Configuring Weblogic Server

Oracle FLEXCUBE Universal Banking Solution 12.0.2

Version 3.0

November 2015

ORACLE®

FINANCIAL SERVICES

Table of Contents

ORACLE®

1. Configuring WebLogic Server

This section of the document provides detailed explanation on the following:

- Domain
- Administration Server
- Managed Server
- Managed Server Cluster
- Node Manager

1.1 Domain

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

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

WebLogic Domain Structure
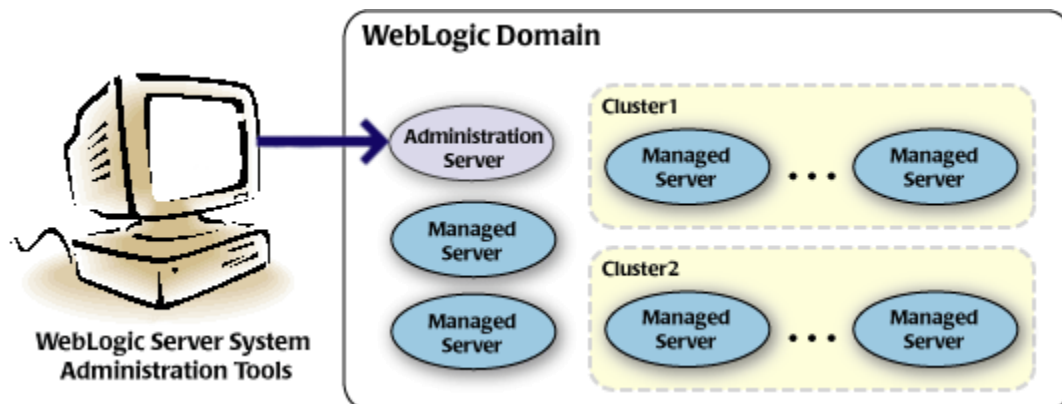
1.2 Administration Server

A domain includes one WebLogic Server instance that is configured as an Administration Server. All changes to configuration and deployment of applications are done through the Administration Server.

The Administration Server provides a central point for managing the domain and providing access to the WebLogic Server administration tools. These tools include the following:

- WebLogic Server Administration Console: Graphical user interface to the Administration Server.

ORACLE®

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

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

1.3 <u>Managed Server</u>

In a domain, server instances other than the Administration Server are referred to as Managed Servers. Managed servers host the components and associated resources that constitute your applications—for example, JSPs and EJBs. When a Managed Server starts up, it connects to the domain's Administration Server to obtain configuration and deployment settings.

In a domain with only a single WebLogic Server instance, that single server works as both the administration server and managed server.

1.4 <u>Managed Server Cluster</u>

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

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

1.5 <u>Node Manager</u>

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

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

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

ORACLE

2. Domain Creation

The following sections provides step-wise WebLogic Server configuration for Oracle FCUBS which covers the following:

- Managed server Creation
- Cluster Creation
- Node Creation

In this example, we are going to create a domain with two managed servers. The managed servers are going to be created on two different physical servers (nodes). This requires Weblogic Server of same version to be installed on both the machines and services (Weblogic and node manager) should be up.

2.1 Environment details

Note that, this document has been prepared based on a test conducted in Linux servers running ORACLE Linux 5 UPDATE 5.

The machine configuration is as given below:

2.1.1 Node1

Hostname: XXX187DOR

IP Address: xx.xx.xx.187

CPU details: 8 core Intel(R) Xeon(R) CPU E5540 @ 2.53GHz

Physical memory: 24GB

2.1.2 Node2

Hostname: XXX186DOR

IP Address: xx.xx.xx.186

CPU details: 8 core Intel(R) Xeon(R) CPU E5540 @ 2.53GHz

Physical memory: 24GB

ORACLE

3. Start Node Manager

You need to start the node manager in both servers before creating a domain. Any domain created before creation of Node Manager Service will not be accessible via node Manager even after restarting node manager. Hence the solution is to run the WLST command "nmEnroll" to enroll that domain with the Node Manager.

If the domain is created after the Node Manager Service has been started should not have to be enrolled against the Node Manager. The Node Manager should automatically be 'reachable' by the domain.

How to check whether nodemanager is already running?

To verify the nodemanager status, execute the following command:

ps –ef|grep weblogic.nodemanager.javaHome

If the above command returns the process details then node manager is running.

If nodemanager is not running in any of the server, start the Nodemanager in both the servers using **startNodeManager.sh** script under $WLSHOME/server/bin directory.

3.1 Enable Graphical User Interface (GUI)

Establish a telnet or SSH connection to xx.xx.xx.187 server. Start X-manager (or any similar tool) in windows desktop. Export DISPLAY environment variable to the machine IP where x-manager is running.

Syntax: export DISPLAY=<ip-address>:<port>

Test using xclock

3.2 Fusion Middleware Configuration Wizard

Weblogic domain creation and configuration will be done from server xx.xx.xx.187.

From server xx.xx.xx.187, launch the fusion Middleware configuration wizard using the command **config.sh** available under $WLS_HOME/common/bin directory.

- In the Welcome screen, select "Create a new Weblogic domain" option. Click on **Next**.
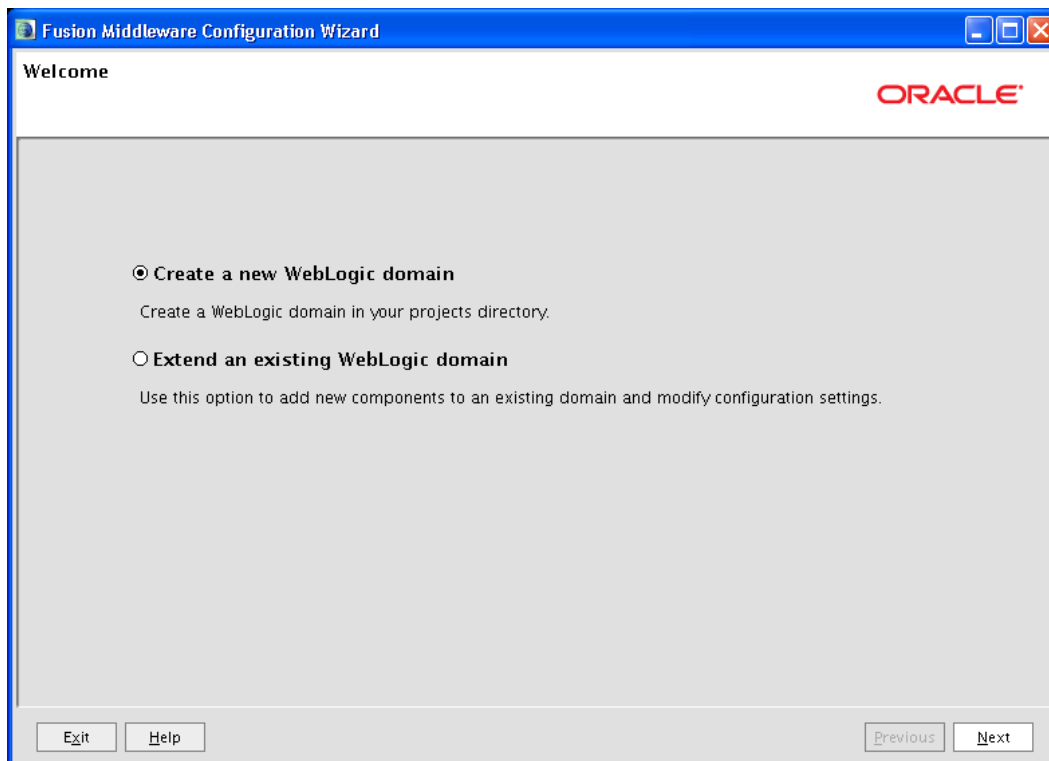
ORACLE

- Select the **Domain Source** and click **Next**.
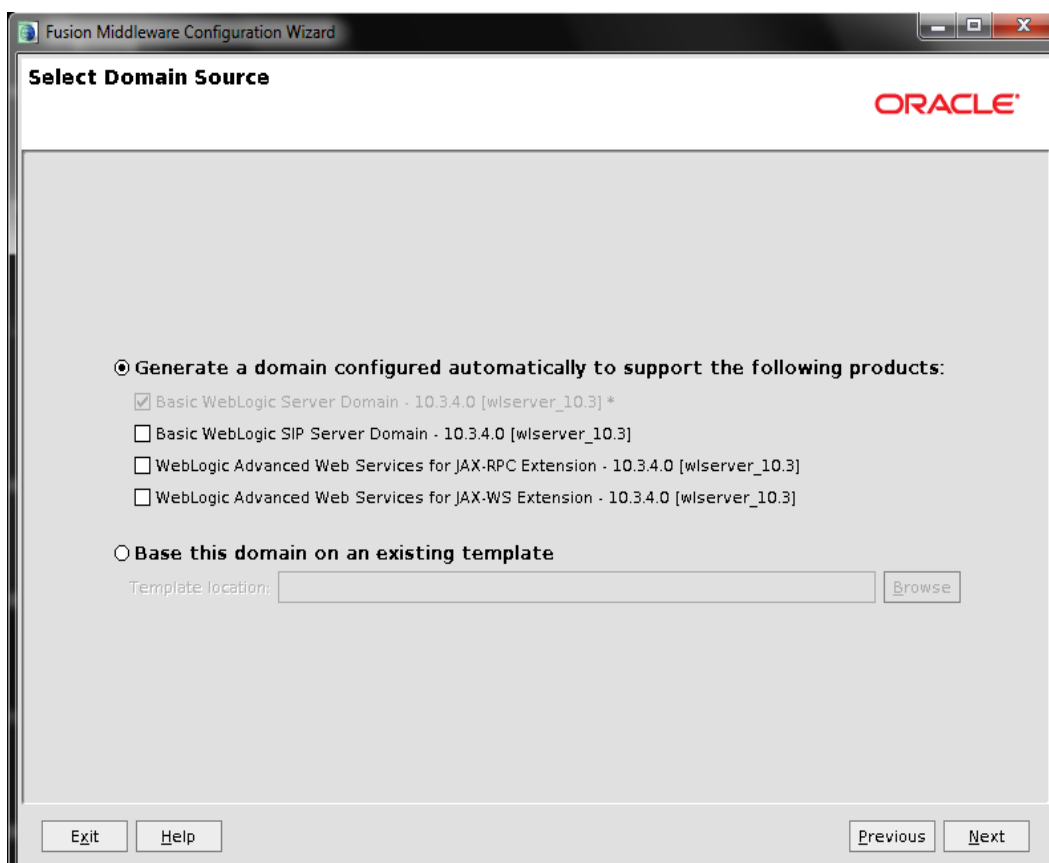
ORACLE

