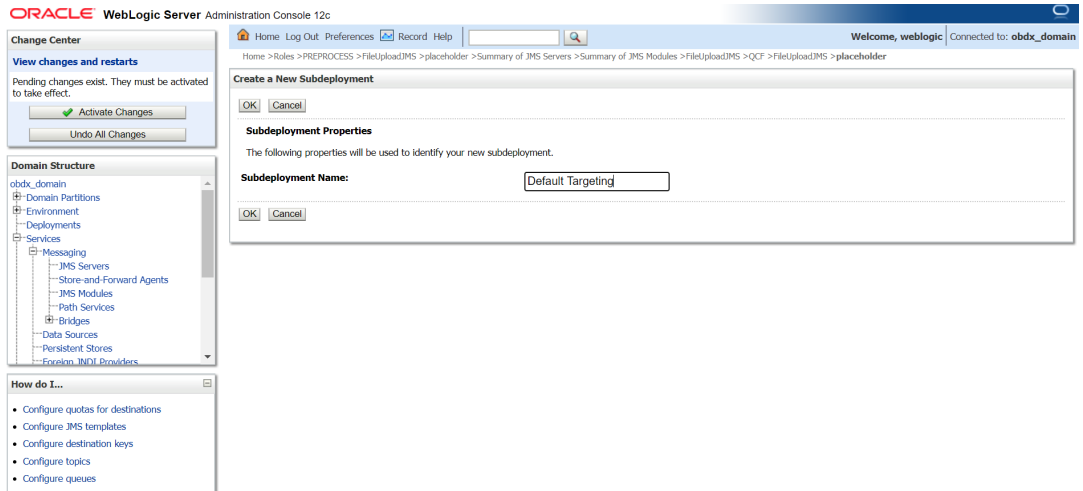
The following JMS module targets will be used as the default targets for your new JMS system module resource. If the module's targets are changed, this resource will also be retargeted appropriately.

Targets :

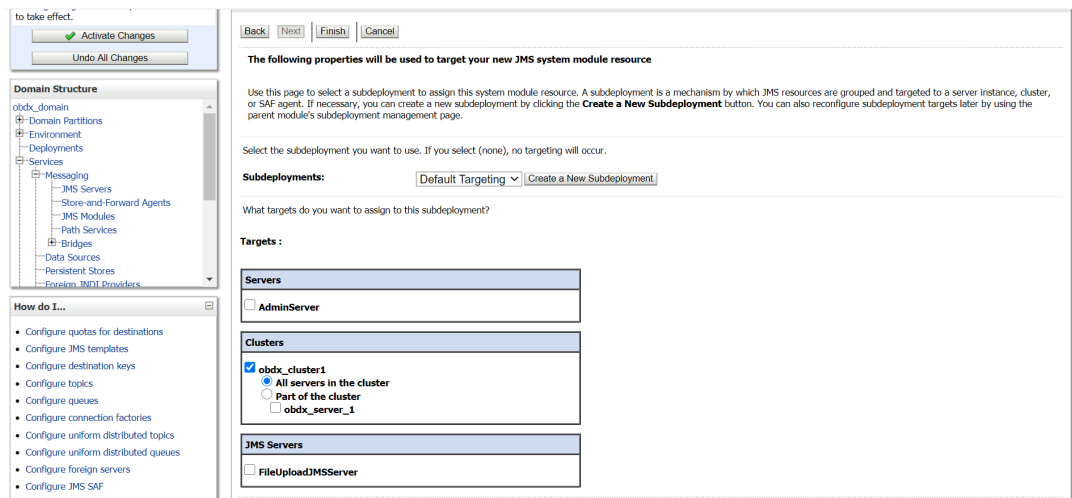
Clusters

- obdx_cluster1
 - All servers in the cluster
 - Part of the cluster
 - obdx_server_1

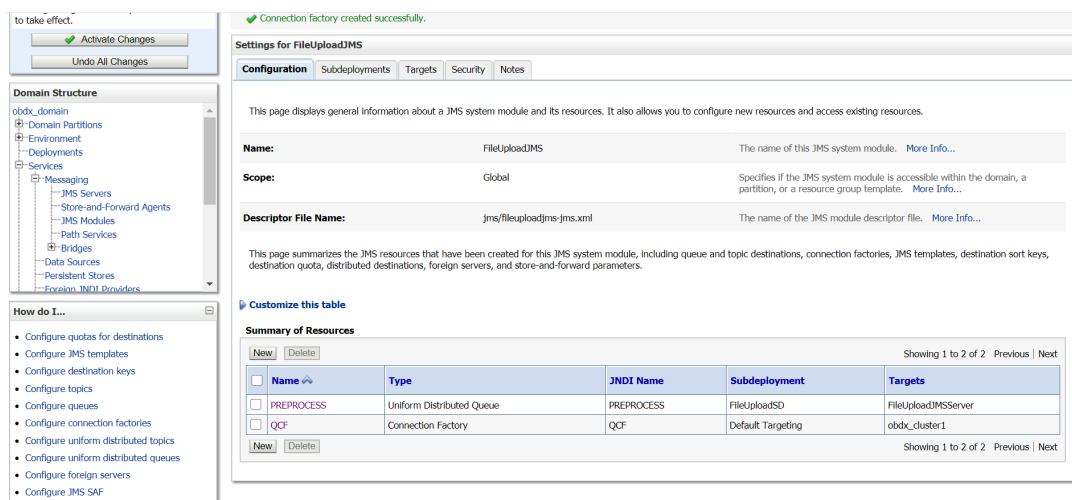
21. Click on Advanced targeting.



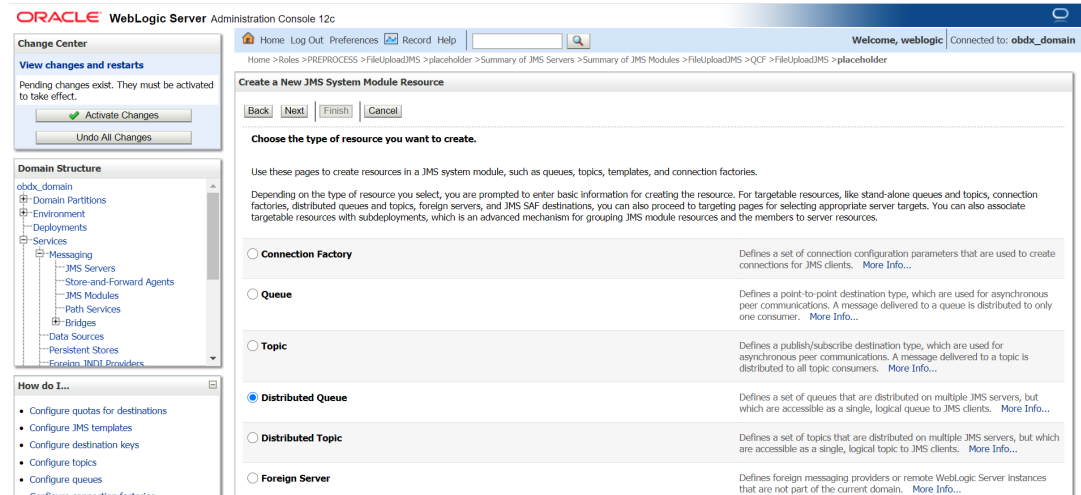
22. Provide Subdeployment Name as Default Targeting.



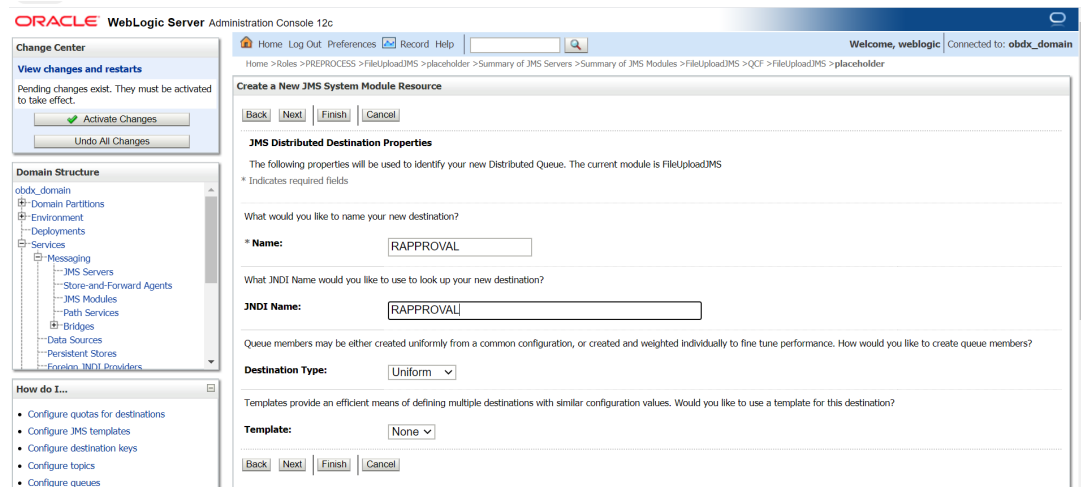
23. Select cluster and click Finish.



