WebLogic Multitenancy Workshop

Contents

INTRODUCTION:

Multitenancy in WebLogic Server provides a sharable infrastructure for use by client organizations (tenants). By allowing one domain to support multiple tenants, WLS MT improves density and achieves a more efficient use of resources while eliminating the trade-offs that are traditionally made in a shared environment: Isolation issues. Multitenancy essentially creates the tension between isolation and sharing. Isolation separates both the administration and runtime of different tenants from each other, where resource sharing among tenants improves efficiency and reduces operation costs.

Domain Partition:

WebLogic Server MT provides resource isolation with in domain partitions, an administrative and runtime slice of a WebLogic domain that is dedicated to running application instances and related resources for a tenant. Domain Partition achieve greater density by allowing application instances and related resources to share the domain, WebLogic Server itself, the Java virtual machine, and the operating system while isolating tenant specific application data, configuration, and runtime traffic. Each domain partition has its own runtime copy of the application and resources.

Resource Groups:

WLS MT introduces resource groups, simply as a convenient way to group together Java EE applications and the resources they use into a distinct administrative unit within the domain. The resources and applications are "fully qualified" in that administrator provides all information needed to start or connect to those resources, including credentials for connecting to data source and targeting information for Java EE application. A resource group will either contain these deployable resources directly or refer to a resource group templates which contain the resources. Resource group can be defined at the domain level, or be specific to domain partition.

All the resources in or referenced by a resource group are targeted together (to the same target). Resource group can be started and stopped.

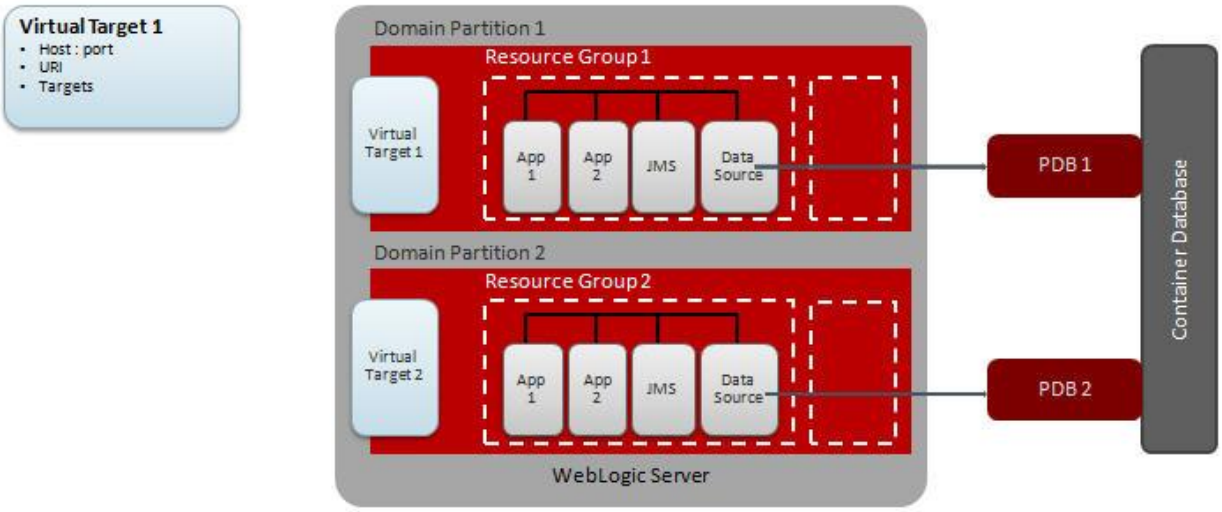
Virtual Target:

Encapsulate where a partition or resource group runs and how to route traffic to them, including addresses, protocol settings, and targeting, Request routing is determined by the host name and optional URI.

May Include:-

- Host name and port
- Optional URI
- Network Access Point or Channel
- Protocol specific configuration
  - T3, IIOP
  - Web Server
- Target Clusters and managed servers

In Multitenant environment, We create Virtual Target first, In previous versions of WebLogic Server, We Targeted our application, system resources (JDBC or JMS Resources) to Clusters, Managed Servers part of Cluster or stand alone managed server. But here we target our resources to Virtual Target. After creation of Virtual Target, we create a domain partition, while creating domain partition we can create resource group. A domain partition can have multiple resource groups. Then we target domain partition to the Virtual Target. Below diagram gives you the picture how they work together.



As in the Above Diagram, We created Virtual target 1, Virtual Target 2 which are Targeted to WebLogic Server. Then we created Two Domain partitions, Domain partition 1 has Resource Group 1 and Domain Partition 2 has Resource Group 2.

To know more about the Multitenancy, you can follow the link here.

We have total 6 Labs; the brief information about each lab is given below.

Lab 1 consist Non MT configuration, in this Lab we create basic configuration required for each lab. First we create **base_domain** using the **Restricted-JRF** template. This is the domain we are using in each lab. Using the Fusion Middleware Control Console, we will create machine and one dynamic cluster of initial size of two managed server inside it.

Lab 2 consist creation of MT configuration, in which we creates virtual target, domain partition and resource group. We show you can easily deploy one application twice in a domain in different domain partition. In this case both applications will be connected to different database. Good thing is that you don't need to modify the application deploying to different domain partition because we have JNDI isolation. We also use day trader application which is build by IBM, we made few changes in the application to run the application in Non MT environment, and we took the same application to deploy in MT environment (inside domain partition). So you don't need any specific application development to deploy the application in Multitenant environment.

Lab3 shows how you can have two or more security realms active inside a domain. Traditionally in WebLogic server, we have more than one Security realm but only one is active for the domain. Here, we have multiple domain partition, so you can have separate security realm for each domain partition. In Lab 2 we deploy Medrec application in domain partition dp1 and dp2. As while creating domain partition dp2 we do not choose any security realm so it uses the default security realm. In Lab 3 we create a new security realm and assign it to domain partition dp1. So both domain partitions will have different set of authorized user.

Lab 4 shows how you can easily move domain partition from one environment to another environment. Generally we work in simple development environment, like no restricted JRF, no cluster. But we need to test our application in testing environment, run in production environment. In Lab 4 we create a dev_domain using a by default templates and it consist only admin server. We do the required configuration for Medrec Application in dev_domain, once we have Medrec application is deployed in domain partition Medrec-Dev in dev_domain, we export the domain and import it in base_domain.