- Specify **Domain Name** and **Location**.
  Ensure to replace <WLS_HOME> with the full path. Click **Next**.
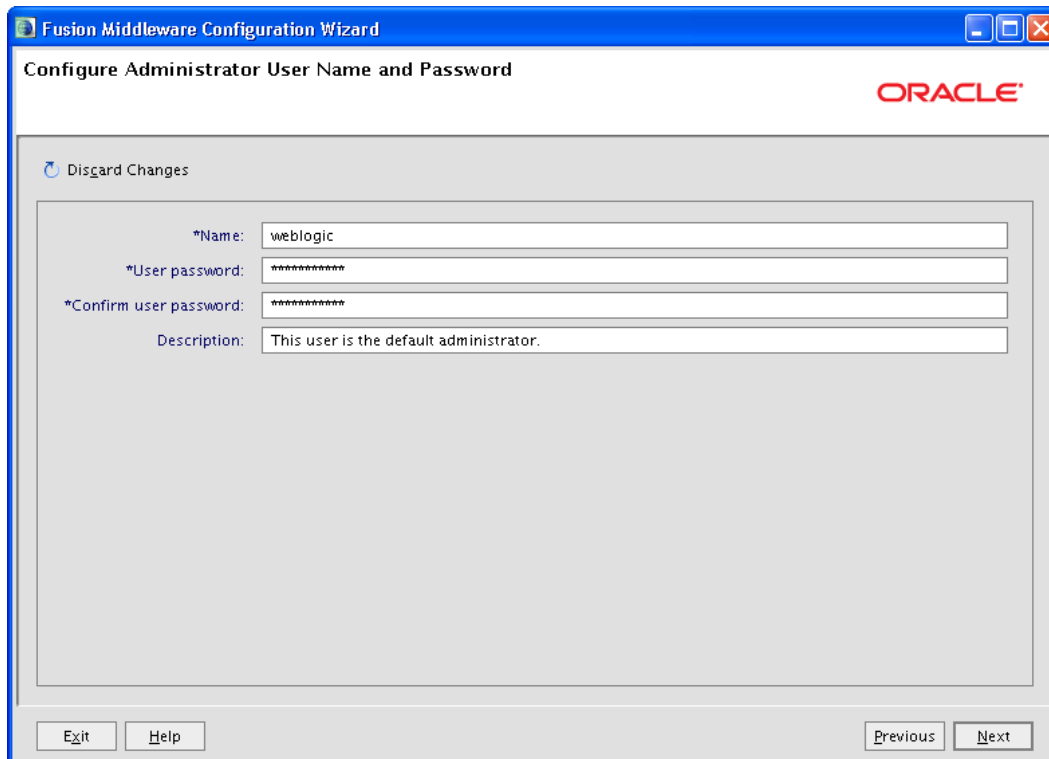
- Specify Administrator **User Name** and **Password**.
  - ➢ The specified credentials are used to access Administration console.
  - ➢ You can use this screen to define the default WebLogic Administrator account for the domain. This account is used to boot and connect to the domain's Administration Server. Click **Next**.
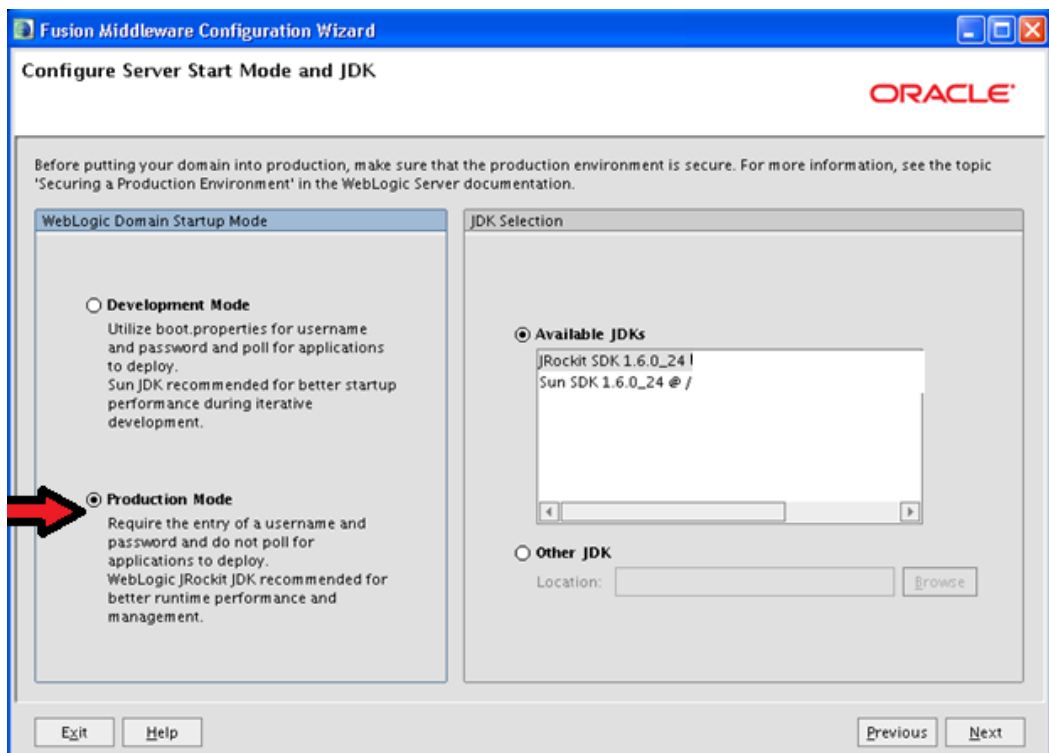
ORACLE®

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

- Select the check box adjacent to **Administration Server** and **Managed Servers, Clusters and Machines**. Click **Next**.

ORACLE®
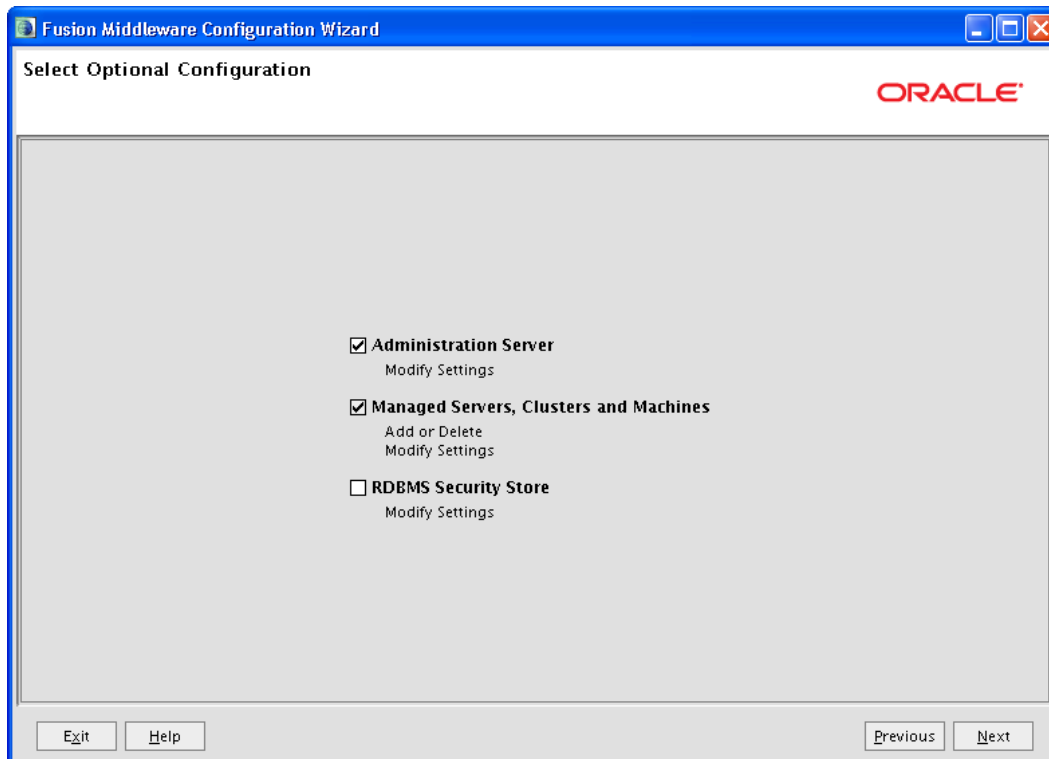
- Specify the Administration server **Listen address** and **Listen port**.

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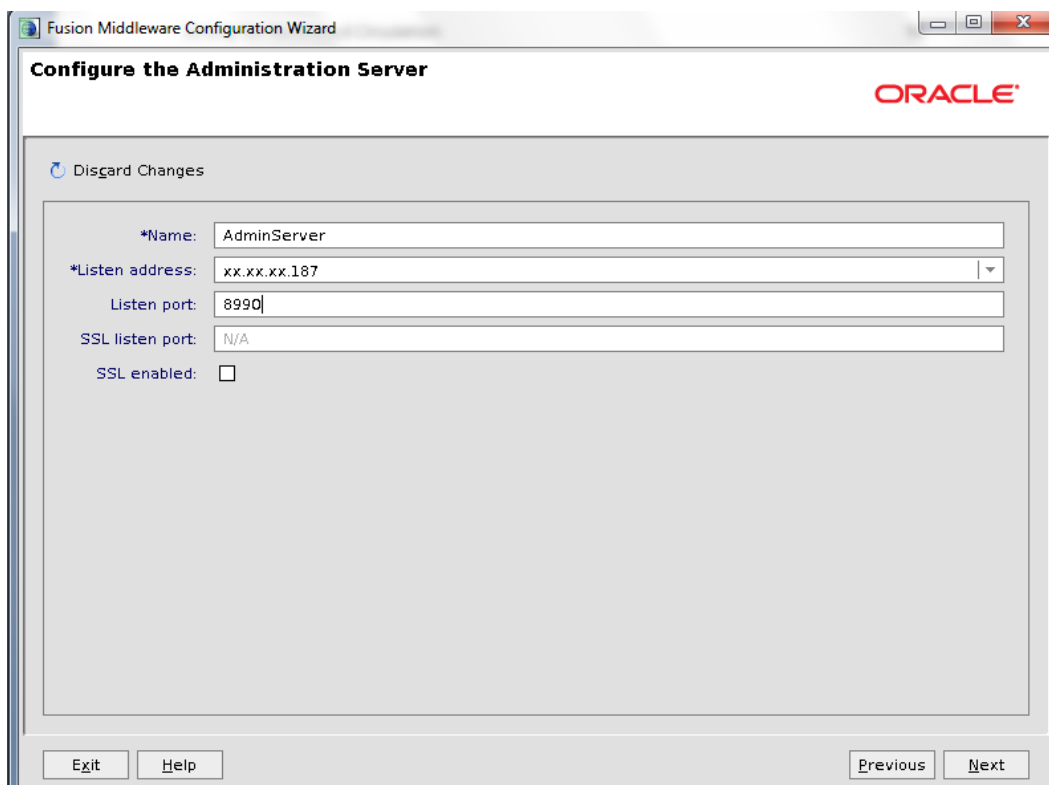
Note: The default Listen port is 7001. This could be changed to any other available port. Ensure to make a note, of this port since the same is required for launching the Admin console, post domain creation.

In this configuration 8990 port is input. This completes Admin Server configuration.

Note: Check for the port availability using the command - **netstat -anp |grep <Port no>**

The next screen navigates to **Managed Server configuration**.

3.2.1 Configure Managed Server

Note that, every 75 concurrent users require one managed server of size 4GB. I.e. for 300 concurrent FCUBS users, it is recommended to have 4 managed servers.

- Click on **Add** to add the desired number of Managed Servers.

This completes managed server configurations.

The next screen navigates to **Cluster creation and configuration**.