24. Go to FileUpload JMS and click New.



25. Select Distributed Queue.



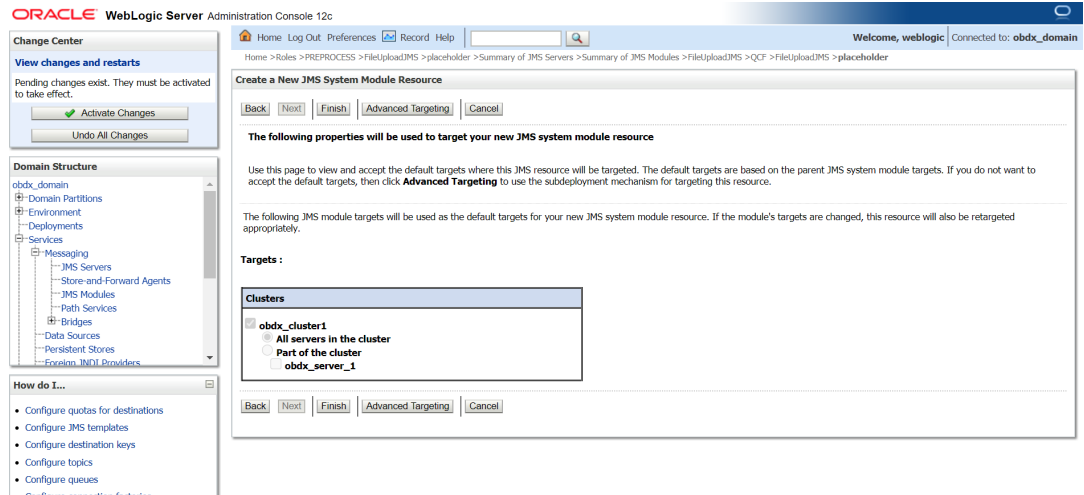
26. Provide

Name : RAPPROVAL

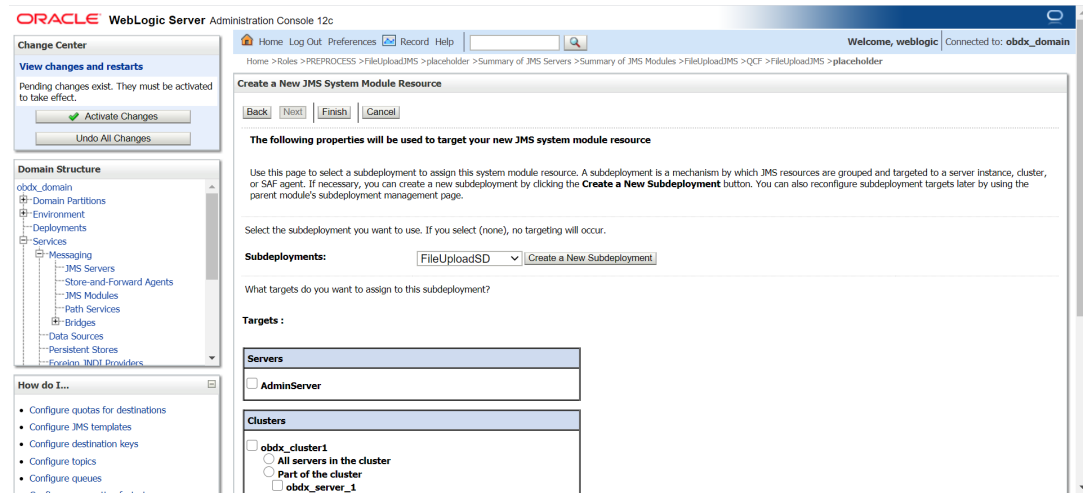
JNDI Name : RAPPROVAL

Destination Type: Uniform

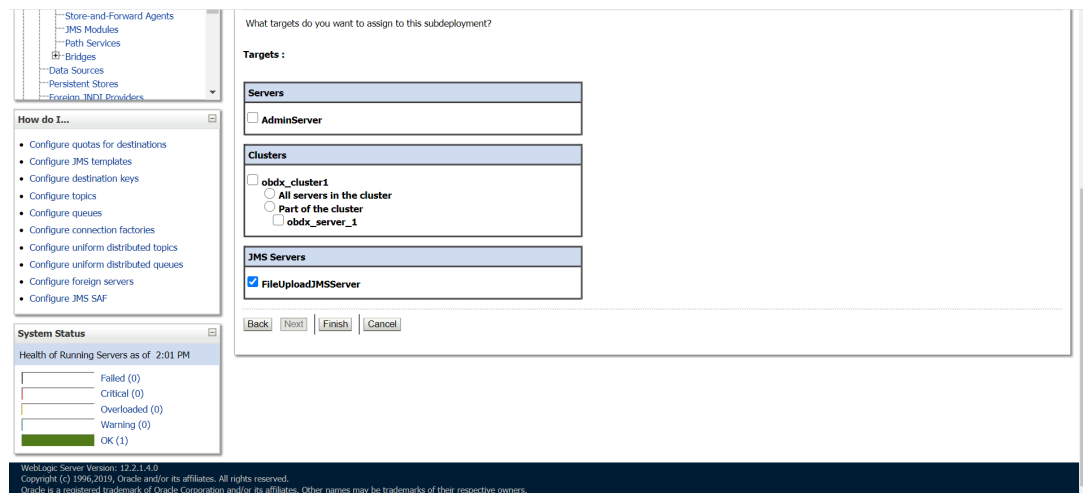
Template : None



27. Select Advance targeting.



28. Select Subdeployment: FileUploadSD.



29. Select FileUploadJMSServer and click Finish.

This page displays general information about a JMS system module and its resources. It also allows you to configure new resources and access existing resources.

Name: FileUploadJMS The name of this JMS system module. [More Info...](#)

Scope: Global Specifies if the JMS system module is accessible within the domain, a partition, or a resource group template. [More Info...](#)

Descriptor File Name: jms/fileuploadjms-jms.xml The name of the JMS module descriptor file. [More Info...](#)

This page summarizes the JMS resources that have been created for this JMS system module, including queue and topic destinations, connection factories, JMS templates, destination sort keys, destination quota, distributed destinations, foreign servers, and store-and-forward parameters.

Customize this table

Summary of Resources

Name	Type	JNDI Name	Subdeployment	Targets
PREPROCESS	Uniform Distributed Queue	PREPROCESS	FileUploadSD	FileUploadJMServer
QCF	Connection Factory	QCF	Default Targeting	obdx_cluster1
RAPPROVAL	Uniform Distributed Queue	RAPPROVAL	FileUploadSD	FileUploadJMServer

2.8 Creating WLS_JMS_AUDIT_PS FileStore

2.9 Creating AuditJMSServer JMS Server

2.10 Creating WLS_JMS_REPORT_PS FileStore

1.

ORACLE WebLogic Server Administration Console 12c

Welcome, weblogic Connected to: obdx_domain

Home > PREPROCESS > FileUploadJMS > placeholder > Summary of JMS Servers > Summary of JMS Modules > FileUploadJMS > QCF > FileUploadJMS > placeholder > Summary of JMS Servers

Create a New JMS Server

Back Next Finish Cancel

JMS Server Properties

The following properties will be used to identify your new JMS Server.
* Indicates required fields

What would you like to name your new JMS server?

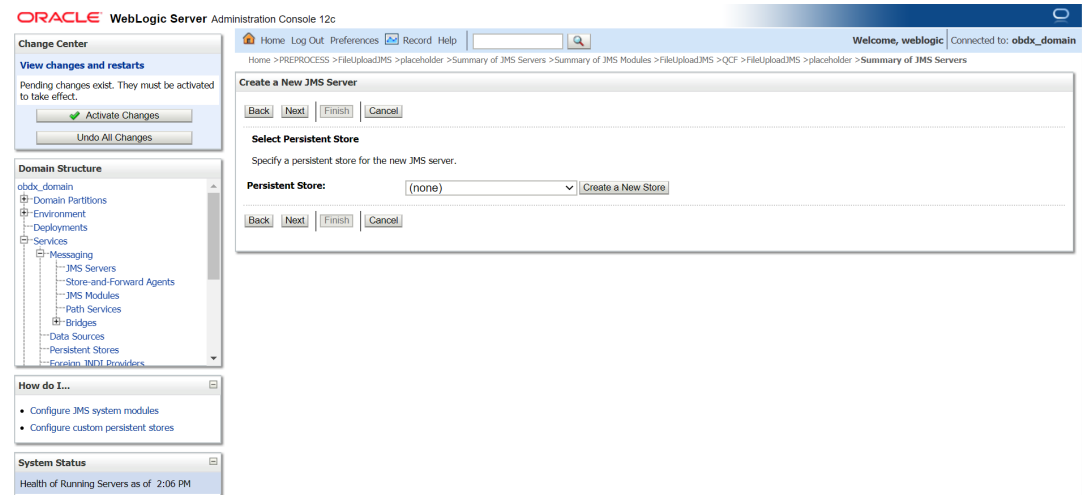
Name: AuditJMSServer

Would you like this new JMS server to be restricted to a specific resource group template or resource group?

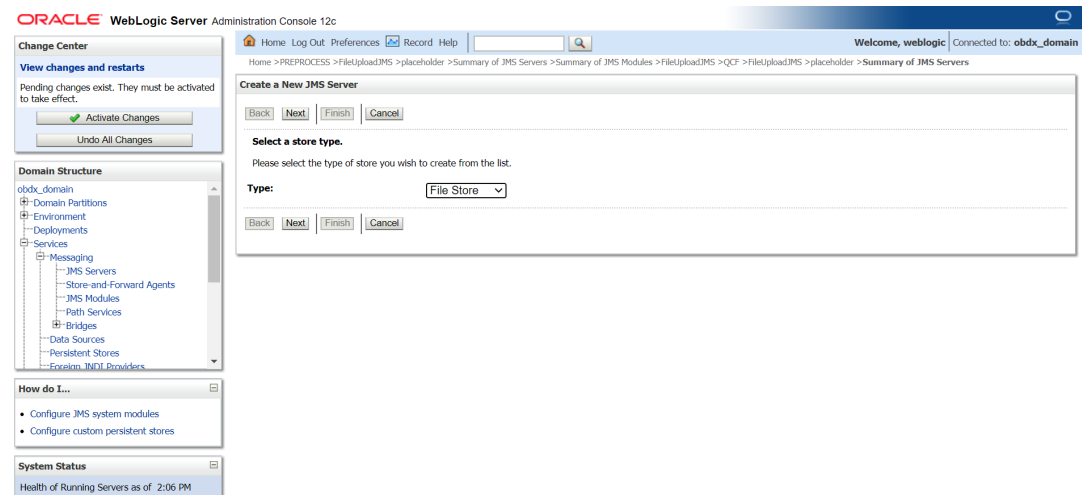
Scope: Global

Back Next Finish Cancel

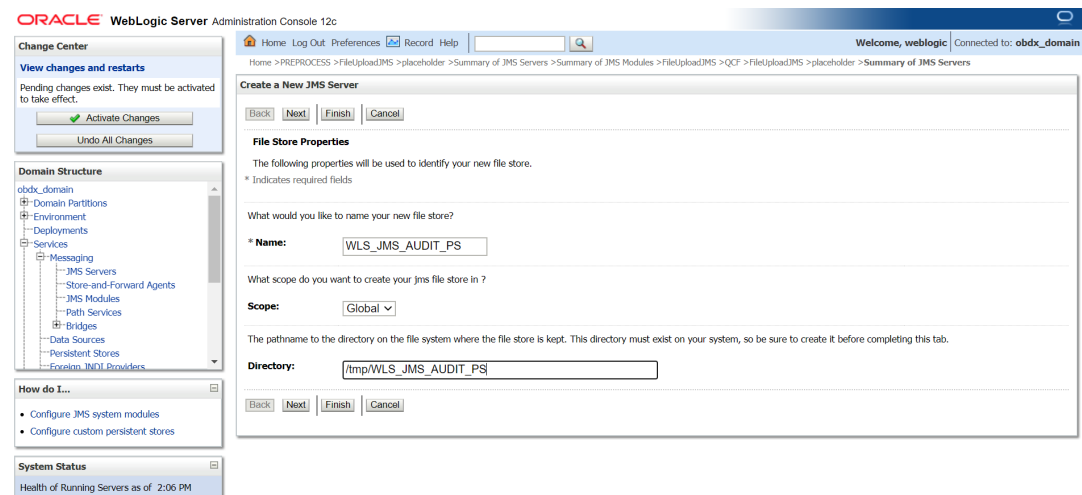
2. Click on JMS server and click **New**.
3. Provide Name as AuditJMSServer, Scope as **Global**.



