Installation Guide- Non-Linux Platforms
Oracle Banking Digital Experience
Patchset Release 22.2.5.0.0

Part No. F72987-01

October 2024

ORACLE

Installation Guide- Non-Linux Platforms

October 2024

Oracle Financial Services Software Limited

Oracle Park

Off Western Express Highway

Goregaon (East)

Mumbai, Maharashtra 400 063

India

Worldwide Inquiries:

Phone: +91 22 6718 3000

Fax:+91 22 6718 3001

www.oracle.com/financialservices/

Table of Contents

1. Preface

1.1 Purpose

Welcome to the User Guide for Oracle Banking Digital Experience. This guide explains the operations that the user will follow while using the application.

1.2 Audience

This manual is intended for Customers and Partners who setup and use Oracle Banking Digital Experience.

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 ensure 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 or the glossary.
Italic	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.

1.7 Screenshot Disclaimer

The images of screens used in this user manual are for illustrative purpose only, to provide improved understanding of the functionality; actual screens that appear in the application may vary based on selected browser, theme, and mobile devices.

1.8 Acronyms and Abbreviations

The list of the acronyms and abbreviations that you are likely to find in the manual are as follows:

Abbreviation	Description
OBDX	Oracle Banking Digital Experience

2. Manual OBDX installation

OBDX Database Installation with OBPM FLAVOR

Once obdx and ehms schema created in base installer, please proceed to below path for patchset scripts execution -

OBDX_Installer/installables/OBDX/<Installation type>/<version>/db/<version>/OBDX/

Inside above path ddl, dml, and constraints folders are present inside which OBDX scripts will be present which needs to be executed manually.

If any place holder or variables that needs to be replaced manually before executing.

Similarly for other modules also you can find scripts those are to be executed in below path -

OBDX_Installer/installables/OBDX/<Installation type>/<version>/db/<version>/

Inside above path ddl, dml, and constraints folders are present inside which OBDX scripts will be present which needs to be executed.

2.1 Policy Seeding

TEMP_PATH=Temparory Path

cp ${OBDX INSTALLER}/installables/OBDX/<Installation type>/<version>/policies/Entitlement_log4j.properties to TEMP_PATH/db/Entitlement_log4j.properties

cp ${OBDX INSTALLER}/installables/OBDX/<Installation type>/<version>/policies /Task_log4j.properties to TEMP_PATH/db/Task_log4j.properties

cp ${OBDX INSTALLER}/installables/OBDX/<Installation type>/<version>/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=OBDX_${POST_FIX} and SCHEMA_PASS= Password of OBDX_${POST_FIX} .

$JAVA_HOME/bin/java -Djava.util.logging.config.file= TEMP_PATH/db/Task_log4j.properties -jar ${OBDX INSTALLER}/OBDX/<Installation type>/<version>/policies/com.ofss.digx.utils.feed.data.task.jar /installables/policies/Task.csv oracle.jdbc.OracleDriver SCHEMA_NAME SCHEMA_PASS 'jdbc:oracle:thin:@OBDX_DATABASE_HOSTNAME:OBDX_DATABASE_PORT/OBDX_DATABASE_SID'

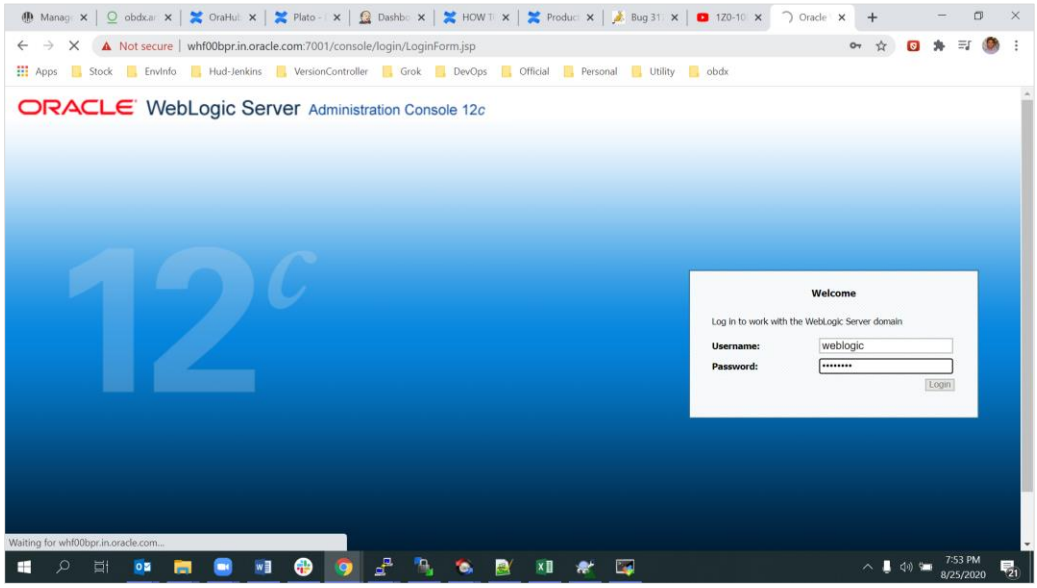
```
# $JAVA_HOME/bin/java -Djava.util.logging.config.file=
TEMP_PATH/db/Dashboard_seed_log4j.properties -jar ${OBDX
INSTALLER}/OBDX/<Installation type>/<version>/policies/com.ofss.digx.utils.dashboard.jar
${OBDX INSTALLER}/}/OBDX/<Installation type>/<version>/policies/dashboard_json/
oracle.jdbc.OracleDriver SCHEMA_NAME SCHEMA_PASS
'jdbc:oracle:thin:@OBDX_DATABASE_HOSTNAME:OBDX_DATABASE_PORT/OBDX_DATABA
SE_SID'

# $JAVA_HOME/bin/java -Djava.util.logging.config.file=
TEMP_PATH/db/Entitlement_log4j.properties -jar ${OBDX INSTALLER}/}/OBDX/<Installation
type>/<version>/policies/com.ofss.digx.utils.entitlement.feed.data.jar  ${OBDX
INSTALLER}/}/OBDX/<Installation type>/<version>/policies/Resources.csv ${OBDX
INSTALLER}/}/OBDX/<Installation type>/<version>/policies/Entitlement.csv ${OBDX
INSTALLER}/}/OBDX/<Installation type>/<version>/policies/Day0Policy.csv KERNEL
oracle.jdbc.OracleDriver SCHEMA_NAME SCHEMA_PASS
'jdbc:oracle:thin:@OBDX_DATABASE_HOSTNAME:OBDX_DATABASE_PORT/OBDX_DATABA
SE_SID'
```

3. WEBLOGIC Setup and Configuration

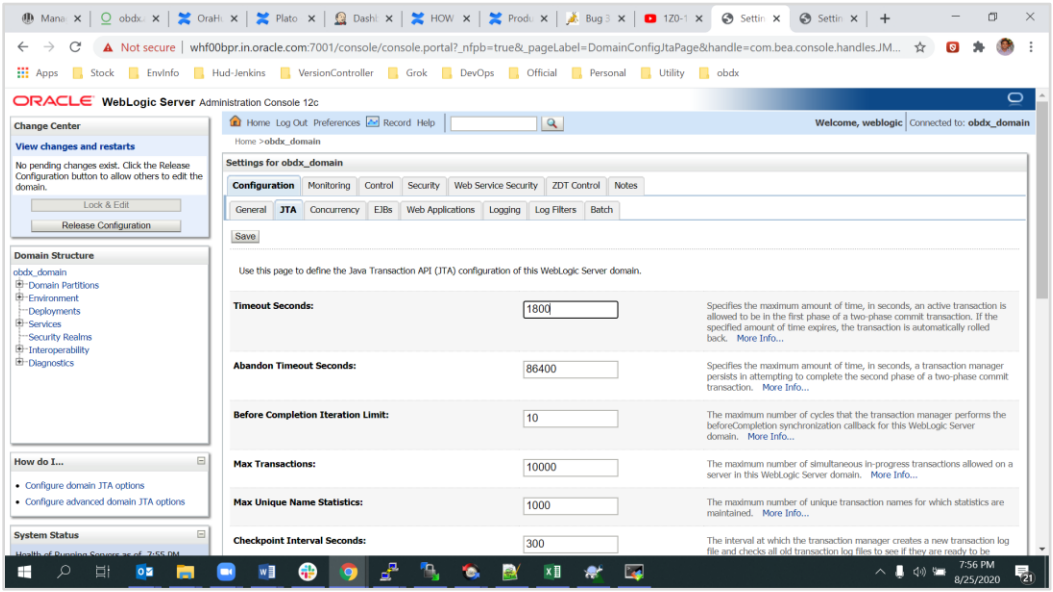
Once OBDX and EHMS schema created, weblogic domain created, managed server, cluster and node manager configured, proceed with below steps.

3.1 Setting Domain JTA Transaction timeout

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



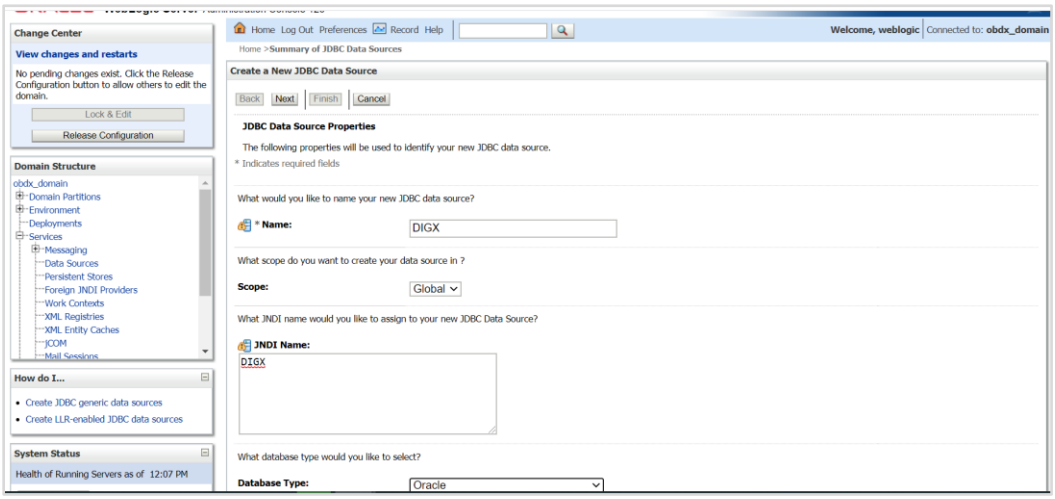
2. click on DOMAIN_NAME→ JTA→ set Timeout Seconds to 1800 → click on save → Activate changes



Please verify once if below datasources are already created post 22.2.0.0.0 base installation and if present proceed to JMS Server and JMS Module creation, if not created proceed with below steps.

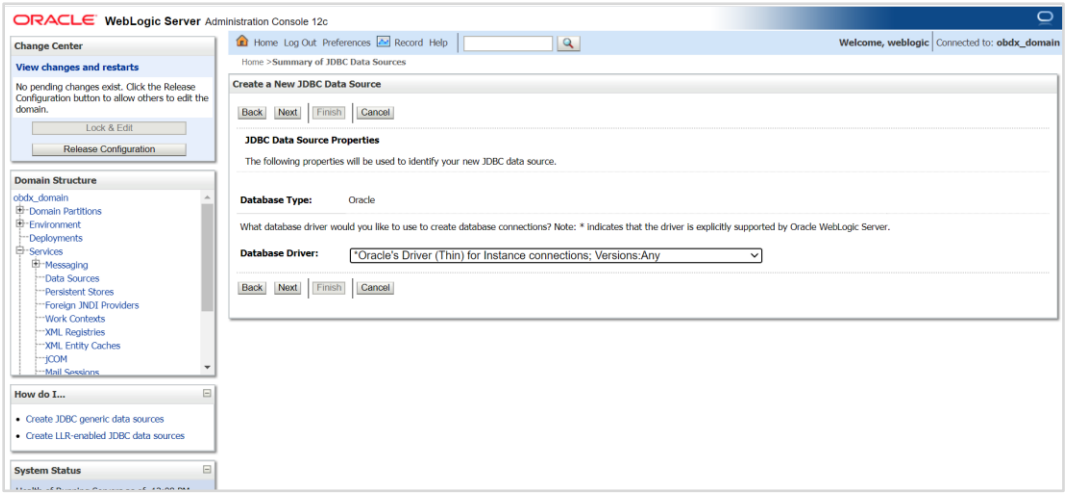
3.2 Creating DIGX data source

1. Navigate to Data Source → click on new → Provide details and click on finish.

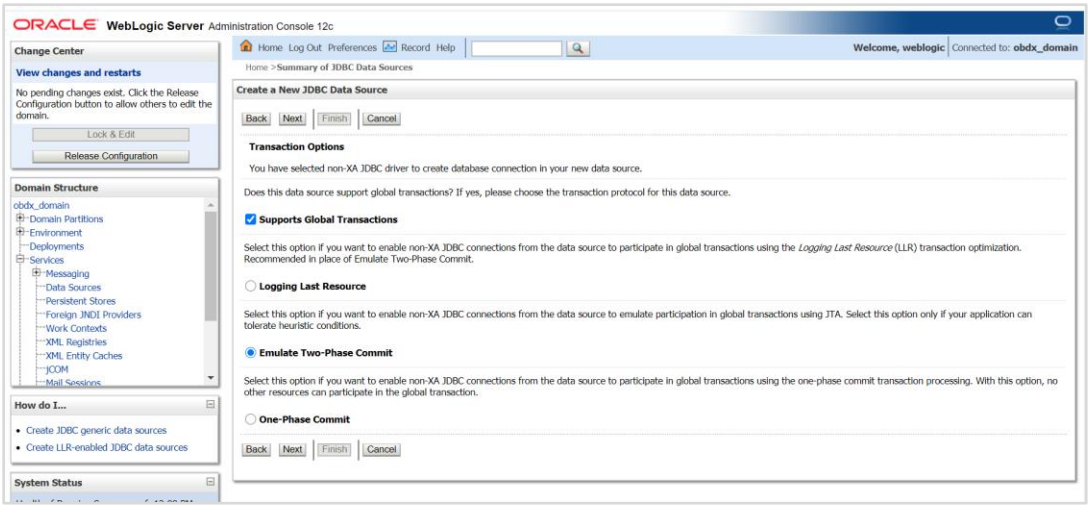


2. **Name: -** DIGX

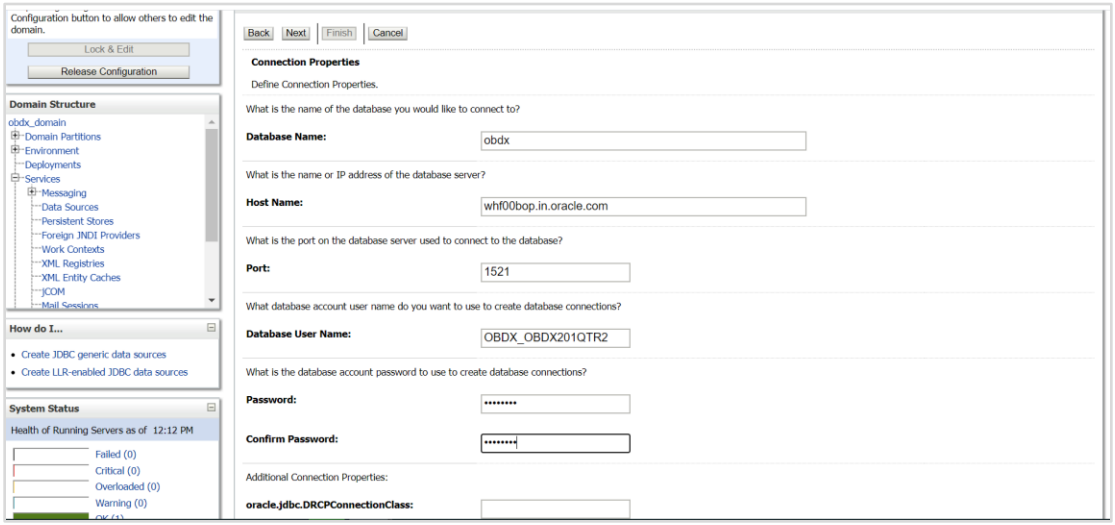
JNDI Name: - DIGX



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



4. Select Emulate Two-Phase Commit



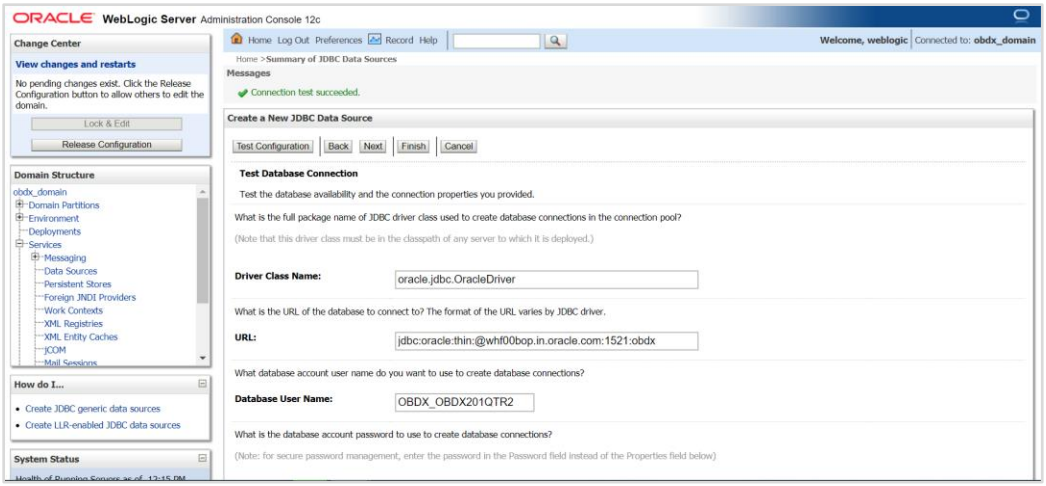
5. Provide

Database Name: - Database SID

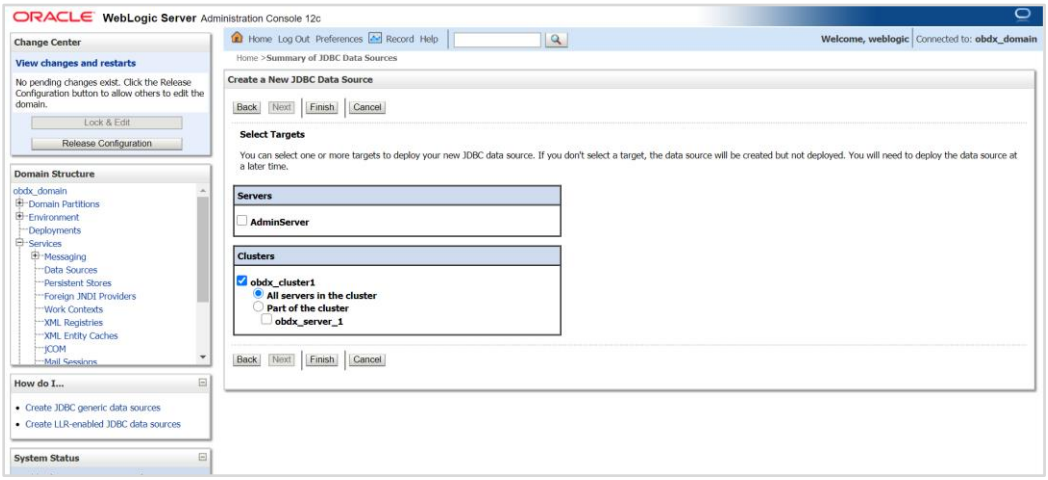
Host Name: - Database hostname

Port: - Database port Number

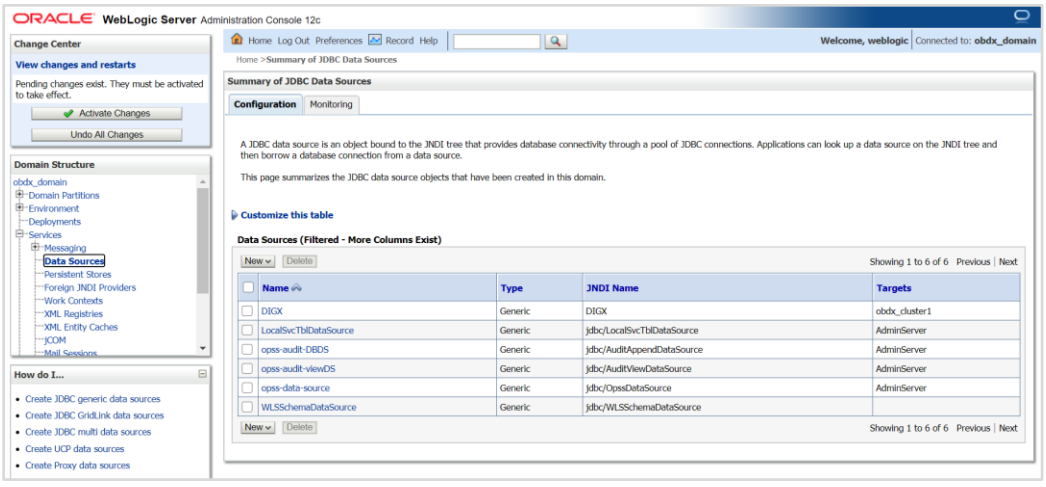
Database user Name: - OBDX_${POST_FIX}



6. Test Configuration

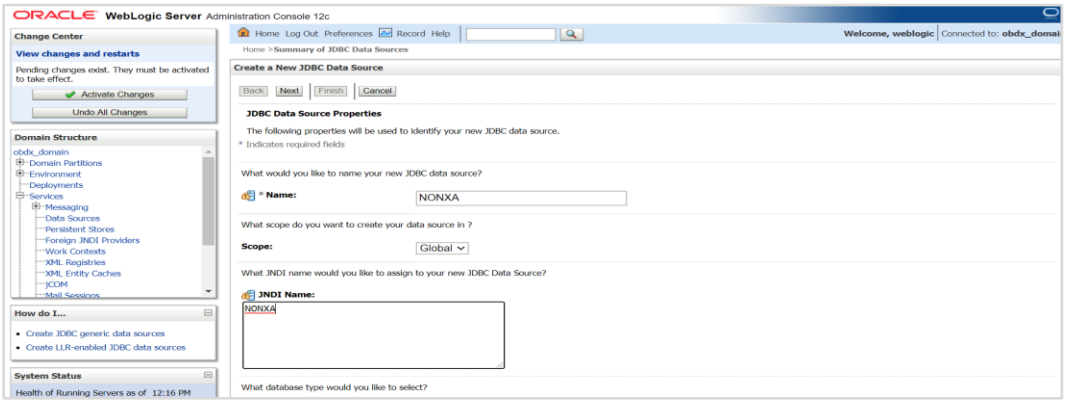


7. Target to cluster



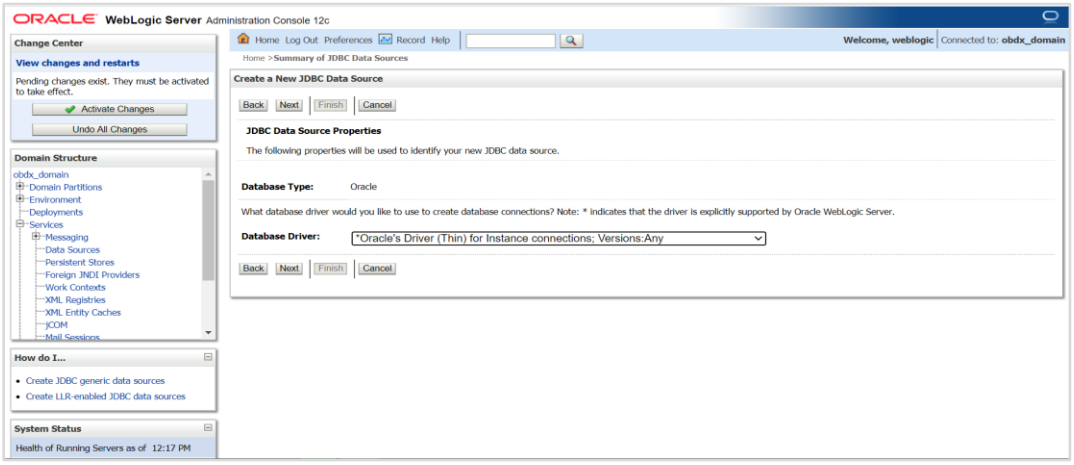
3.3 Creating NONXA data source

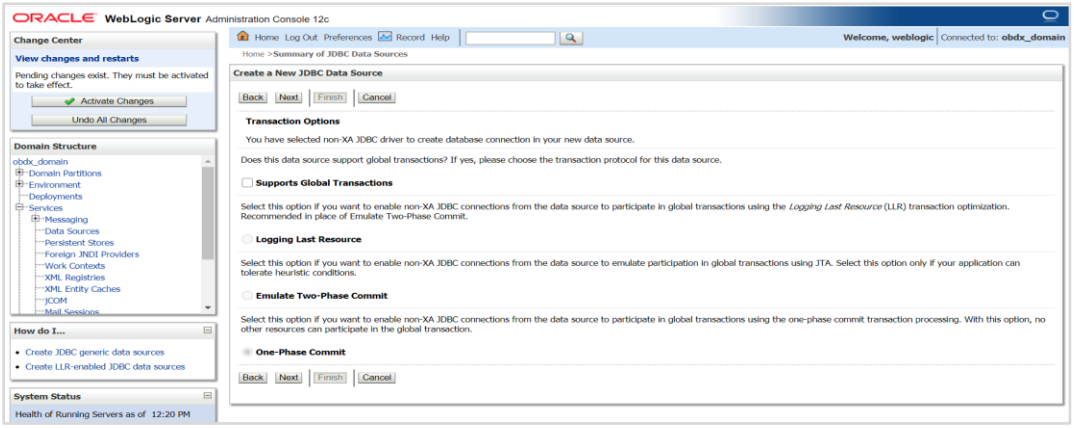
1. Navigate to Data Source → click on new → Provide details and click on finish