Lab 5 shows how you can configure the resource configuration manager. As the domain partition is targeted to the Virtual target which is targeted to a cluster. So multiple domain partition may be using the same cluster, in that scenario to avoid one partition to use the resources excessively, we set the limit using resource configuration management, we specify the limit for notify, slow and shutdown action. As the resource uses reaches that limit, particular action happens.

Lab 6 shows how you can integrate OTD as a front end in domain partition. It shows the steps how to integrate OTD while creating the domain partition. This Lab also shows you the Resource group migration from one cluster to other cluster. As OTD is front ending your domain partition, so without any interruption you can access the application using the same URL.

You create three domains as part of this workshop. There role are as given:

base_domain: We create this domain as part of Lab 1. This is the main domain which we are using in each lab. We create machine, dynamic cluster and domain partitions.

dev_domain: We create this domain as part of Lab 4. This domain uses default template and contain Admin Server only. This domain represent domain which we generally uses for development purposes.
We create a domain partition Medrec-Dev inside it, which we later export and import it to base_domain.

otd_domain: We create this domain as part of Lab 6. This domain consist Admin Server, and Load Balance instance.

How to Use Individual Labs:

Lab 1 and Lab 2 are compulsory.
Lab 3: You need to run Lab 1 and Lab2 before executing Lab 3.
Lab 4: You need to run Lab 1 and Lab 2 before executing Lab 4.
Lab 5: You need to run Lab 1 and Lab 2 compulsory for it. And if you do not execute Lab 4 before running Lab 5, then instead of modifying the changes to Medrec-Dev domain, you need to modify the domain partition dp1.
Lab 6: You need to run Lab 1 before executing the Lab 6.

Virtual Box Environment

The Hands on Lab Environment

Operating System Details

Operating System	Oracle Linux 6.4 x86_64
Hostname	localhost, wins-vbox
Root User	root/oracle
Oracle User	oracle/welcome1

Note: For this hand on lab you should only need to use **oracle** user account.

Installation Directories

JDK 1.8.0_60	/usr/java/jdk1.8.0_60/
WebLogic Server 12.2.1	/u01/wins/wls1221/
Oracle Traffic Director 12.2.1	/u01/wins/wls1221/
Oracle Database 12c	/u01/app/oracle/product/12.1.0/dbhome_1/

Workshop Content:

Labs Directory	/u01/content/weblogic-innovation-seminars/WInS_Demos/MT-Workshop/

LAB 1: DOMAIN CREATION AND NON-MT CONFIG

Overview

In this lab, we are going to perform the below operations.

- We create a WebLogic domain, in that domain we create machine, dynamic cluster.

Start the database

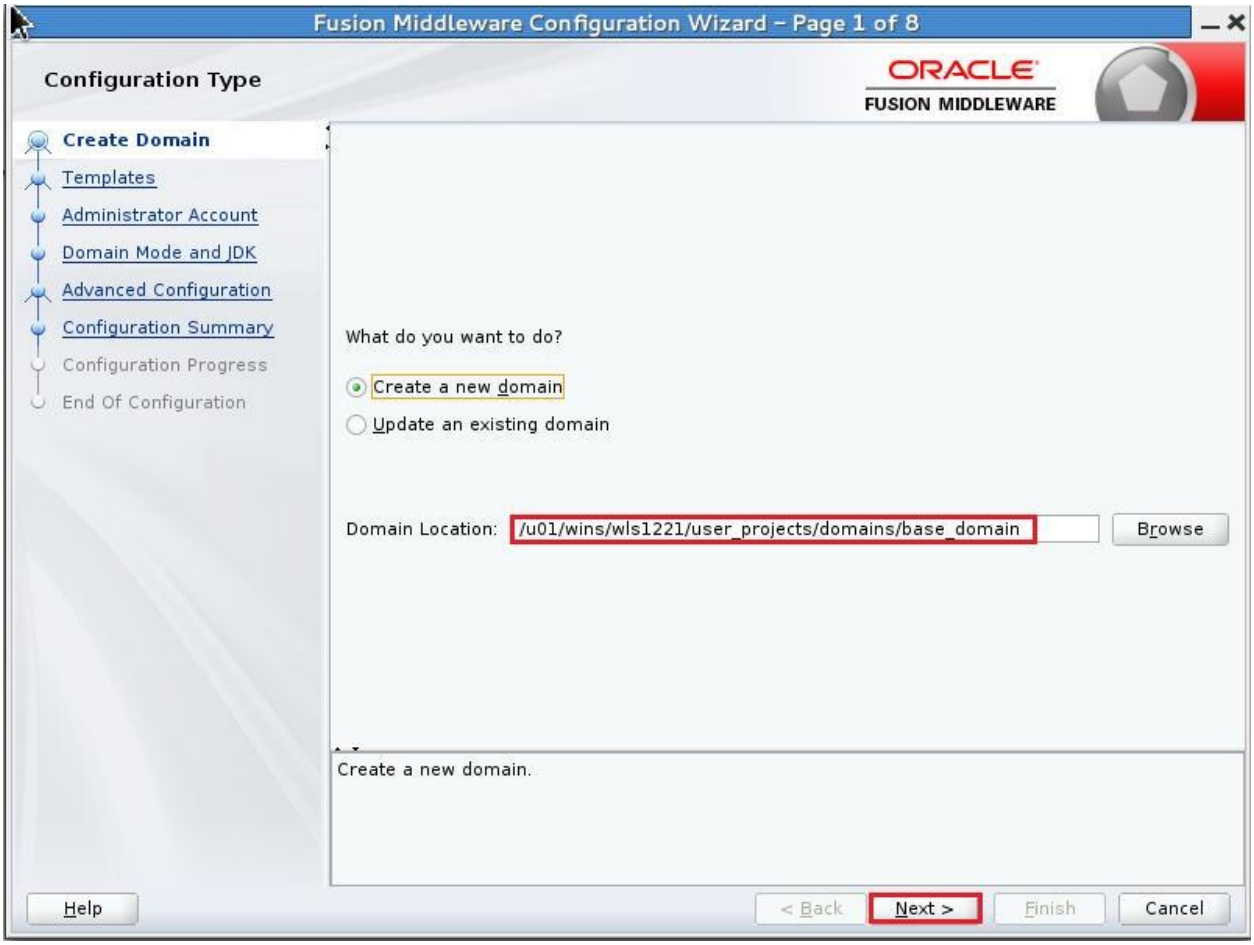
We have two Pluggable database **pdborcl** and **pdb2**; we are going to start both the database.

a. In Desktop, Double Click on Icon "**Start Database**".