3.2.2 Cluster Creation and Configuration

- Click on **Add** to add a new cluster and specify the **Cluster Name** and select the Cluster messaging mode as **unicast**.
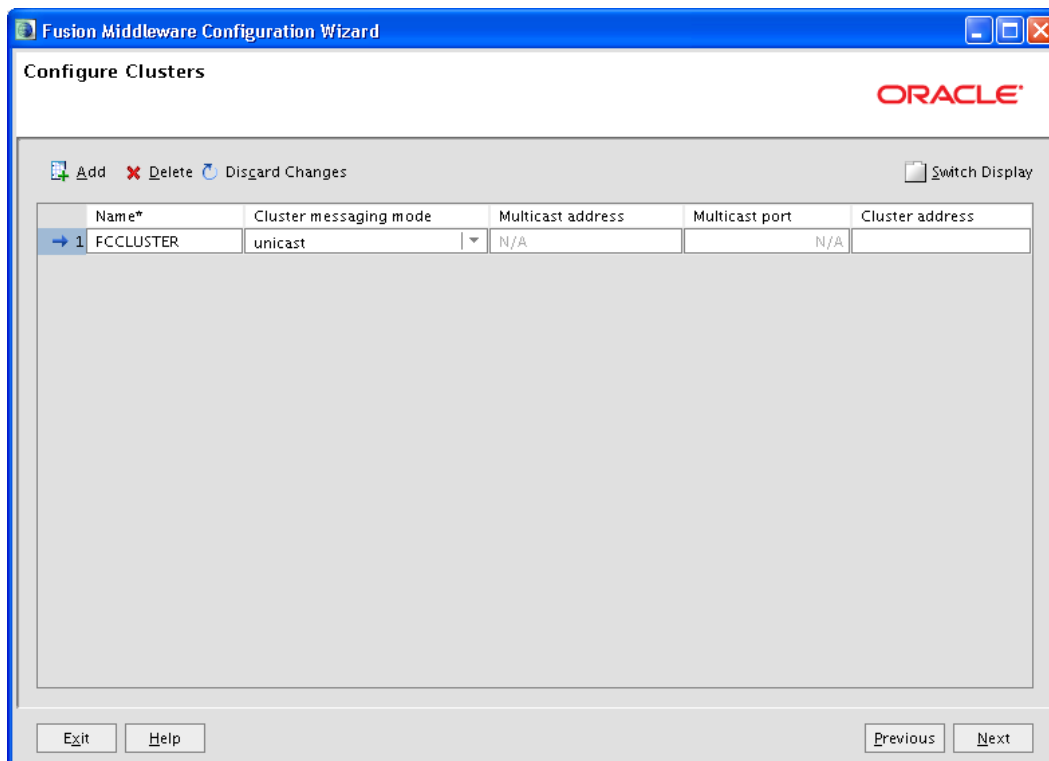
- Select the managed servers listed in the left pane and assign them to the cluster by clicking on the right arrow button.
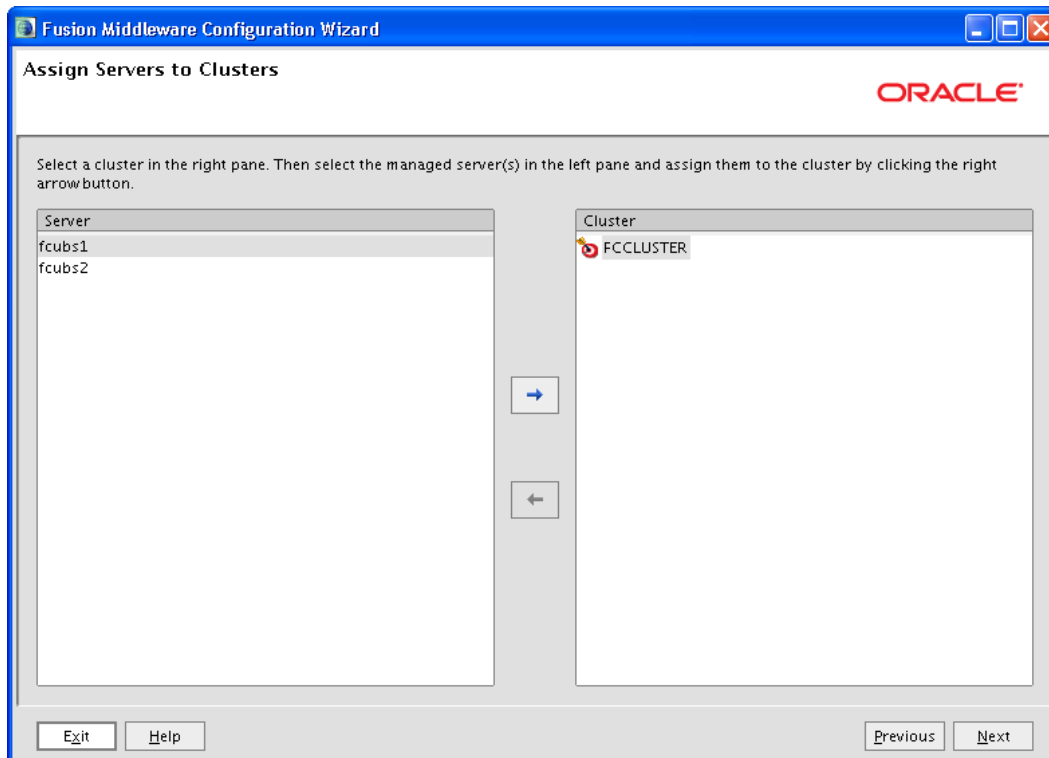
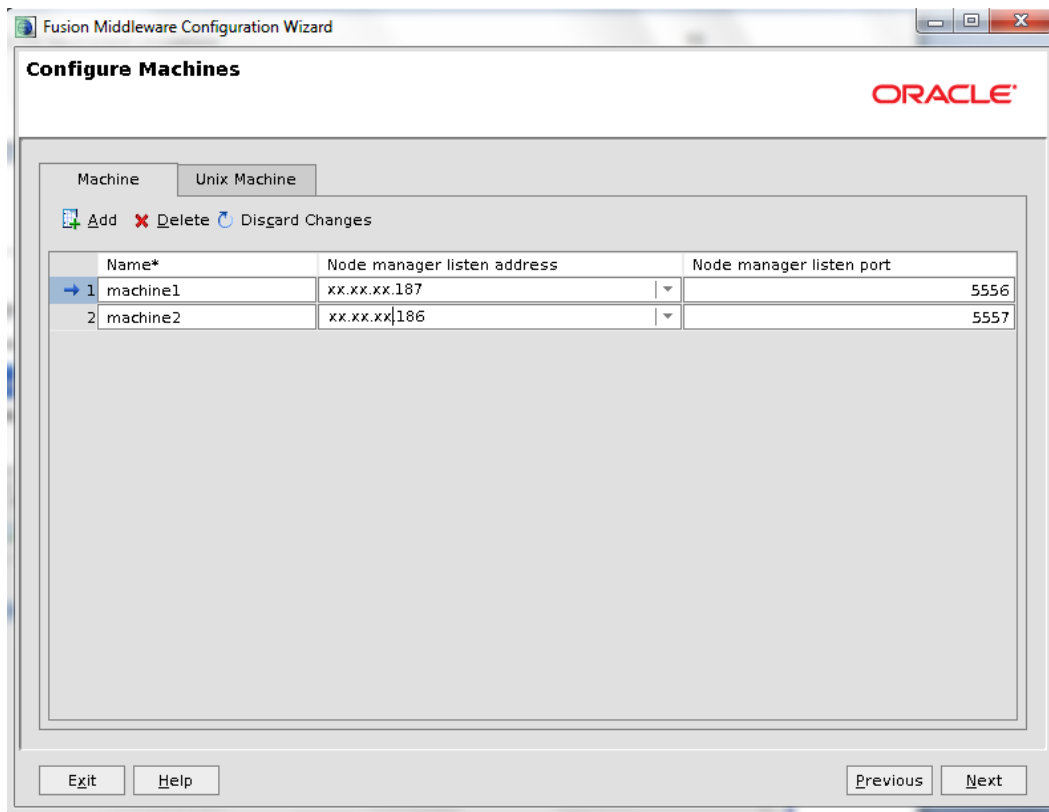- Following screen indicates post assigning managed servers to cluster.

ORACLE®
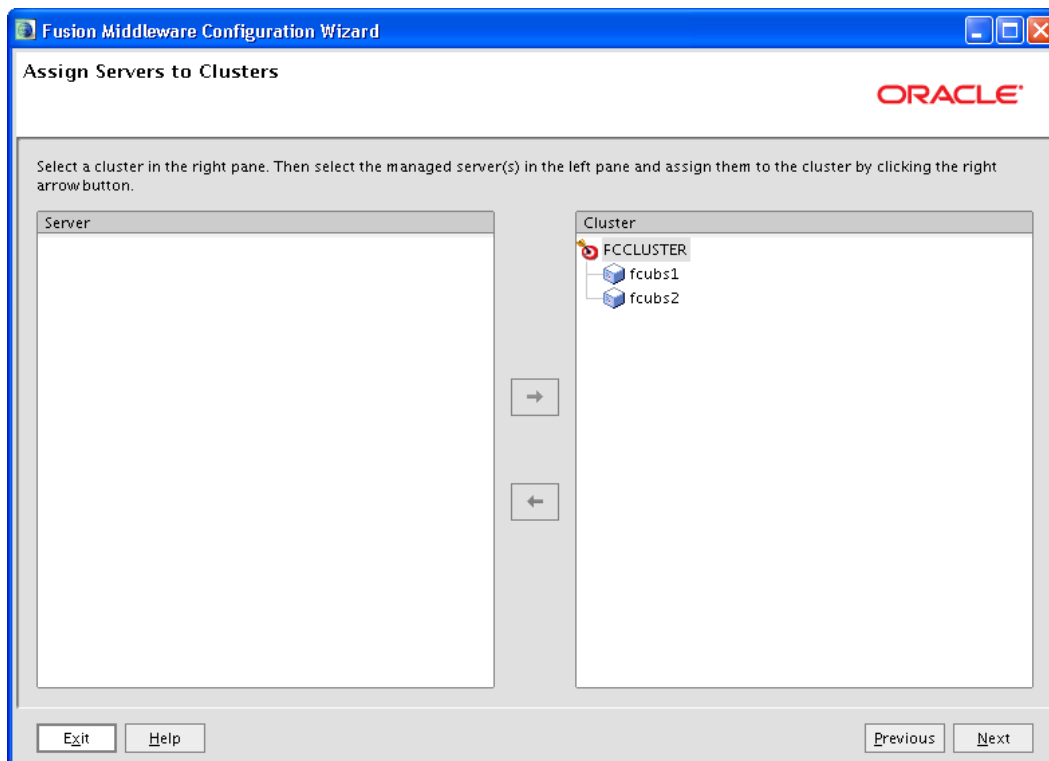
This completes Cluster configuration.

The next screen navigates to *Machine Configuration* screen.

3.2.3 Configure machines

- Click on **Add** to add the desired number of nodes.

- Depending on the host on which the managed server is created, the managed servers are assigned to respective Nodes. Select the managed Server(s) in the left pane and assign them to the desired Node by clicking the right arrow button.
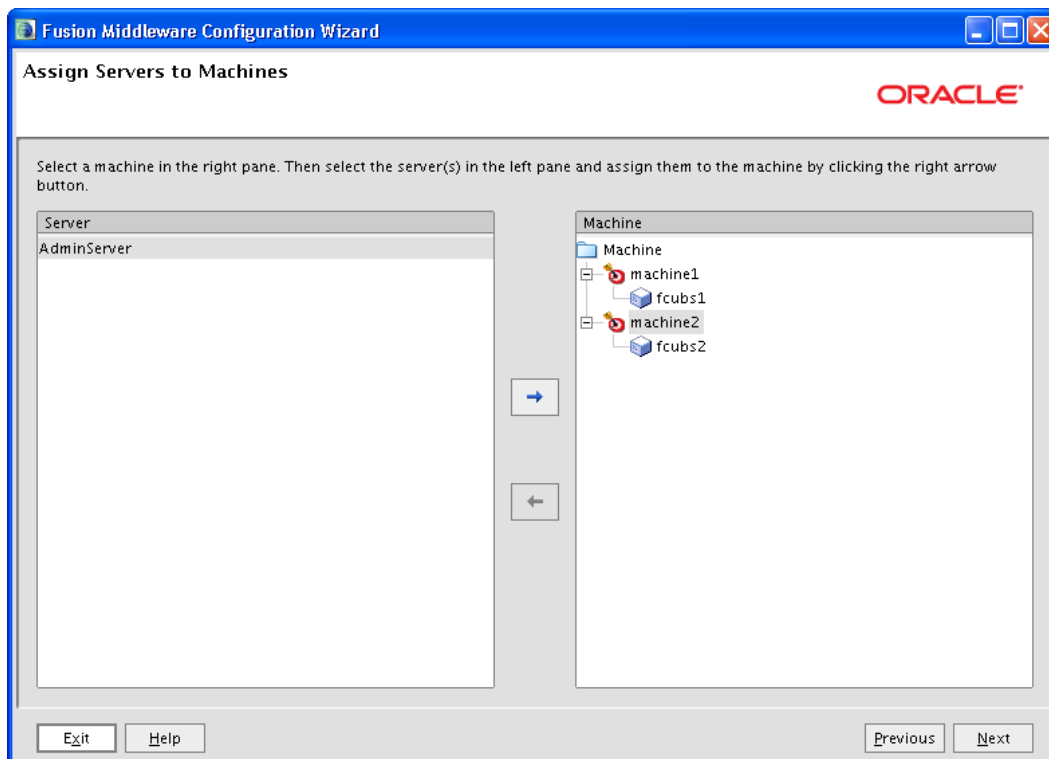
As per the above configuration,

- Managed server fcubs1 will start in server xx.xx.xx.187 and listen to port 8991.
- Managed server fcubs2 will start in server xx.xx.xx.186 and listen to port 8992.