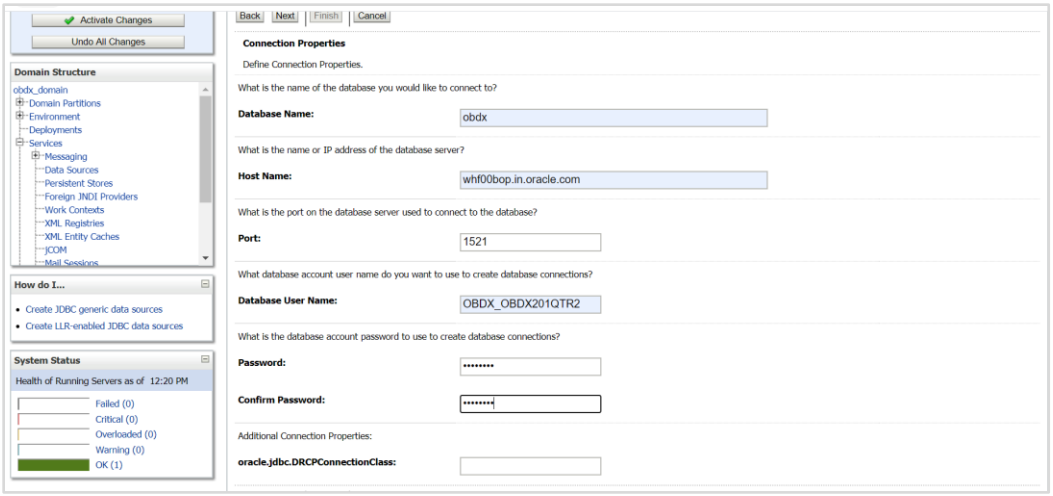
2. **Name :-** NONXA

JNDI Name :- NONXA





3. Click **Next**



4. Provide

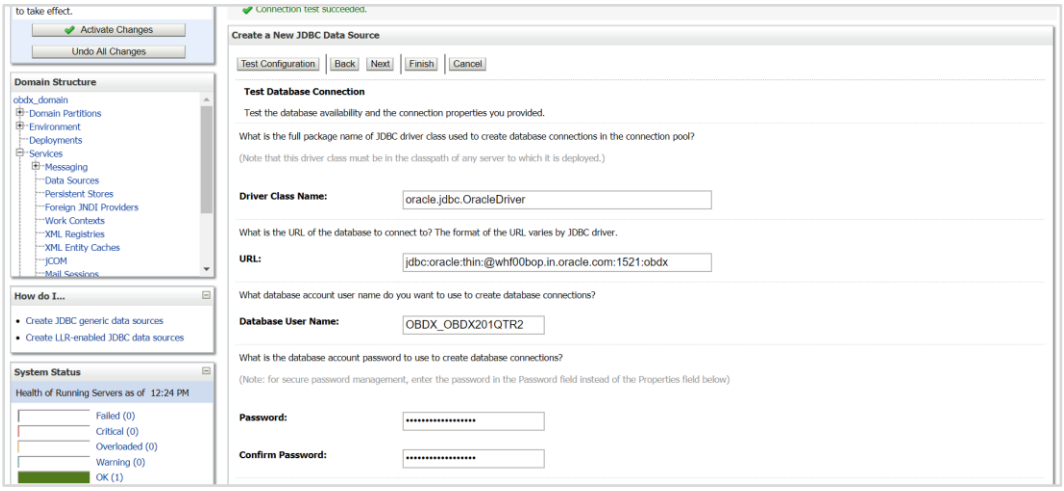
Database Name: - Database SID

Host Name: - Database hostname

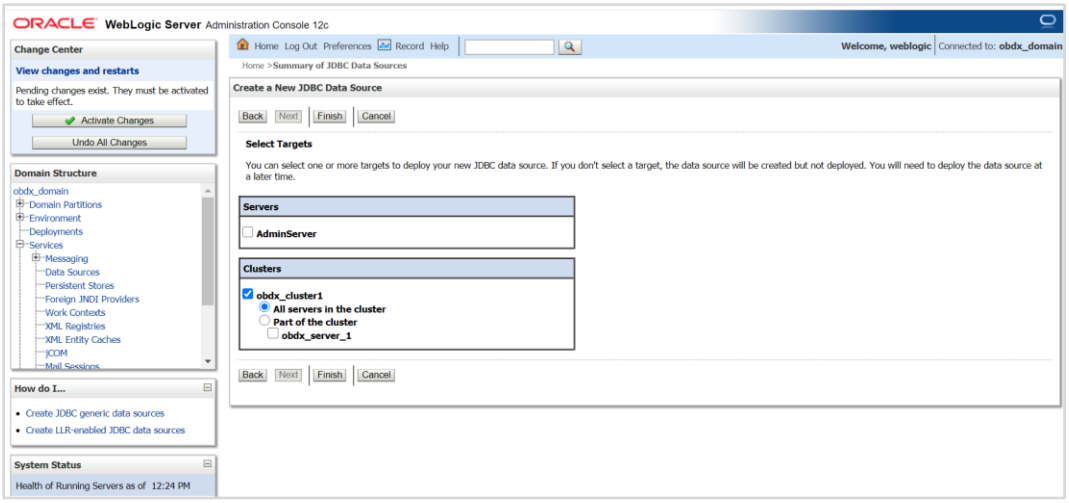
Port: - Database port Number

Database user Name: - OBDX_${POST_FIX}

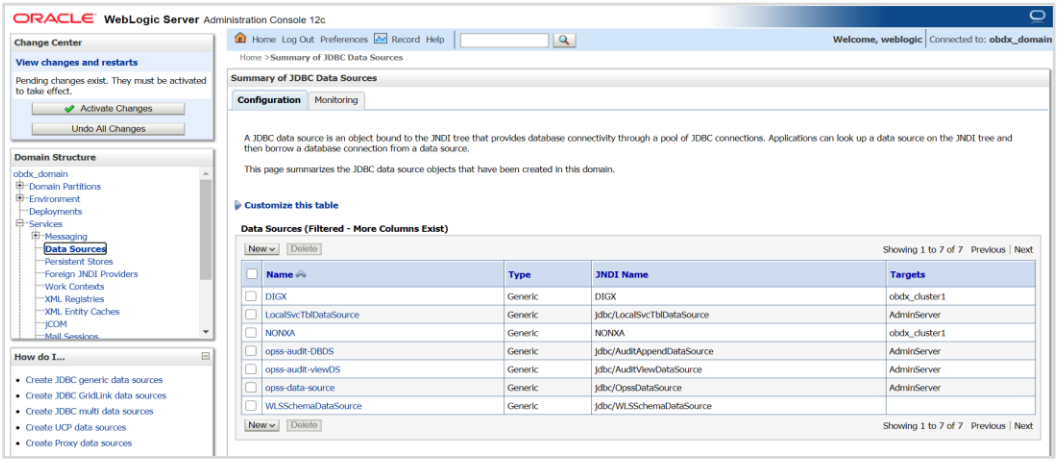
Password:- Database user password



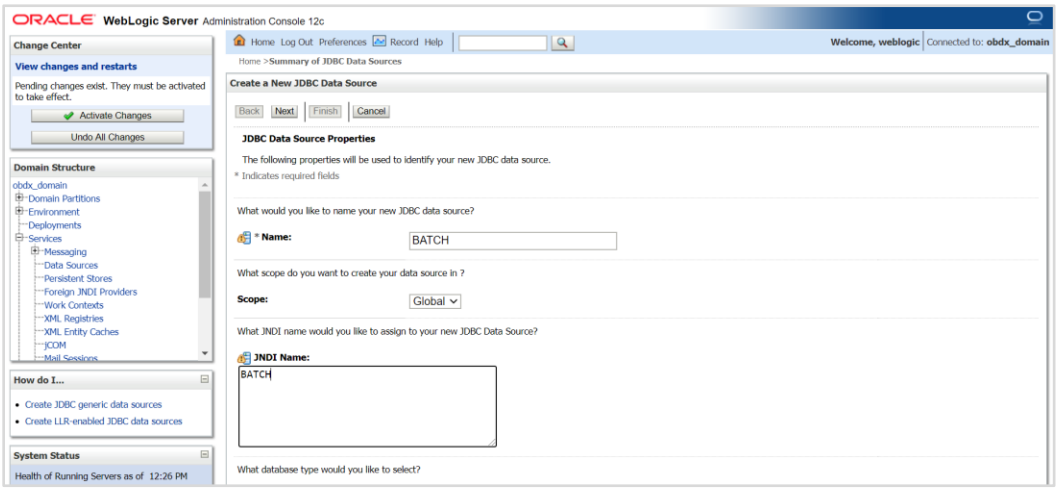
5. Test Configuration



6. Select target as cluster -- > Finish

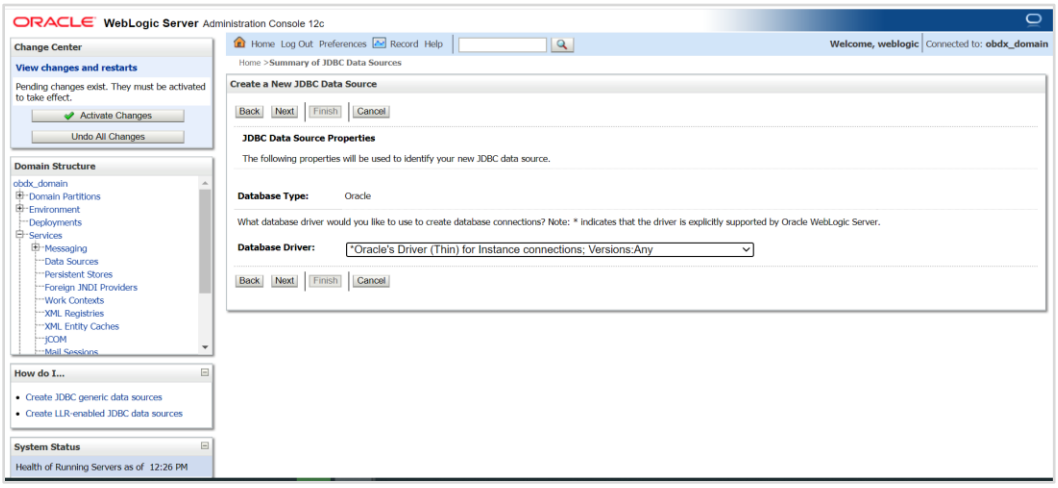


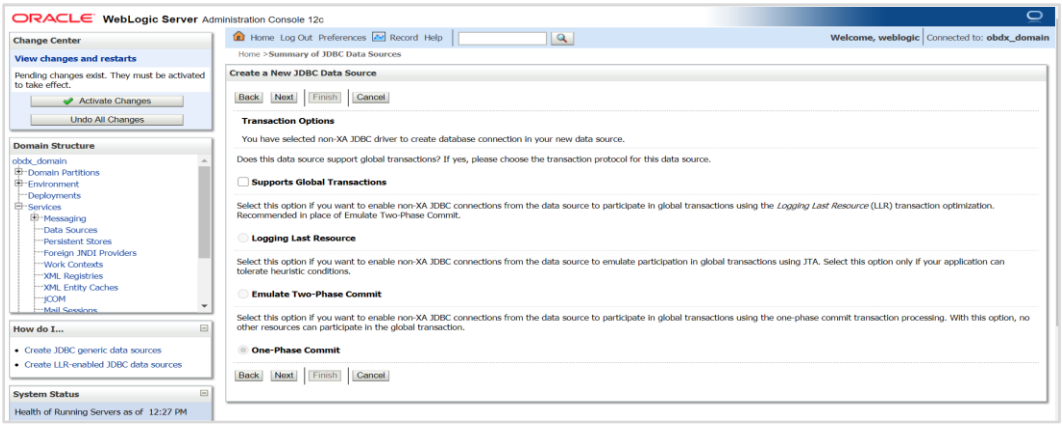
3.4 Creating BATCH data source



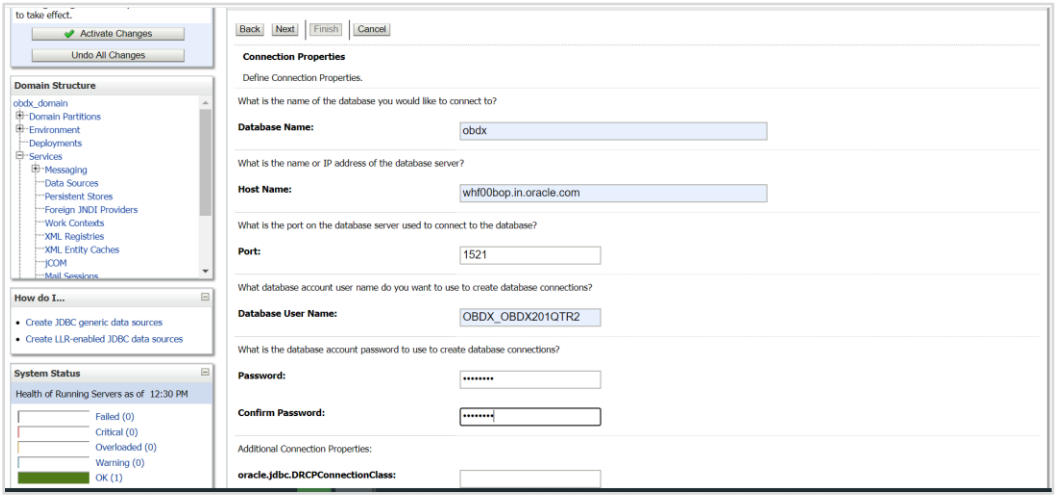
7. **Name :-** BATCH

JNDI Name :- BATCH





8. Click **Next**



9. **Provide**

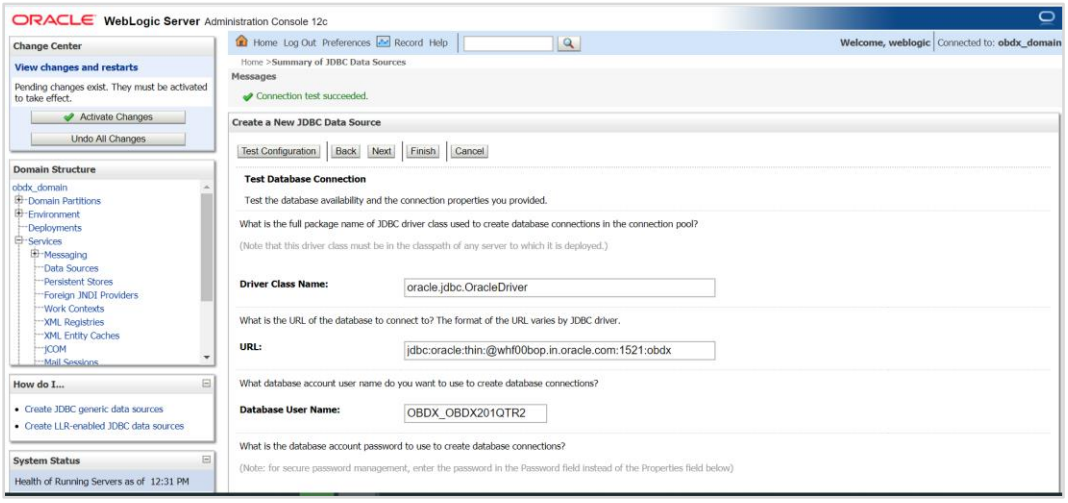
Database Name: - Database SID

Host Name: - Database hostname

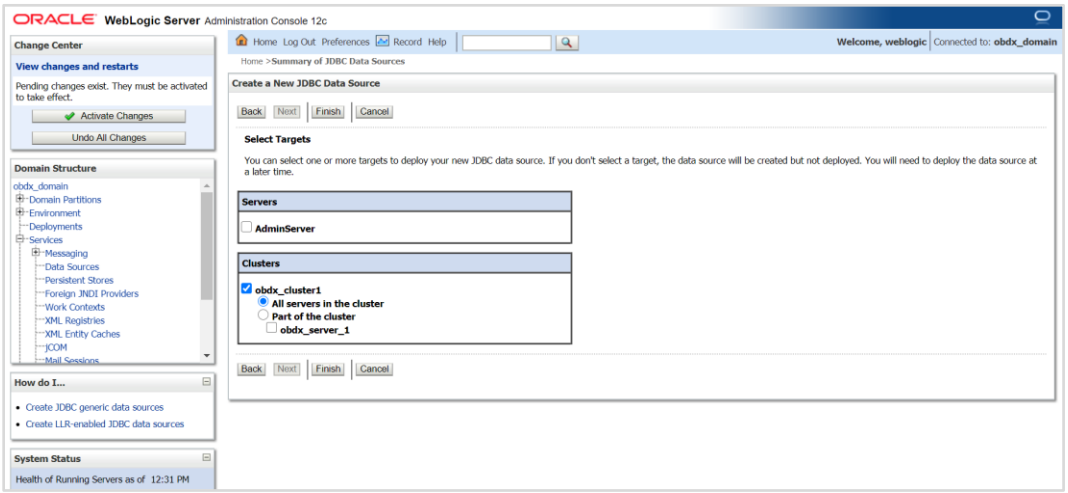
Port: - Database port Number

Database user Name: - OBDX_${POST_FIX}

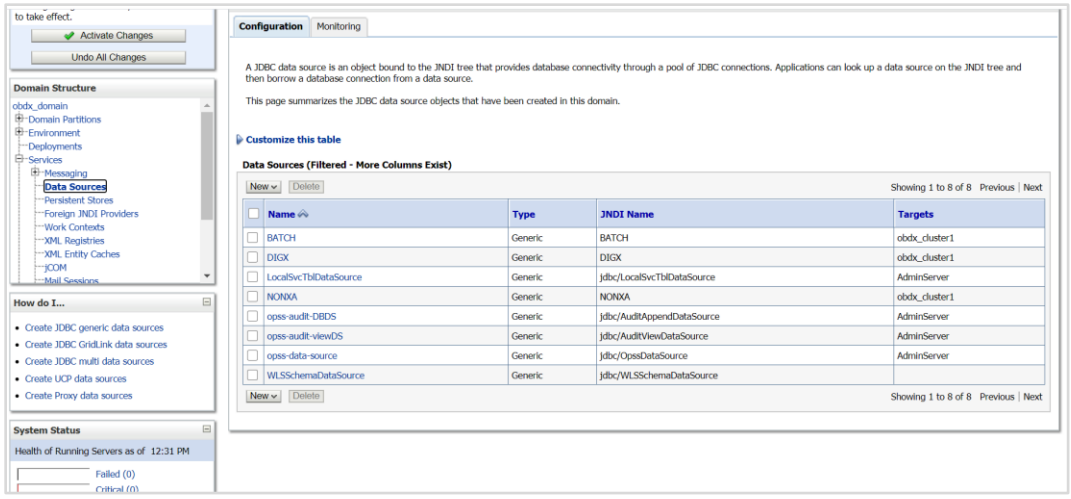
Password:- Database user password



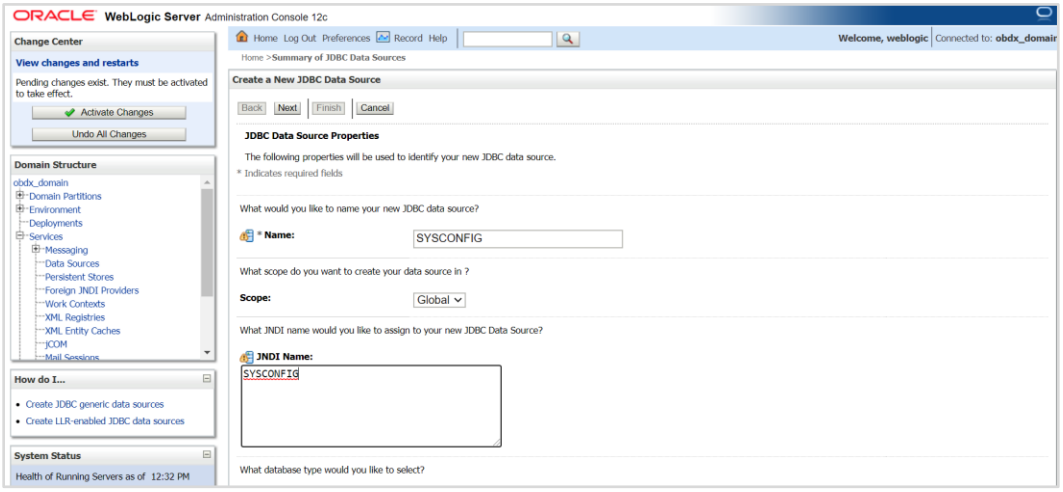
10. Test Configuration



11. Target Cluster and click on Finish

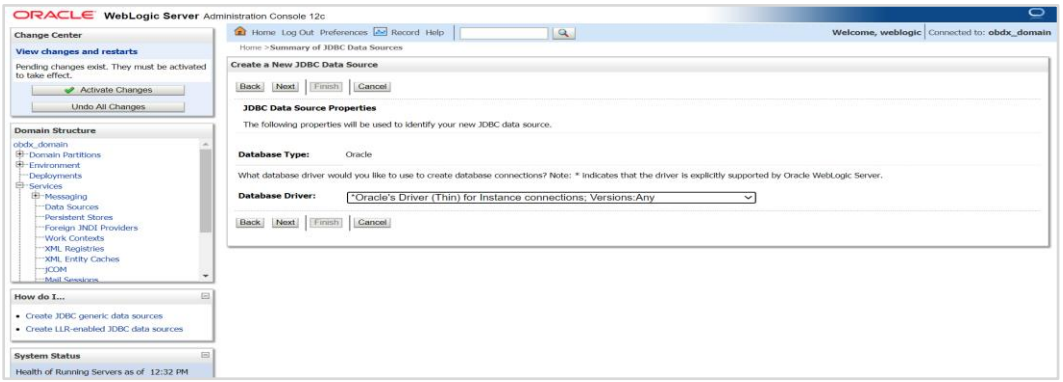

to take effect.
Activate Changes
Undo All Changes
Domain Structure
obdx_domain
Domain Partitions
Environment
Deployments
Services
Messaging
Data Sources
Persistent Stores
Foreign JNDI Providers
Work Contexts
XML Registries
XML Entity Caches
jCOM
Mail Sessions
How do I...
Create JDBC generic data sources
Create JDBC GridLink data sources
Create JDBC multi data sources
Create UCP data sources
Create Proxy data sources
System Status
Health of Running Servers as of 12:31 PM
Failed (0)
Critical (0)
Configuration
Monitoring
A JDBC data source is an object bound to the JNDI tree that provides database connectivity through a pool of JDBC connections. Applications can look up a data source on the JNDI tree and then borrow a database connection from a data source.
This page summarizes the JDBC data source objects that have been created in this domain.
Customize this table
Data Sources (Filtered - More Columns Exist)
New
Delete
Showing 1 to 8 of 8 Previous | Next
Name
Type
JNDI Name
Targets
BATCH Generic BATCH obdx_cluster1
DIGX Generic DIGX obdx_cluster1
LocalSvcTblDataSource Generic jdbc/LocalSvcTblDataSource AdminServer
NONXA Generic NONXA obdx_cluster1
opss-audit-DBDS Generic jdbc/AuditAppendDataSource AdminServer
opss-audit-viewDS Generic jdbc/AuditViewDataSource AdminServer
opss-data-source Generic jdbc/OpssDataSource AdminServer
WLSSchemaDataSource Generic jdbc/WLSSchemaDataSource
New
Delete
Showing 1 to 8 of 8 Previous | Next