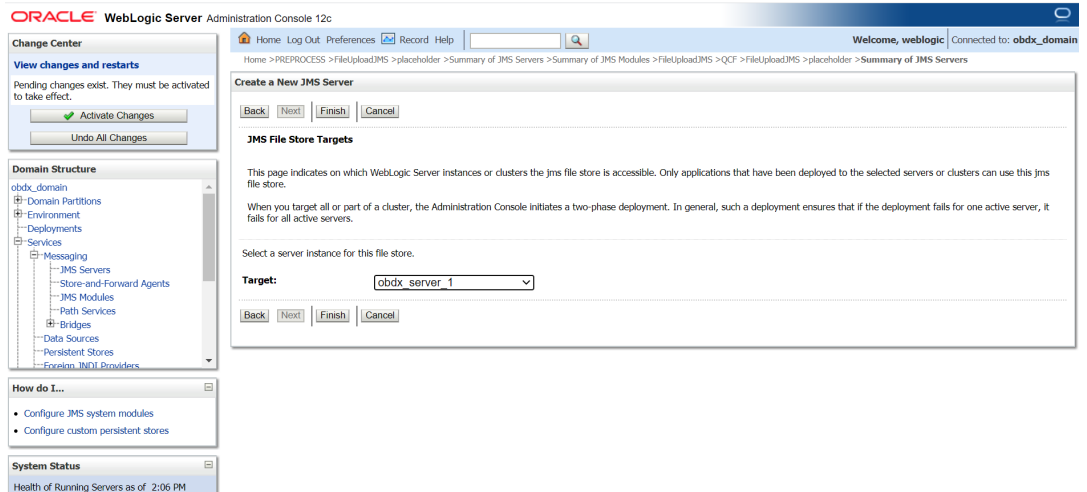
4. Click on **Create a New Store.**



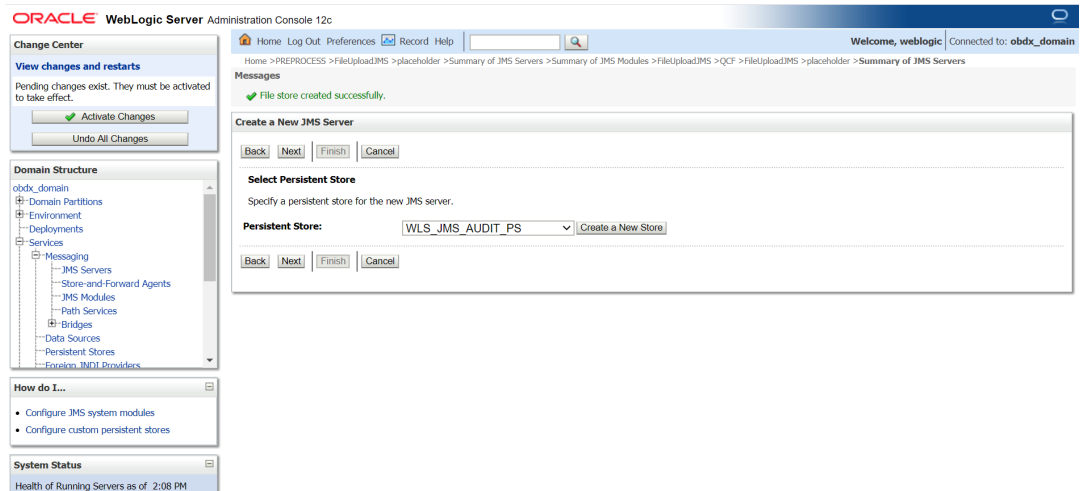
5. Select **File Store.**



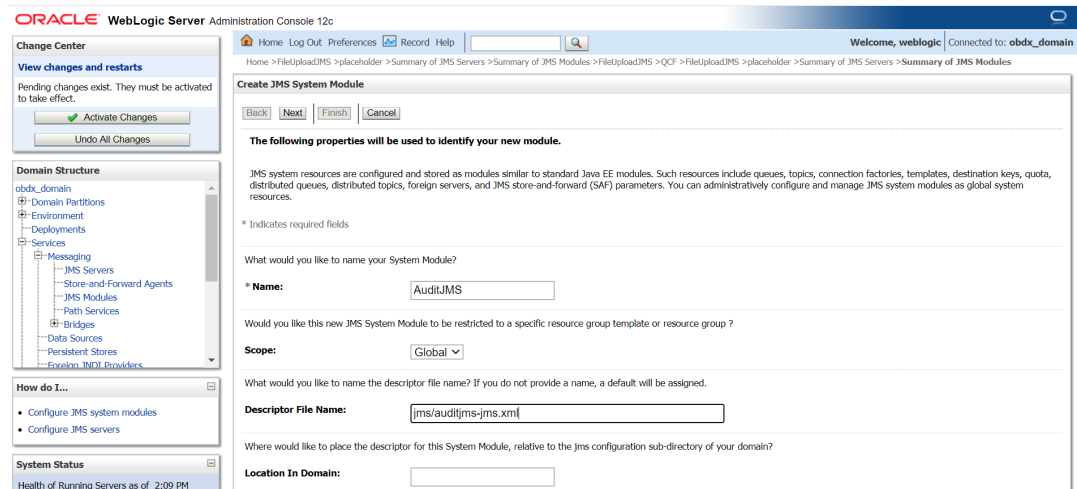
- Provide**
Name : WLS_JMS_AUDIT_PS.
Scope : Global
Directory : /tmp/WLS_JMS_AUDIT_PS.



- Select Target as managed server and click **Finish**.**



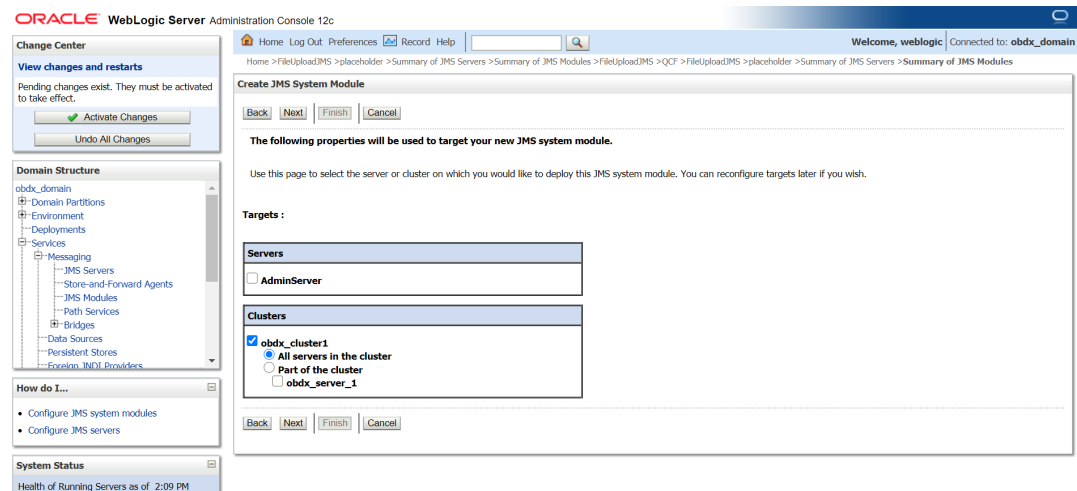
- Select the new store created WLS_JMS_AUDIT_PS and click **Next**.**



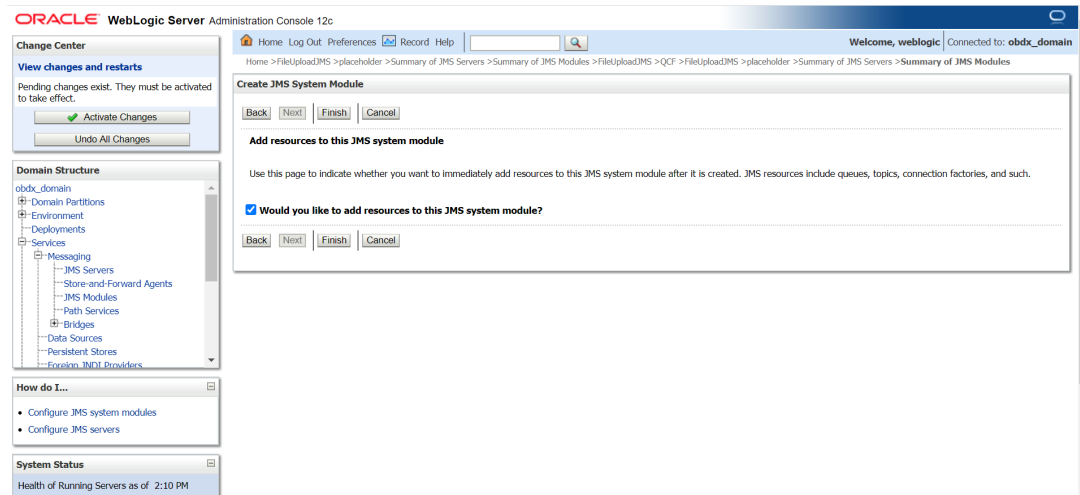
9. Provide
Name : AuditJMS

Scope : Global

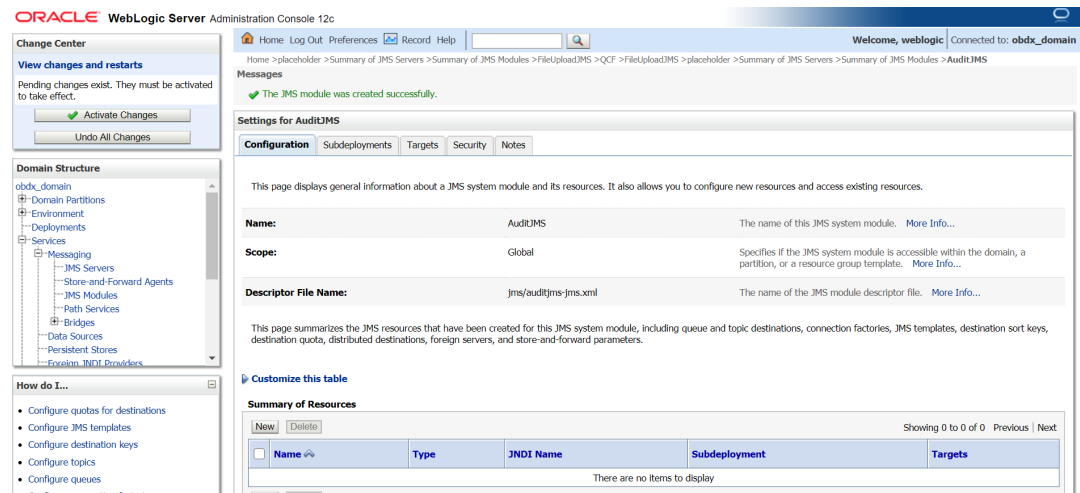
Descriptor File Name: jms/auditjms-jms.xml



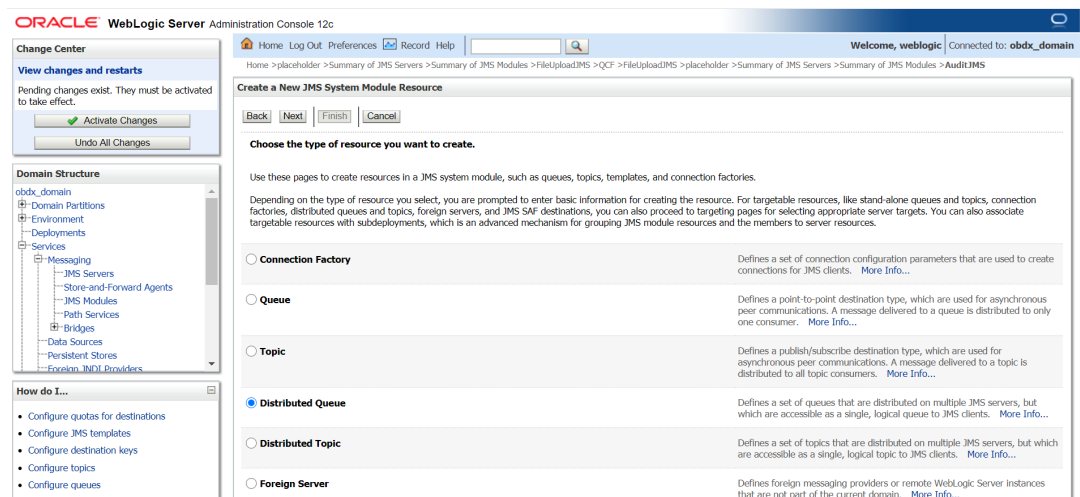
10. Select Cluster as a target.