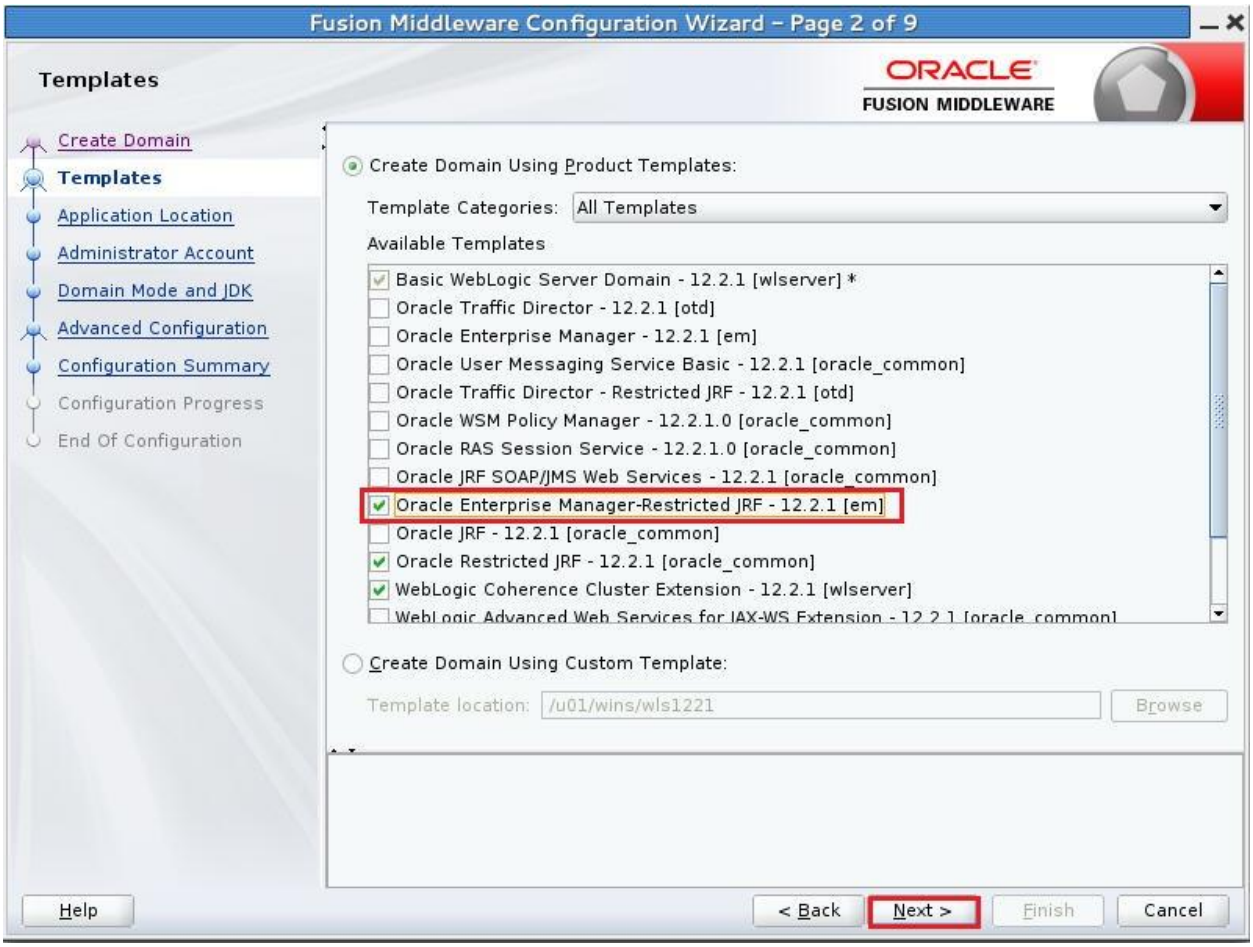
Note: Wait until the Window disappears.

Create WebLogic Restricted JRF Domain

a. Open a new terminal.
b. cd /u01/wins/wls1221/oracle_common/common/bin/
c. ./config.sh
d. Select **"Create a new domain"** and Enter **"/u01/wins/wls1221/user_projects/domains/base_domain**" as Domain Location then click on **Next.**



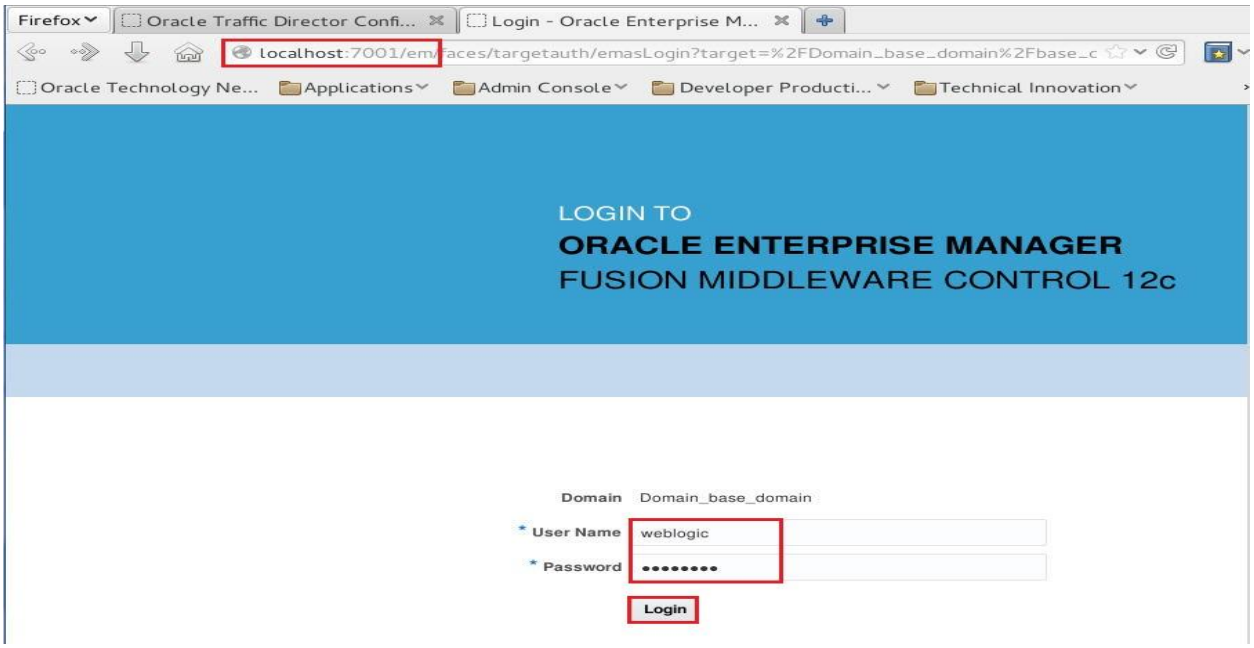
e. Select "**Oracle Enterprise Manager –Restricted JRF-12.2.1 [em]**" as it also select the remaining required check boxes then click on **Next.**