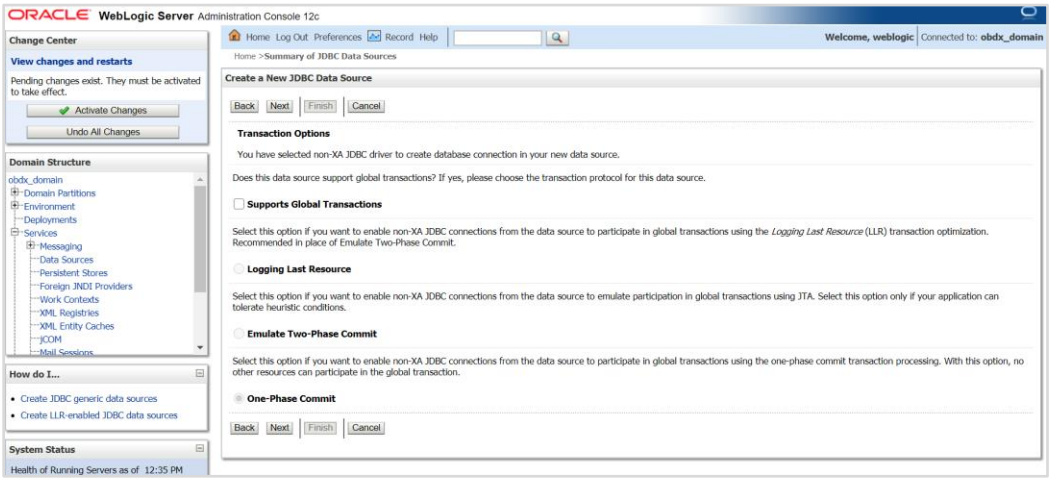
3.5 Creating SYSCONFIG data source



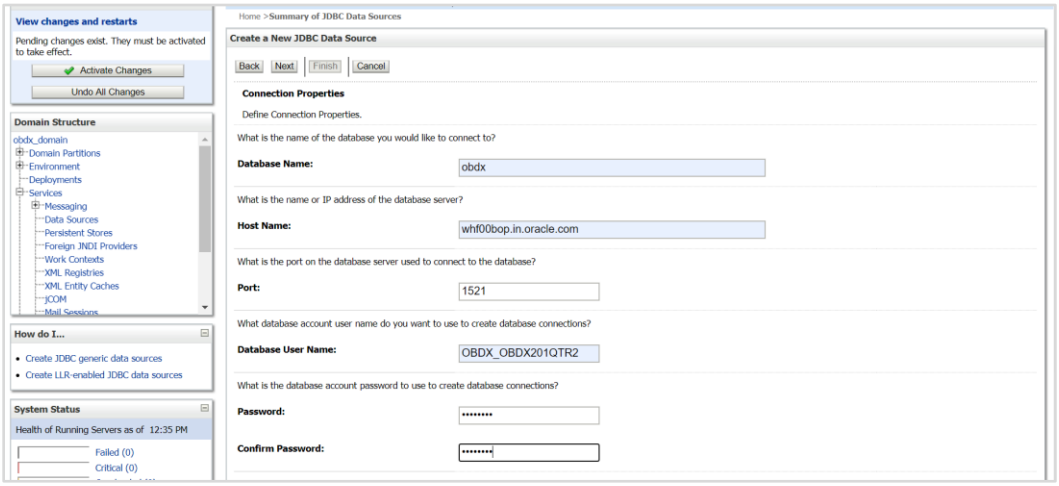
12. **Name :-** SYSCONFIG

JNDI Name :- SYSCONFIG





13. Click on Next



14. Provide

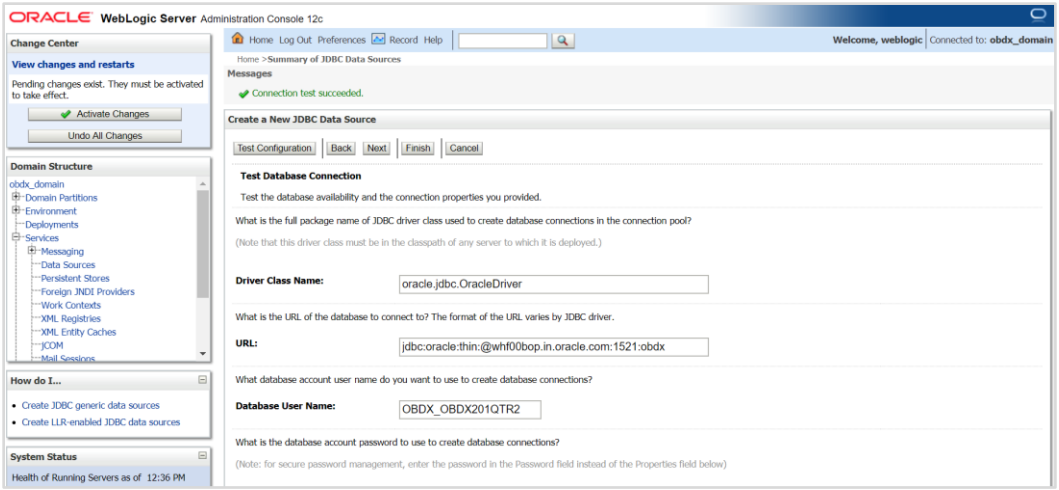
Database Name: - Database SID

Host Name: - Database hostname

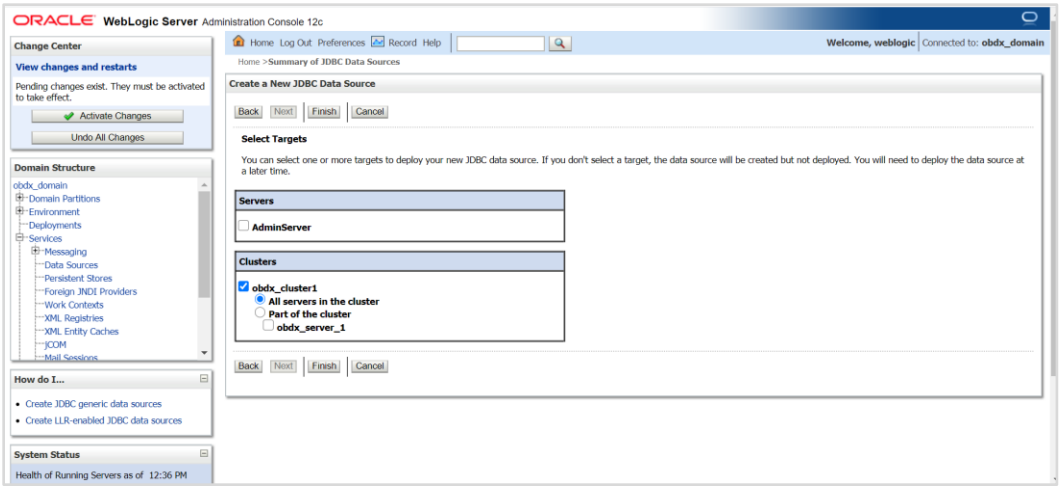
Port: - Database port Number

Database user Name: - OBDX_${POST_FIX}

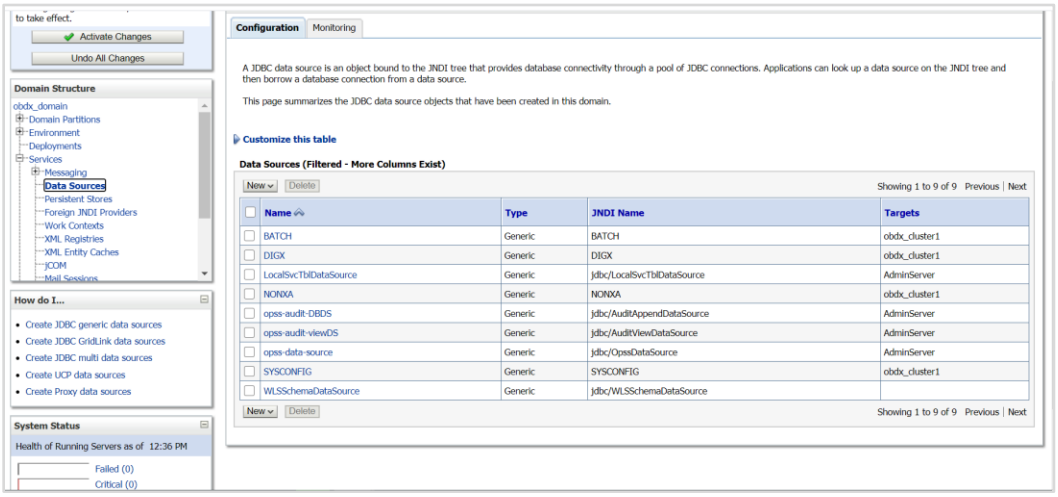
Password:- Database user password



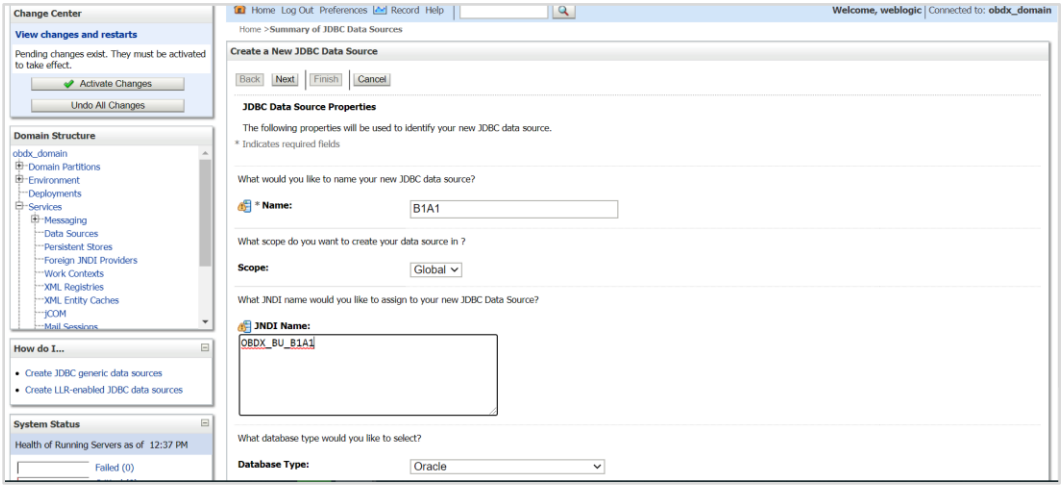
15. Test Configuration



16. Select target as cluster and click on Finish

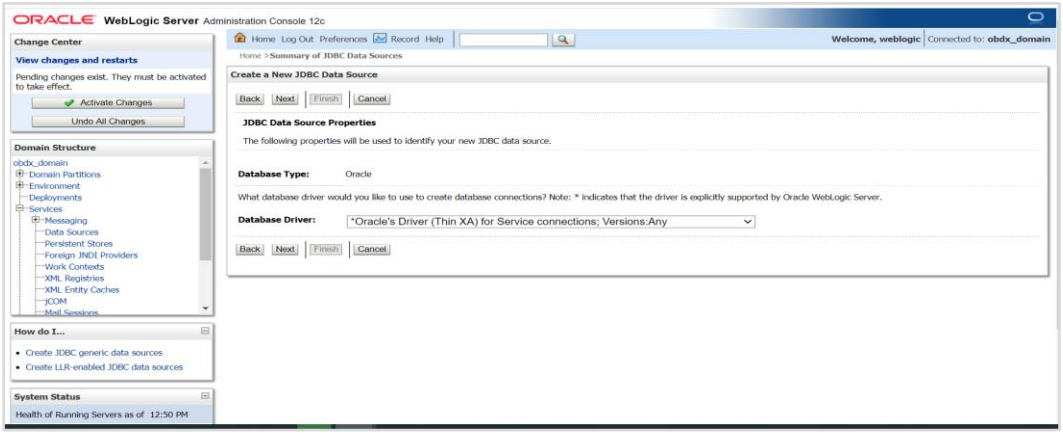


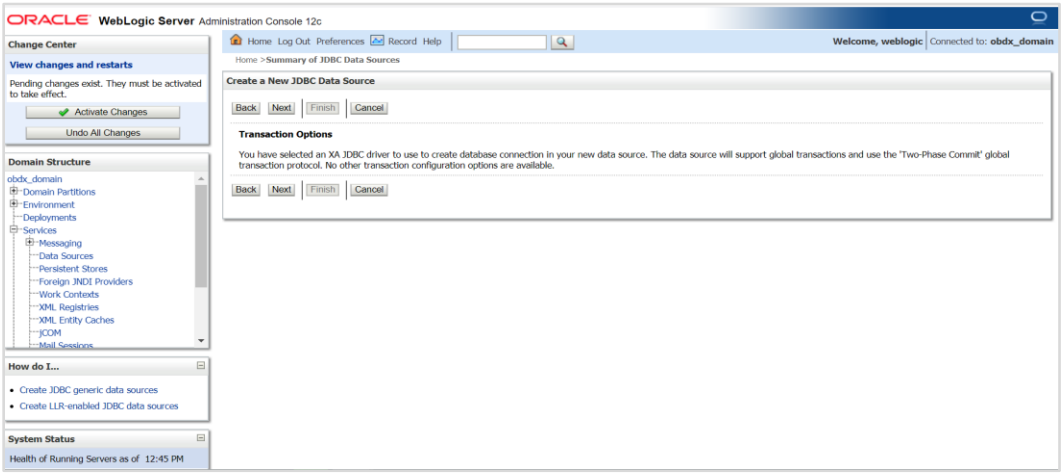
3.6 Creating B1A1 data source



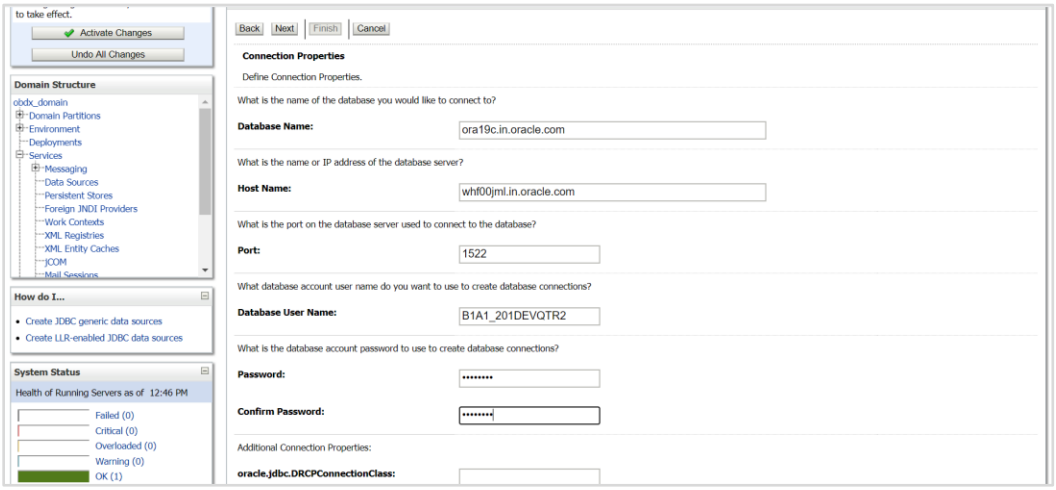
17. **Name:-** B1A1

JNDI Name :- OBDX_BU_B1A1





18. Click on Next



19. Provide

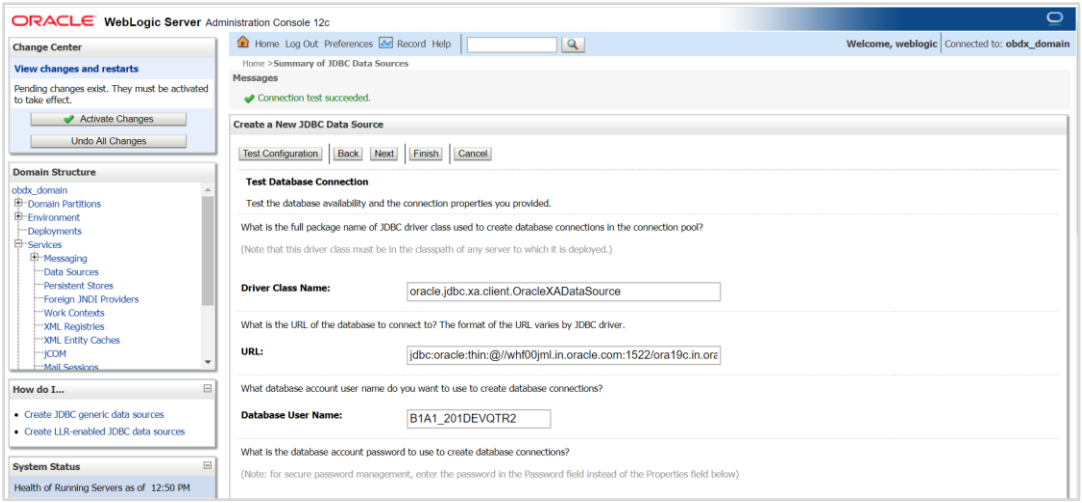
Database Name: - Database SID ($EHMS_DATABASE_SID)

Host Name: - Database hostname ($EHMS_DATABASE_HOSTNAME)

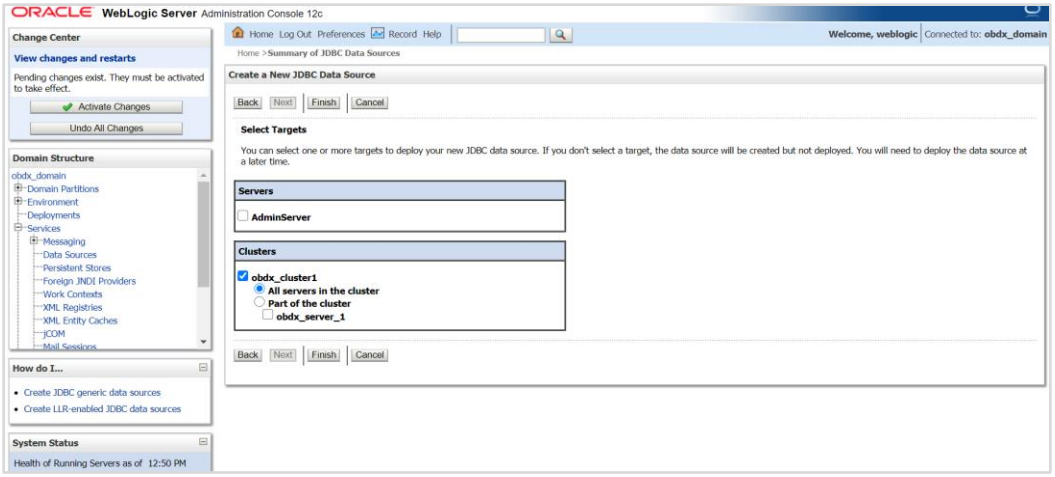
Port: - Database port Number ($EHMS_DATABASE_PORT)

Database user Name: - ${ EHMS_SCHEMA_NAME }

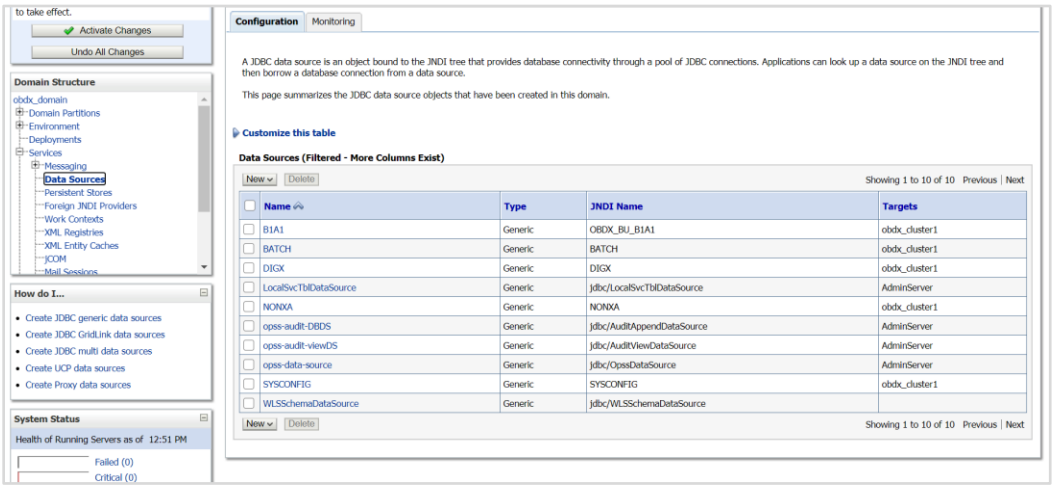
Password: - Database user ${ EHMS_SCHEMA_NAME } password



20. Test Configuration



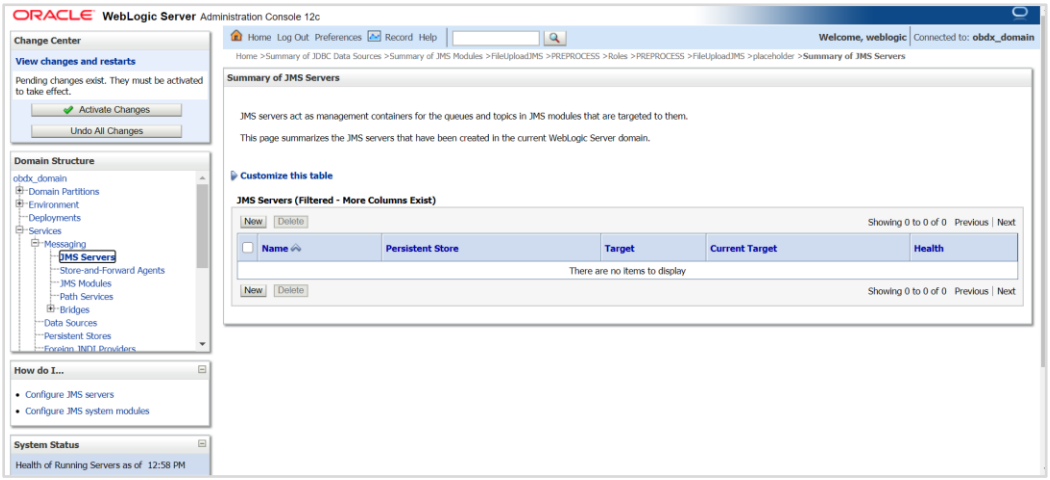
21. Set target as cluster and click on Finish

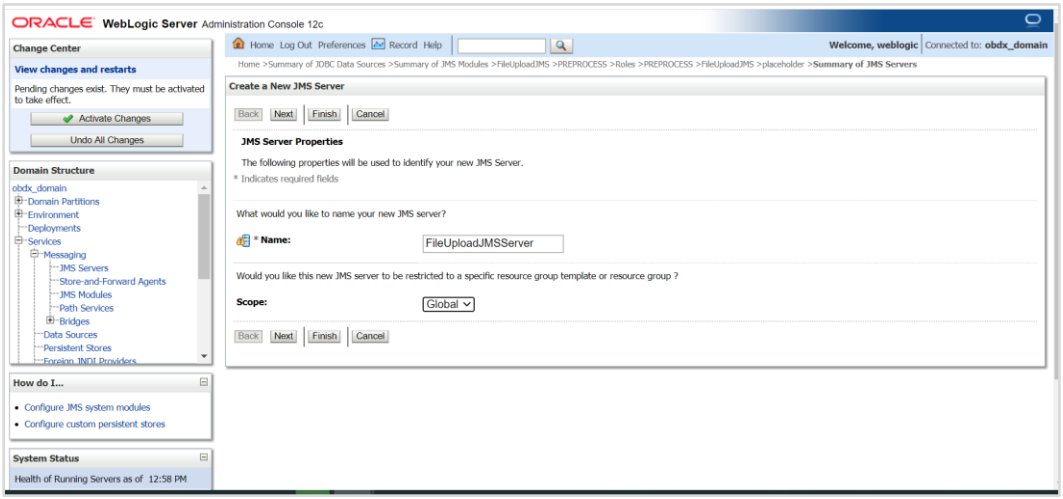


Before starting with below step please verify if below mentioned JMS Servers and Modules are present, if not please refer to jms.xml file present in path - OBDX_Installer\installables\OBDX\<Installation type>\<version>/ config/xml/jms