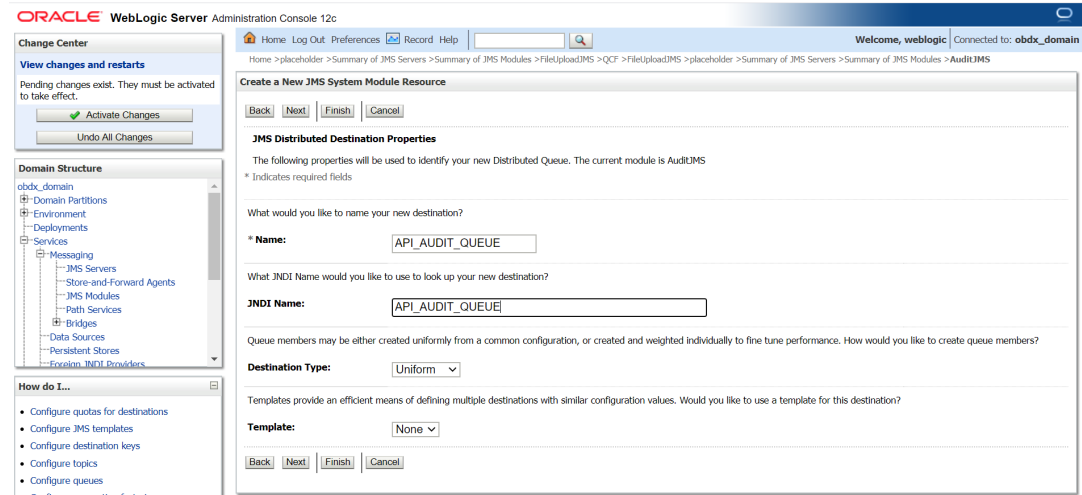
11. Select would you like to add resource to this JMS system module?.



12. Click New.



13. Select Distributed Queue.



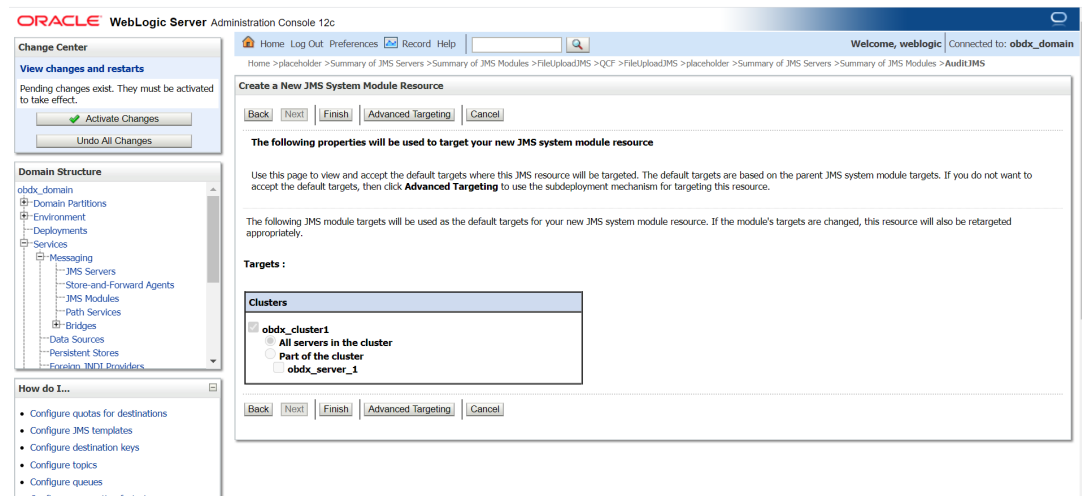
14. Provide:

Name: API_AUDIT_QUEUE

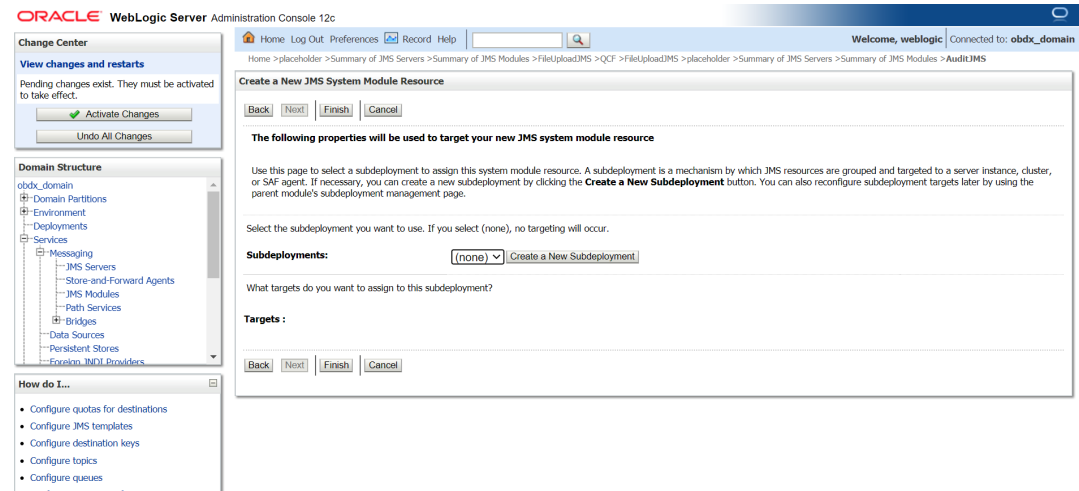
JNDI Name: API_AUDIT_QUEUE

Destination Type : Uniform

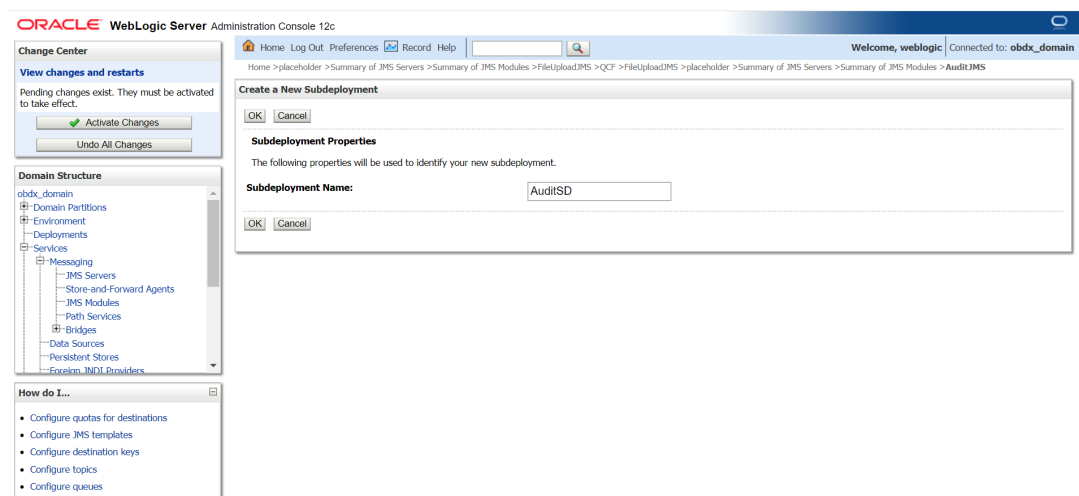
Template:- None



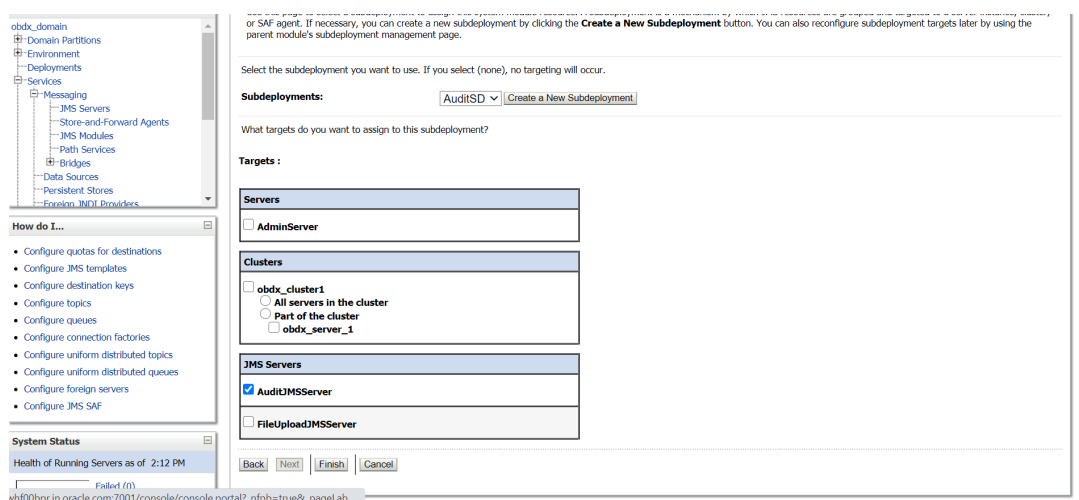
15. Select Advance targeting.



16. Click on Create a New Subdeployment.



17. Provide Subdeployment Name as AuditSD.



18. Select Target as AuditJMSServer.

ORACLE WebLogic Server Administration Console 12c

Home > placeholder > Summary of JMS Servers > Summary of JMS Modules > FileUploadJMS > QCF > FileUploadJMS > placeholder > Summary of JMS Servers > Summary of JMS Modules > AuditJMS

Welcome, weblogic | Connected to: obdx_domain

Create a New JMS System Module Resource

Back | Next | Finish | Cancel

JMS Distributed Destination Properties

The following properties will be used to identify your new Distributed Queue. The current module is AuditJMS

* Indicates required fields

What would you like to name your new destination?

Name:

What JNDI Name would you like to use to look up your new destination?

JNDI Name:

Queue members may be either created uniformly from a common configuration, or created and weighted individually to fine tune performance. How would you like to create queue members?

Destination Type:

Templates provide an efficient means of defining multiple destinations with similar configuration values. Would you like to use a template for this destination?

Template:

Back | Next | Finish | Cancel

ORACLE WebLogic Server Administration Console 12c

Home > placeholder > Summary of JMS Servers > Summary of JMS Modules > FileUploadJMS > QCF > FileUploadJMS > placeholder > Summary of JMS Servers > Summary of JMS Modules > AuditJMS

Welcome, weblogic | Connected to: obdx_domain

Create a New JMS System Module Resource

Back | Next | Finish | Advanced Targeting | Cancel

The following properties will be used to target your new JMS system module resource

Use this page to view and accept the default targets where this JMS resource will be targeted. The default targets are based on the parent JMS system module targets. If you do not want to accept the default targets, then click **Advanced Targeting** to use the subdeployment mechanism for targeting this resource.

The following JMS module targets will be used as the default targets for your new JMS system module resource. If the module's targets are changed, this resource will also be retargeted appropriately.

Targets :

Clusters

obdx_cluster1
 All servers in the cluster
 Part of the cluster
 obdx_server_1

Back | Next | Finish | Advanced Targeting | Cancel

ORACLE WebLogic Server Administration Console 12c

Home > placeholder > Summary of JMS Servers > Summary of JMS Modules > FileUploadJMS > QCF > FileUploadJMS > placeholder > Summary of JMS Servers > Summary of JMS Modules > AuditJMS

Welcome, weblogic | Connected to: obdx_domain

Create a New JMS System Module Resource

Back | Next | Finish | Cancel

The following properties will be used to target your new JMS system module resource

Use this page to select a subdeployment to assign this system module resource. A subdeployment is a mechanism by which JMS resources are grouped and targeted to a server instance, cluster, or SAF agent. If necessary, you can create a new subdeployment by clicking the **Create a New Subdeployment** button. You can also reconfigure subdeployment targets later by using the parent module's subdeployment management page.