The following configurations are completed:

1. Admin Server Configuration

2. Cluster Configuration

3. Machine configuration

- Click **Next** and view the summary screen.
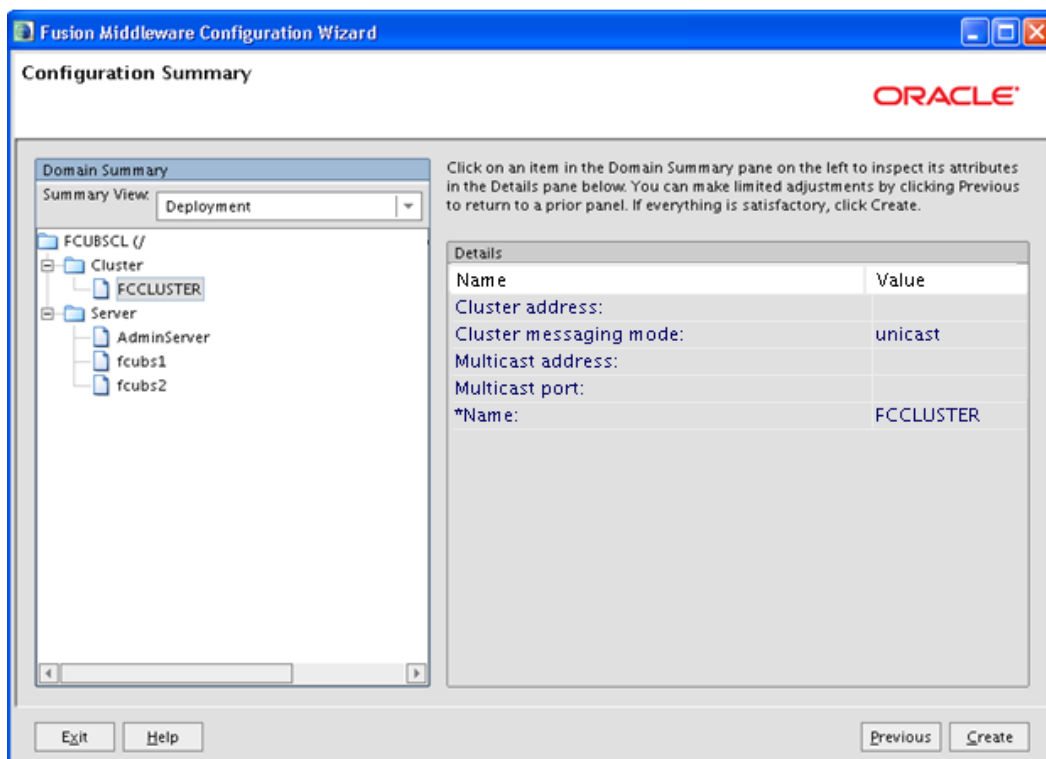
- Verify the details and click **Create**. The domain creation process is initiated and the progress of completion is indicated.
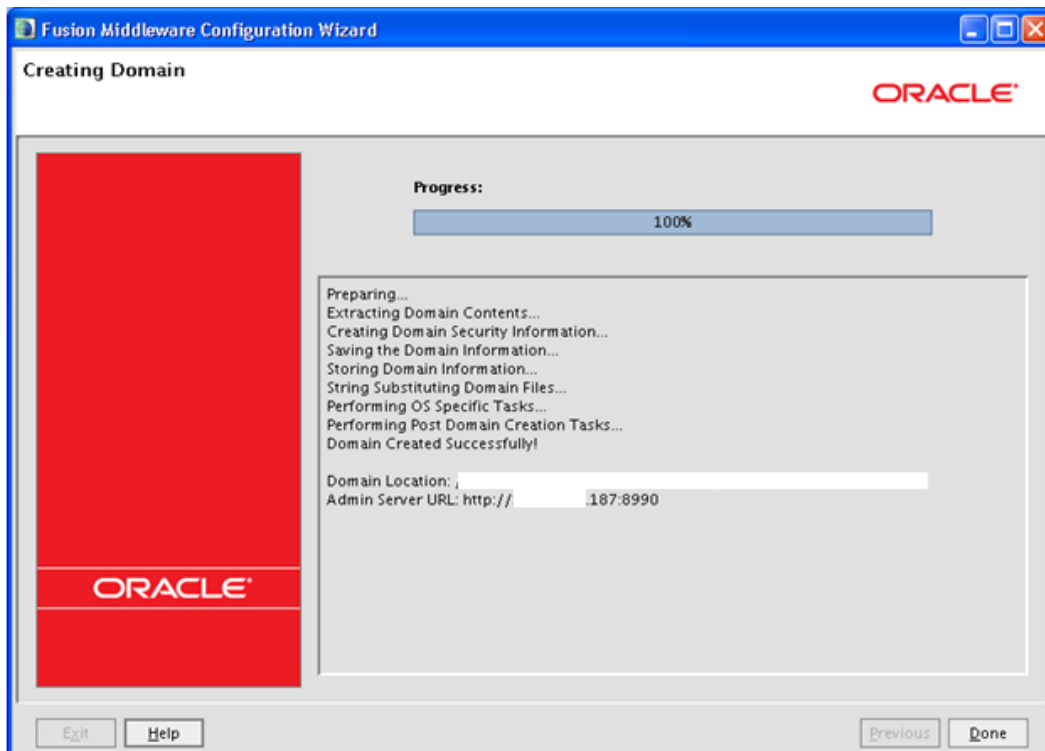
The Admin Server console URL is as indicated below:

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

- <IP address >: Host on which domain was created.
- <admin console port> : Port specified in Administration Server configuration page.

In this case the Admin Console URL is: http://xx.xx.xx.187:8990/console

4. Pack and Unpack Domain

The domain structure is expected to be copied to the second server (XXX186DOR) during domain creation. In some cases, the domain structure (domain directories and scripts) will not be copied to the second node (XXX186DOR) of the weblogic clusters. Hence to copy the same, you can use pack and unpack utility provided under $WLSHOME/common/bin.

4.1 <u>Pack</u>

Pack domain in xx.xx.xx.187 server:

```
./pack.sh -managed=true -
domain=$WLS_HOME/../user_projects/domains/FCUBSCL -
template=/tmp/FCUBSCL.jar -template_name="FCUBSCL"
```

4.2 <u>Unpack</u>

Unpack FTP FCUBSCL.jar in binary mode to server xx.xx.xx.186 under /tmp area and unpack the domain using unpack utility provided under $WLSHOME/common/bin

```
./unpack.sh -domain=$WLS_HOME/.. /user_projects/domains/FCUBSCL -
template=/tmp/FCUBSCL.jar
```

ORACLE

5. JVM Tuning

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

Note that, every 75 concurrent FLEXCUBE users require one managed server of size 4GB. I.e. for 300 concurrent FLEXCUBE users, it is recommended to have 4 managed servers.

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

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

Go to $JAVA_HOME/bin directory. Check java version using command **./java –d64 –version**

64 bit JVM shows the version details where as 32bit throws an error.

How to modify the JVM heap parameters?

To change the JVM heap parameters modify setDomainEnv.sh under domain FCUBSCL in both servers. This file is located at "$WL_HOME/user_projects/domains/$WLS_DOMAIN/bin"

Use USER_MEM_ARGS variable to override the standard memory arguments passed to java.

5.1 32 bit JVM

5.1.1 SUN JDK

```
USER_MEM_ARGS="-
Dorg.apache.xml.dtm.DTMManager=org.apache.xml.dtm.ref.DTMManagerDefault
-
Dorg.apache.xerces.xni.parser.XMLParserConfiguration=org.apache.xerces.
parsers.XML11Configuration -Dweblogic.threadpool.MinPoolSize=100 -
Dweblogic.threadpool.MaxPoolSize=100 –Xms1536M –Xmx1536M -
XX:MaxPermSize=256m -server -XX:+UseParallelOldGC -
XX:ParallelGCThreads=4"

export USER_MEM_ARGS
```

5.1.2 JROCKIT JDK

```
USER_MEM_ARGS="-
Dorg.apache.xml.dtm.DTMManager=org.apache.xml.dtm.ref.DTMManagerDefault
-
Dorg.apache.xerces.xni.parser.XMLParserConfiguration=org.apache.xerces.
parsers.XML11Configuration -Dweblogic.threadpool.MinPoolSize=100 -
Dweblogic.threadpool.MaxPoolSize=100 –Xms1536M –Xmx1536M -server"

export USER_MEM_ARGS
```

ORACLE

5.2 64 bit JVM

5.2.1 SUN JDK

```
USER_MEM_ARGS="-
Dorg.apache.xml.dtm.DTMManager=org.apache.xml.dtm.ref.DTMManagerDefault
-
Dorg.apache.xerces.xni.parser.XMLParserConfiguration=org.apache.xerces.
parsers.XML11Configuration -Dweblogic.threadpool.MinPoolSize=100 -
Dweblogic.threadpool.MaxPoolSize=100 -Xms4g -Xmx4g -XX:MaxPermSize=512m
-server -d64 -XX:+UseParallelOldGC -XX:ParallelGCThreads=4"

export USER_MEM_ARGS
```

5.2.2 JROCKIT JDK

```
USER_MEM_ARGS="-
Dorg.apache.xml.dtm.DTMManager=org.apache.xml.dtm.ref.DTMManagerDefault
-
Dorg.apache.xerces.xni.parser.XMLParserConfiguration=org.apache.xerces.
parsers.XML11Configuration -Dweblogic.threadpool.MinPoolSize=100 -
Dweblogic.threadpool.MaxPoolSize=100 -Xms4g -Xmx4g -server -d64"

export USER_MEM_ARGS
```

In server XXX187DOR,

File setDomain.sh under directory $WLS_HOME/../user_projects/domains/FCUBSCL/bin

In server XXX186DOR,

File setDomain.sh under directory $WLS_HOME/../user_projects/domains/FCUBSCL/bin

Note: Take a backup of the file before modifying the same.

ORACLE

6.   Managed server startup using script

It is recommended to start managed servers using scripts (startWeblogic.sh) when the managed servers are started from weblogic console.

To enable this option, set the following property in Nodemanager.properties file available under $WLSHOME/ common/ nodemanager in both servers:

```
StartScriptEnabled=true
```

In server xx.xx.xx.187 - Nodemanager.properties location: $WLS_HOME/common/nodemanager/

In server xx.xx.xx.186 - Nodemanager.properties location: $WLS_HOME/common/nodemanager

Note: Node manager must be restarted in all servers (xx.xx.xx.187 & xx.xx.xx.186) to activate these settings.

How to restart node manager?

- Locate node manager pid using *ps -ef|grep weblogic.nodemanager.javaHome*
- Change directory to $WLSHOME/server/bin
- Kill the unix process using *kill -9 <pid>*
- Verify that the node manager is killed by *tail –f nohup.out*
- Start node manager using *nohup ./startNodeManager.sh &*
- Verify nodemanager is started using *tail –f nohup.out*

Similarly, restart nodemanager in server xx.xx.xx.186.

6.1   Starting Admin server

To start the Admin server in xx.xx.xx.187 server, execute **startWeblogic.sh** available under *$DOMAIN_HOME/bin.*

In this example startWeblogic.sh script is under $WLS_HOME/../user_projects/domains/FCUBSCL/bin/ under server xx.xx.xx.187

Note: Refer to Appendix section to know about the possible admin server startup problems.

6.2   Starting managed servers

To start the managed servers, do the following:

- Login to Administration console using URL http://xx.xx.xx.187:8990/console. The User name and password set during domain creation are auto populated.