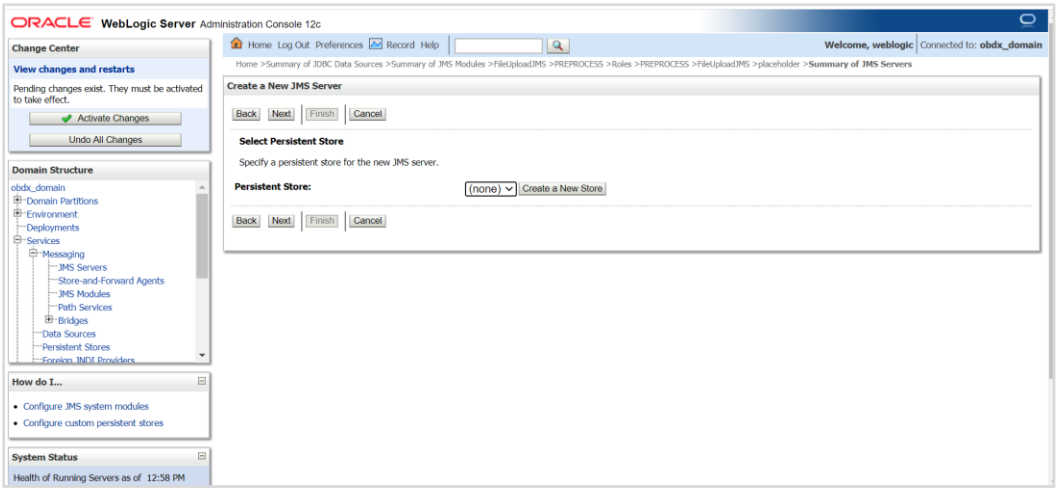
3.7 Create JMS server and JMS Module

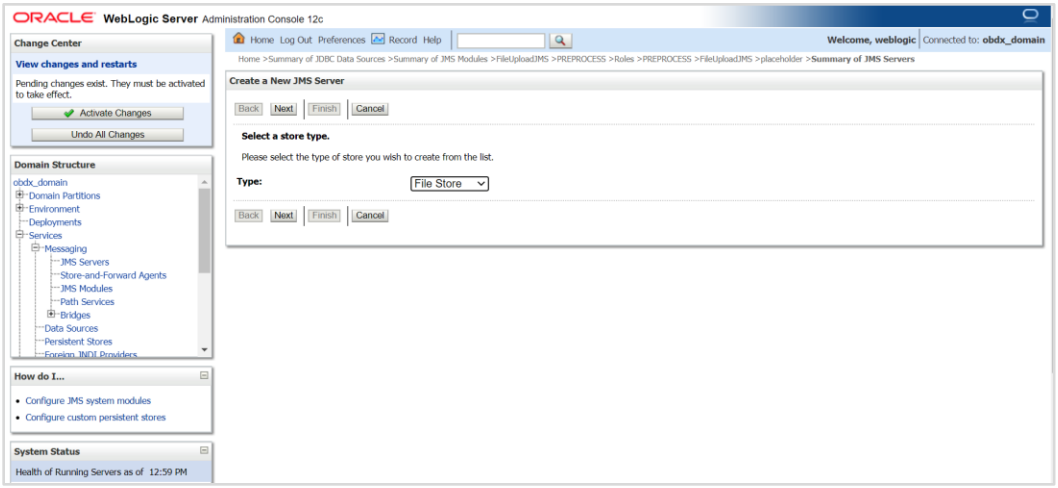
- Creating FileUploadJMS JSM Module
- Creating WLS_JMS_FILEUPLOAD_PS FileStore
- Creating FileUploadJMSServer JMS Server



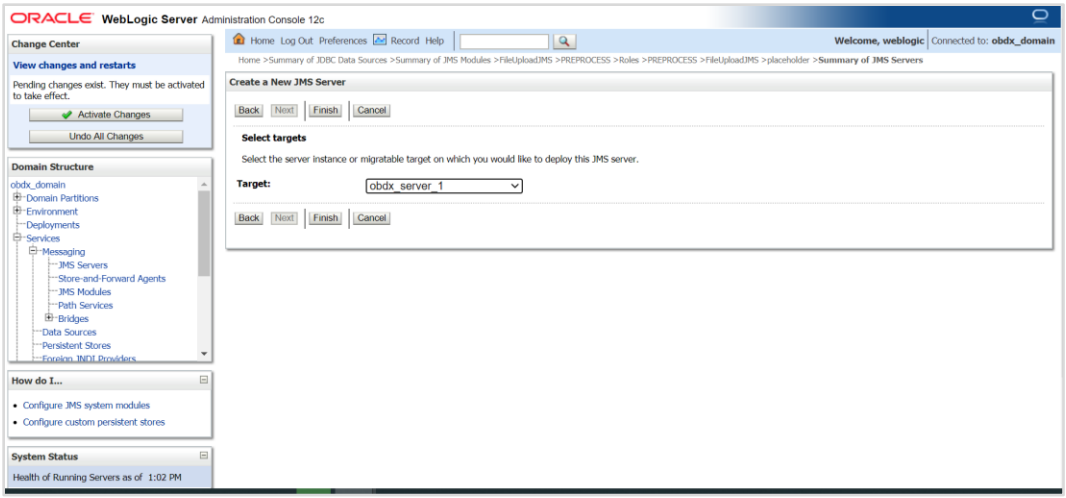


22. Click on JMS Servers → Name – FileUploadJMSServer -- > Click on Next

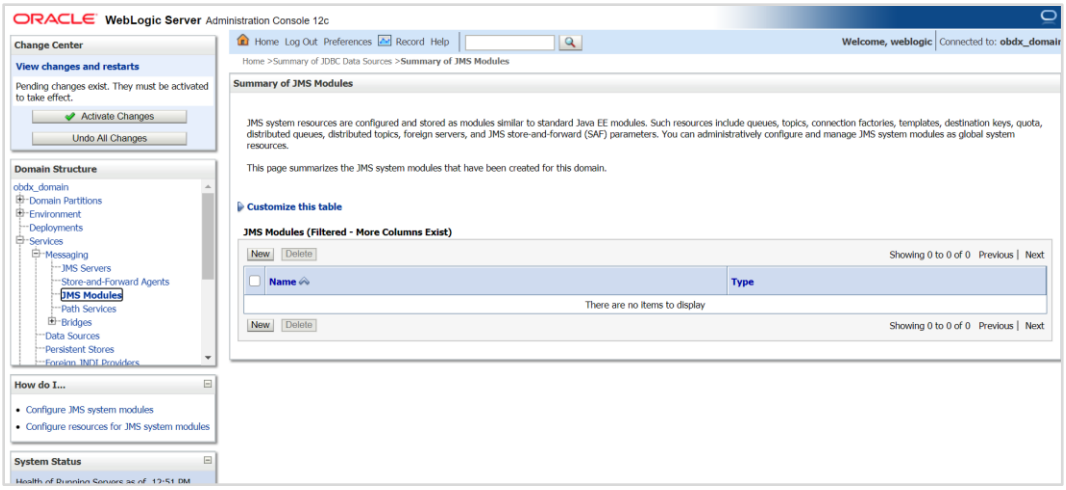




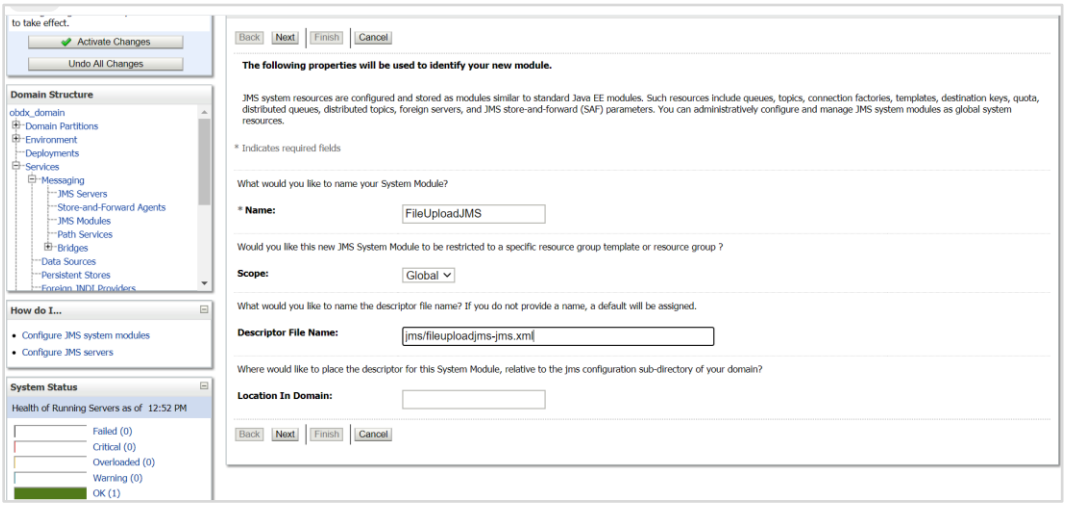
23. Select Type as File Store and click on Next



24. Select target as managed server and click on Finish



25. Left hand side click on JMS Module -- click on New

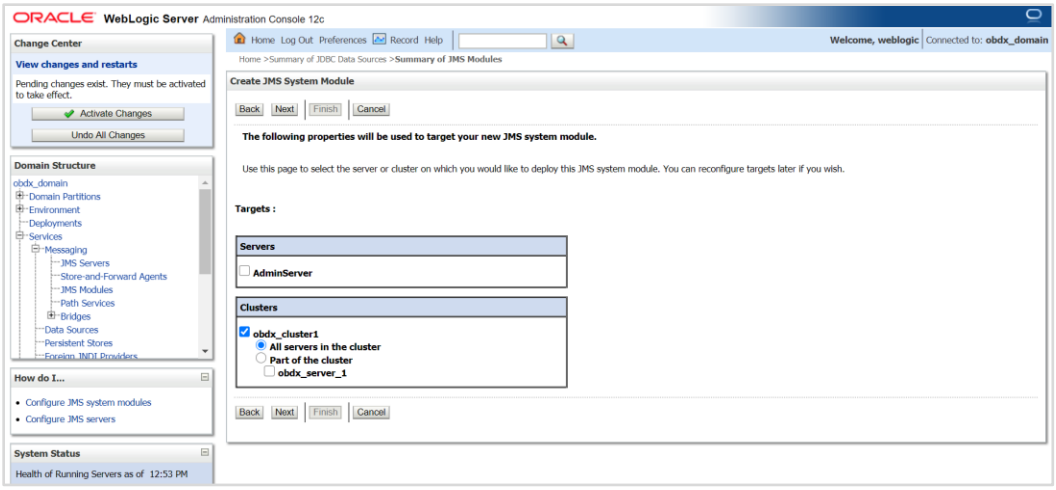


26. **Name :-** FileUploadJMS

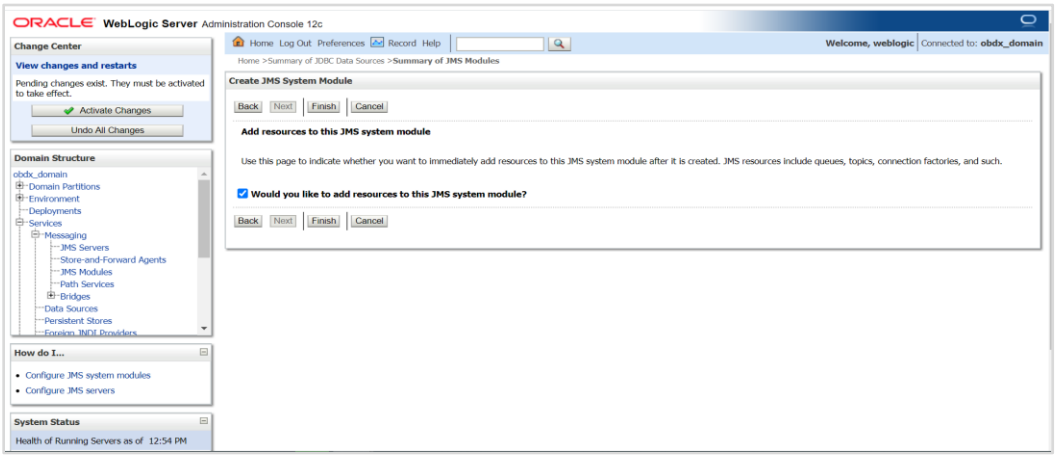
    Scope:- Global

    Descriptor File Name:- jms/fileuploadjms-jms.xml

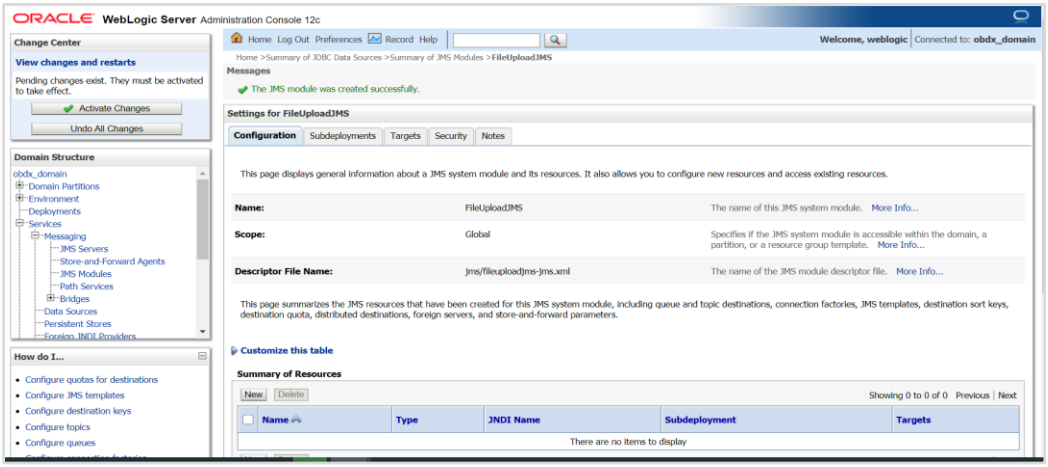
27. Click on Next



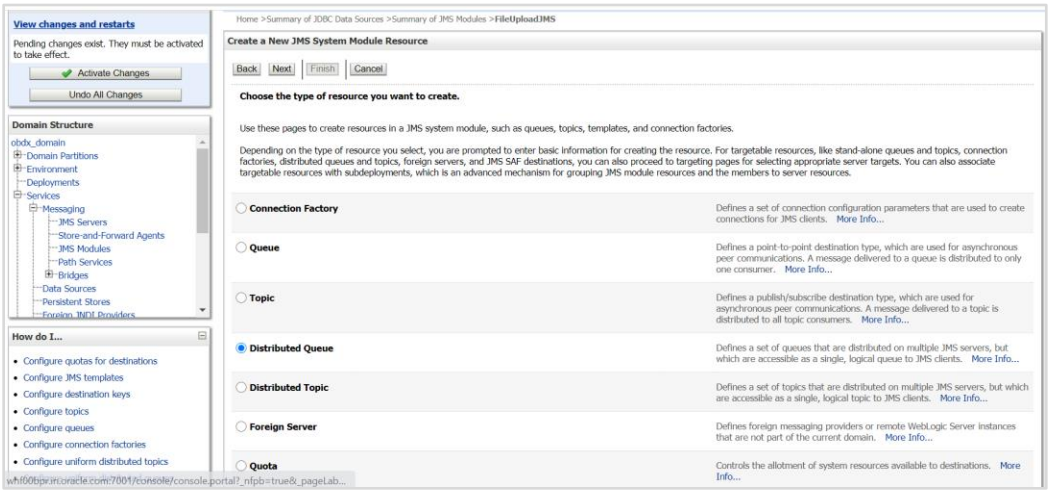
28. Set target as cluster → click on Next



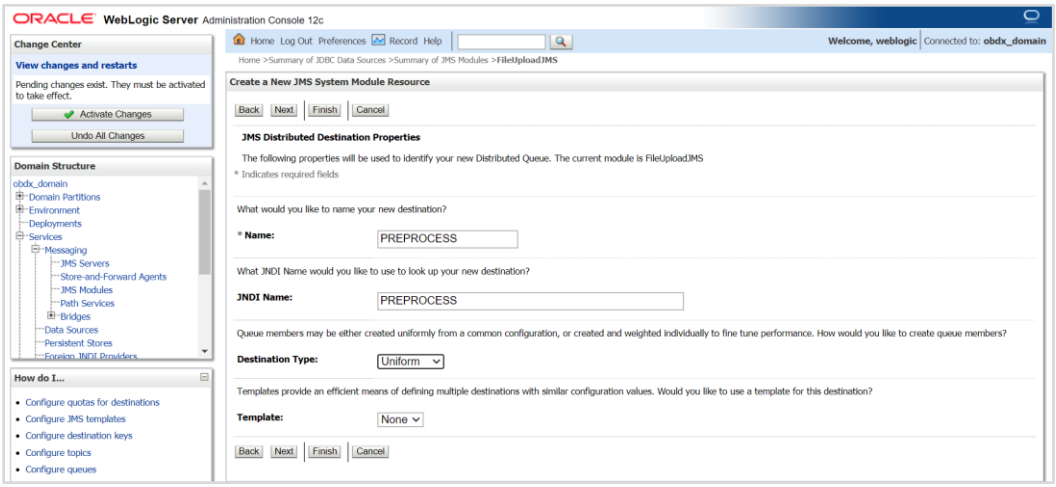
29. Select Would you like to add resources to this JMS system module and click on finish



30. Select new



31. Select Distributed Queue and click next



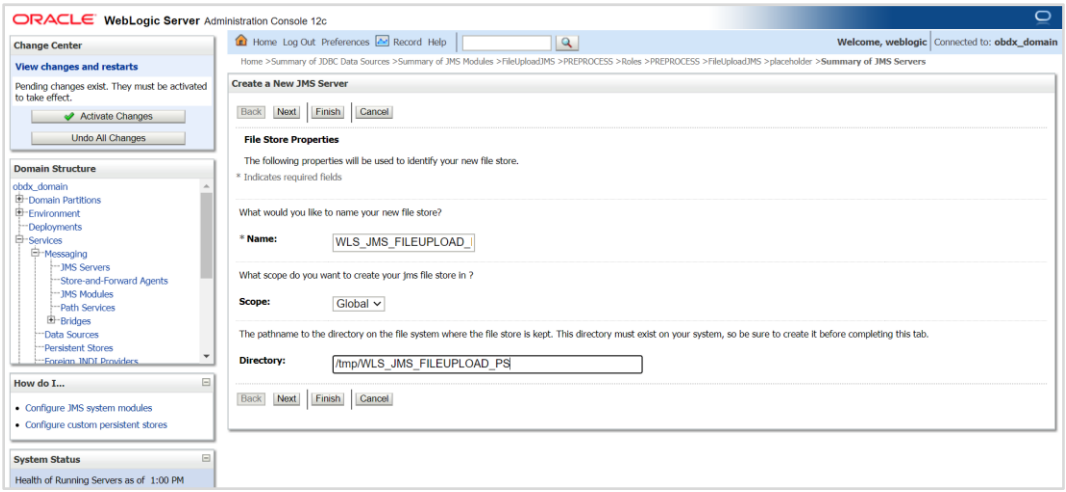
32. Provide

Name: - PREPROCESS

JNDI Name: - PREPROCESS

Destination Type: - Uniform

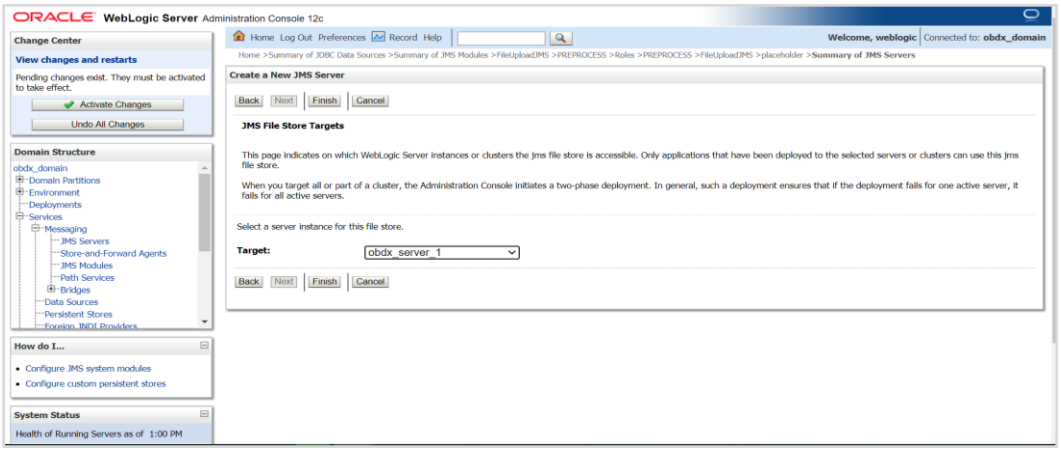
Template: - None



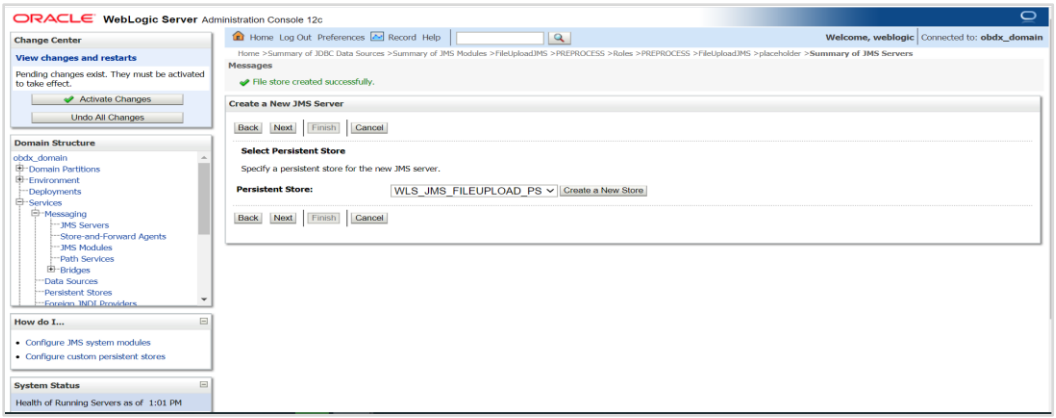
33. **Name :-** WLS_JMS_FILEUPLOAD_PS

    Scope :- Global

    Directory :- /tmp/WLS_JMS_FILEUPLOAD_PS

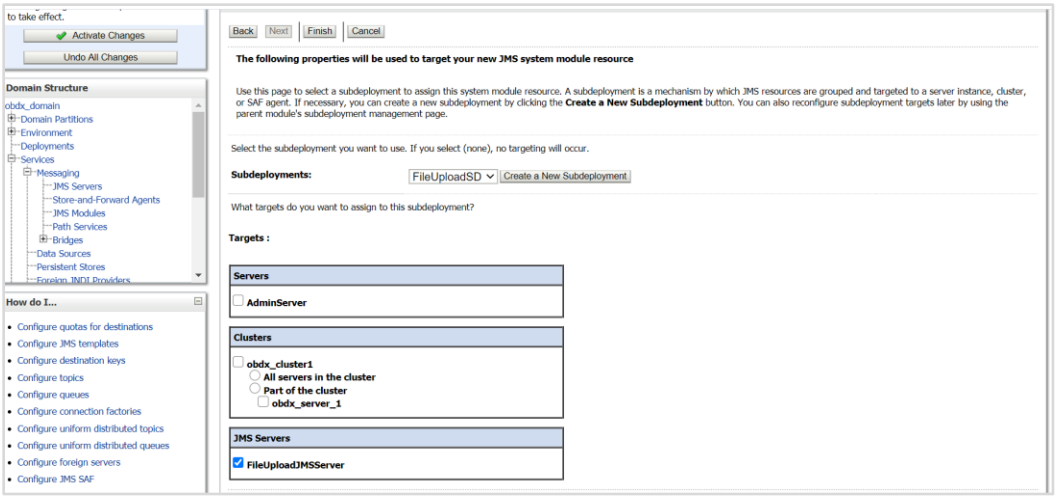


34. Select target as managed server

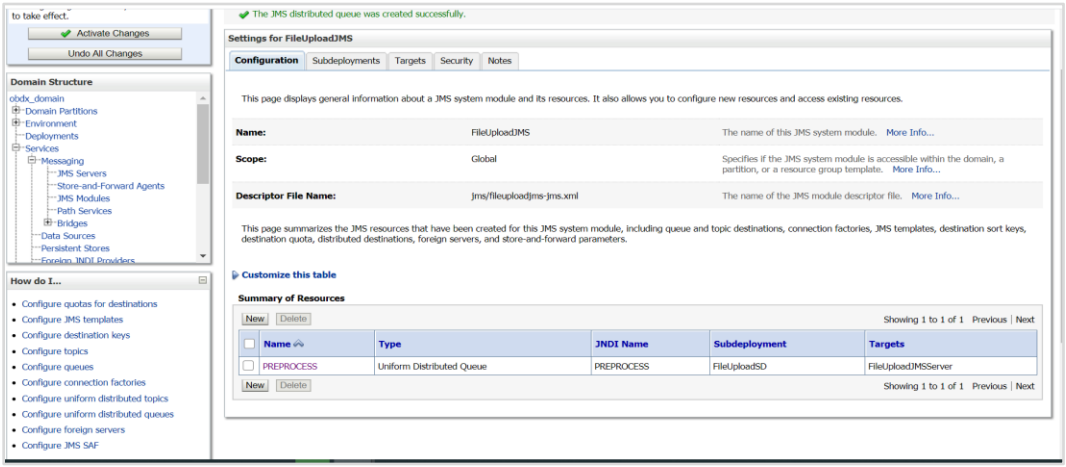


35. Select WLS_JMS_FILEUPLOAD_PS and click on Next

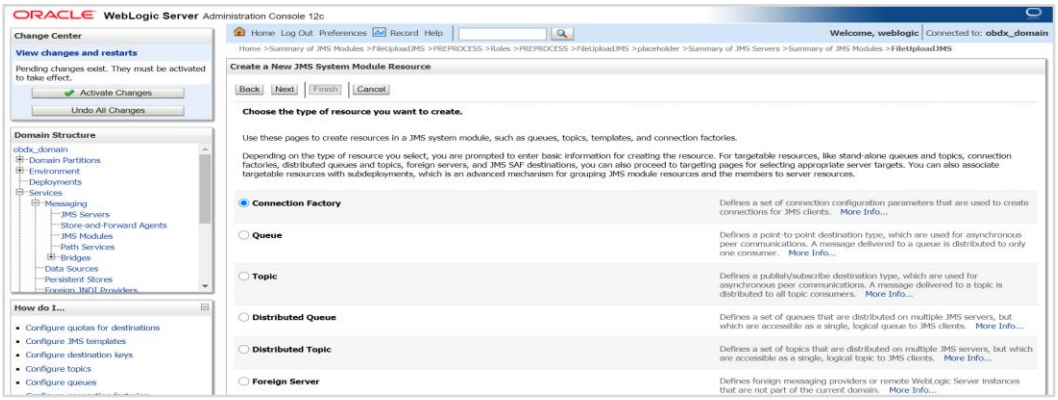
36. Select Create a New Subdeploymeny and create FileUploadSD