Select the subdeployment you want to use. If you select (none), no targeting will occur.

Subdeployments:

What targets do you want to assign to this subdeployment?

Targets :

Servers

AdminServer

Clusters

obdx_cluster1
 All servers in the cluster
 Part of the cluster
 obdx_server_1

JMS Servers

AuditJMSServer

ORACLE WebLogic Server Administration Console 12c

Welcome, weblogic | Connected to: obdx_domain

Home > placeholder > Summary of JMS Servers > Summary of JMS Modules > FileUploadJMS > QCF > FileUploadJMS > placeholder > Summary of JMS Servers > Summary of JMS Modules > AuditJMS

Create a New JMS System Module Resource

Back | Next | Finish | Cancel

Choose the type of resource you want to create.

Use these pages to create resources in a JMS system module, such as queues, topics, templates, and connection factories.

Depending on the type of resource you select, you are prompted to enter basic information for creating the resource. For targetable resources, like stand-alone queues and topics, connection factories, distributed queues and topics, foreign servers, and JMS SAF destinations, you can also proceed to targeting pages for selecting appropriate server targets. You can also associate targetable resources with subdeployments, which is an advanced mechanism for grouping JMS module resources and the members to server resources.

- Connection Factory** Defines a set of connection configuration parameters that are used to create connections for JMS clients. [More Info...](#)
- Queue** Defines a point-to-point destination type, which are used for asynchronous peer communications. A message delivered to a queue is distributed to only one consumer. [More Info...](#)
- Topic** Defines a publish/subscribe destination type, which are used for asynchronous peer communications. A message delivered to a topic is distributed to all topic consumers. [More Info...](#)
- Distributed Queue** Defines a set of queues that are distributed on multiple JMS servers, but which are accessible as a single, logical queue to JMS clients. [More Info...](#)
- Distributed Topic** Defines a set of topics that are distributed on multiple JMS servers, but which are accessible as a single, logical topic to JMS clients. [More Info...](#)
- Foreign Server** Defines foreign messaging providers or remote WebLogic Server instances that are not part of the current domain. [More Info...](#)

ORACLE WebLogic Server Administration Console 12c

Welcome, weblogic | Connected to: obdx_domain

Home > Summary of JMS Modules > FileUploadJMS > QCF > FileUploadJMS > placeholder > Summary of JMS Servers > Summary of JMS Modules > AuditJMS > Summary of JMS Modules > AuditJMS

Create a New JMS System Module Resource

Back | Next | Finish | Cancel

Choose the type of resource you want to create.

Use these pages to create resources in a JMS system module, such as queues, topics, templates, and connection factories.

Depending on the type of resource you select, you are prompted to enter basic information for creating the resource. For targetable resources, like stand-alone queues and topics, connection factories, distributed queues and topics, foreign servers, and JMS SAF destinations, you can also proceed to targeting pages for selecting appropriate server targets. You can also associate targetable resources with subdeployments, which is an advanced mechanism for grouping JMS module resources and the members to server resources.

- Connection Factory** Defines a set of connection configuration parameters that are used to create connections for JMS clients. [More Info...](#)
- Queue** Defines a point-to-point destination type, which are used for asynchronous peer communications. A message delivered to a queue is distributed to only one consumer. [More Info...](#)
- Topic** Defines a publish/subscribe destination type, which are used for asynchronous peer communications. A message delivered to a topic is distributed to all topic consumers. [More Info...](#)
- Distributed Queue** Defines a set of queues that are distributed on multiple JMS servers, but which are accessible as a single, logical queue to JMS clients. [More Info...](#)
- Distributed Topic** Defines a set of topics that are distributed on multiple JMS servers, but which are accessible as a single, logical topic to JMS clients. [More Info...](#)
- Foreign Server** Defines foreign messaging providers or remote WebLogic Server instances that are not part of the current domain. [More Info...](#)

19. Click on connection Factory.

to take effect.

Activate Changes | Undo All Changes

Domain Structure

obdx_domain

- Domain Partitions
- Environment
- Deployments
- Services
 - Messaging
 - JMS Servers
 - Store-and-Forward Agents
 - JMS Modules
 - Path Services
 - Bridges
 - Data Sources
 - Persistent Stores
 - Foreign JNDI Providers

How do I...
 • Configure quotas for destinations
 • Configure JMS templates
 • Configure destination keys
 • Configure topics
 • Configure queues
 • Configure connection factories
 • Configure uniform distributed topics
 • Configure uniform distributed queues
 • Configure foreign servers
 • Configure JMS SAF

Back | Next | Finish | Cancel

Connection Factory Properties

The following properties will be used to identify your new connection factory. The current module is AuditJMS.
 * Indicates required fields

What would you like to name your new connection factory?

Name:

What JNDI Name would you like to use to look up your new connection factory?

JNDI Name:

The Connection Factory Subscription Sharing Policy Subscribers can be used to control which subscribers can access new subscriptions. Should subscriptions created using this factory be sharable?

Subscription Sharing Policy:

The Client ID Policy indicates whether more than one JMS connection can use the same Client ID. Oracle recommends setting the Client ID policy to Unrestricted if sharing durable subscribers. Subscriptions created with different Client ID policies are always treated as independent subscriptions. What Client ID Policy would you like to use?

Client ID Policy:

A connection factory can limit the number of messages that can be queued for an asynchronous session. Should this connection factory impose a limit?

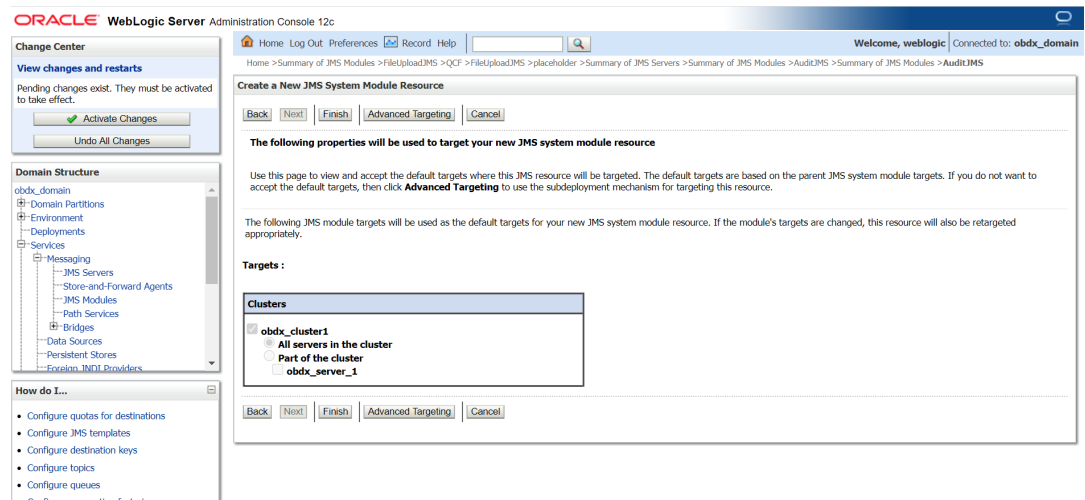
Maximum Messages per Session:

Should this connection factory create sessions that are JTA aware, and create XA queues and XA topics?

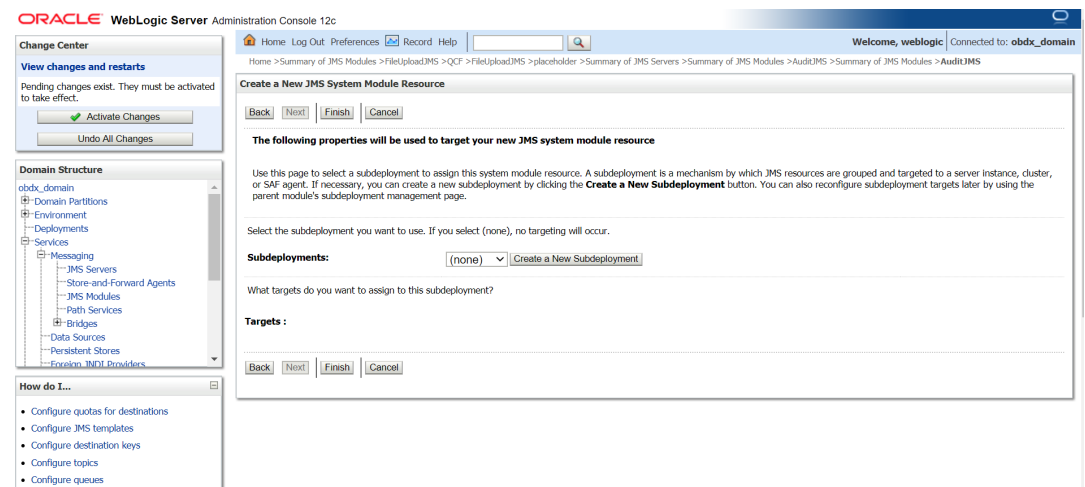
XA Connection Factory Enabled

20. Provide
Name : AUDITQCF

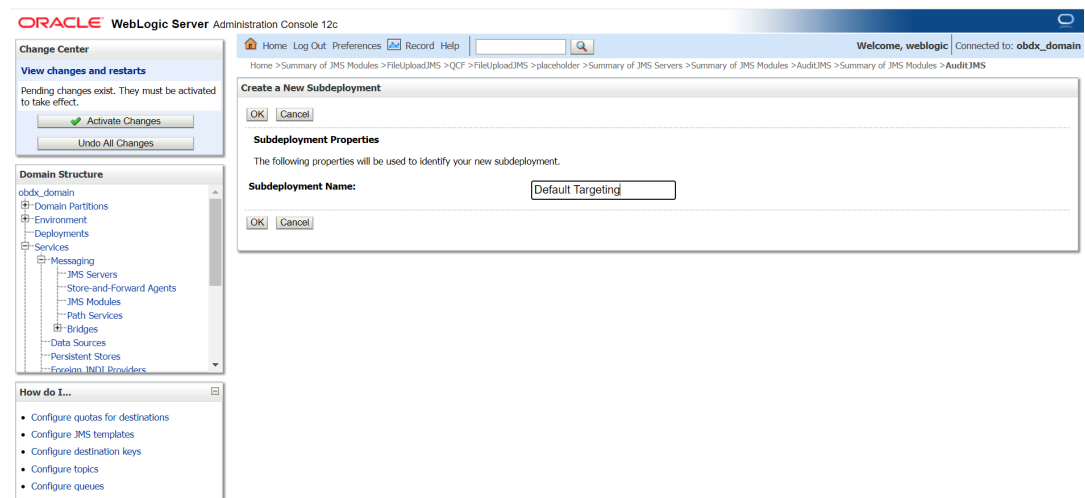
JNDI Name : AUDITQCF



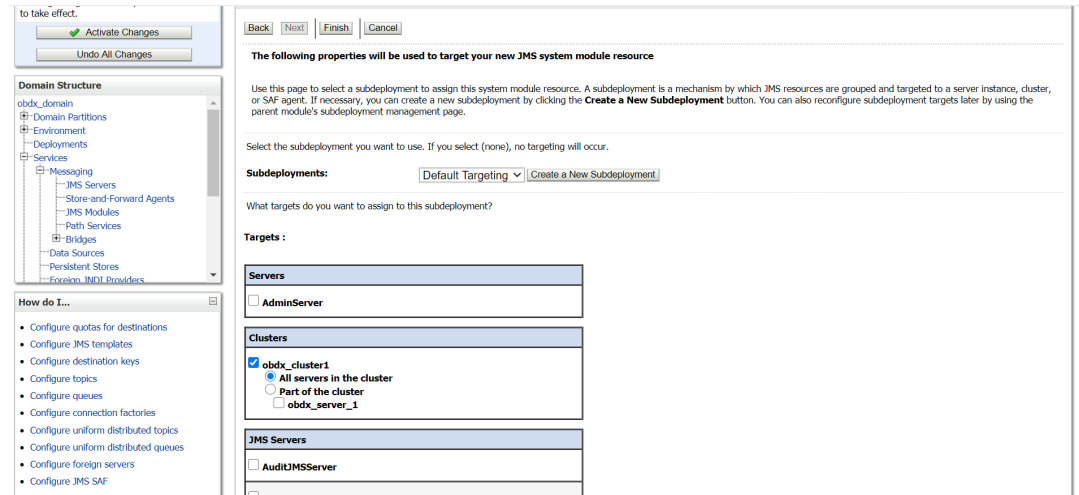
21. Click on Advanced Targeting.