ORACLE
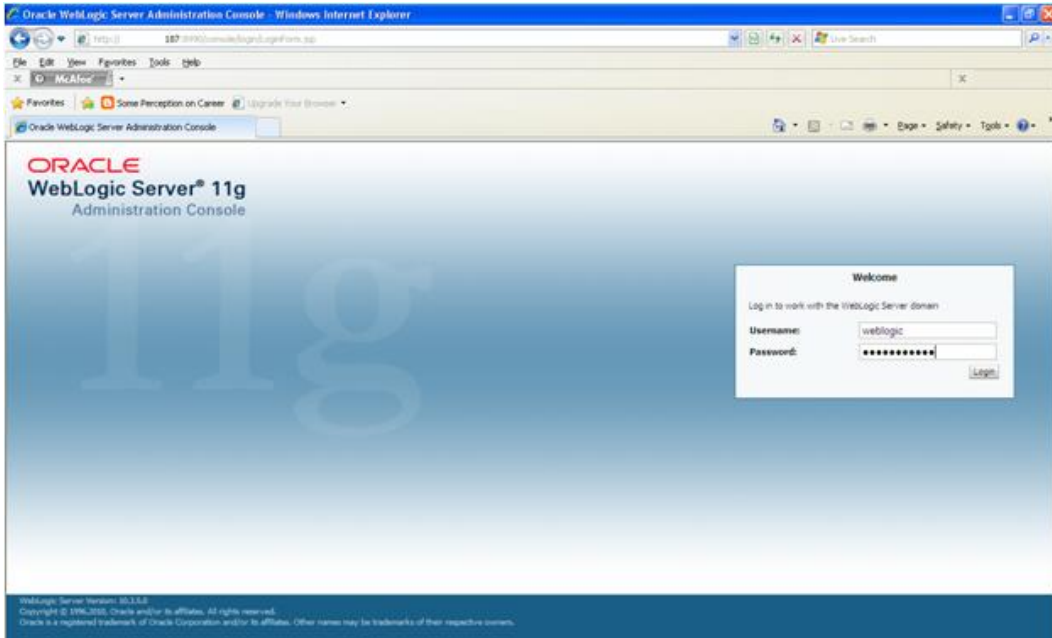
- Click **Login**.
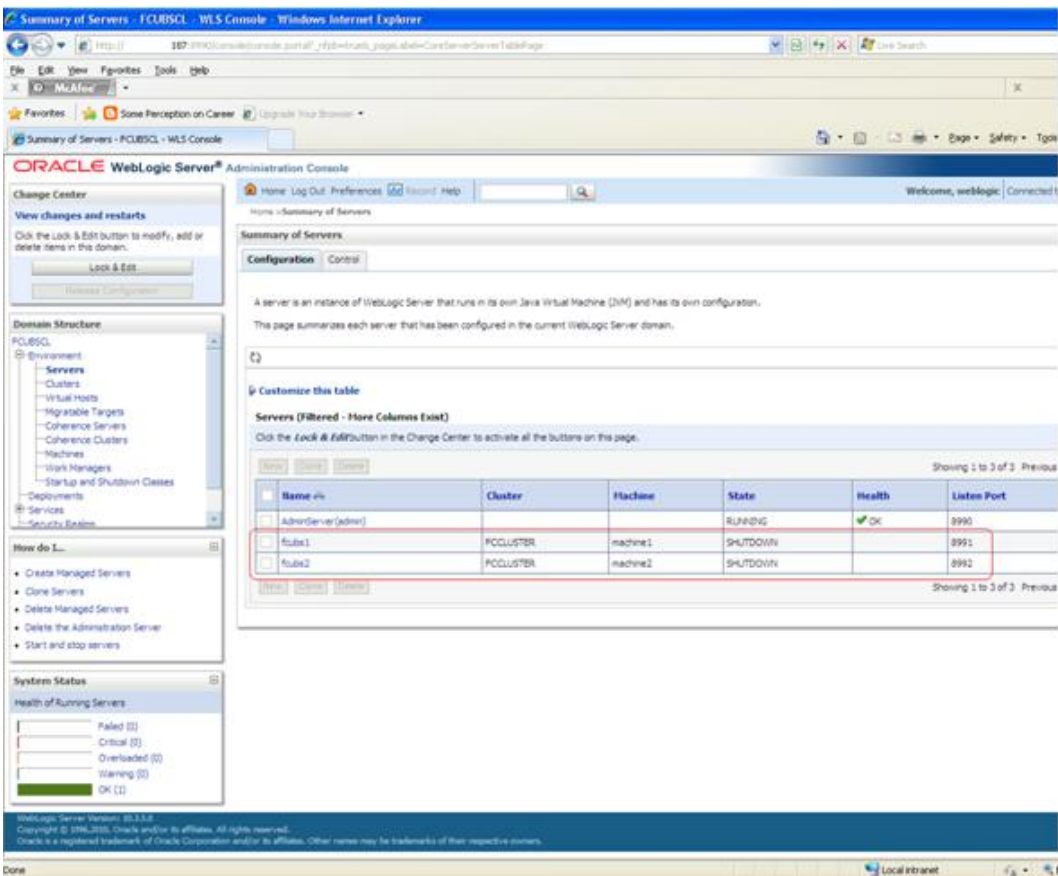
- This screen provides details of both managed servers which are currently down. Both managed servers are part of FCUBSCL domain.
    - Fcubs1 managed server is associated with machine1 that is xx.xx.xx.187 server.

ORACLE

➢ Fcubs2 managed server is associated with machine2 that is xx.xx.xx.186 server.

6.3 <u>Verifying machine status</u>

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

- In the console, navigate through Domain structure > Machines > machine1 > Monitoring > Node Manager Status.
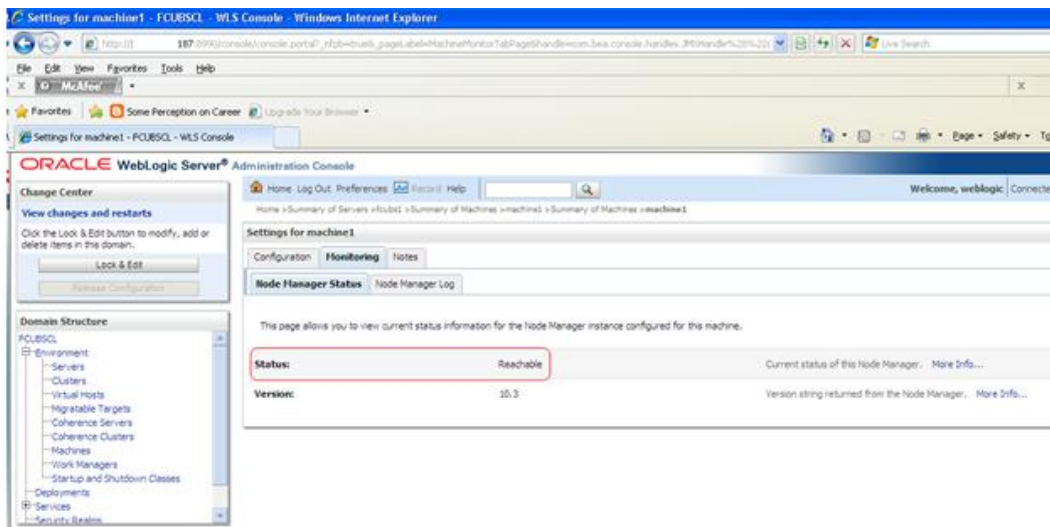
- In the console, navigate through Domain structure > Machines > machine2 > Monitoring > Node Manager Status.
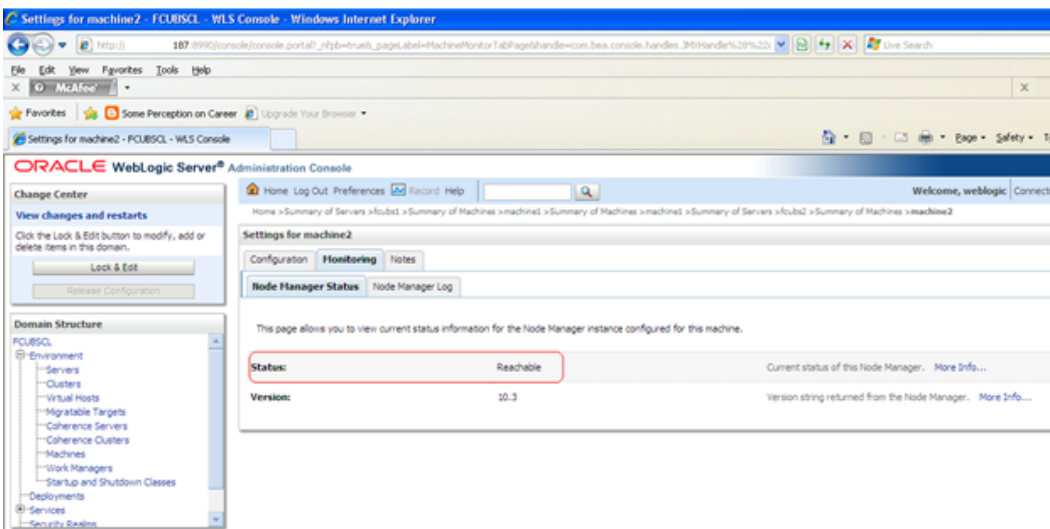
To start the managed server:

- In the console, navigate through Domain structure > Environment > servers > control.
- Select the adjacent checkbox and click **Start**.

ORACLE
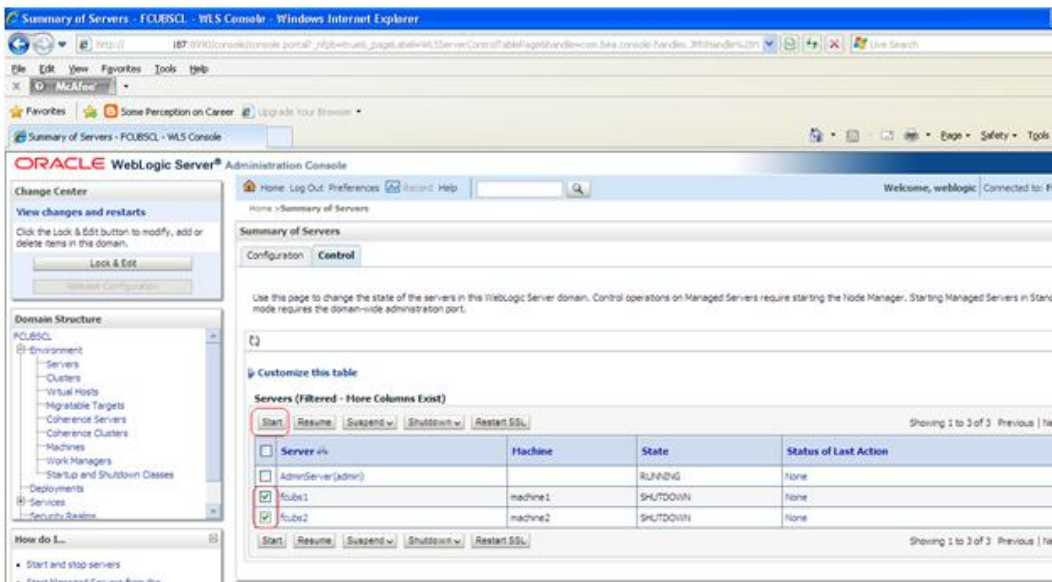
- The status of Managed server is indicated as "**RUNNING**".