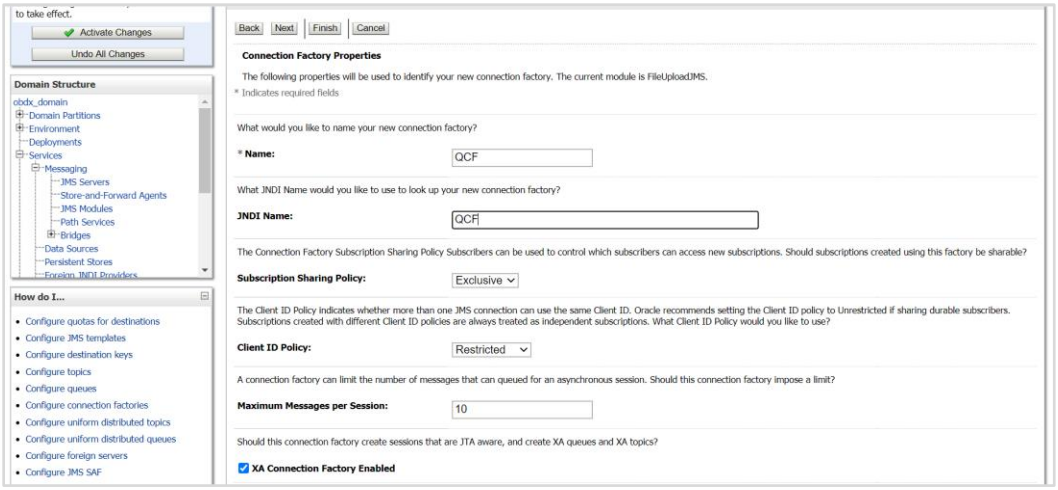
37. Select FileUploadJMSServer and click on Finish



38. Similarly Go into FileuploadJMS module and click on Next



39. Select Connection factory –> Click Next



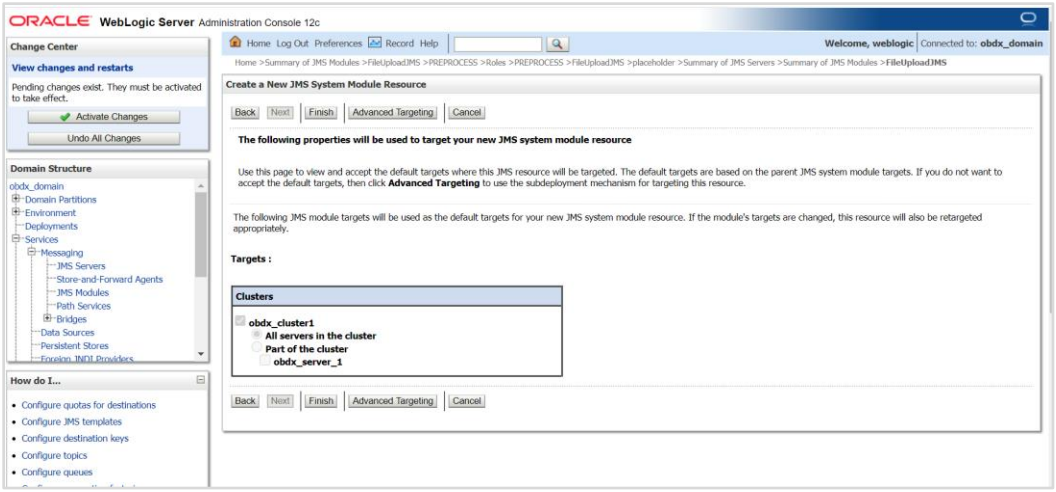
40. Provide

Name :- OCF

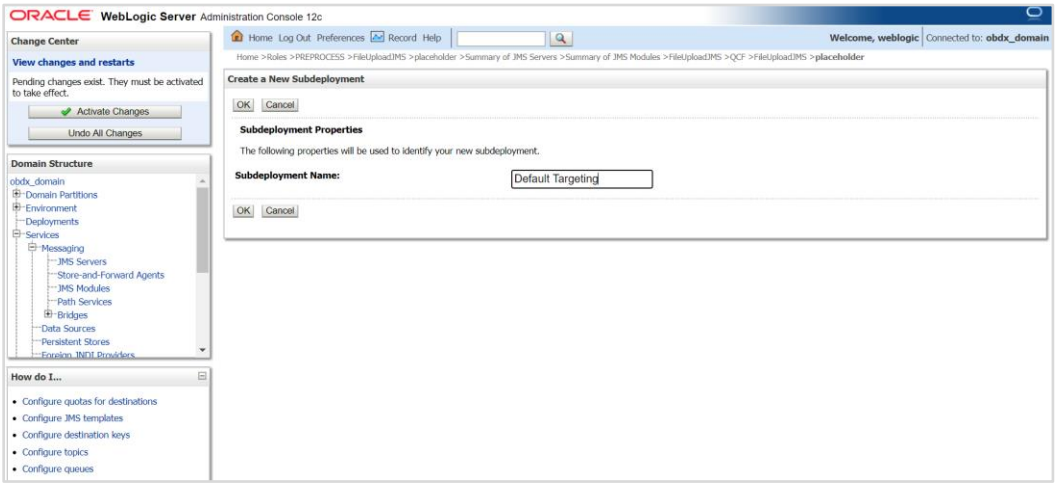
JNDI Name :- OCF

Subscription Sharing Policy :- Exclusive

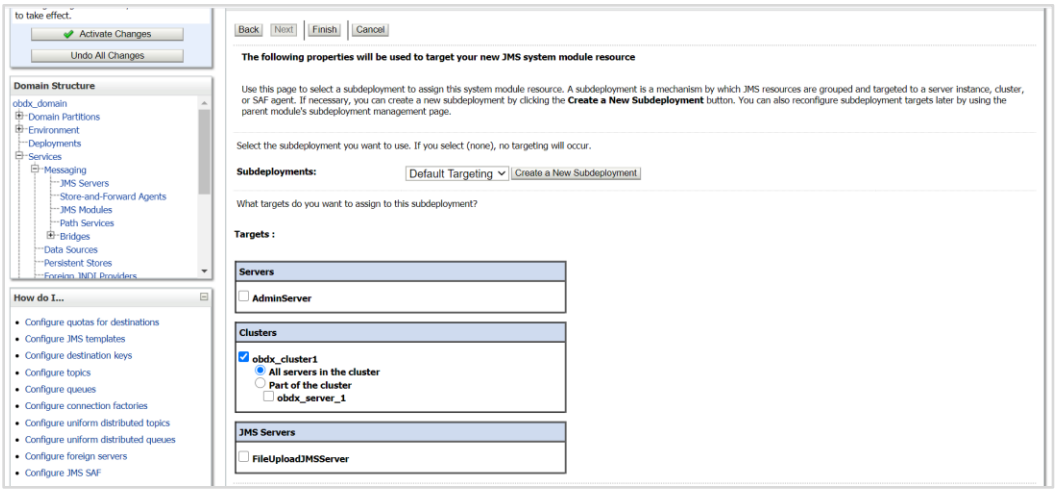
Client ID Policy :- Restricted



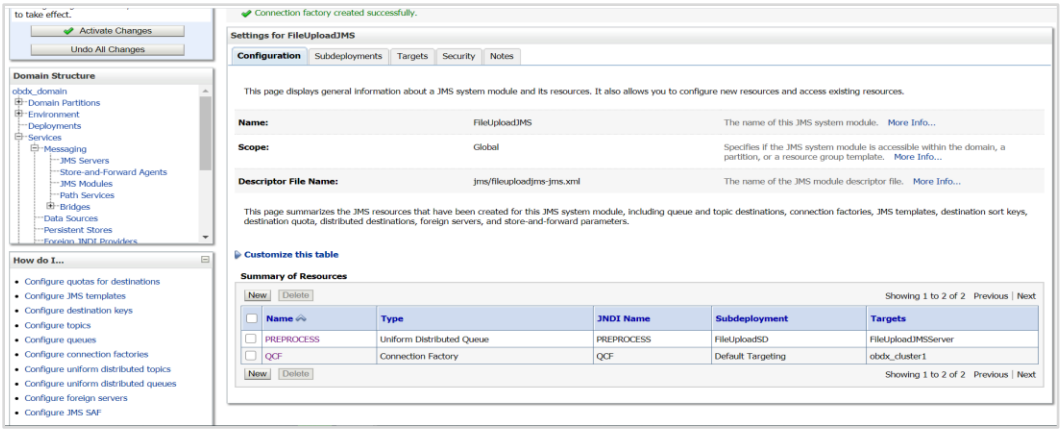
41. Click on Advanced targeting



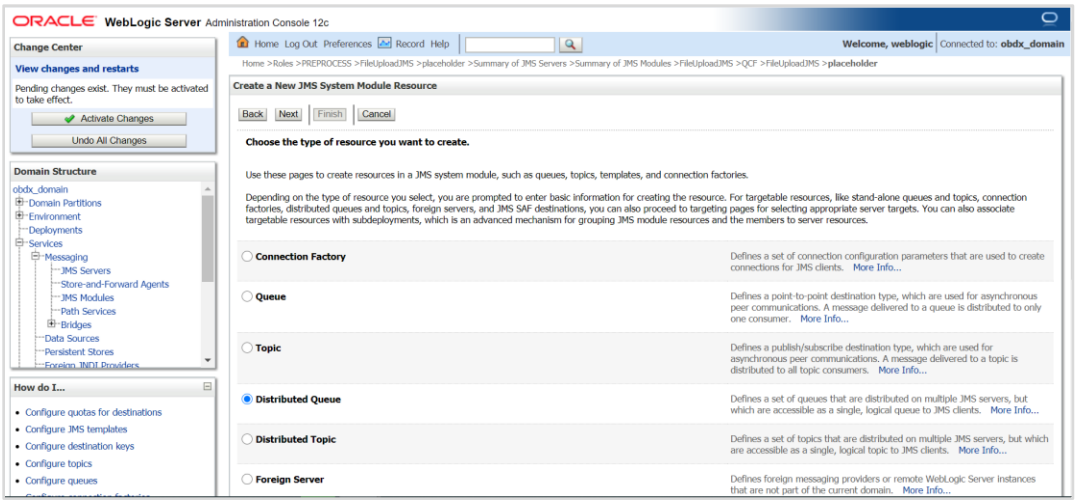
42. Provide Subdeployment Name as Default Targeting



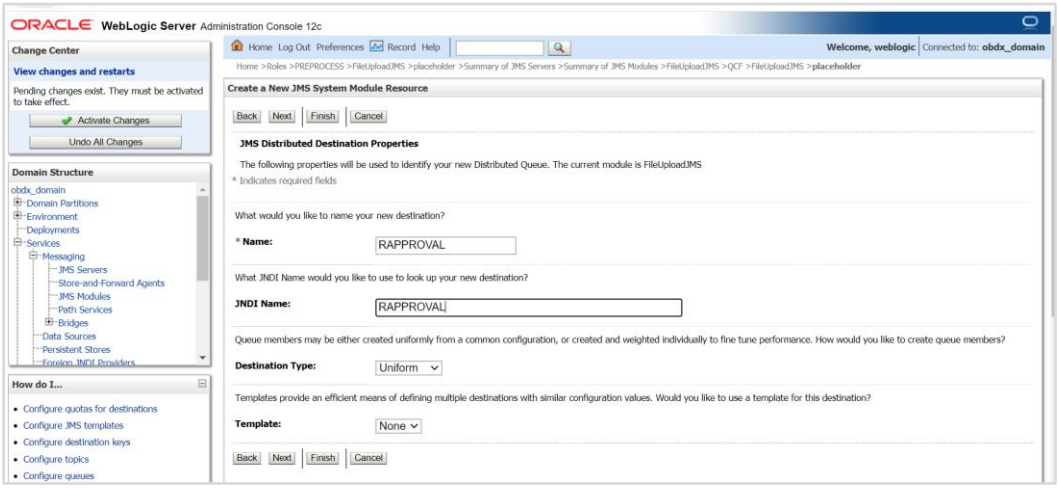
43. Select cluster and click on Finish



44. Go to FileUpload JMS click on New



45. Select Distributed Queue



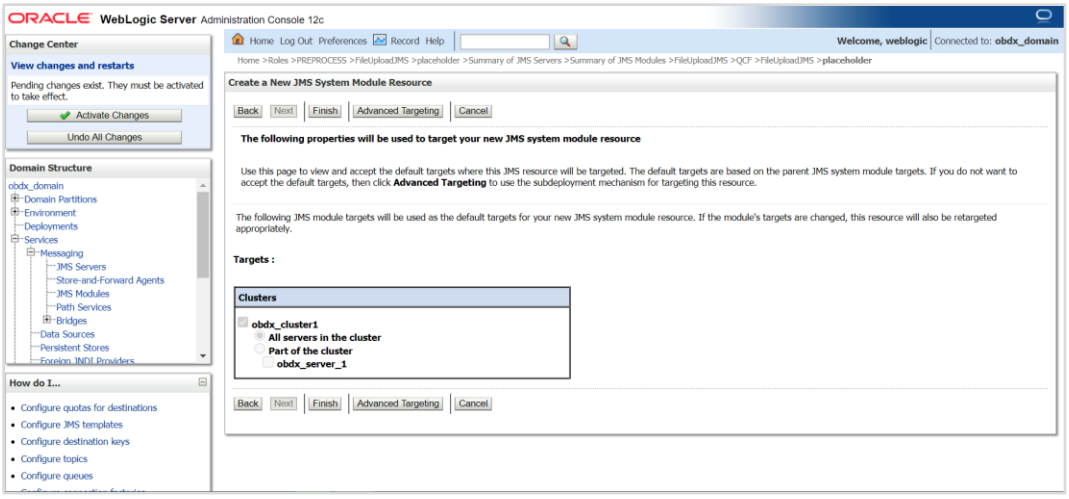
46. Provide

Name :- RAPPROVAL

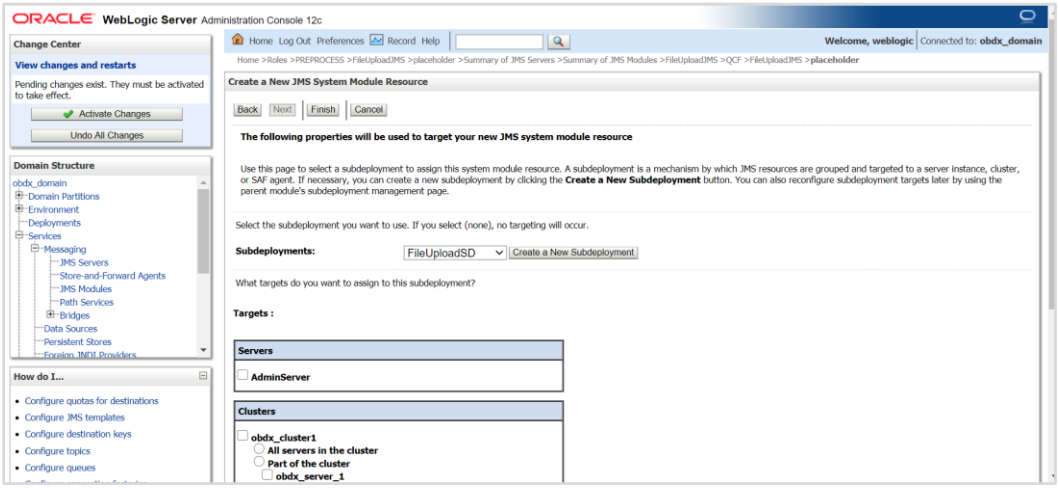
JNDI Name :- RAPPROVAL

Destination Type:- Uniform

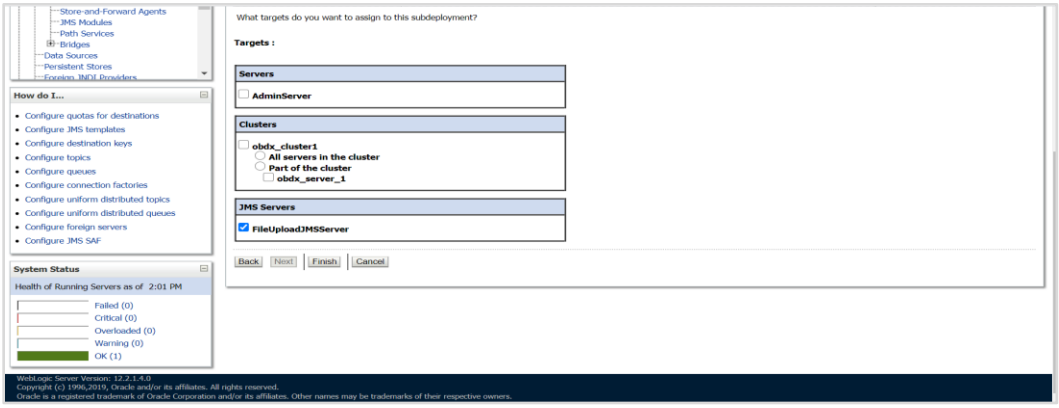
Template :- None



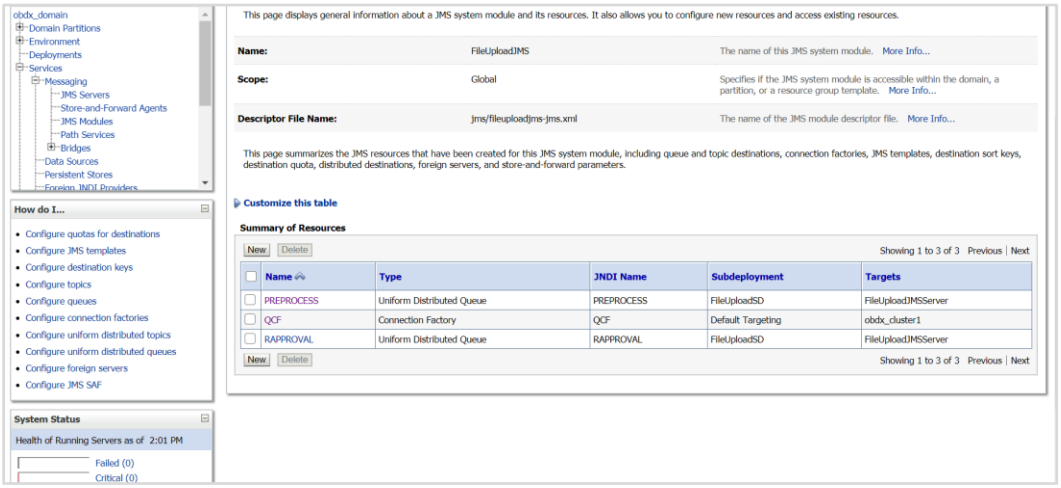
47. Select Advance targeting



48. Select Subdeployment :- FileUploadSD



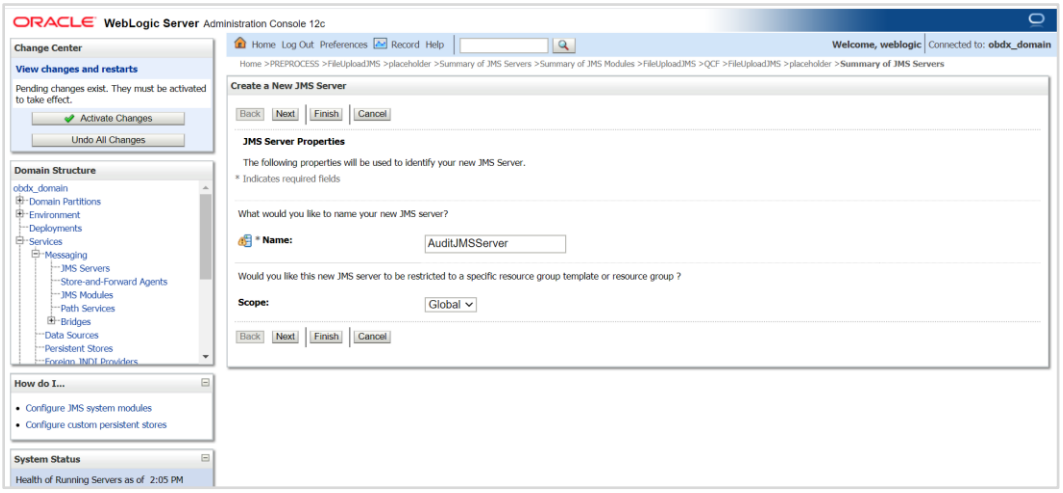
49. Select FileUploadJMSServer and click on Finish



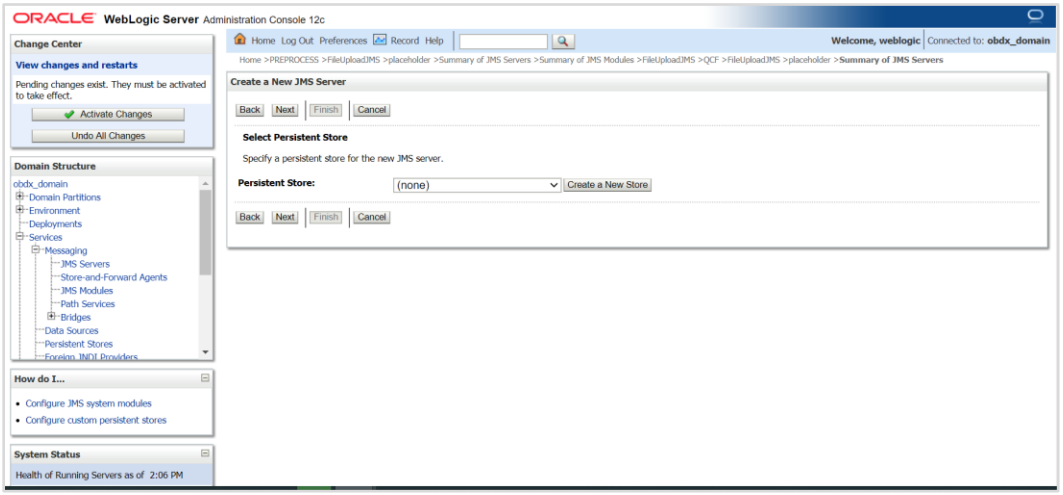
3.8 Creating WLS_JMS_AUDIT_PS FileStore

3.9 Creating AuditJMSServer JMS Server

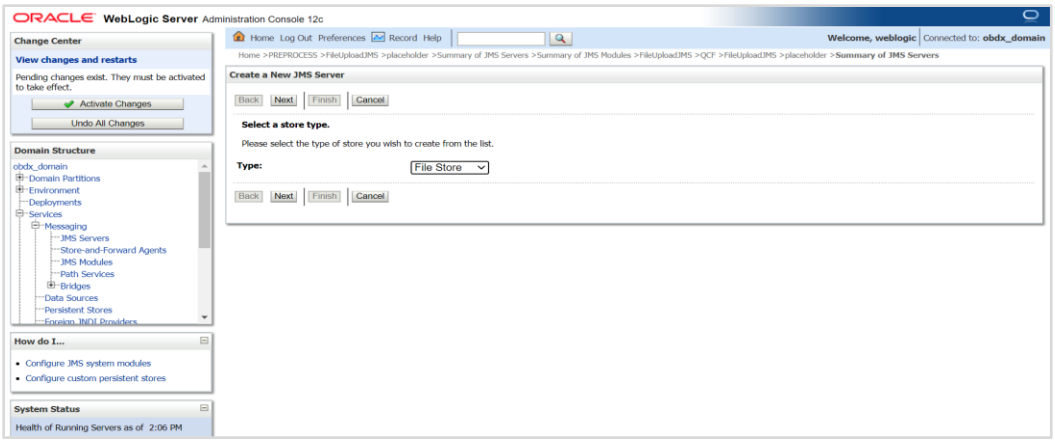
3.10 Creating WLS_JMS_REPORT_PS FileStore