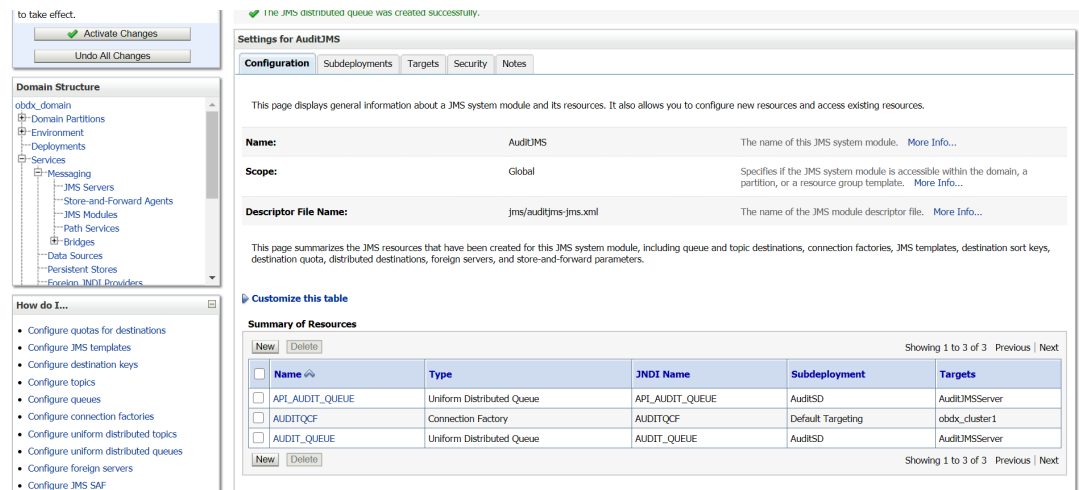
22. Click on Create a New Subdeployment.



23. Give Subdeployment Name as Default Targeting.



24. Under AuditJMS module Create Uniform Distrubuted Queue and connection Factory as show below in the screen shot.



2.11 Creating ReportsJMSServer JMS Server

1. Similarly create ReportsJMSServer under JMS Server and ReportsJMSModule under JMS Module.

Change Center
View changes and restarts
Pending changes exist. They must be activated to take effect.
[Activate Changes] [Undo All Changes]

Domain Structure
obdx_domain
- Domain Partitions
- Environment
- Deployments
- Services
- Messaging
- JMS Servers
- Store-and-Forward Agents
- JMS Modules
- Path Services
- Bridges
- Data Sources
- Persistent Stores
- Foreign JNDI Providers

How do I...
- Configure JMS servers

Summary of JMS Servers
JMS servers act as management containers for the queues and topics in JMS modules that are targeted to them. This page summarizes the JMS servers that have been created in the current WebLogic Server domain.

Customize this table
JMS Servers (Filtered - More Columns Exist)
Showing 1 to 3 of 3 Previous | Next

Name	Persistent Store	Target	Current Target	Health
AuditJMServer	WLS_JMS_AUDIT_PS	obdx_server_1	obdx_server_1	
FileUploadJMSserver	WLS_JMS_FILEUPLOAD_PS	obdx_server_1	obdx_server_1	
ReportsJMSserver	WLS_JMS_REPORT_PS	obdx_server_1	obdx_server_1	

Change Center
View changes and restarts
Pending changes exist. They must be activated to take effect.
[Activate Changes] [Undo All Changes]

Domain Structure
obdx_domain
- Domain Partitions
- Environment
- Deployments
- Services
- Messaging
- JMS Servers
- Store-and-Forward Agents
- JMS Modules
- Path Services
- Bridges
- Data Sources
- Persistent Stores
- Foreign JNDI Providers

How do I...
- Configure JMS system modules
- Configure resources for JMS system modules

System Status
Health of Running Servers as of 3:58 PM

Summary of JMS Modules
JMS system resources are configured and stored as modules similar to standard Java EE modules. Such resources include queues, topics, connection factories, templates, destination keys, quota, distributed queues, distributed topics, foreign servers, and JMS store-and-forward (SAF) parameters. You can administratively configure and manage JMS system modules as global system resources. This page summarizes the JMS system modules that have been created for this domain.

Customize this table
JMS Modules (Filtered - More Columns Exist)
Showing 1 to 3 of 3 Previous | Next

Name	Type
AuditJMS	JMSSystemResource
FileUploadJMS	JMSSystemResource
ReportsJMSModule	JMSSystemResource

Settings for ReportsJMSModule
Configuration | Subdeployments | Targets | Security | Notes

This page displays general information about a JMS system module and its resources. It also allows you to configure new resources and access existing resources.

Name: ReportsJMSModule
The name of this JMS system module. [More Info...](#)

Scope: Global
Specifies if the JMS system module is accessible within the domain, a partition, or a resource group template. [More Info...](#)

Descriptor File Name: Jms/reportsjmsmodule-jms.xml
The name of the JMS module descriptor file. [More Info...](#)

This page summarizes the JMS resources that have been created for this JMS system module, including queue and topic destinations, connection factories, JMS templates, destination sort keys, destination quota, distributed destinations, foreign servers, and store-and-forward parameters.

Customize this table
Summary of Resources
Showing 1 to 3 of 3 Previous | Next

Name	Type	JNDI Name	Subdeployment	Targets
REPORTADHOC	Uniform Distributed Queue	REPORTADHOC	ReportsSubdeployment	ReportsJMSServer
REPORTSCHEDULED	Uniform Distributed Queue	REPORTSCHEDULED	ReportsSubdeployment	ReportsJMSServer
ReportsQCF	Connection Factory	ReportsQCF	Default Targeting	obdx_cluster1

- Under ReportsJMSModule create UniformDistributed Queue and connection factory as show above in the screen shot.
REPORTADHOC – Uniform Distributed Queue
REPORTSCHEDULED - Uniform Distributed Queue
ReportsQCF – Connection Factory

2.12 Creating jpa-cache JMS Server

2.13 Creating WLS_JPA_PS FileStore

1. Create jpa-cache JMS server and jpa-cache JMS Module as show in below screen shot.

Summary of JMS Servers

JMS servers act as management containers for the queues and topics in JMS modules that are targeted to them. This page summarizes the JMS servers that have been created in the current WebLogic Server domain.

Customize this table

JMS Servers (Filtered - More Columns Exist)

Name	Persistent Store	Target	Current Target	Health
AuditJMServer	WLS_JMS_AUDIT_PS	obdx_server_1	obdx_server_1	
FileUploadJMSserver	WLS_JMS_FILEUPLOAD_PS	obdx_server_1	obdx_server_1	
jpa-cache	WLS_JPA_PS	obdx_server_1	obdx_server_1	
ReportsJMServer	WLS_JMS_REPORT_PS	obdx_server_1	obdx_server_1	

Summary of JMS Modules

JMS system resources are configured and stored as modules similar to standard Java EE modules. Such resources include queues, topics, connection factories, templates, destination keys, quota, distributed queues, distributed topics, foreign servers, and JMS store-and-forward (SAF) parameters. You can administratively configure and manage JMS system modules as global system resources. This page summarizes the JMS system modules that have been created for this domain.

Customize this table

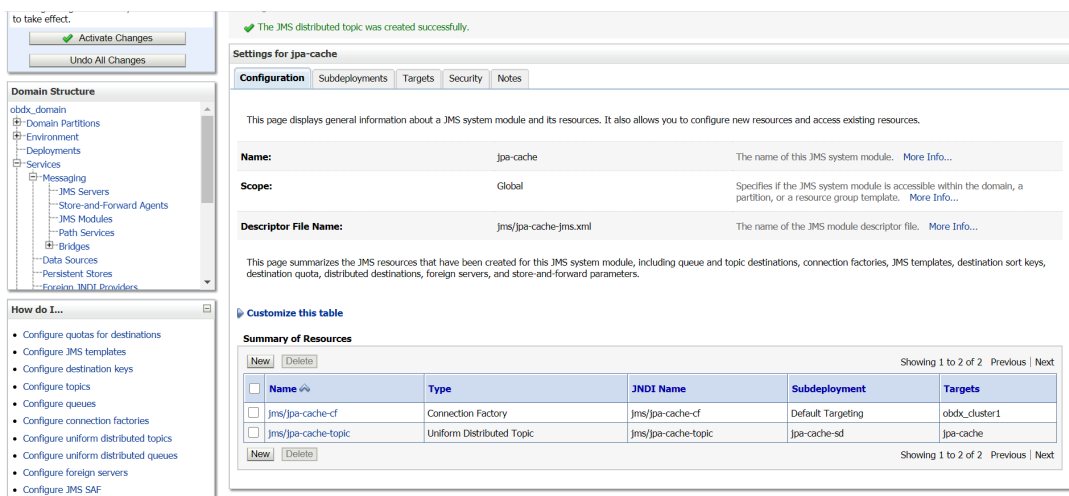
JMS Modules (Filtered - More Columns Exist)

Name	Type
AuditJMS	JMSSystemResource
FileUploadJMS	JMSSystemResource
jpa-cache	JMSSystemResource
ReportsJMSModule	JMSSystemResource

2. Under jpa-cache JMS Module create connection Factory and Uniform Distributed topic as shown in below screen shot.

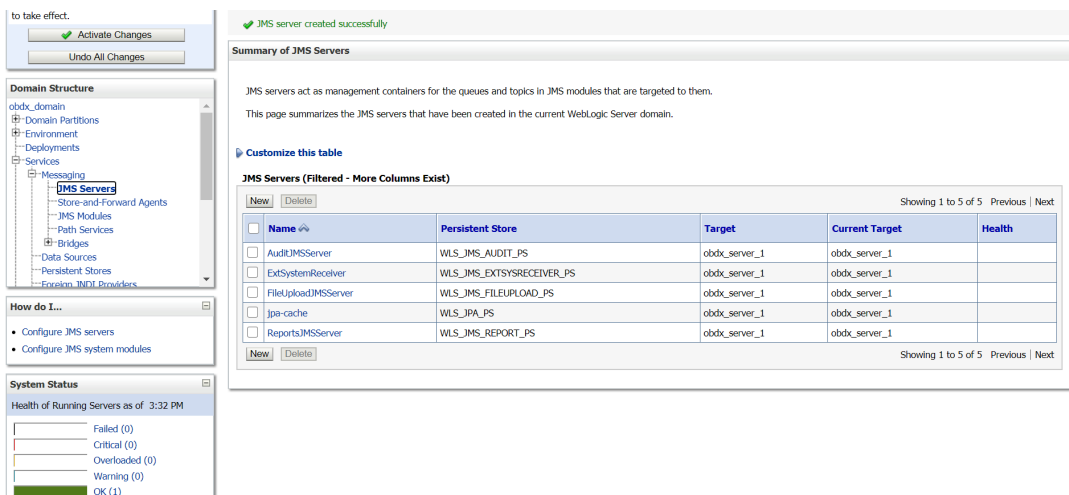
Jms/jpa-cache-cf --- Connection Factory

Jms/jpa-cache-topic --- Uniform Distributed Topic



2.14 Creating ExtSystemReceiver JMS Server - WLS_JMS_EXTSYSRECEIVER_PS FileStore

1. Create ExtSystemReceiver JMS Server Persistent store file store as WLS_JMS_EXTSYSRECEIVER_PS as show in below screen shot.