f. Leave Default on **Application Location** then click on **Next.**
g. Enter **weblogic/welcome1** as Name/Password then click on **Next.**
h. Leave Default on **Domain Mode and JDK Screen** then click on **Next.**
i. Leave Default on **Advanced Configuration** then click on **Next.**
j. Click on **Create**.
k. Click on **Next** then click on **Finish**.

Configuration of WebLogic base_domain

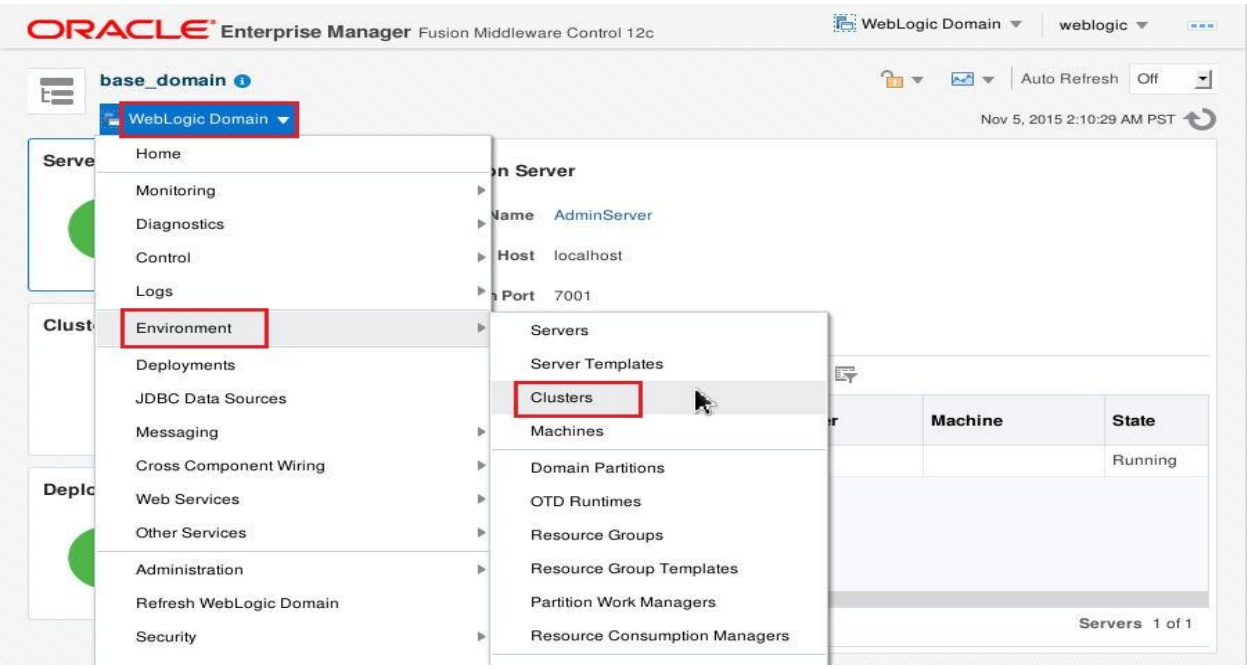
a. cd /u01/wins/wls1221/user_projects/domains/base_domain/bin/
b. ./**startNodeManager.sh**
c. In tab, Click on **Terminal -> Set Title**. Enter **base_nm** as Title then click on OK.
d. Open a new tab.
e. cd /u01/wins/wls1221/user_projects/domains/base_domain/bin/
f. ./**startWebLogic.sh**
g. In tab, Click on **Terminal -> Set Title**. Enter **base_admin** as Title then click on OK.
h. Go back to Firefox and type the Fusion Middleware Control URL http://localhost:7001/em
i. Enter **weblogic/welcome1** as **username/password** then click on **Login.**



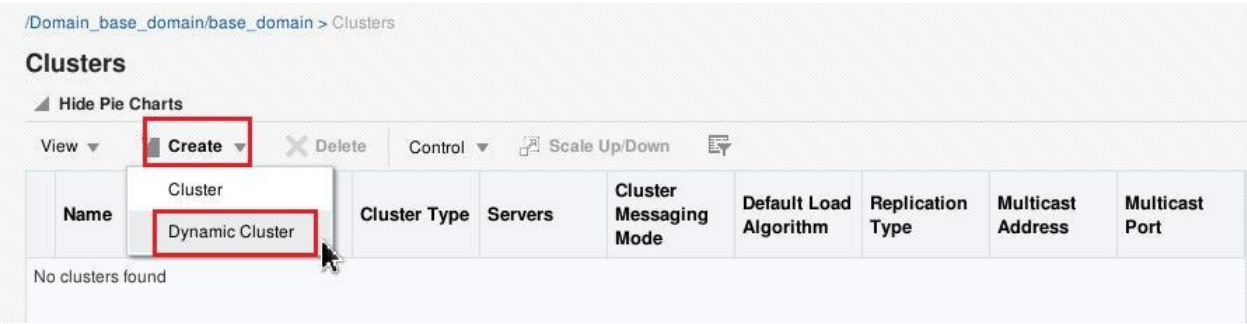
j. Create a Machine.
   i. Click on **WebLogic Domain-> Environment -> Machine.**
   ii. Click on **Create**.
   iii. Enter "**machine**" as Name, Select "**Unix"** as Machine OS, then click on **Next**.
   iv. Leave Defaults on Node Manager Properties then click on **Create.**
   v. Click on the Machine name **machine.**
   vi. Click on **Monitoring** tab and verify the status as **Reachable.**