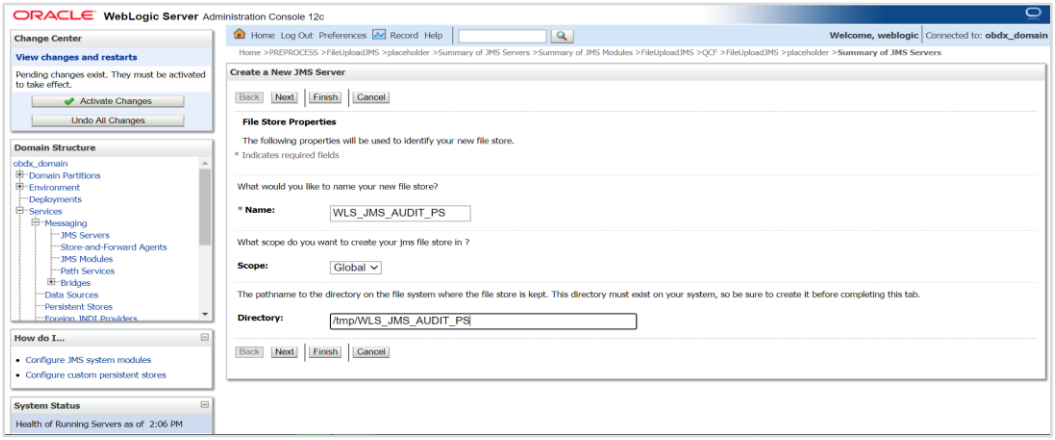
1. Click on JMS server and click on New
2. Provide Name as AuditJMSServer , Scope as Global



3. Click on Create a New Store



4. Select File Store

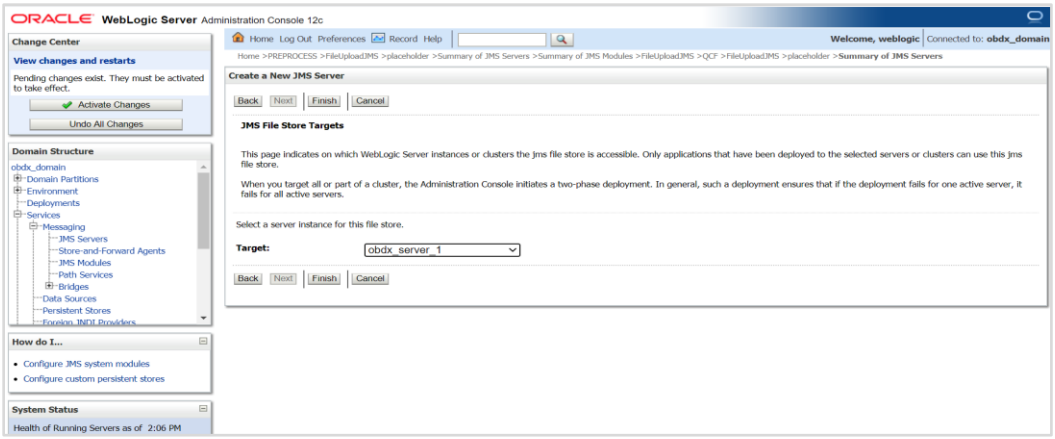


5. Provide

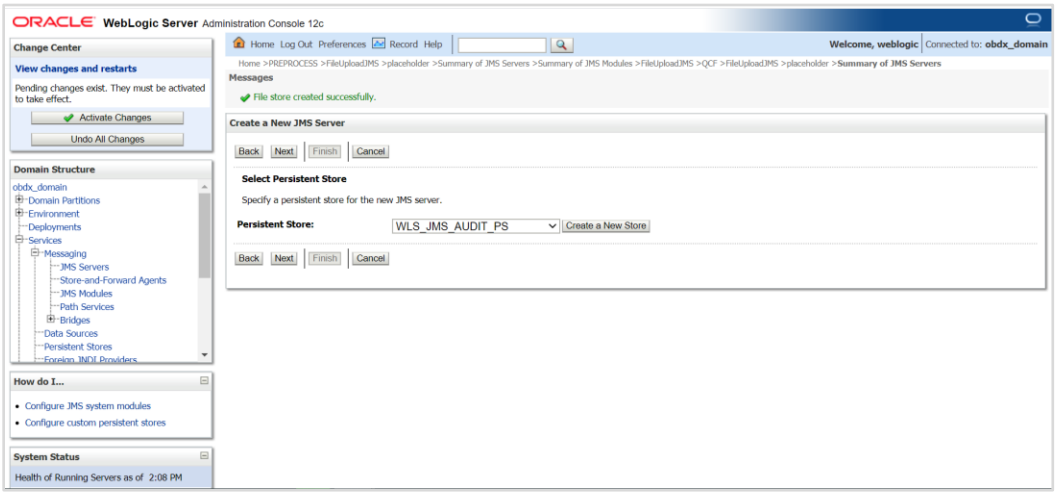
Name :- WLS_JMS_AUDIT_PS

Scope :- Global

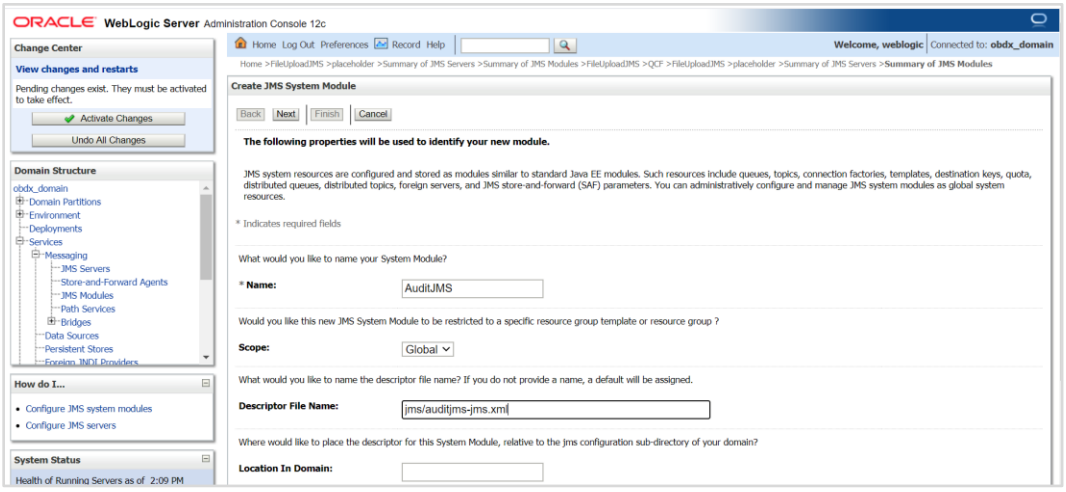
Directory :- /tmp/WLS_JMS_AUDIT_PS



6. Select Target as managed server and click on Finish



7. Select the new store created WLS_JMS_AUDIT_PS and click on Next

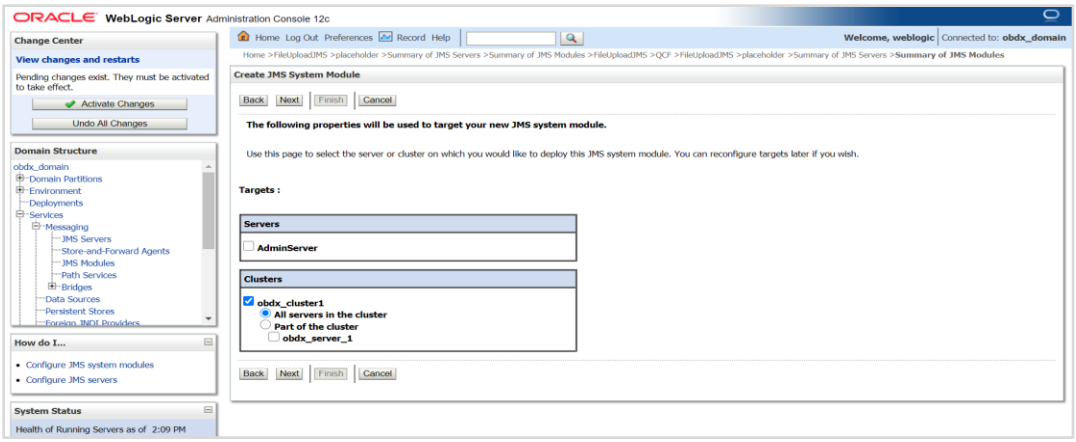


8. **Provide**

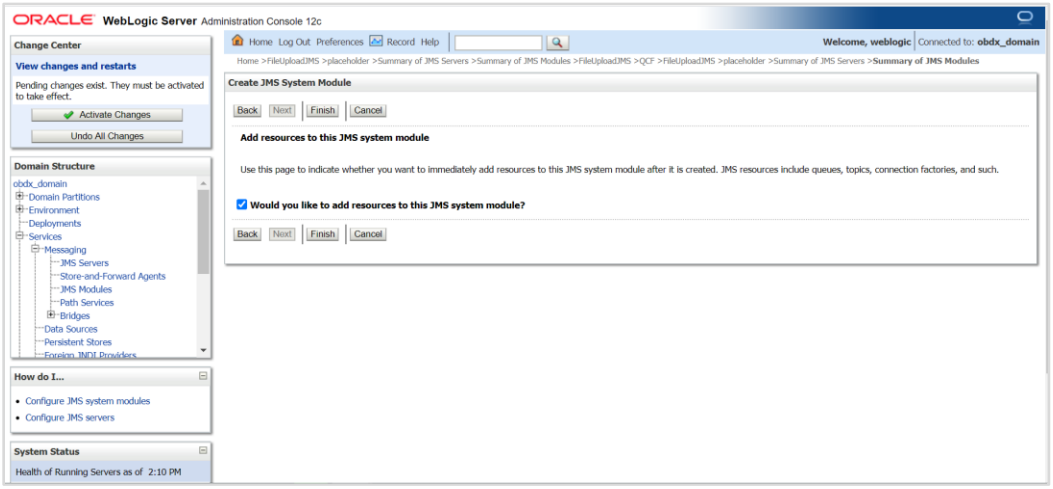
Name :- AuditJMS

Scope :- Global

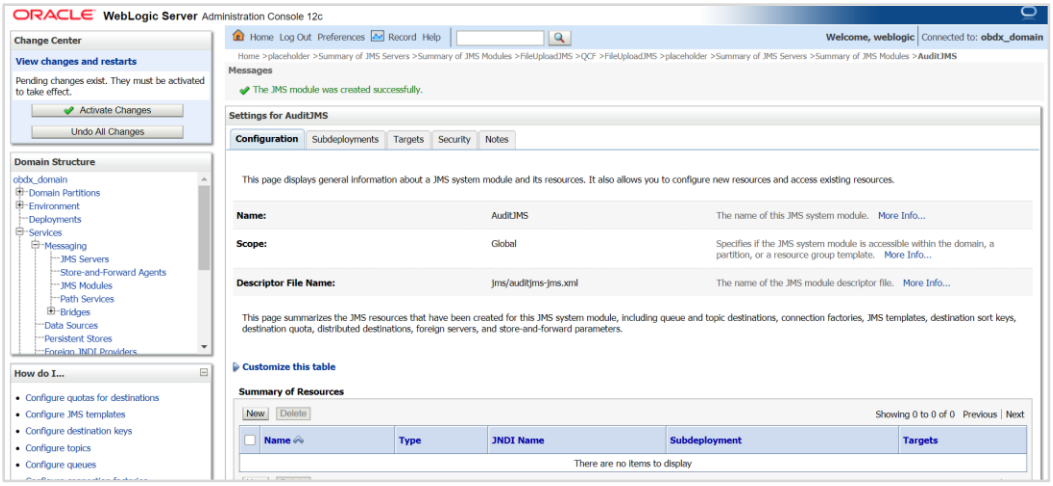
Descriptor File Name:- jms/auditjms-jms.xml



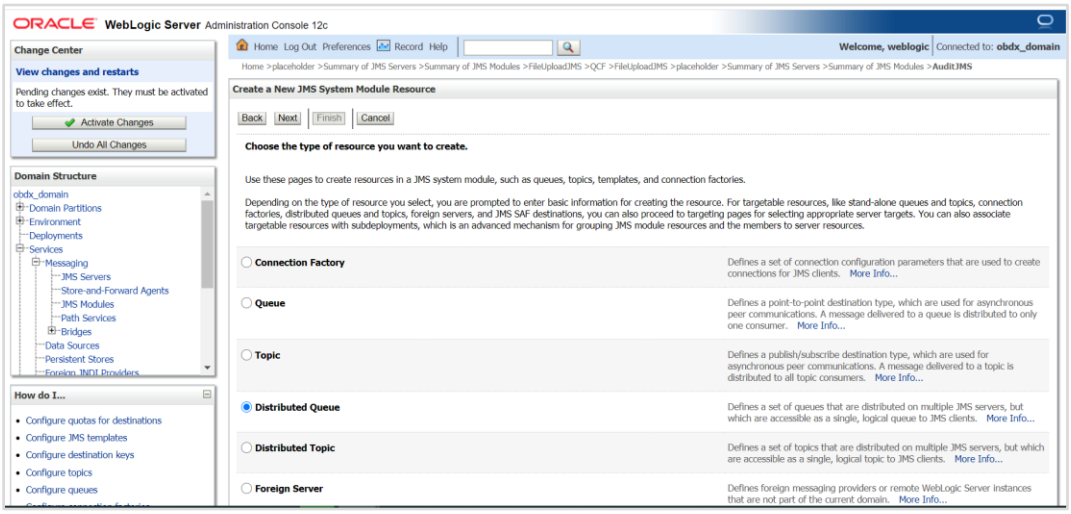
9. Select Cluster as a target



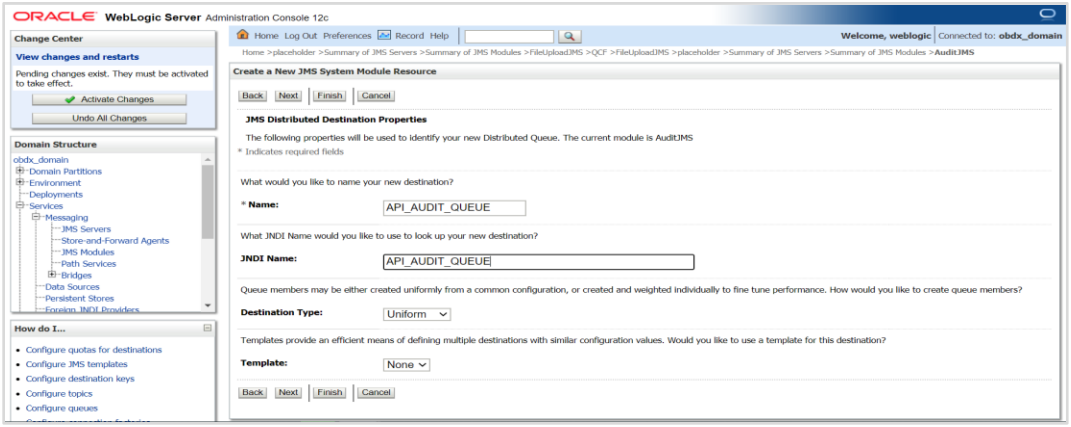
10. Select would you like to add resource to this JMS system module?



11. Click on new



12. Select Distributed Queue

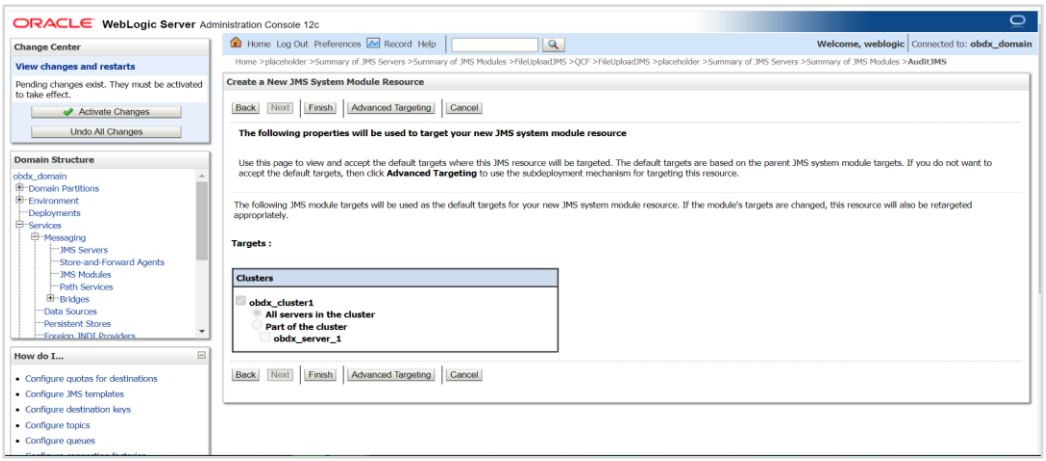


13. **Name:-** API_AUDIT_QUEUE

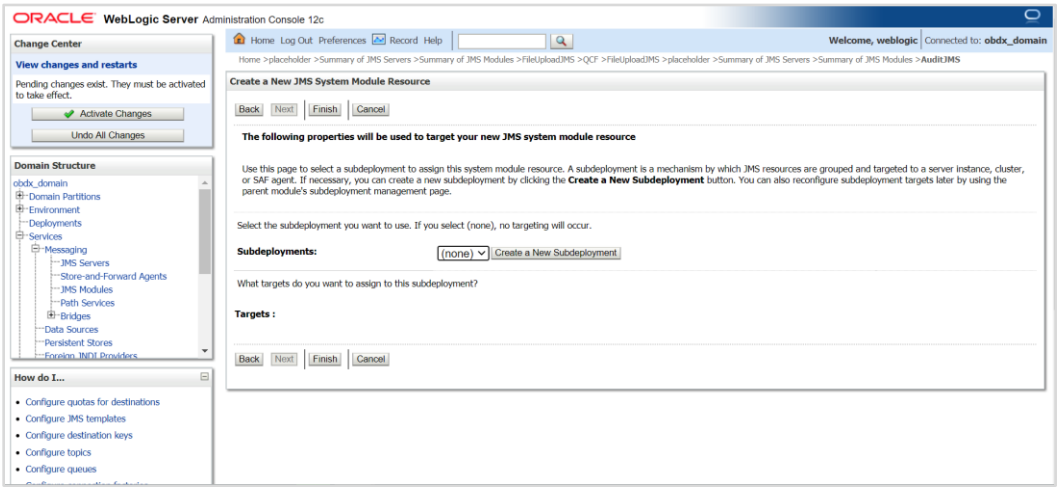
    JNDI Name:- API_AUDIT_QUEUE

    Destination Type :- Uniform

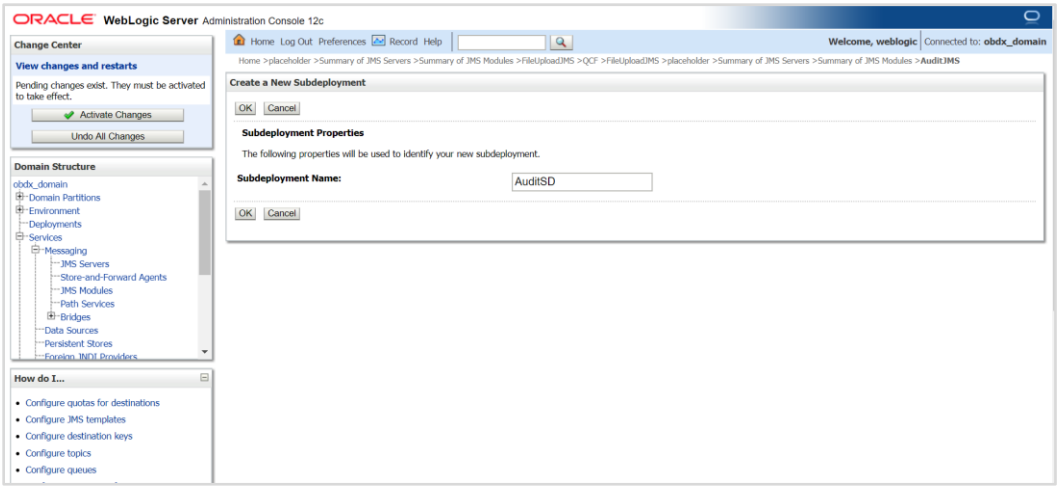
    Template:- None



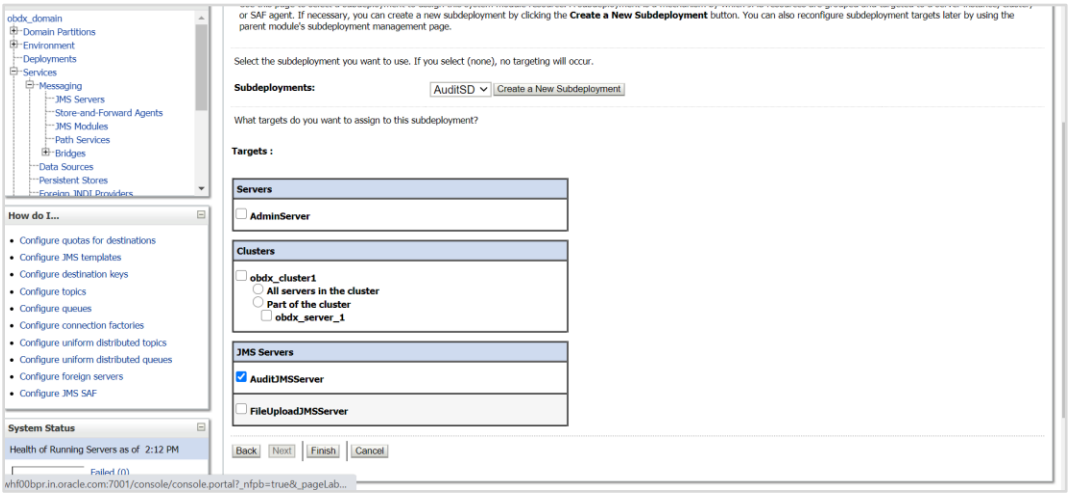
14. Select Advance targeting



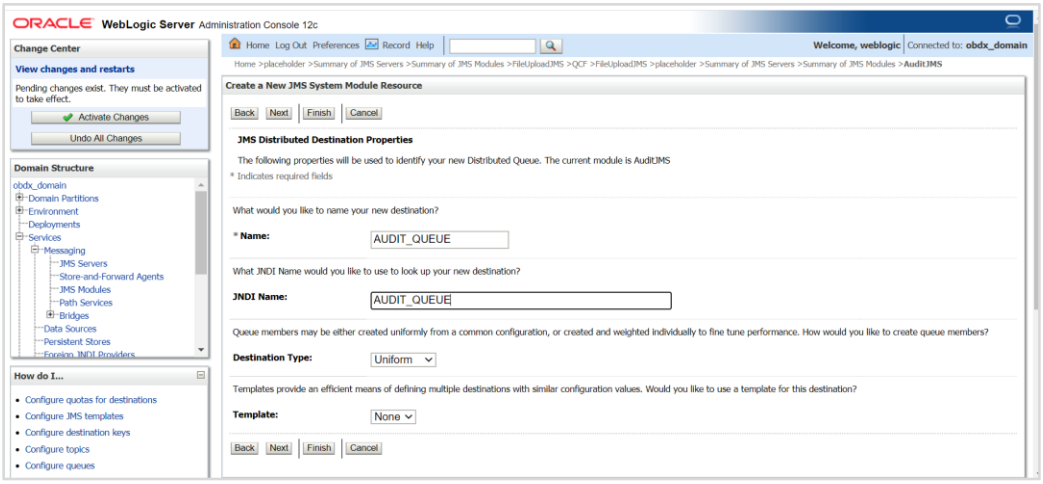
15. Click on Create a New Subdeployment

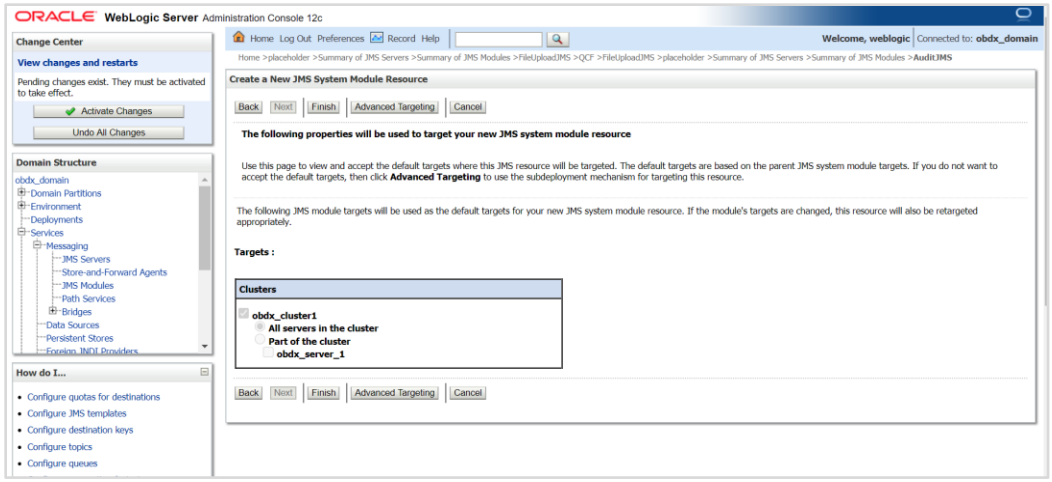


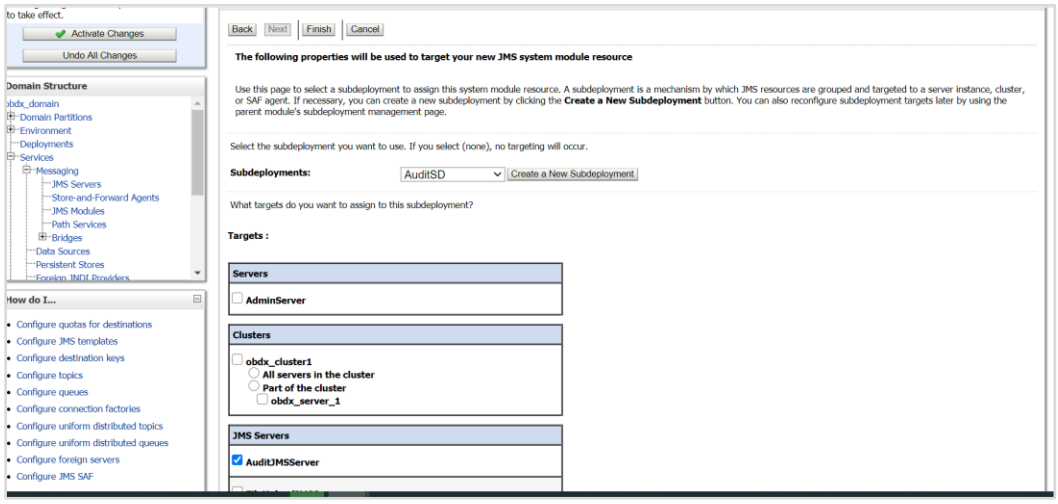
16. Provide Subdeployment Name as AuditSD