2. Create ExtSystemReceiver JMS Module as below.

to take effect.

Domain Structure

obdx_domain

- Domain Partitions
- Environment
 - Deployments
 - Services
 - Messaging
 - JMS Servers
 - Store-and-Forward Agents
 - JMS Modules**
 - Path Services
 - Bridges
 - Data Sources
 - Persistent Stores
 - Foreign JNDI Providers

How do I...

- Configure JMS system modules
- Configure resources for JMS system modules

System Status

Health of Running Servers as of 4:16 PM

Failed (0)
 Critical (0)
 Overloaded (0)
 Warning (0)
 OK (1)

Summary of JMS Modules

JMS system resources are configured and stored as modules similar to standard Java EE modules. Such resources include queues, topics, connection factories, templates, destination keys, quota, distributed queues, distributed topics, foreign servers, and JMS store-and-forward (SAF) parameters. You can administratively configure and manage JMS system modules as global system resources.

This page summarizes the JMS system modules that have been created for this domain.

Customize this table

JMS Modules (Filtered - More Columns Exist)

Showing 1 to 5 of 5 Previous | Next

Name	Type
AuditJMS	JMSSystemResource
ExtSystemReceiver	JMSSystemResource
FileUploadJMS	JMSSystemResource
jpa-cache	JMSSystemResource
ReportsJMSModule	JMSSystemResource

Showing 1 to 5 of 5 Previous | Next

3. Create ExtSystemReceiverQCF – connection Factory and ExtSystemReceiverQueue – uniform Distributed Queue in ExtSystemReceiver JMS Module refer below screen shot.

to take effect.

Domain Structure

obdx_domain

- Domain Partitions
- Environment
 - Deployments
 - Services
 - Messaging
 - JMS Servers
 - Store-and-Forward Agents
 - JMS Modules
 - Path Services
 - Bridges
 - Data Sources
 - Persistent Stores
 - Foreign JNDI Providers

How do I...

- Configure quotas for destinations
- Configure JMS templates
- Configure destination keys
- Configure topics
- Configure queues
- Configure connection factories
- Configure uniform distributed topics
- Configure uniform distributed queues
- Configure foreign servers
- Configure JMS SAF

Settings for ExtSystemReceiver

Configuration | Subdeployments | Targets | Security | Notes

This page displays general information about a JMS system module and its resources. It also allows you to configure new resources and access existing resources.

Name: ExtSystemReceiver The name of this JMS system module. [More Info...](#)

Scope: Global Specifies if the JMS system module is accessible within the domain, a partition, or a resource group template. [More Info...](#)

Descriptor File Name: Jms/extsystemreceiver-jms.xml The name of the JMS module descriptor file. [More Info...](#)

This page summarizes the JMS resources that have been created for this JMS system module, including queue and topic destinations, connection factories, JMS templates, destination sort keys, destination quota, distributed destinations, foreign servers, and store-and-forward parameters.

Customize this table

Summary of Resources

Showing 1 to 2 of 2 Previous | Next

Name	Type	JNDI Name	Subdeployment	Targets
ExtSystemReceiverQCF	Connection Factory	ExtSystemReceiverQCF	Default Targeting	obdx_cluster1
ExtSystemReceiverQueue	Uniform Distributed Queue	ExtSystemReceiverQueue	ExtSystemReceiverSub	ExtSystemReceiver

Showing 1 to 2 of 2 Previous | Next

2.15 Creating ExtSystemSender JMS Server Persistent Store FileStore as WLS_JMS_EXTSYSENDER_PS

1. As show below create JMS Server ExtSystemSender.

to take effect.

Activate Changes
Undo All Changes

Domain Structure

- obdx_domain
 - Domain Partitions
 - Environment
 - Deployments
 - Services
 - Messaging
 - JMS Servers
 - Store-and-Forward Agents
 - JMS Modules
 - Path Services
 - Bridges
 - Data Sources
 - Persistent Stores
 - Foreign JNDI Providers

How do I...
 • Configure JMS servers
 • Configure JMS system modules

System Status
 Health of Running Servers as of 3:33 PM

- Failed (0)
- Critical (0)
- Overloaded (0)
- Warning (0)
- OK (1)

Summary of JMS Servers

JMS servers act as management containers for the queues and topics in JMS modules that are targeted to them. This page summarizes the JMS servers that have been created in the current WebLogic Server domain.

Customize this table

JMS Servers (Filtered - More Columns Exist)

Name	Persistent Store	Target	Current Target	Health
AuditJMSReceiver	WLS_JMS_AUDIT_PS	obdx_server_1	obdx_server_1	
ExtSystemReceiver	WLS_JMS_EXTSYSRECEIVER_PS	obdx_server_1	obdx_server_1	
ExtSystemSender	WLS_JMS_EXTSYSENDER_PS	obdx_server_1	obdx_server_1	
FileUploadJMSReceiver	WLS_JMS_FILEUPLOAD_PS	obdx_server_1	obdx_server_1	
java-cache	WLS_JPA_PS	obdx_server_1	obdx_server_1	
ReportsJMSReceiver	WLS_JMS_REPORT_PS	obdx_server_1	obdx_server_1	

2. Create ExtSystemSender JMS Module.

ORACLE WebLogic Server Administration Console 12c

Home Log Out Preferences Record Help

Welcome, weblogic Connected to: obdx_domain

Home > Summary of Persistent Stores > Summary of JMS Modules > ReportsJMSModule > Summary of JMS Modules > java-cache > Summary of JMS Modules > ExtSystemReceiver > Summary of JMS Modules > ExtSystemSender > Summary of JMS Modules

Summary of JMS Modules

JMS system resources are configured and stored as modules similar to standard Java EE modules. Such resources include queues, topics, connection factories, templates, destination keys, quota, distributed queues, distributed topics, foreign servers, and JMS store-and-forward (SAF) parameters. You can administratively configure and manage JMS system modules as global system resources. This page summarizes the JMS system modules that have been created for this domain.

Customize this table

JMS Modules (Filtered - More Columns Exist)

Name	Type
AuditJMS	JMSSystemResource
ExtSystemReceiver	JMSSystemResource
ExtSystemSender	JMSSystemResource
FileUploadJMS	JMSSystemResource
java-cache	JMSSystemResource
ReportsJMSModule	JMSSystemResource

3. Under ExtSystemSender JMS Module create ExtSystemSenderQCF – connection Factory and ExtSystemSenderQueue – Uniform Distributed Queue as show below.

obdx_domain

- Environment
 - Deployments
 - Services
 - Messaging
 - JMS Servers
 - Store-and-Forward Agents
 - JMS Modules
 - Path Services
 - Bridges
 - Data Sources
 - Persistent Stores
 - Foreign JNDI Providers

How do I...
 • Configure quotas for destinations
 • Configure JMS templates
 • Configure destination keys
 • Configure topics
 • Configure queues
 • Configure connection factories
 • Configure uniform distributed topics
 • Configure uniform distributed queues
 • Configure foreign servers
 • Configure JMS SAF

System Status
 Health of Running Servers as of 4:22 PM

- Failed (0)
- Warning (0)
- OK (1)

This page displays general information about a JMS system module and its resources. It also allows you to configure new resources and access existing resources.

Name: ExtSystemSender The name of this JMS system module. More Info...

Scope: Global Specifies if the JMS system module is accessible within the domain, a partition, or a resource group template. More Info...

Descriptor File Name: jms/extsystemsender-jms.xml The name of the JMS module descriptor file. More Info...

This page summarizes the JMS resources that have been created for this JMS system module, including queue and topic destinations, connection factories, JMS templates, destination sort keys, destination quota, distributed destinations, foreign servers, and store-and-forward parameters.

Customize this table

Summary of Resources

Name	Type	JNDI Name	Subdeployment	Targets
ExtSystemSenderQCF	Connection Factory	ExtSystemSenderQCF	Default Targeting	obdx_cluster1
ExtSystemSenderQueue	Uniform Distributed Queue	ExtSystemSenderQueue	ExtSystemSenderSub	ExtSystemSender

2.16 Creating UBSForeignServer JMS Server

1.

to take effect.

Domain Structure

obdx_domain

- Domain Partitions
- Environment
- Deployments
- Services
 - Messaging
 - JMS Servers
 - Store-and-Forward Agents
 - JMS Modules**
 - Path Services
 - Bridges
 - Data Sources
 - Persistent Stores
 - Foreign JNDI Providers

How do I...

- Configure JMS system modules
- Configure resources for JMS system modules

System Status

Health of Running Servers as of 4:24 PM

- Failed (0)
- Critical (0)
- Overloaded (0)
- Warning (0)
- OK (1)

Summary of JMS Modules

JMS system resources are configured and stored as modules similar to standard Java EE modules. Such resources include queues, topics, connection factories, templates, destination keys, quota, distributed queues, distributed topics, foreign servers, and JMS store-and-forward (SAF) parameters. You can administratively configure and manage JMS system modules as global system resources.

This page summarizes the JMS system modules that have been created for this domain.

Customize this table

JMS Modules (Filtered - More Columns Exist)

Showing 1 to 7 of 7 Previous Next

Name	Type
<input type="checkbox"/> AuditJMS	JMSSystemResource
<input type="checkbox"/> ExtSystemReceiver	JMSSystemResource
<input type="checkbox"/> ExtSystemSender	JMSSystemResource
<input type="checkbox"/> FileUploadJMS	JMSSystemResource
<input type="checkbox"/> jpa-cache	JMSSystemResource
<input type="checkbox"/> ReportsJMSModule	JMSSystemResource
<input type="checkbox"/> UBSSystemModule	JMSSystemResource

Showing 1 to 7 of 7 Previous Next

2. Under UBSSystemModule create UBSForeignServer – Foreign Server as shown below.

to take effect.

Domain Structure

obdx_domain

- Domain Partitions
- Environment
- Deployments
- Services
 - Messaging
 - JMS Servers
 - Store-and-Forward Agents
 - JMS Modules
 - Path Services
 - UBSForeignServer
 - Bridges
 - Data Sources
 - Persistent Stores
 - Foreign JNDI Providers

How do I...

- Configure quotas for destinations
- Configure JMS templates
- Configure destination keys
- Configure topics
- Configure queues
- Configure connection factories
- Configure uniform distributed topics
- Configure uniform distributed queues
- Configure foreign servers
- Configure JMS SAF

Settings for UBSSystemModule

The foreign server was created successfully.

Configuration Subdeployments Targets Security Notes

This page displays general information about a JMS system module and its resources. It also allows you to configure new resources and access existing resources.