k. Create a Dynamic Cluster

i. Click on **WebLogic Domain -> Environment -> Clusters.**



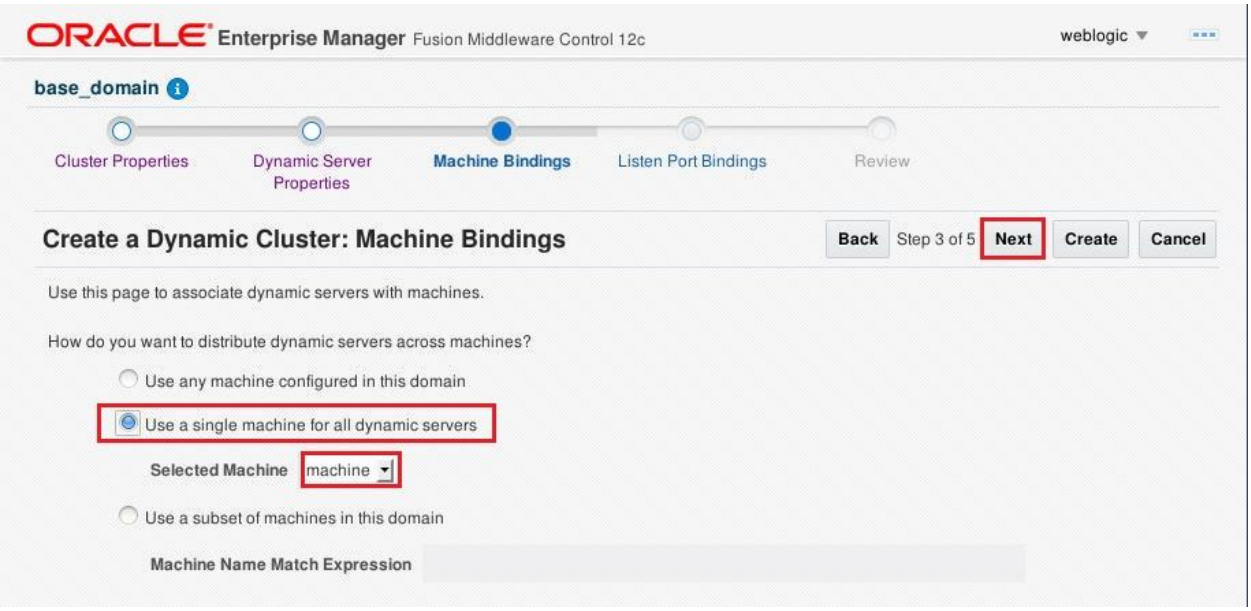
ii. Click on **Create -> Dynamic Cluster.**



iii. Enter **app-cluster** as Name then click on **Next.**

iv. Leave Default on **Dynamic Server Properties** page and click on **Next.**

v. Select "**Use a single machine for all dynamic servers**" and choose the **machine**, then click on **Next**.

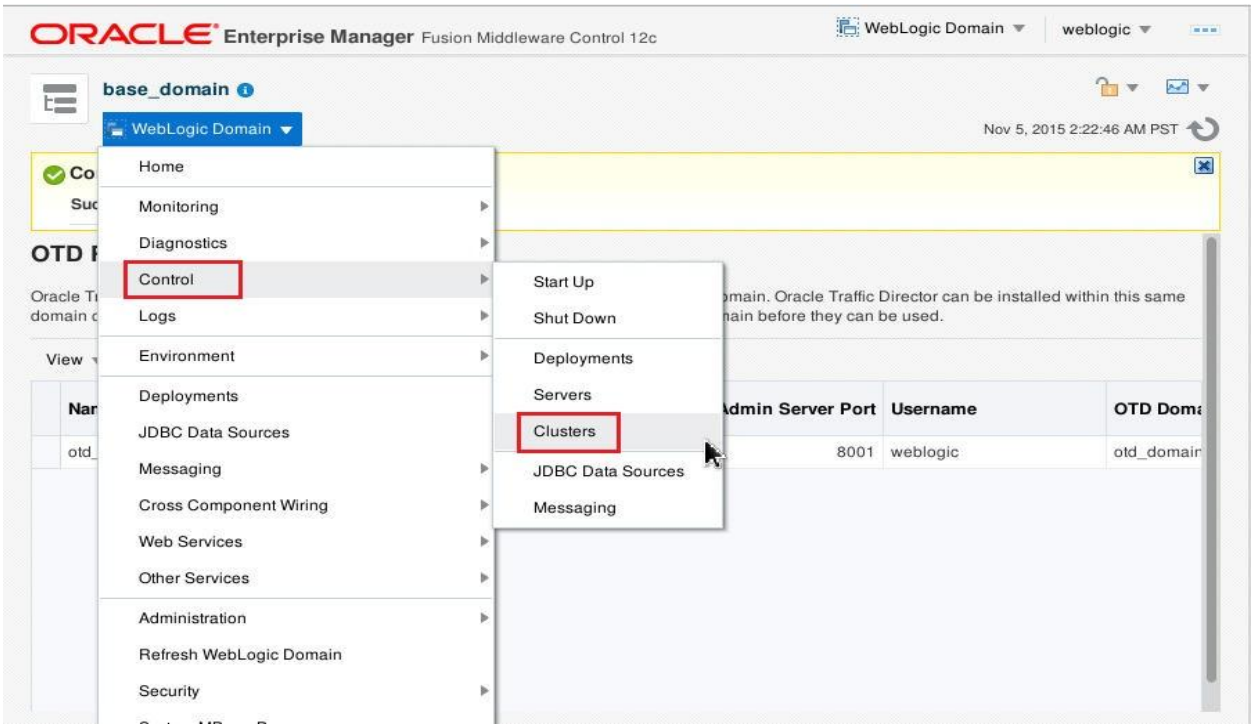


vi. Leave Default on Listen Port Bindings, and then click on **Next**.