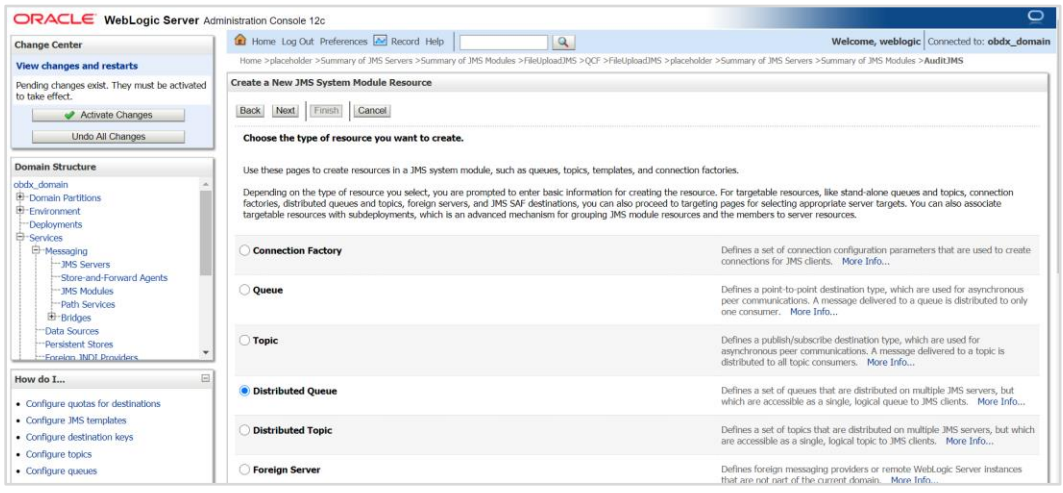


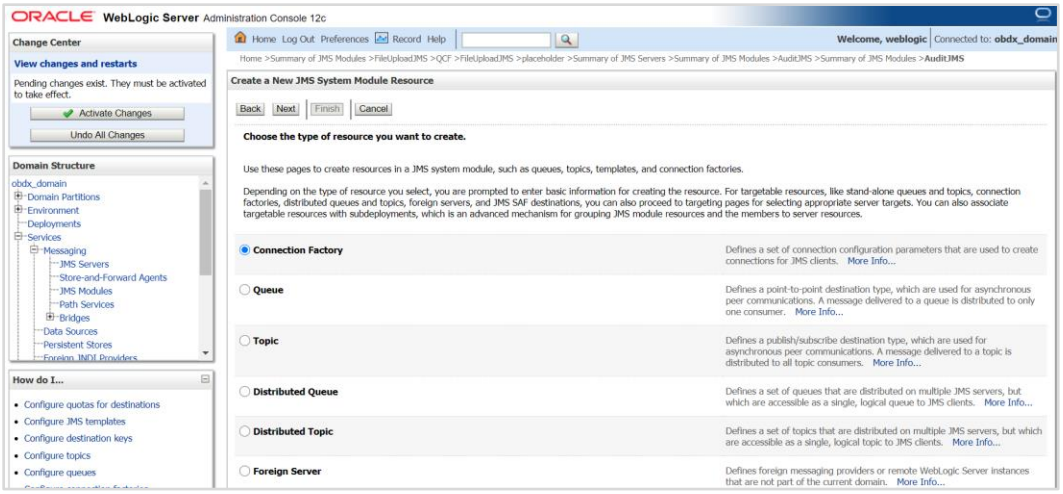
17. Select Target as AuditJMSServer



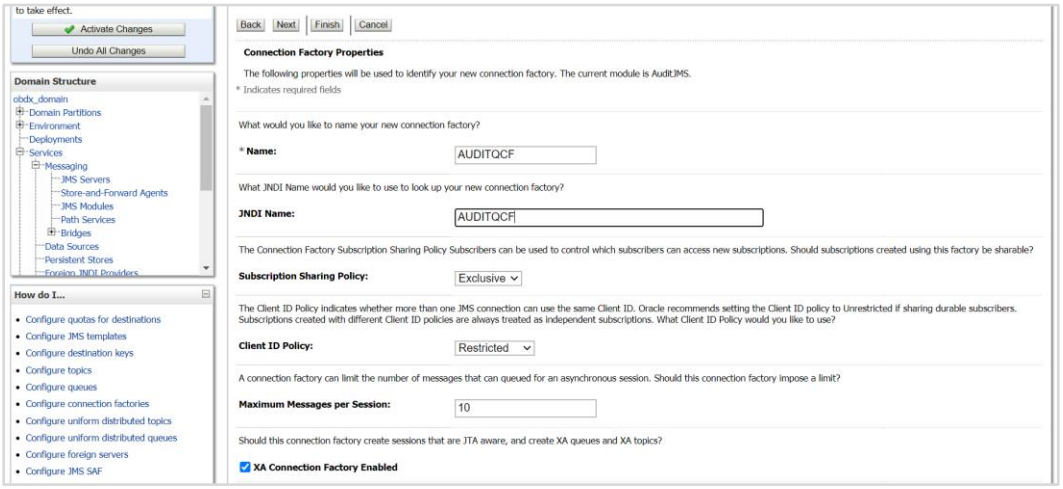

ORACLE WebLogic Server Administration Console 12c
Change Center
View changes and restarts
Pending changes exist. They must be activated to take effect.
Activate Changes
Undo All Changes
Domain Structure
obdx_domain
Domain Partitions
Environment
Deployments
Services
Messaging
JMS Servers
Store-and-Forward Agents
JMS Modules
Path Services
Bridges
Data Sources
Persistent Stores
Foreign JNDI Providers
How do I...
Configure quotas for destinations
Configure JMS templates
Configure destination keys
Configure topics
Configure queues
Home Log Out Preferences Record Help
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 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


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
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: AuditSD Create a New Subdeployment
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
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 quotas for destinations
Configure JMS templates
Configure destination keys
Configure topics
Configure queues
Home Log Out Preferences Record Help
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...



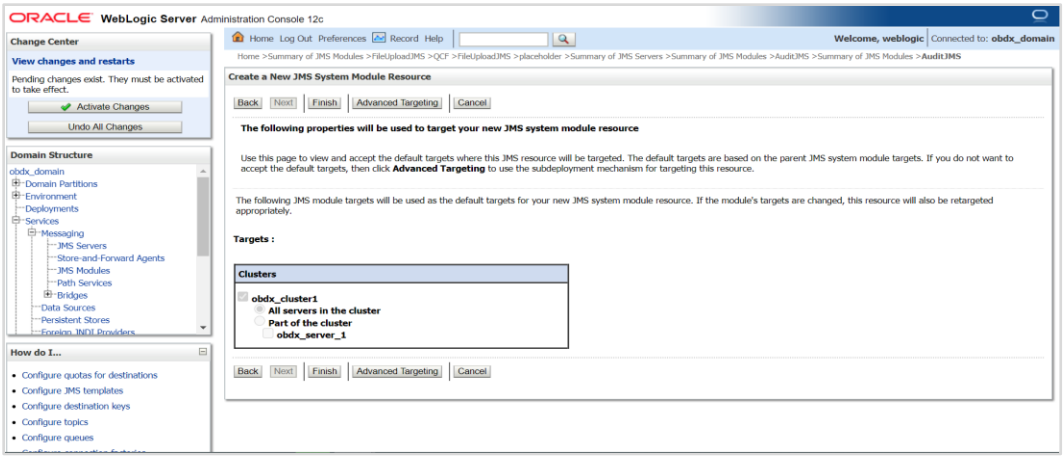
18. Click on connection Factory



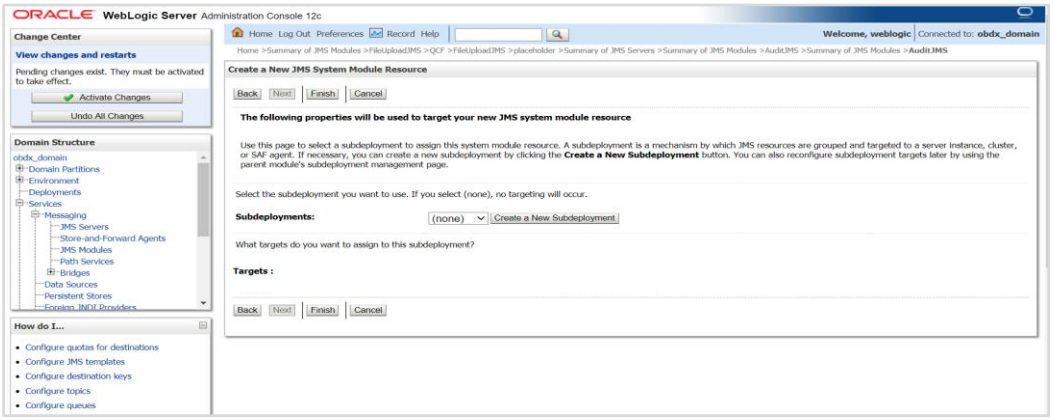
19. Provide

Name :- AUDITQCF

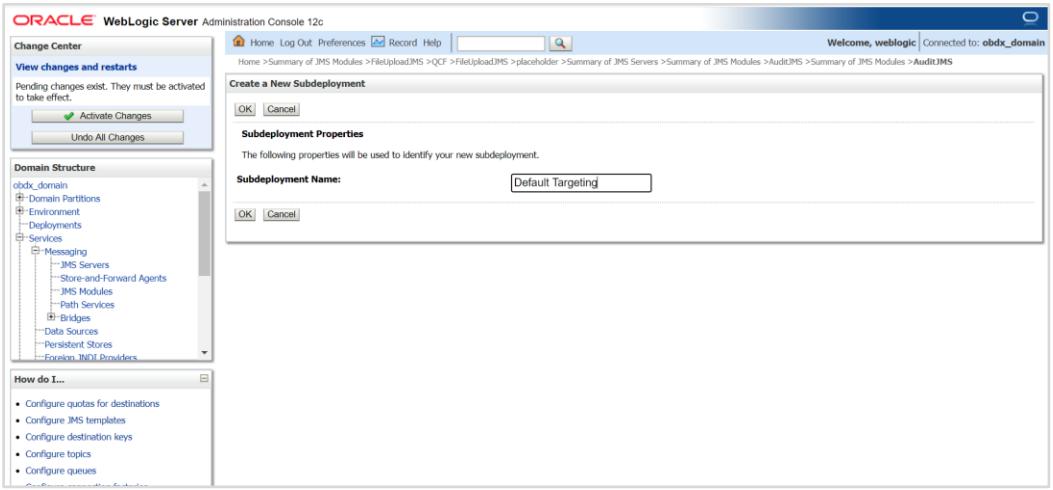
JNDI Name :- AUDITQCF



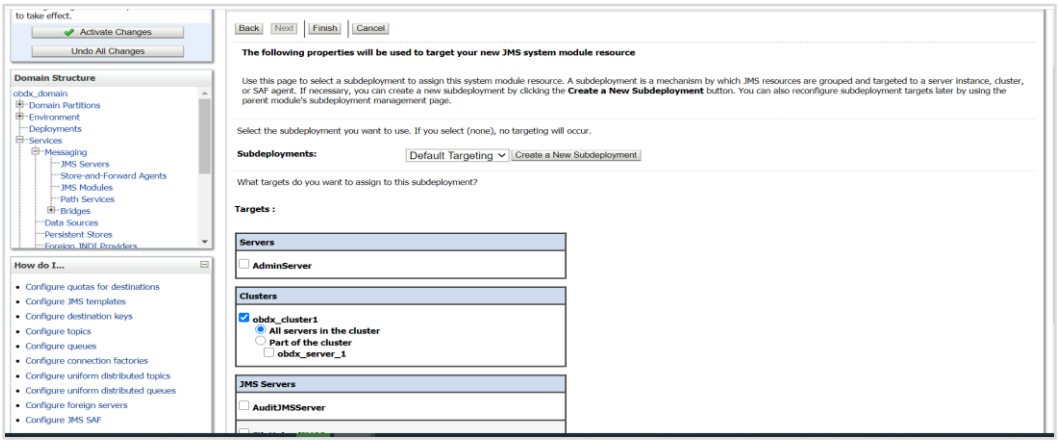
20. Click on Advanced Targeting



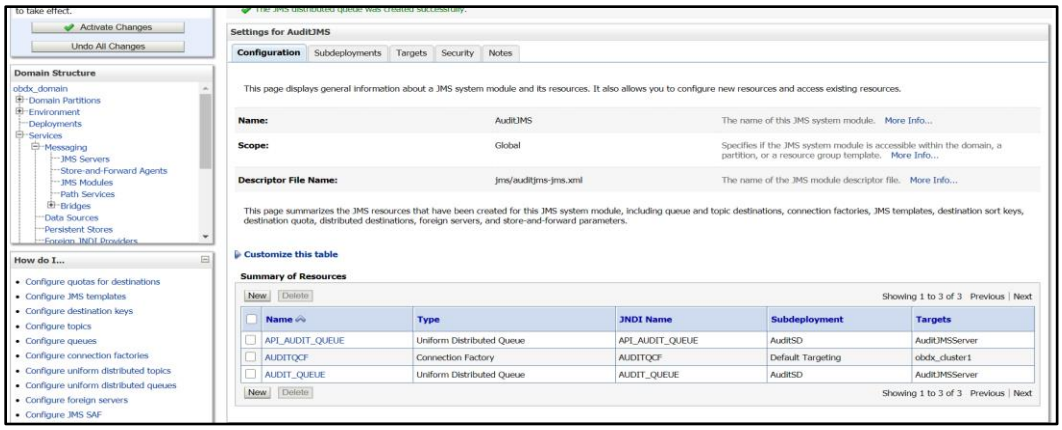
21. Click on Create a New Subdeployment



22. Give Subdeployment Name as Default Targeting

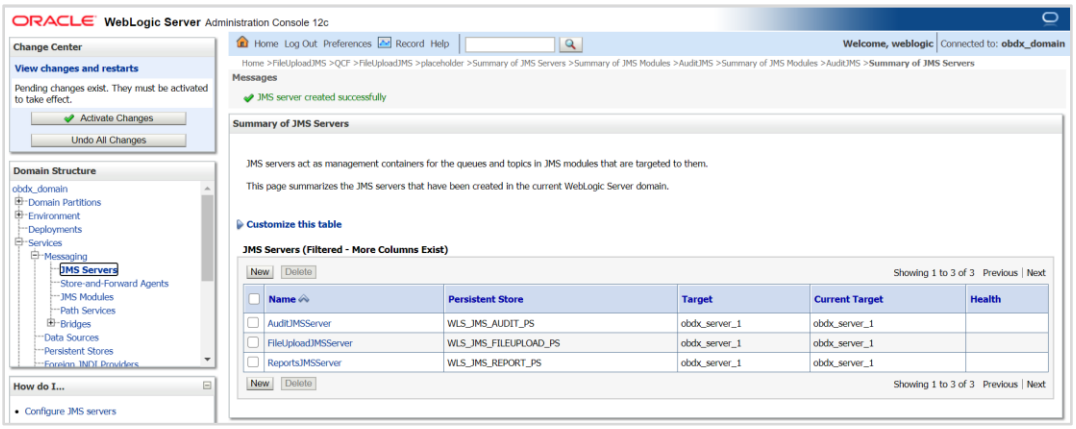


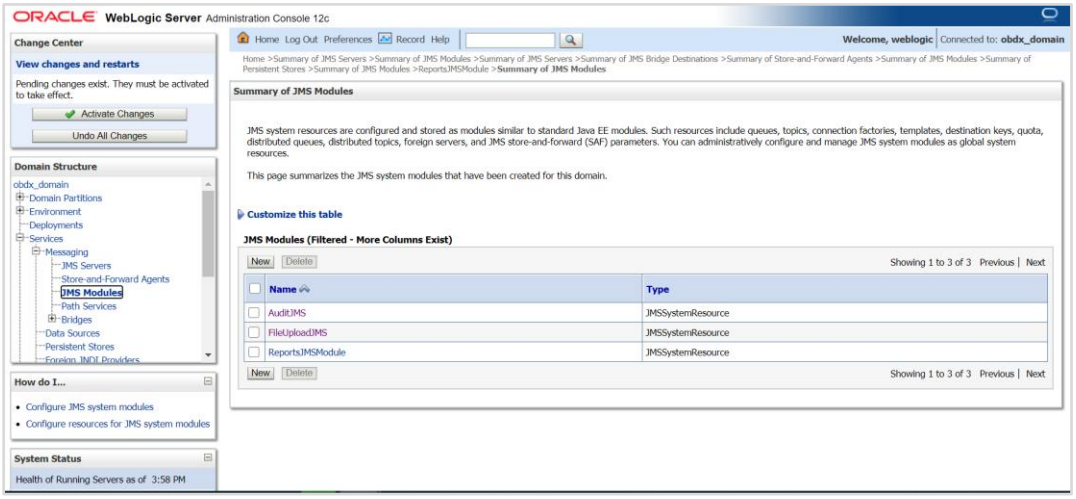
23. Under AuditJMS module Create Uniform Distrubuted Queue and connection Factory as show below in the screen shot

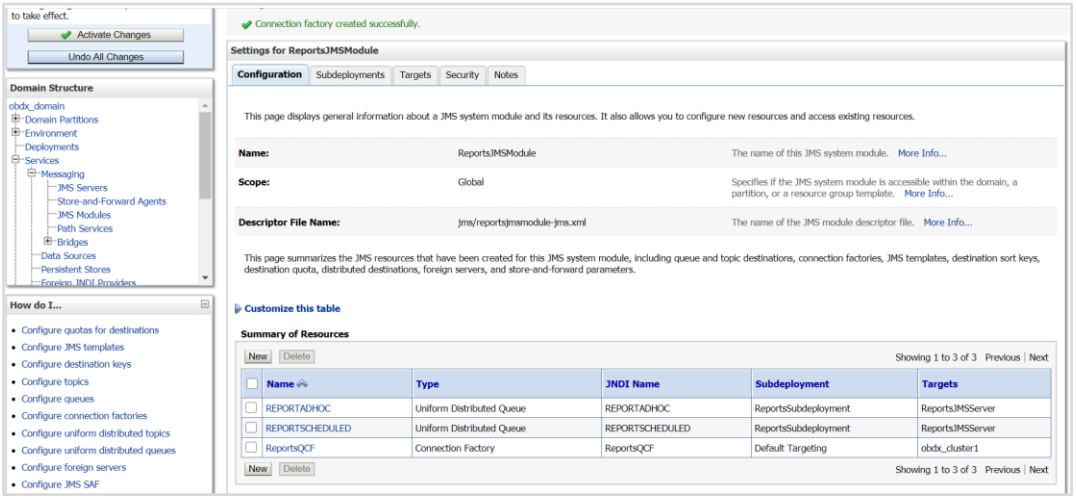


3.11 Creating ReportsJMSServer JMS Server

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







2. Under ReportsJMSModule create UniformDistributed Queue and connection factory as show above in the screen shot.

   REPORTADHOC – Uniform Distributed Queue

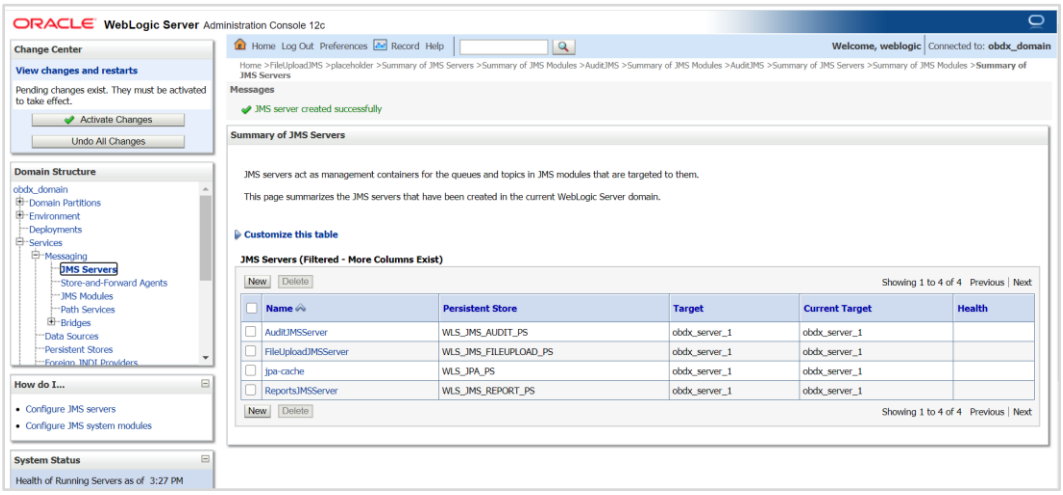
   REPORTSCHEDULED -- Uniform Distributed Queue

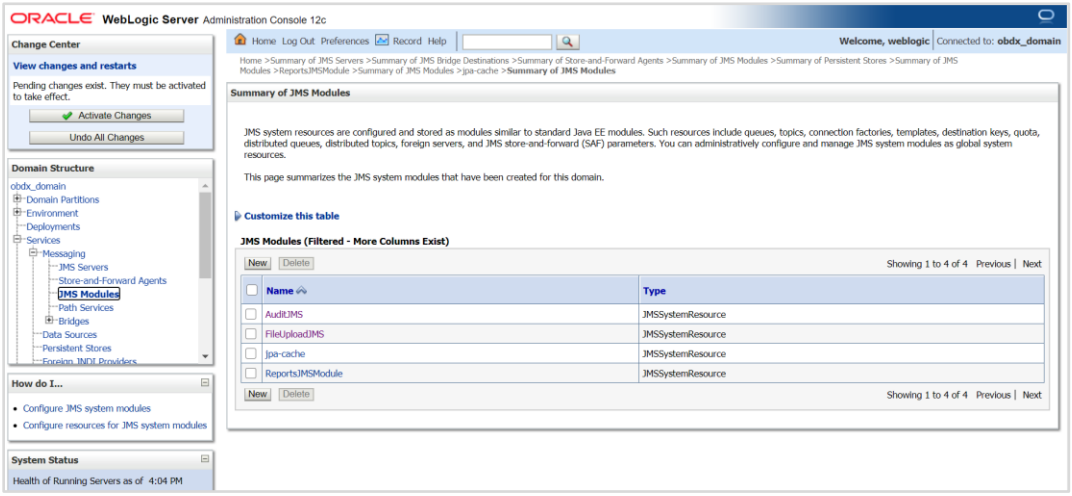
   ReportsQCF – Connection Factory

3.12 Creating jpa-cache JMS Server

3.13 Creating WLS_JPA_PS FileStore

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

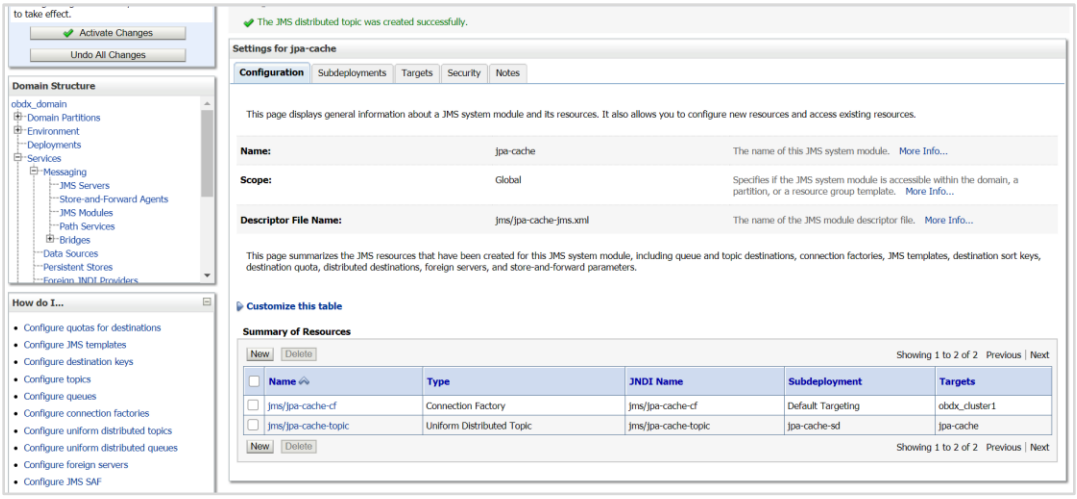




1. Under jpa-cache JMS Module create connection Factory and Uniform Distrbuted topic as shown in below screen shot

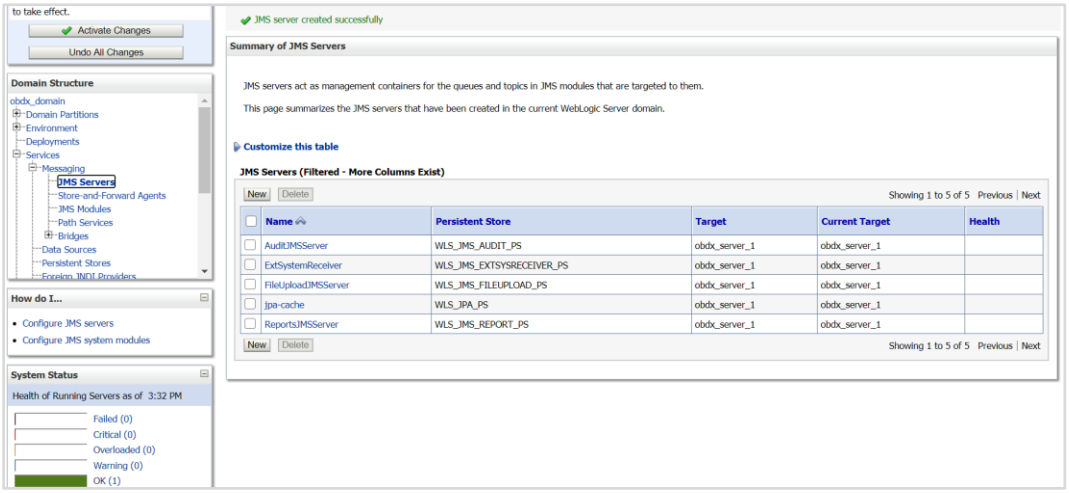
   Jms/jpa-cache-cf --- Connection Factory