ORACLE®
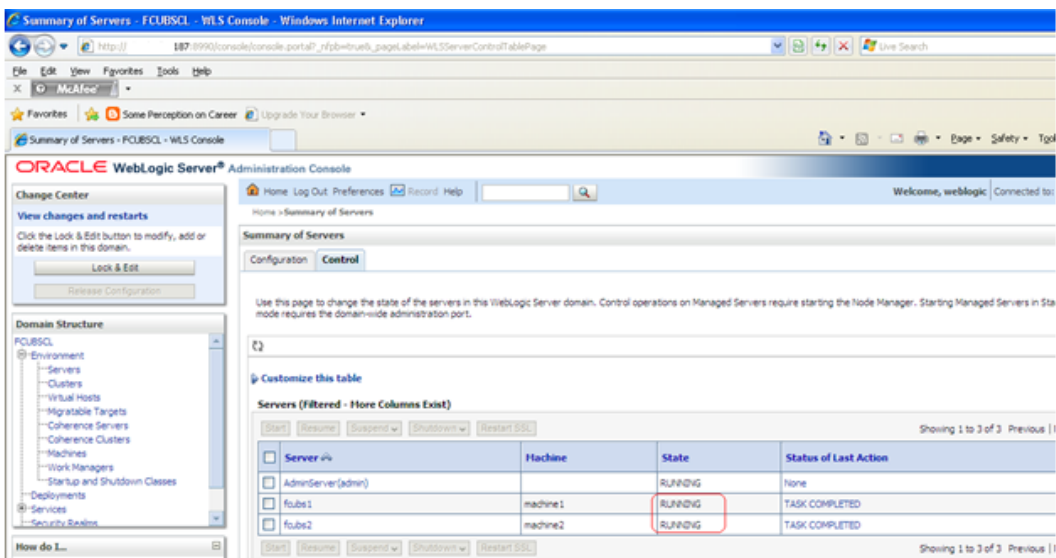
7.   Data Source creation and JDBC Configuration

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

- jdbc/fcjdevDS - This datasource is used by FLEXCUBE online screen excluding branch screens.
- jdbc/fcjdevDSBranch - This datasource is used by Branch screens.
- jdbc/fcjSchedulerDS - This datasource is used by Quartz scheduler.

Note:

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

7.1 <u>For Oracle Single Instance Database</u>

- Click on Services > Data Sources on the left pane.
- Click on Lock & edit button and select New > Generic data source. (jdbc/fcjdevDS)
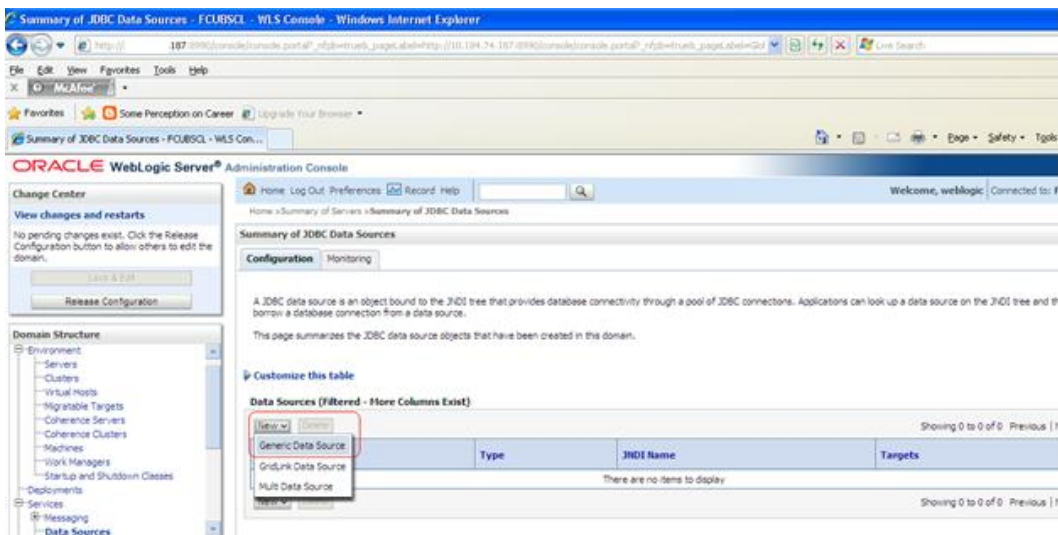
- Provide the following details,
    - Name: FCUBSDS
    - JNDI Name: jdbc/fcjdevDS
    - Database Type: Oracle
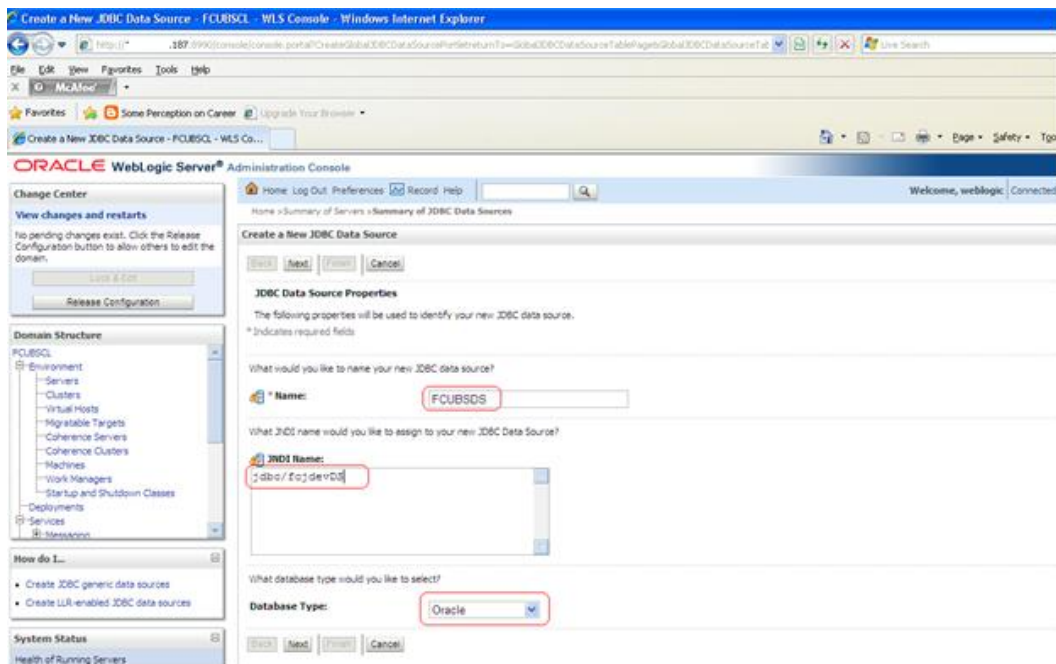
ORACLE

- **Select the Database Driver**: Oracle Driver (Thin) for instance connections; Versions:9.0.1 and later. Ensure that the Driver selected has "Oracle Driver (Thin)", the version can be the latest available.
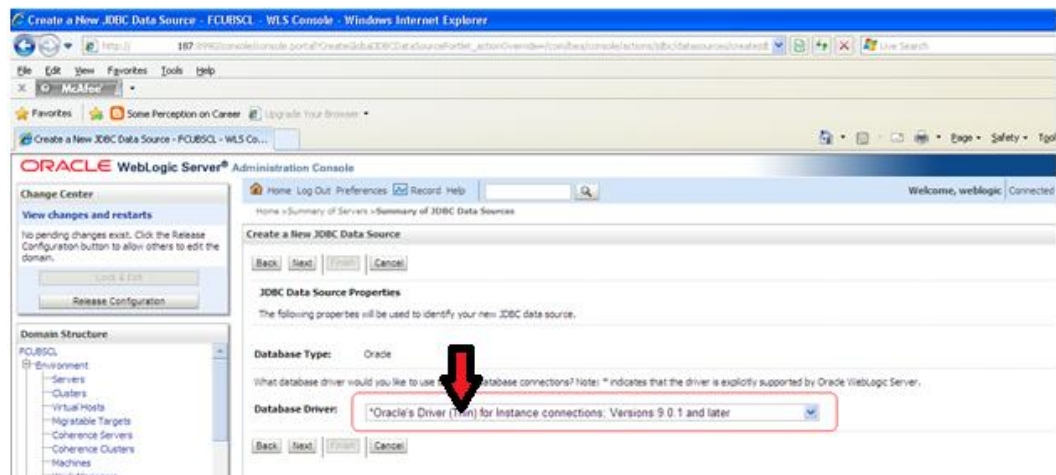
- Uncheck **Support Global Transaction** checkbox. Click **Next**.
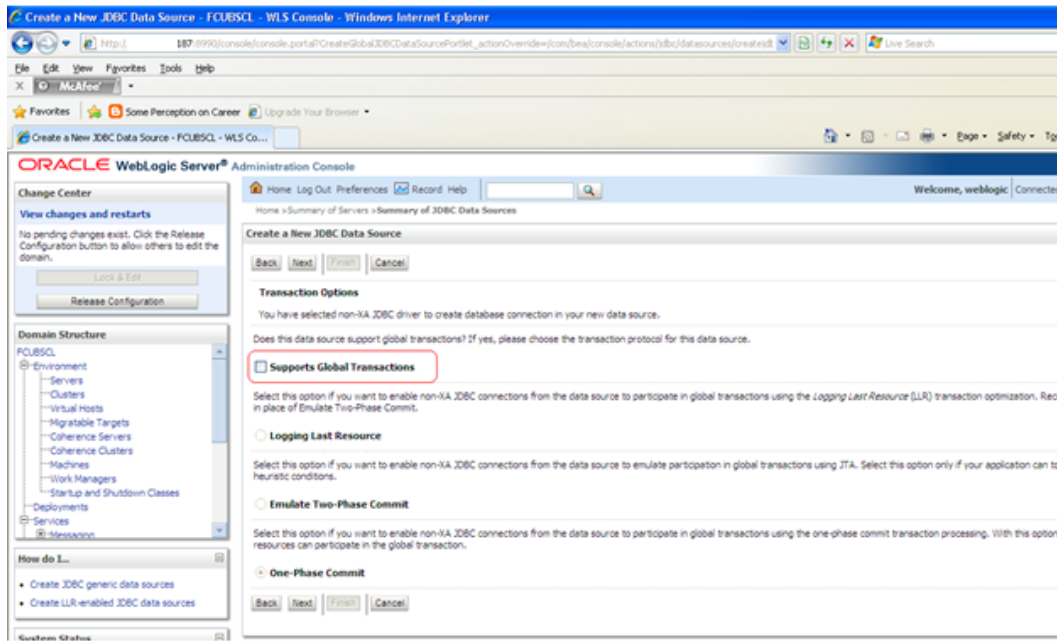
ORACLE

- Specify the Oracle database details such as Database Name, Host Name, Port, Database User Name, and Password. Click **Next**.
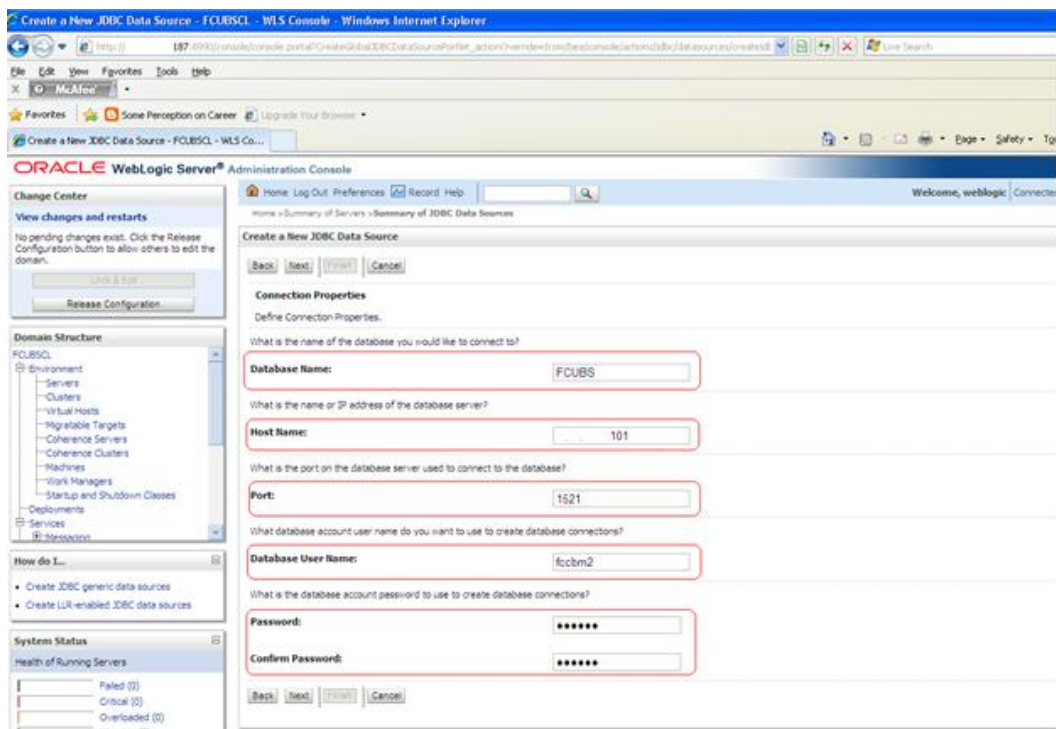
Make sure that in URL, we make the changes to reflect oci ,

Default URL: jdbc:oracle:thin:@<IP_Adress>:<Port>:<INSTANCE_NAME>.