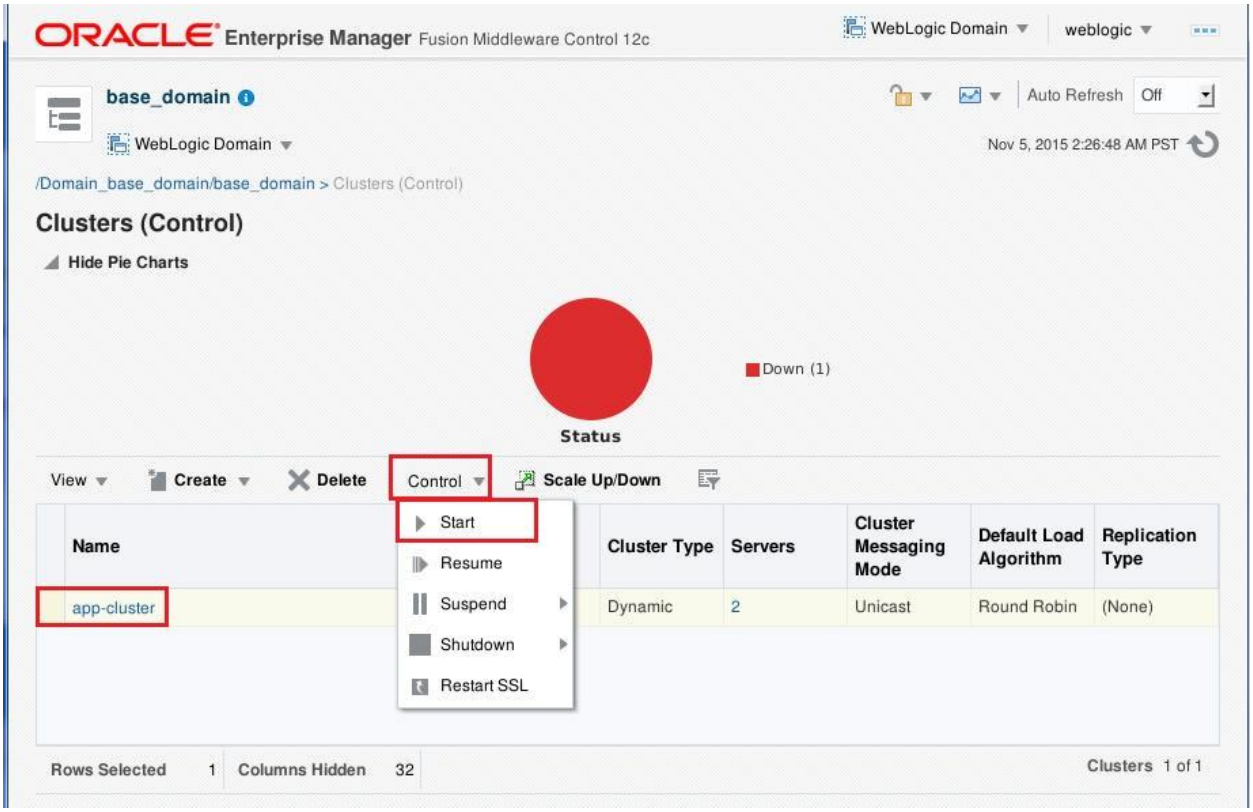
vii. Review the Configuration and click on **Create**.

l. Start Managed Servers.

i. Click on **WebLogic Domain -> Control ->Clusters.**



ii. Check the box near to **app-cluster** to make it highlighted and then Click on **Control -> Start -> Start Servers.**



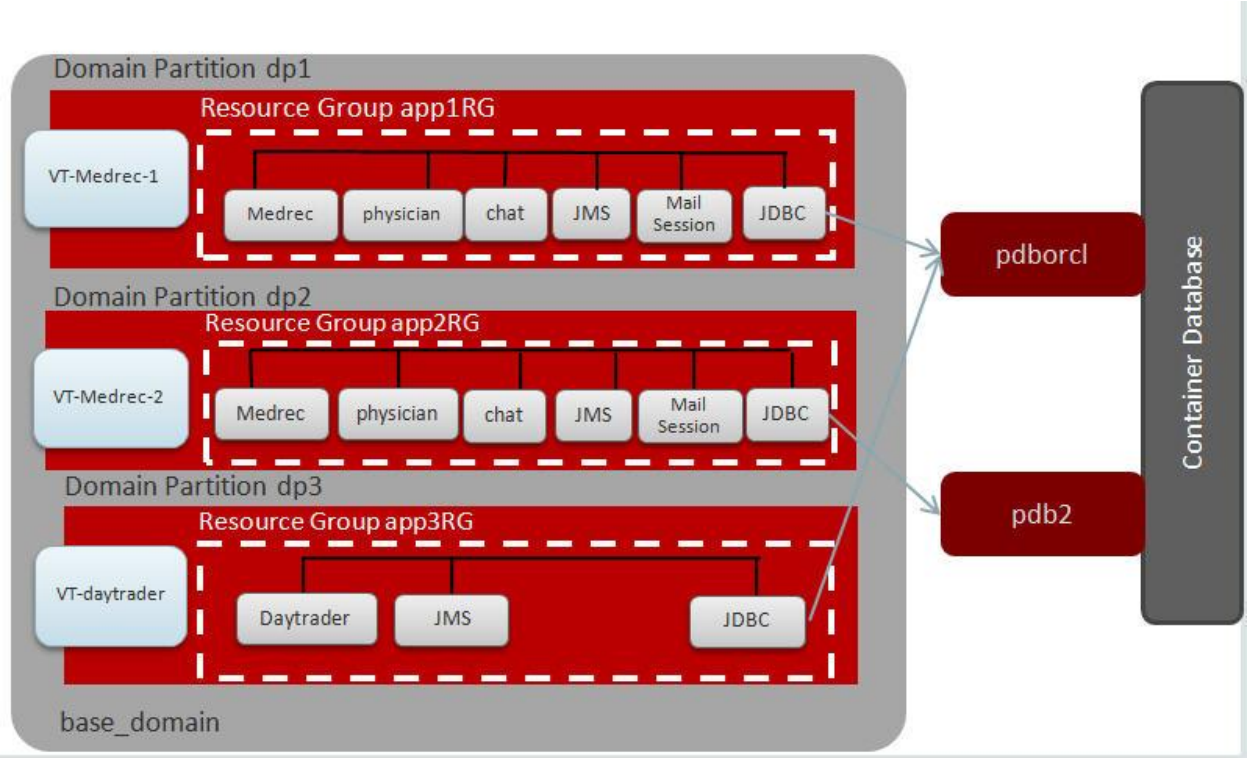
LAB 2: MULTITENANCY CONFIGURATION

Overview:

In this lab, we are going to learn the following:

- Configuration of Virtual Target, Domain Partition and Resource Group
- Run multiple instances of Medrec application in different domain partition without modifying the application. We try to show **JNDI Isolation** for that we are using the same Medrec applications and same JNDI name for the Datasources, connection factory and Distributed queue in both the domain partition.
- Run Day trader application which is build by WebSphere to WebLogic 12.2.1.
- You don't need to modify your application to run in Multitenant environment. So no special application development needed.