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

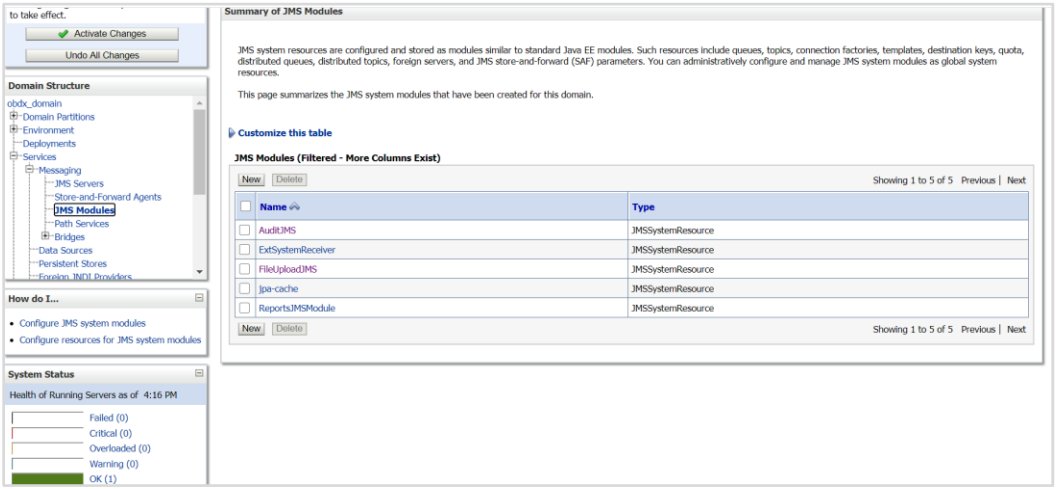

to take effect.
Activate Changes
Undo All Changes
Domain Structure
obdx_domain
Domain Partitions
Environment
Deployments
Services
Messaging
JMS Servers
Store-and-Forward Agents
JMS Modules
Path Services
Bridges
Data Sources
Persistent Stores
Foreign JNDI Providers
How do I...
Configure quotas for destinations
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
The JMS distributed topic was created successfully.
Settings for jpa-cache
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: jpa-cache
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/jpa-cache-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
New
Delete
Showing 1 to 2 of 2 Previous | Next
Name
Type
JNDI Name
Subdeployment
Targets
jms/jpa-cache-cf
Connection Factory
jms/jpa-cache-cf
Default Targeting
obdx_cluster1
jms/jpa-cache-topic
Uniform Distributed Topic
jms/jpa-cache-topic
jpa-cache-sd
jpa-cache
New
Delete
Showing 1 to 2 of 2 Previous | Next

3.14 Creating ExtSystemReceiver JMS Server -- WLS_JMS_EXTSYSRECEIVER_PS FileStore

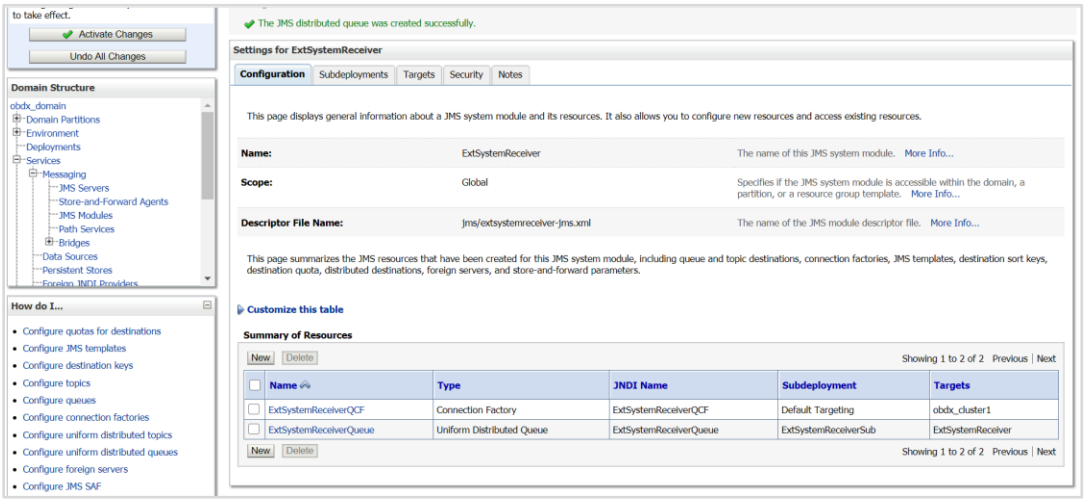
1. Create ExtSystemReceiver JMS Server Persistent store file store as WLS_JMS_EXTSYSRECEIVER_PS as show in below screen shot.



2. Create ExtSystemReceiver JMS Module as below

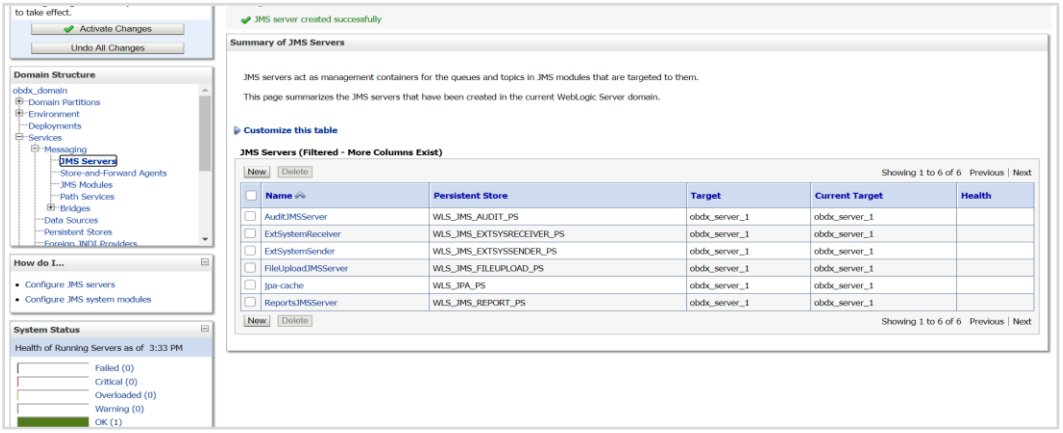


3. Create ExtSystemReceiverQCF – connection Factory and ExtSystemReceiverQueue – uniform Distributed Queue in ExtSystemReceiver JMS Module refer below screen shot

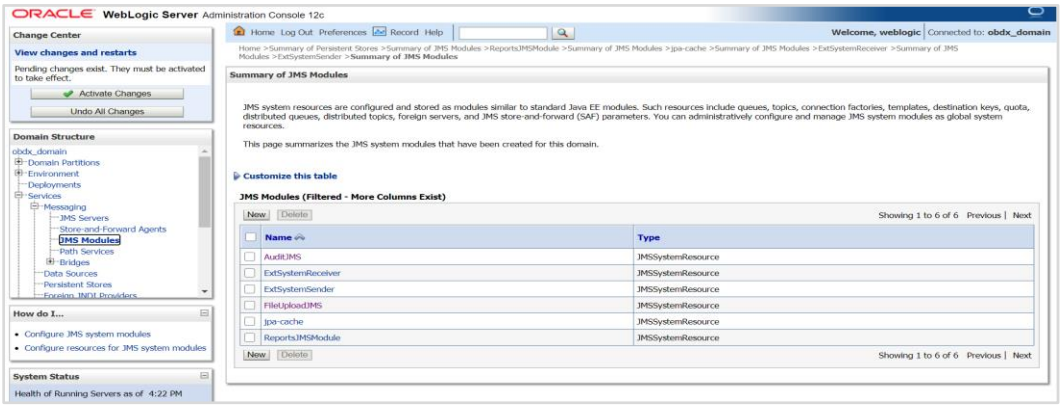


3.15 Creating ExtSystemSender JMS Server Persistent Store FileStore as WLS_JMS_EXTSYSSENDER_PS

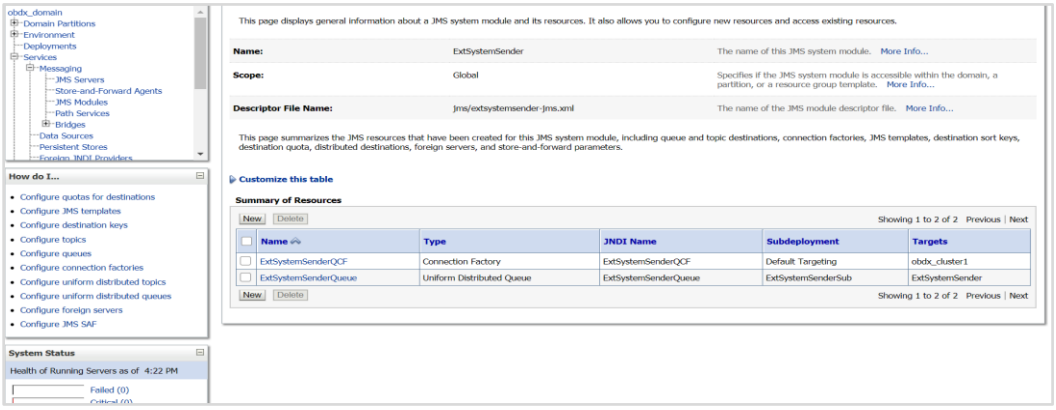
As show below create JMS Server ExtSystemSender



1. Create ExtSystemSender JMS Module

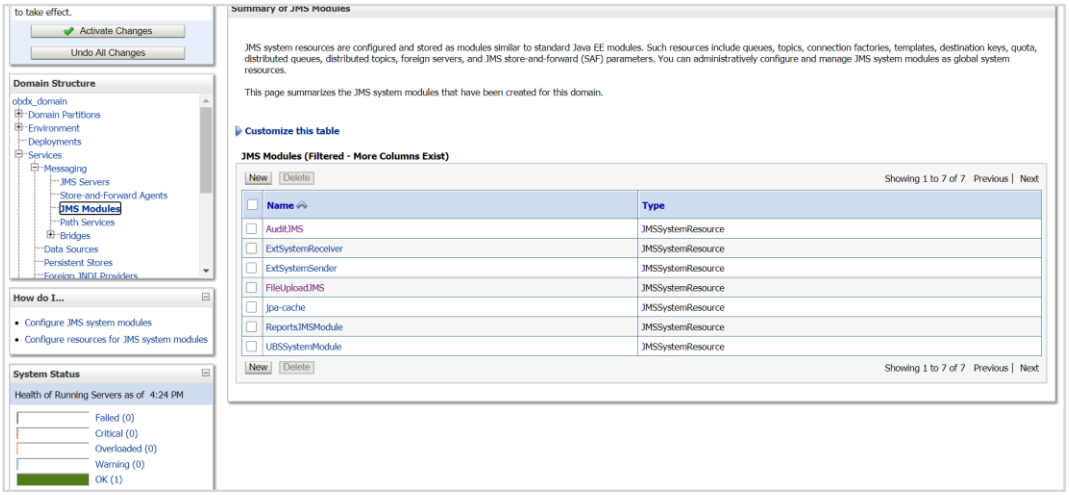


2. Under ExtSystemSender JMS Module create ExtSystemSenderQCF – connection Factory and ExtSystemSenderQueue – Uniform Distributed Queue as show below

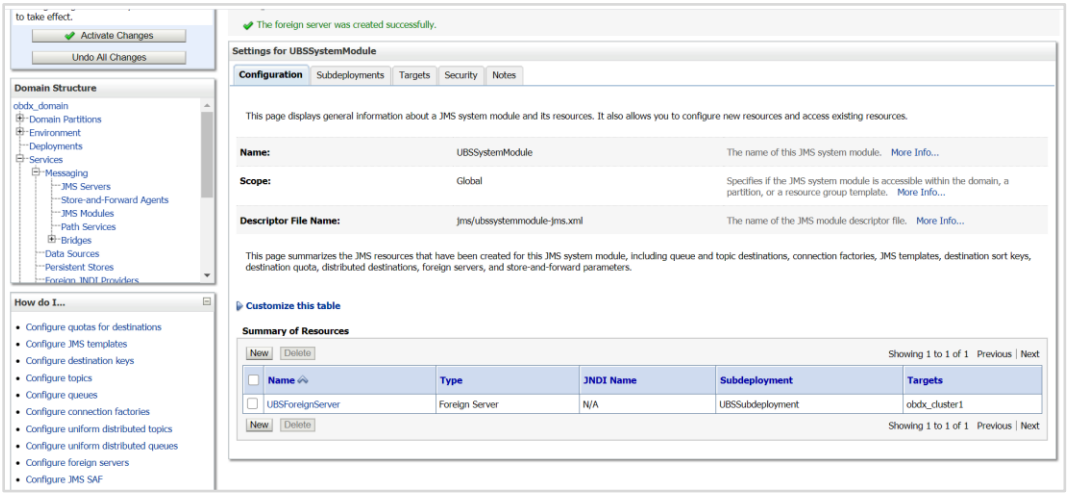


3.16 Creating UBSForeignServer JMS Server

1. In JMSModule create UBSSystemModule

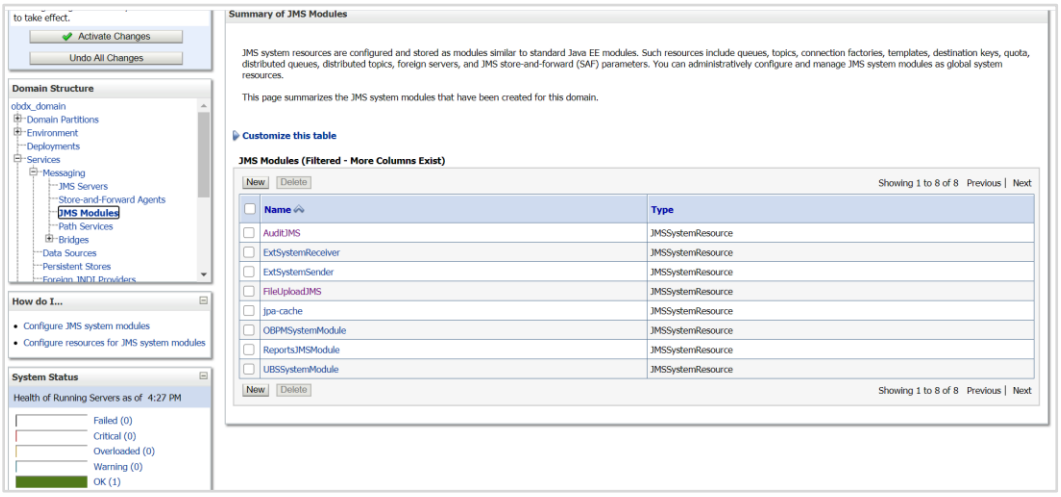


2. Under UBSSystemModule create UBSForeignServer – Foreign Server as shown below

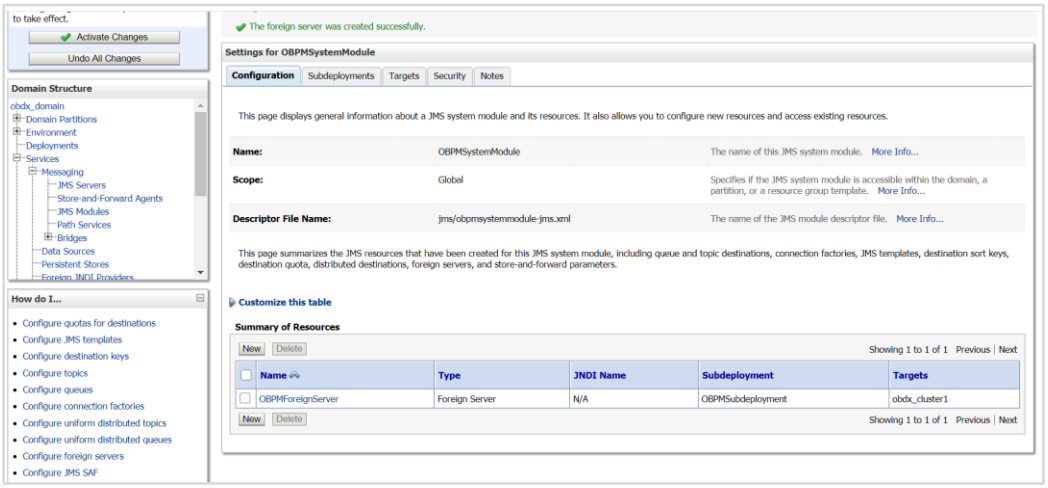


3.17 Creating OBPMForeignServer JMS Server

1. In JMSModule create OBPMSystemModule



2. Under OBPMSystemModule create OBPMForeignServer – Foregin Server as show below in screen shot



4. Deploying Applications

Deployment of Lib and Wars

Wars and Libs which are independent are present in path- OBDX_Installer\installables\OBDX\<Installation type>\<version>/ app/components/common

Wars that are created on runtime will be available in path- OBDX_Installer/ OBDX_Installer/ExecInstances/<date>/app/wars. Please refer below XML file for list of wars to be deployed.

```
<application name="digx-cms.war" displayedName="digx-cms"
target="@wls_cluster_name@" location="@deploy_path@" type="" deployOrder="100"/>

<application name="digx-corporateloan.war" displayedName="digx-corporateloan"
target="@wls_cluster_name@" location="@deploy_path@" type="" deployOrder="100"/>

<application name="digx-creditfacility.war" displayedName="digx-creditfacility"
target="@wls_cluster_name@" location="@deploy_path@" type="" deployOrder="100"/>

<application name="digx-edx.war" displayedName="digx-edx"
target="@wls_cluster_name@" location="@deploy_path@" type="" deployOrder="100"/>

<application name="digx-liquiditymanagement.war" displayedName="digx-
liquiditymanagement" target="@wls_cluster_name@" location="@deploy_path@" type=""
deployOrder="100"/>

<!-- <application name="digx-loanapplication.war" displayedName="digx-loanapplication"
target="@wls_cluster_name@" location="@deploy_path@" type="" deployOrder="100"/> -
->

<application name="digx-payments.war" displayedName="digx-payments"
target="@wls_cluster_name@" location="@deploy_path@" type="" deployOrder="100"/>

<application name="digx-pfm.war" displayedName="digx-pfm"
target="@wls_cluster_name@" location="@deploy_path@" type="" deployOrder="100"/>

<!-- <application name="digx-pm.war" displayedName="digx-pm"
target="@wls_cluster_name@" location="@deploy_path@" type="" deployOrder="100"/> -
->

<application name="digx-processmanagement.war" displayedName="digx-
processmanagement" target="@wls_cluster_name@" location="@deploy_path@" type=""
deployOrder="100"/>

<application name="digx-retail.war" displayedName="digx-retail"
target="@wls_cluster_name@" location="@deploy_path@" type="" deployOrder="100"/>
```

```
<application name="digx-scf.war" displayedName="digx-scf"
target="@wls_cluster_name@" location="@deploy_path@" type="" deployOrder="100"/>

<application name="digx-scfcm.war" displayedName="digx-scfcm"
target="@wls_cluster_name@" location="@deploy_path@" type="" deployOrder="100"/>

<application name="digx-tradefinance.war" displayedName="digx-tradefinance"
target="@wls_cluster_name@" location="@deploy_path@" type="" deployOrder="100"/>

<application name="digx-virtual-account.war" displayedName="digx-virtual-account"
target="@wls_cluster_name@" location="@deploy_path@" type="" deployOrder="100"/>

<application name="digx-kafkanotification.war" displayedName="digx-kafkanotification"
target="@wls_cluster_name@"
location="@installerhome@/installables/app/components/common" type=""
deployOrder="100"/>

<application name="digx-common.war" displayedName="digx-common"
target="@wls_cluster_name@" location="@deploy_path@" type="common"
deployOrder="100"/>

<application name="digx-admin.war" displayedName="digx-admin"
target="@wls_cluster_name@" location="@deploy_path@" type="common"
deployOrder="99"/>

<application name="digx-infra.war" displayedName="digx-infra"
target="@wls_cluster_name@" location="@deploy_path@" type="common"
deployOrder="100"/>

<library name="digx-shared-libs.war" displayedName="digx-shared-libs"
target="@wls_cluster_name@,AdminServer" location="@deploy_path@" type="common"
deployOrder="0"/>

<application name="digx-eurekaserver.war" displayedName="digx-eurekaserver"
target="@wls_cluster_name@"
location="@installerhome@/installables/app/components/common" type="common"
deployOrder="100"/>

<application name="digx-webauthn.war" displayedName="digx-webauthn"
target="@wls_cluster_name@"
location="@installerhome@/installables/app/components/common" type="common"
deployOrder="100"/>

<application name="digx-coherence.war" displayedName="digx-coherence"
target="@wls_cluster_name@"
location="@installerhome@/installables/app/components/common" type="common"
deployOrder="0"/>
```

<application name="digx-extxfacesimulator.war" displayedName="digx-extxfacesimulator" target="@wls_cluster_name@" location="@installerhome@/installables/app/components/common" type="common" deployOrder="100"/>

<library name="digx-lzn-libs.war" displayedName="digx-lzn-libs" target="@wls_cluster_name@,AdminServer" location="@installerhome@/installables/app/components/common" type="common" deployOrder="0"/>

<application name="digx-ukob.war" displayedName="digx-ukob" target="@wls_cluster_name@" location="@deploy_path@" type="common" deployOrder="100"/>

<application name="digx-berlinob.war" displayedName="digx-berlinob" target="@wls_cluster_name@" location="@deploy_path@" type="common" deployOrder="100"/>

<application name="digx-genai.war" displayedName="digx-genai" target="@wls_cluster_name@" location="@deploy_path@" type="common" deployOrder="100"/>

<application name="digx-finlimit.war" displayedName="digx-finlimit" target="@wls_cluster_name@" location="@installerhome@/installables/app/components/common" type="common" deployOrder="100"/>

<application name="digx-em.war" displayedName="digx-em" target="@wls_cluster_name@" location="@installerhome@/installables/app/components/common" type="common" deployOrder="100"/>

<application name="digx-sms.war" displayedName="digx-sms" target="@wls_cluster_name@" location="@deploy_path@" type="common" deployOrder="99"/>

<application name="digx-configserver.war" displayedName="digx-configserver" target="@wls_cluster_name@" location="@installerhome@/installables/app/components/common" type="common" deployOrder="100"/>

<!-- <application name="digx-approval.war" displayedName="digx-approval" target="@wls_cluster_name@" location="@installerhome@/installables/app/components/common" type="common" deployOrder="100"/> -->

Please use the wars present in above location and deploy the wars accordingly in weblogic.

5. 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.GenericIdentityStoreProvider">

   <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"/>