Name: UBSSystemModule The name of this JMS system module. [More Info...](#)

Scope: Global Specifies if the JMS system module is accessible within the domain, a partition, or a resource group template. [More Info...](#)

Descriptor File Name: jms/ubssystemmodule-jms.xml The name of the JMS module descriptor file. [More Info...](#)

This page summarizes the JMS resources that have been created for this JMS system module, including queue and topic destinations, connection factories, JMS templates, destination sort keys, destination quota, distributed destinations, foreign servers, and store-and-forward parameters.

Customize this table

Summary of Resources

Showing 1 to 1 of 1 Previous Next

Name	Type	JNDI Name	Subdeployment	Targets
<input type="checkbox"/> UBSForeignServer	Foreign Server	N/A	UBSSubdeployment	obdx_cluster1

Showing 1 to 1 of 1 Previous Next

2.17 Creating OBPMForeignServer JMS Server

1. In JMSModule create OBPMSystemModule.

to take effect.

Domain Structure

obdx_domain

- Domain Partitions
- Environment
- Deployments
- Services
 - Messaging
 - JMS Servers
 - Store-and-Forward Agents
 - JMS Modules**
 - Path Services
 - Bridges
 - Data Sources
 - Persistent Stores
 - Foreign JNDI Providers

How do I...

- Configure JMS system modules
- Configure resources for JMS system modules

System Status

Health of Running Servers as of 4:27 PM

- Failed (0)
- Critical (0)
- Overloaded (0)
- Warning (0)
- OK (1)

Summary of JMS Modules

JMS system resources are configured and stored as modules similar to standard Java EE modules. Such resources include queues, topics, connection factories, templates, destination keys, quota, distributed queues, distributed topics, foreign servers, and JMS store-and-forward (SAF) parameters. You can administratively configure and manage JMS system modules as global system resources.

This page summarizes the JMS system modules that have been created for this domain.

Customize this table

JMS Modules (Filtered - More Columns Exist)

Showing 1 to 8 of 8 Previous | Next

<input type="checkbox"/>	Name ↕	Type
<input type="checkbox"/>	AuditJMS	JMSSystemResource
<input type="checkbox"/>	ExtSystemReceiver	JMSSystemResource
<input type="checkbox"/>	ExtSystemSender	JMSSystemResource
<input type="checkbox"/>	FileUploadJMS	JMSSystemResource
<input type="checkbox"/>	jpa-cache	JMSSystemResource
<input type="checkbox"/>	OBPMSystemModule	JMSSystemResource
<input type="checkbox"/>	ReportsJMSModule	JMSSystemResource
<input type="checkbox"/>	UBSSystemModule	JMSSystemResource

Showing 1 to 8 of 8 Previous | Next

- Under OBPMSystemModule create OBPMForeignServer – Foreign Server as show below in screen shot.

to take effect.

Domain Structure

obdx_domain

- Domain Partitions
- Environment
- Deployments
- Services
 - Messaging
 - JMS Servers
 - Store-and-Forward Agents
 - JMS Modules
 - Path Services
 - Bridges
 - Data Sources
 - Persistent Stores
 - Foreign JNDI Providers

How do I...

- Configure quotas for destinations
- Configure JMS templates
- Configure destination keys
- Configure topics
- Configure queues
- Configure connection factories
- Configure uniform distributed topics
- Configure uniform distributed queues
- Configure foreign servers
- Configure JMS SAF

Settings for OBPMSystemModule

The foreign server was created successfully.

Configuration Subdeployments Targets Security Notes

This page displays general information about a JMS system module and its resources. It also allows you to configure new resources and access existing resources.

Name: OBPMSystemModule The name of this JMS system module. [More Info...](#)

Scope: Global Specifies if the JMS system module is accessible within the domain, a partition, or a resource group template. [More Info...](#)

Descriptor File Name: jms/obpmsystemmodule-jms.xml The name of the JMS module descriptor file. [More Info...](#)

This page summarizes the JMS resources that have been created for this JMS system module, including queue and topic destinations, connection factories, JMS templates, destination sort keys, destination quota, distributed destinations, foreign servers, and store-and-forward parameters.

Customize this table

Summary of Resources

Showing 1 to 1 of 1 Previous | Next

<input type="checkbox"/>	Name ↕	Type	JNDI Name	Subdeployment	Targets
<input type="checkbox"/>	OBPMForeignServer	Foreign Server	N/A	OBPMSubdeployment	obdx_cluster1

Showing 1 to 1 of 1 Previous | Next

3

Deploying Applications

Deployment of Lib and Apps

```
${MW_HOME}/wlserver/common/deployable-libraries/jax-rs-2.0.war  
(Target - obapi_cluster, AdminServer)
```

```
${OBAPI_INSTALLER}/installables/app/components/obapi/deploy/  
obapi.app.domain.ear  
(Target - obapi_cluster, AdminServer)
```

```
${OBAPI_INSTALLER}/installables/app/components/obapi/deploy/  
obapi.app.framework.ear  
(Target - obapi_cluster, AdminServer)
```

```
${OBAPI_INSTALLER}/installables/app/components/obapi/deploy/  
obapi.extsystem.domain.ear  
(Target - obapi_cluster, AdminServer)
```

```
${OBAPI_INSTALLER}/installables/app/components/obapi/deploy/  
obapi.app.core.domain.ear  
(Target - obapi_cluster ,AdminServer)
```

```
${OBAPI_INSTALLER}/installables/app/components/obapi/deploy/  
obapi.thirdparty.app.domain.ear  
(Target - obapi_cluster,AdminServer)
```

```
${OBAPI_INSTALLER}/installables/app/components/obapi/deploy/  
obapi.app.rest.idm.ear  
(Target - obapi_cluster)
```

```
${OBAPI_INSTALLER}/installables/app/components/obapi/deploy/  
BatchResourceAdapter.ear  
(Target - obapi_cluster)
```

```
${OBAPI_INSTALLER}/installables/app/components/obapi/deploy/AuditMDBEAR.ear  
(Target - obapi_cluster)
```

```
${OBAPI_INSTALLER}/installables/app/components/obapi/deploy/  
com.ofss.digx.app.connector.ear  
(Target - obapi_cluster)
```

```
${OBAPI_INSTALLER}/installables/app/components/obapi/deploy/  
obapi.app.mdb.report.ear  
(Target - obapi_cluster)
```

```
${OBAPI_INSTALLER}/installables/app/components/obapi/deploy/  
obapi.app.timer.ear  
(Target - obapi_cluster)
```

```
{OBAPI INSTALLER}/installables/app/components/obapi/deploy/
obapi.app.oauth.ear
(Target - obapi_cluster,AdminServer)

{OBAPI INSTALLER}/installables/app/components/obapi/deploy/
obapi.app.oauth.rest.ear
(Target - obapi_cluster)

{OBAPI INSTALLER}/installables/app/cz/obapi.cz.app.domain.ear
(Target - obapi_cluster,AdminServer)

{OBAPI INSTALLER}/installables/app/cz/obapi.cz.extsystem.domain.ear
(Target - obapi_cluster,AdminServer)

{OBAPI INSTALLER}/installables/app/cz/obapi.cz.thirdparty.app.domain.ear
(Target - obapi_cluster,AdminServer)

{OBAPI INSTALLER}/installables/app/components/ubs/deploy/obapi.app.soap.ear
(Target - obapi_cluster)

{OBAPI INSTALLER}/installables/app/components/thp/deploy/
ExtxfaceSimulatorMDB.ear
(Target - obapi_cluster)
```

4

Configured jps-config.xml

Update the jps-config.xml

Edit \$DOMAIN_HOME/config/fmwconfig/jps-config.xml file and add following entries.

1. Find <serviceProviders> tag in the file, add below serviceProvider between <serviceProviders></serviceProviders>.

```
<serviceProvider type="IDENTITY_STORE" name="custom.provider"
class="oracle.security.jps.internal.idstore.generic.GenericIdentityStorePro
vider">
<description>Custom IdStore Provider</description></serviceProvider>
```

2. Find <serviceInstances> tag in the file, add below serviceInstances between <serviceInstances></serviceInstances>.

```
<serviceInstance name="idstore.custom" provider="custom.provider"
location="dumb">
<description>Custom Identity Store Service Instance</description>
<property name="idstore.type" value="CUSTOM"/>
<property name="ADF_IM_FACTORY_CLASS"
value="com.ofss.sms.dbAuthenticator.providers.db.DBIdentityStoreFactory"/>

<property name="DATASOURCE_NAME" value="DIGX"/>
</serviceInstance>
```

3. Find <jpsContext name="default"> tag in the file, add below serviceInstanceRef between <jpsContext name="default"></jpsContext>.

```
<serviceInstanceRef ref="idstore.custom"/>
```

5

List of Topics

This user manual is organized as follows:

Table 5-1 List of Topics

Topics	Description
Preface	This topic provides information on the introduction, intended audience, list of topics, and acronyms covered in this guide.
Manual OBAPI installation	This topic provides a step to install OBDX database manually.
WEBLOGIC Setup and Configuration	This topic provides information about the creations of weblogic domain , managed server, creations of cluster, configuration of node manager.
Deploying Applications	This topic describes deployment of Lib and Apps.
Configured jps-config.xml	This topic explains the configuration of jps-config.xml.

Index

C

- Configured jps-config.xml, [4-1](#)
- Create Audit Tablespace (file obapi_audit_create_tablespace.sql), [1-2](#)
- Create Bigfile Tablespace TBS_\${EHMS_SCHEMA_NAME}, [1-3](#)
- Create JMS Server and JMS Module, [2-17](#)
- Create OBAPI Tablespace (file obapi_create_tablespace.sql), [1-1](#)
- Create Role (file obapi_create_role.sql), [1-2](#)
- Create Role (file obpm_create_role.sql), [1-4](#)
- Create Role ROLE_\${EHMS_SCHEMA_NAME} NOT IDENTIFIED, [1-4](#)
- Create User (file obapi_create_user.sql), [1-2](#)
- Creating B1A1 Data Source, [2-14](#)
- Creating BATCH Data Source, [2-8](#)
- Creating DIGX Data Source, [2-2](#)
- Creating ExtSystemReceiver JMS Server - WLS_JMS_EXTSYSRECEIVER_PS FileStore, [2-42](#)
- Creating ExtSystemSender JMS Server Persistent Store FileStore as WLS_JMS_EXTSYSSENDER_PS, [2-43](#)
- Creating NONXA Data Source, [2-5](#)
- Creating OBPMForeignServer JMS Server, [2-45](#)
- Creating ReportsJMSServer JMS Server, [2-39](#)
- Creating SYSCONFIG Data Source, [2-11](#)
- Creating UBSForeignServer JMS Server, [2-45](#)
- Creating WLS_JMS_REPORT_PS FileStore, [2-29](#)
- Creating WLS_JPA_PS FileStore, [2-41](#)

D

- Deploying Applications, [3-1](#)

F

- Files execution in sequences on above schema (ex. OBAPI_\${POST_FIX}), [1-3](#)

G

- Grants Execitions, [1-4](#)

M

- Manual OBAPI Installation, [1-1](#)

O

- OBPM Database Installation (OBPM Favor), [1-3](#)

P

- Policy Seeding, [1-5](#)

S

- Scripts Execution, [1-4](#)
- Setting Domain JTA Transaction Timeout, [2-1](#)

T

- Tablespace Creation (file obpm_create_tablespace.sql), [1-3](#)

U

- User Creation (file obpm_create_user.sql), [1-3](#)

W

- WEBLOGIC Setup and Configuration, [2-1](#)