Change the default URL to:

jdbc:oracle:oci: @<IP_Adress>:<Port>:<INSTANCE_NAME>

then click **Test Configuration**.

ORACLE

- In **Select Targets** section, select the target as cluster **FCCLUSTER**. Click **Finish**.

- Click on **Activate Changes** on the LHS menu to configure the changes.
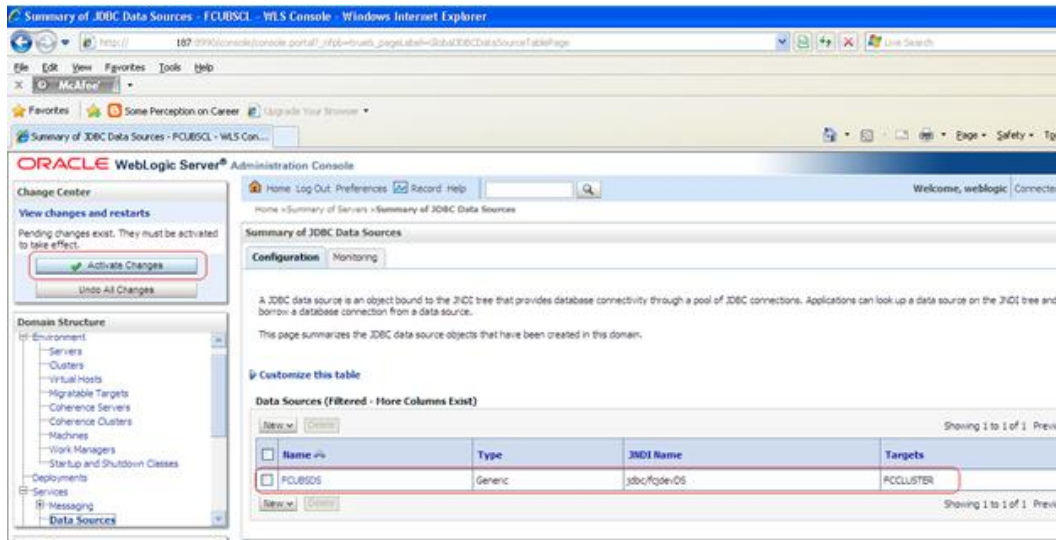
ORACLE®

7.1.1 Data source creation : jdbc/fcjdevDSBranch

- Select the Data Source from the drop down list.
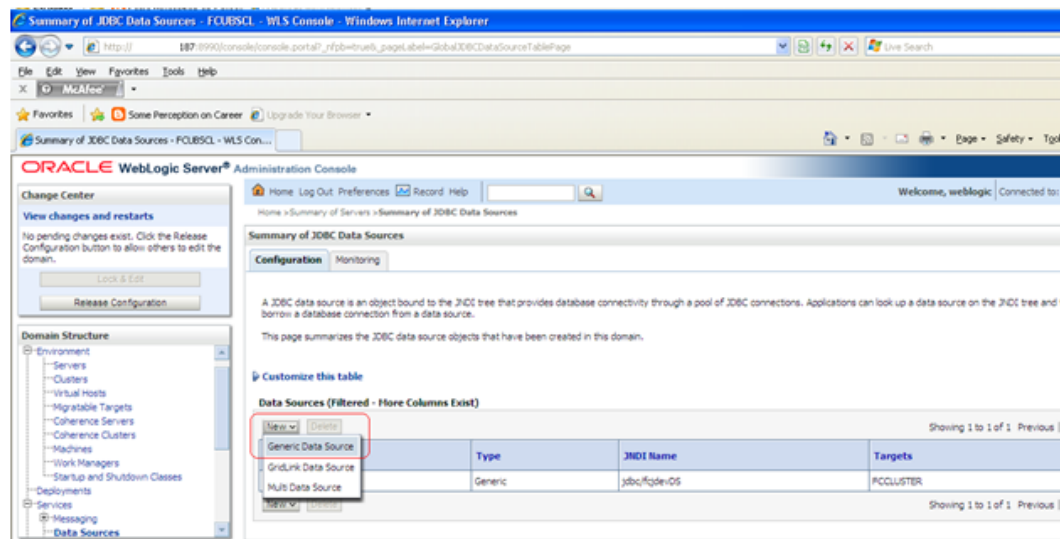
- Enter JDBC data Source Properties such as JDBC data Source Name, JNDI Name, and Database Type. Click **Next**.
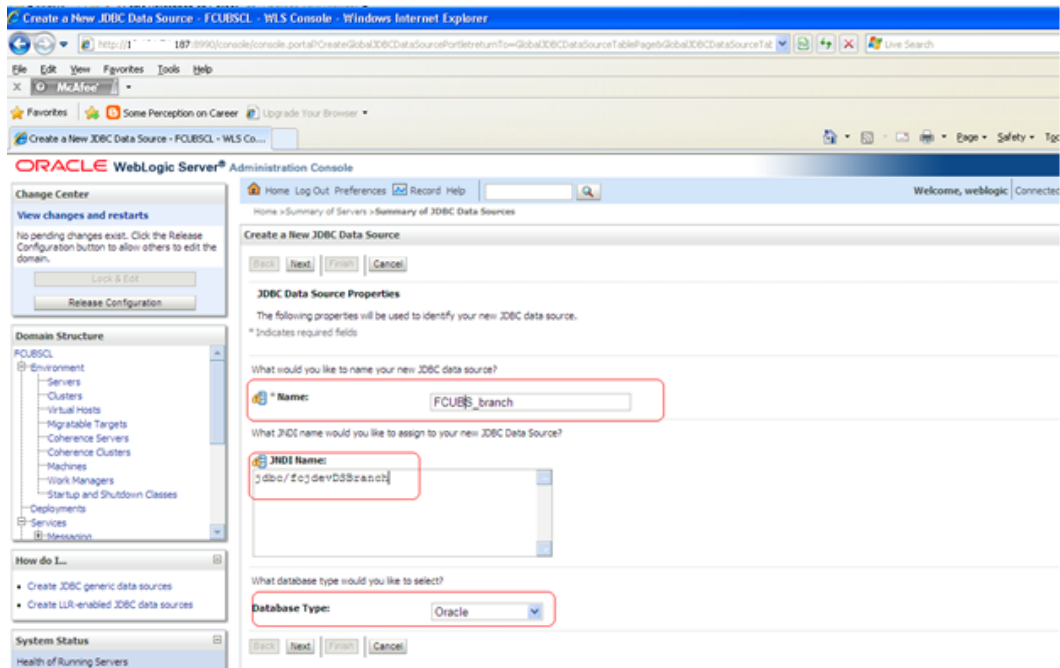
ORACLE®

- Select the Database Driver as "Oracle's Driver (Thin XA) for instance connections; Versions 9.0.1 and later". Ensure to select the latest available version of the database driver.
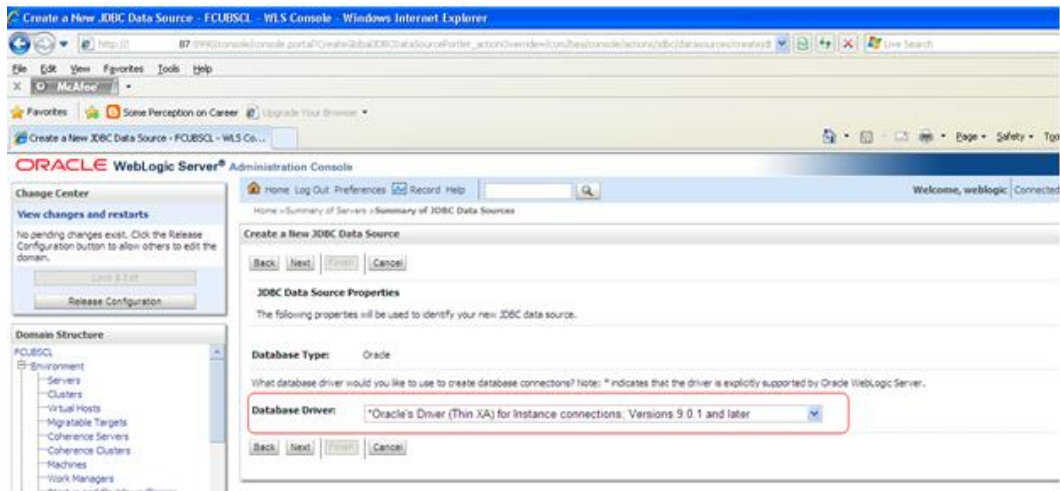
- Enter database connection details such as Database Name, Host Name, Port, Database User Name, and Password. Click **Next**.

ORACLE

- Click **Test configuration**.

- In **Select Targets** section, select the target as cluster **FCCLUSTER**. Click **Finish**.

ORACLE

- Click on **Activate Changes** to activate the configuration.

7.1.2 Data source creation : jdbc/fcjSchedulerDS

Create a data source jdbc/fcjSchedulerDS similar to jdbc/fcjdevDSBranch created above.

Data sources are created as follows,

Data Sources (Filtered - More Columns Exist)

Click the *Lock & Edit* button in the Change Center to activate all the buttons on this page.

New ↴	Delete			Showing 1 to 5 of 5 Previo
☐ Name △	Type	JNDI Name		Targets
☐ FCUBSDS	Generic	jdbc/fcjdevDS		FCCLUSTER
☐ fcubs_async	Generic	jdbc/fcjdevDS_ASYNC		FCCLUSTER
☐ FCUBS_Branch	Generic	jdbc/fcjdevDSBranch		FCCLUSTER
☐ FCUBS_sch	Generic	jdbc/fcjSchedulerDS		FCCLUSTER
New ↴	Delete			Showing 1 to 5 of 5 Previo

7.1.3 Data source creation: for Oracle RAC database

You can create JDBC datasource for RAC databases using Single Client Access Name IP or SCAN – IP. 11gR2 Grid Infrastructure (CRS) introduced Single Client Access Name (SCAN) to simplify client access to database services.

Note: From oracle database 11g Release 2 onwards. it is highly recommend using SCAN IP.

- Choose **New** > **Generic Data Source**.