The final deployment architecture will look like the below where we will create three domain partitions .Deploy Medrec application in domain partition dp1 and dp2. And Day Trader application on domain partition dp3.

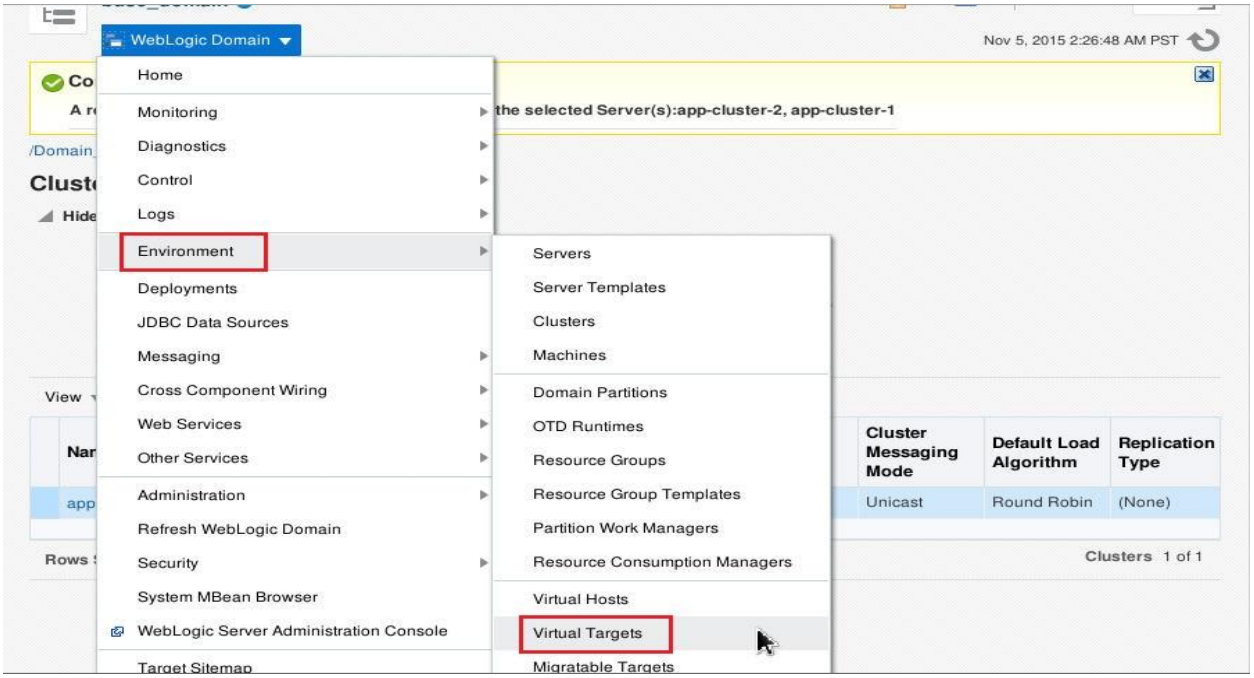


Configuration of Medrec Application in Domain Partition 1

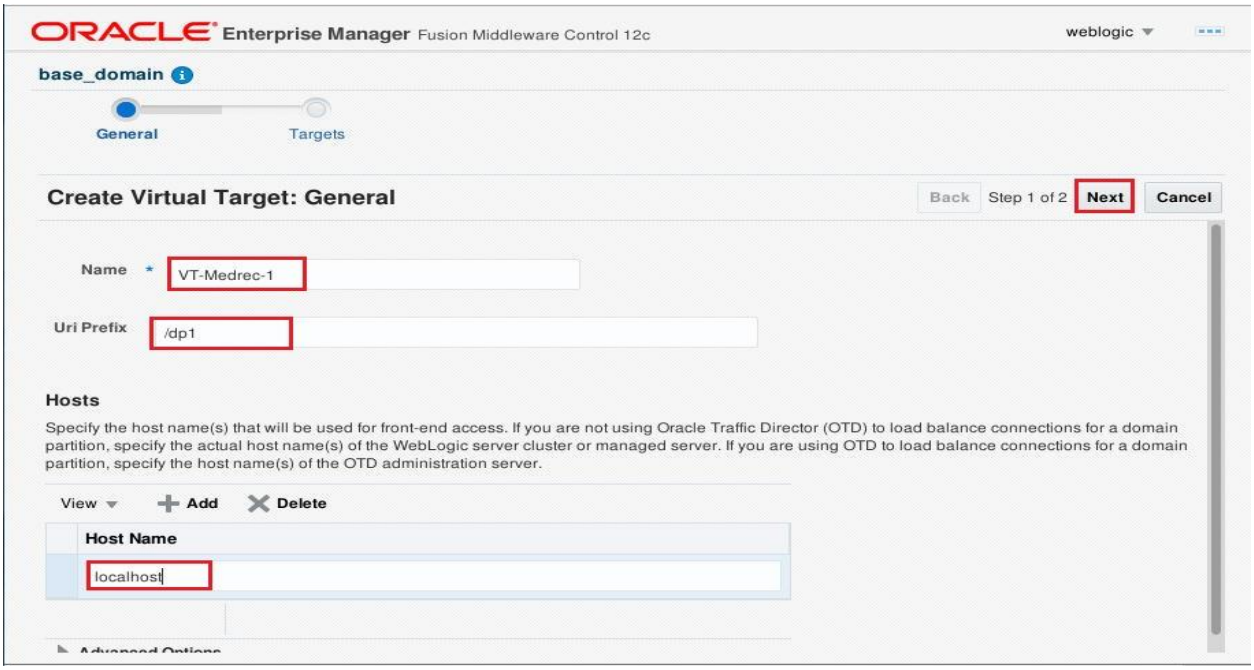
In the next step we are creating the below configuration for Medrec application in domain partition dp1.