ORACLE®

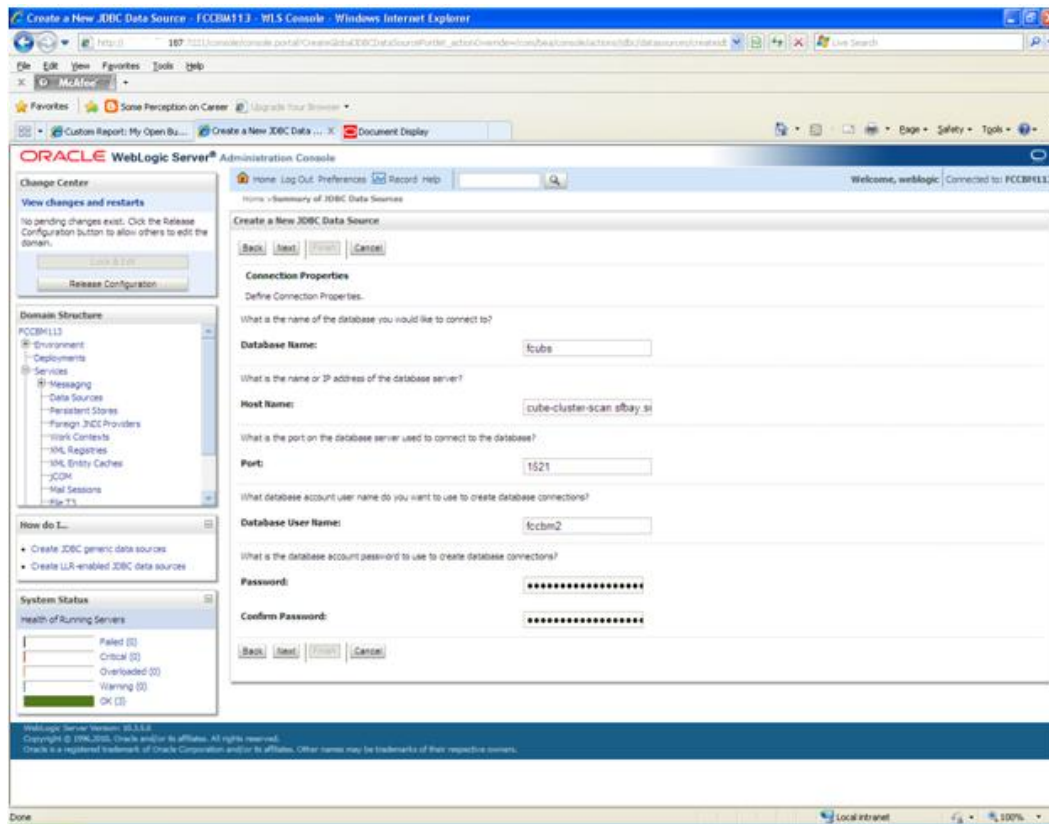- Specify the **Name** and **JNDI Name**. Click **Next**.

- Select the appropriate Database Driver. Ensure to select the latest available driver and click **Next**.
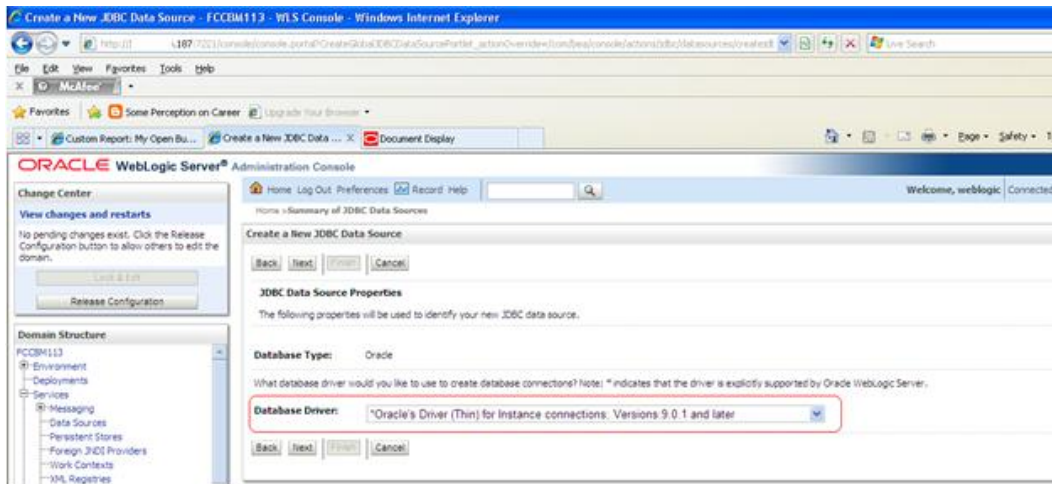
ORACLE

- Deselect the **Support Global Transactions** checkbox. Click **Next**.
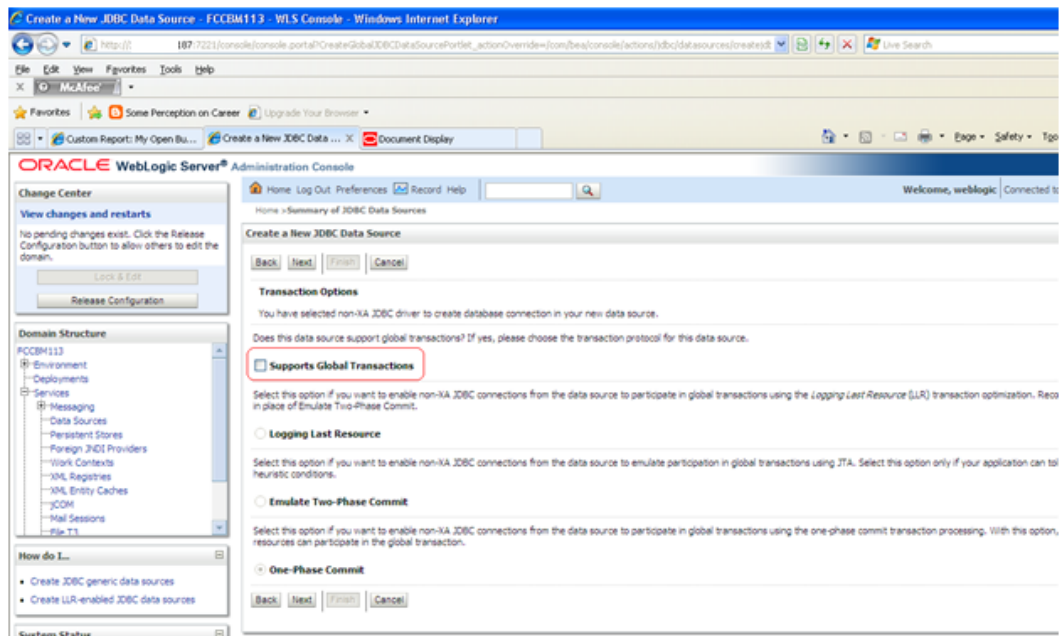
- Specify the **Host Name** and the **SCAN IP** details. Click **Next**.

ORACLE

- Replace the URL in the following way:

jdbc:oracle:thin:@(DESCRIPTION=(ADDRESS_LIST=(ADDRESS=(PROTOCOL=TCP)(HOST=cube-cluster-scan.sfbay.sun.com)(PORT=1521)))(CONNECT_DATA=(SERVICE_NAME=fcubs)))

Where,

- Scan IP = cube-cluster-scan.sfbay.sun.com
- Service Name = fcubs
- Port = 1521

ORACLE

Make sure that in URL, we make the changes to reflect oci ,

Default URL: jdbc:oracle:thin:@<IP_Adress>:<Port>:<INSTANCE_NAME>.

Change the default URL to:

 jdbc:oracle:oci: @<IP_Adress>:<Port>:<INSTANCE_NAME>

then Click on **Test Configuration**. The connection test should be successful.

- In **Select Targets** section, select the target as cluster **FCCLUSTER**. Click **Finish**.

- Click **Finish** to complete the data source creation.

Similarly you can create other data sources with appropriate driver.

ORACLE

8. JDBC Tuning

The following JDBC tuning parameters are detailed below:

- General Parameters
- Managed Server & Http Logging
  - General
  - HTTP

8.1 General Parameters

PARAMETER	VALUE	Navigate To
JTA Time out seconds	18000	• Login to Weblogic Server console. • Click on the domain name (ex: FCUBS) which is under 'Domain Structure'. • Go to Configuration > JTA, parameter and values is found on the right side panel of console.
Session Timeout	900	• Login to Weblogic Server console • Click on Deployments which is under 'Domain Structure'. • Click on the deployed FCJ application from right side panel. • Click on FCJNeoWeb from 'Modules and components' • Go to Configuration General, the parameter values can be found here.
Stuck Thread Max Time	18000	Server > Configuration > Tuning > Stuck Thread Max Time (Should be set for all managed servers)

jdbc/fcjdevDS

jdbc/fcjdevDSBranch

PARAMETER	VALUE	Navigate To
Connection Reserve time out	30	Connection Pool->Advance
Test Frequency	60	Connection Pool->Advance
Inactive connection time out	30	Connection Pool->Advance

ORACLE

Initial Capacity	1	Connection Pool
Max capacity	100	Connection Pool
Capacity Increment	5	Connection Pool
Shrink Frequency	1800	Connection Pool->Advance
Test Connection on Reserve	Checked	Connection Pool->Advance

jdbc/fcjSchedulerDS

PARAMETER	VALUE	Navigate To
Connection Reserve time out	30	Connection Pool->Advance
Test Frequency	60	Connection Pool->Advance
Inactive connection time out	30	Connection Pool->Advance
Initial Capacity	1	Connection Pool
Max capacity	20	Connection Pool
Capacity Increment	2	Connection Pool
Shrink Frequency	900	Connection Pool->Advance
Test Connection on Reserve	Checked	Connection Pool->Advance

8.2   Managed server & HTTP logging

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

The following changes are to be done for all the managed servers.

8.2.1   General

Navigate to Managed server > Logging > General > Advanced option.

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

ORACLE

8.2.2 <u>HTTP</u>

Navigate to Managed server > Logging > HTTP

Access Logs	Unchecked

ORACLE

9.   Oracle WebLogic Loadbalancing

For Weblogic Load balancing, use Oracle HTTP Server.

Best practice document for Oracle HTTP Server is available.

ORACLE®

10. Appendix

This section of the documents explains about the resolved issues for reference.

10.1 <u>Known Issues</u>

Following are the issues that you might face during the Admin server startup.

1. Server is running in Production Mode and Native Library (terminalio) to read the password securely from command line is not found.

Solution:

*Add -**Dweblogic.management.allowPasswordEcho=true** in Startweblogic.sh as per Metalink doc Id **1354736.1,***

2. Admin server failed to start with following errors:

Error1

Error2

ORACLE

```
>
<Jan 13, 2012 7:32:34 PM IST> <Critical> <WebLogicServer> <BEA-000386> <Server subsystem failed.
Reason: weblogic.security.SecurityInitializationException: Authentication for user  denied
weblogic.security.SecurityInitializationException: Authentication for user  denied
        at weblogic.security.service.CommonSecurityServiceManagerDelegateImpl.doBootAuthorizatio
(CommonSecurityServiceManagerDelegateImpl.java:965)
        at weblogic.security.service.CommonSecurityServiceManagerDelegateImpl.initialize(CommonS
curityServiceManagerDelegateImpl.java:1050)
        at weblogic.security.service.SecurityServiceManager.initialize(SecurityServiceManager.ja
a:873)
        at weblogic.security.SecurityService.start(SecurityService.java:141)
        at weblogic.t3.srvr.SubsystemRequest.run(SubsystemRequest.java:64)
        Truncated. see log file for complete stacktrace
Caused By: javax.security.auth.login.FailedLoginException: [Security:090304]Authentication Faile
: User  javax.security.auth.login.LoginException: [Security:090301]Password Not Supplied
        at weblogic.security.providers.authentication.LDAPAtnLoginModuleImpl.login(LDAPAtnLoginM
duleImpl.java:261)
        at com.bea.common.security.internal.service.LoginModuleWrapper$1.run(LoginModuleWrapper.
ava:110)
        at java.security.AccessController.doPrivileged(Native Method)
        at com.bea.common.security.internal.service.LoginModuleWrapper.login(LoginModuleWrapper.
ava:106)
        at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
```

Solution

To solve the above errors,

1. Create a new directory **security** under
   $WLS_HOME/../user_projects/domains/FCUBSCL/servers/AdminServer and create a file
   boot.properties. Add weblogic console username and password in it
2. If there is a directory ldap exists under
   $WLS_HOME/../user_projects/domains/FCUBSCL/servers/AdminServer/data, then
   rename directory ldap to ldap_old and start the Admin server.

This recreates the ldap setting with new password and starts the admin server.

ORACLE

ORACLE®

FCUBS Weblogic Middleware Practices
[Novmeber] [2015]
Version 3.0
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/

ORACLE®