Virtual Target: VT-Medrec-1
Domain Partition: dp1
Resource Group app1RG
app1RG:
- Datasource:
  - Name: MedRecGlobalDataSourceXA
  - JNDI Name: jdbc/MedRecGlobalDataSourceXA
- Mail Session:
  - Name: MedRecMailSession
  - JNDI Name: mail/MedRecMailSession
- Persistence Store: MedRec-fs
- JMS Server: MedRecJMSServer
- JMS Module: MedRecModule
- MedRecModule:
  - Subdeployment: MedRecJMS
  - Connection Factory:
    - Name: MedRecConnectionFactory
  - JNDI Name: com.oracle.medrec.jms.connectionFactory
  - Distributed Queue:
    - Name: PatientNotificationQueue
    - JNDI Name: com.oracle.medrec.jms.PatientNotificationQueue
- Applications: medrec.ear
  physician.ear
  chat.war

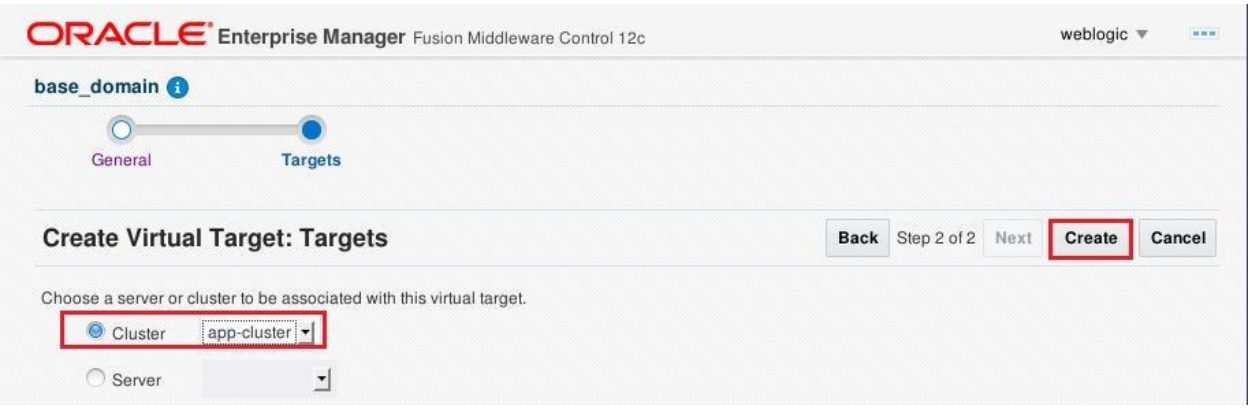
a. Open a new tab.
b. cd /u01/content/weblogic-innovation-seminars/WInS_Demos/MT-Workshop/Lab2/
c. **./Medrec1DB.sh** (It creates the required database user and populate sample data into database)
d. Close the tab.
e. In Enterprise Manager, Click on **WebLogic Domain -> Environment -> Virtual Targets**.



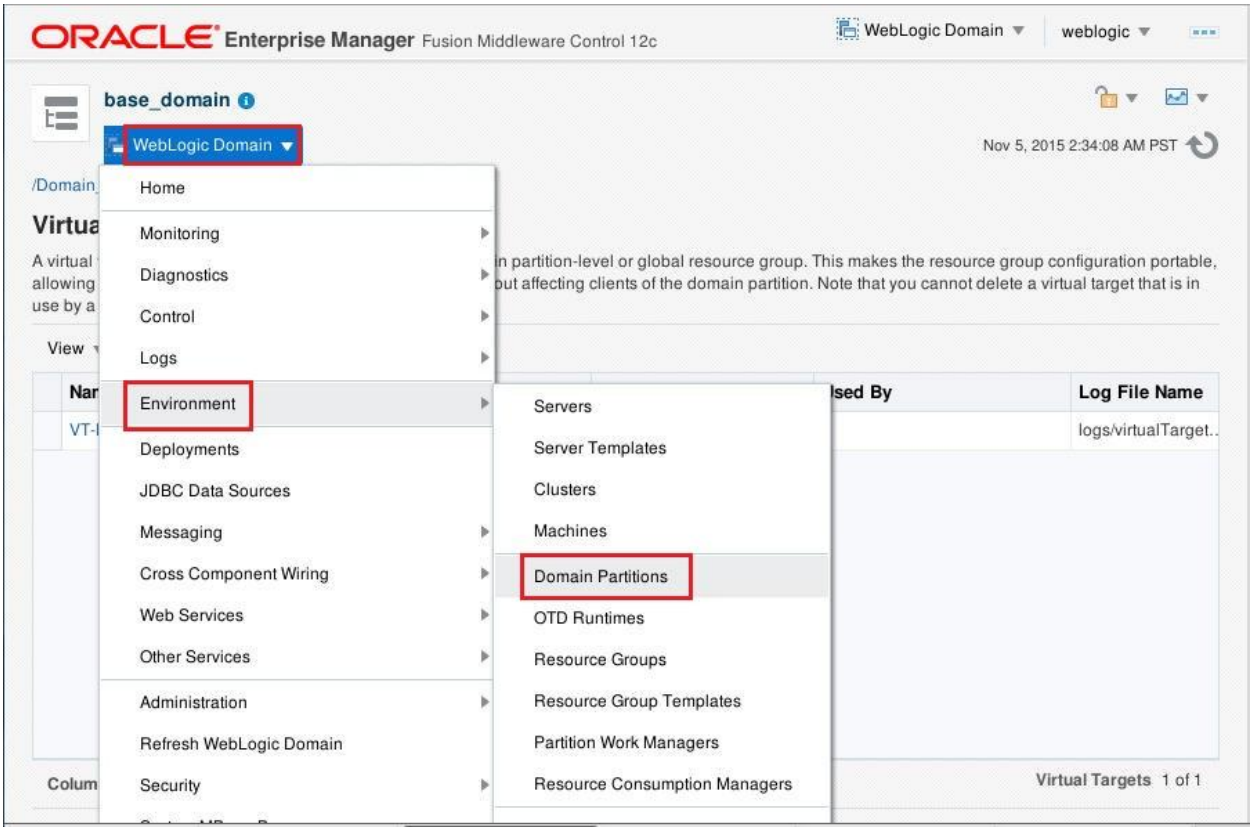
f. Click on **Create**.
g. Enter **VT-Medrec-1** as Name and **/dp1** as Uri Prefix and Add **localhost** as Hosts then click on **Next**.



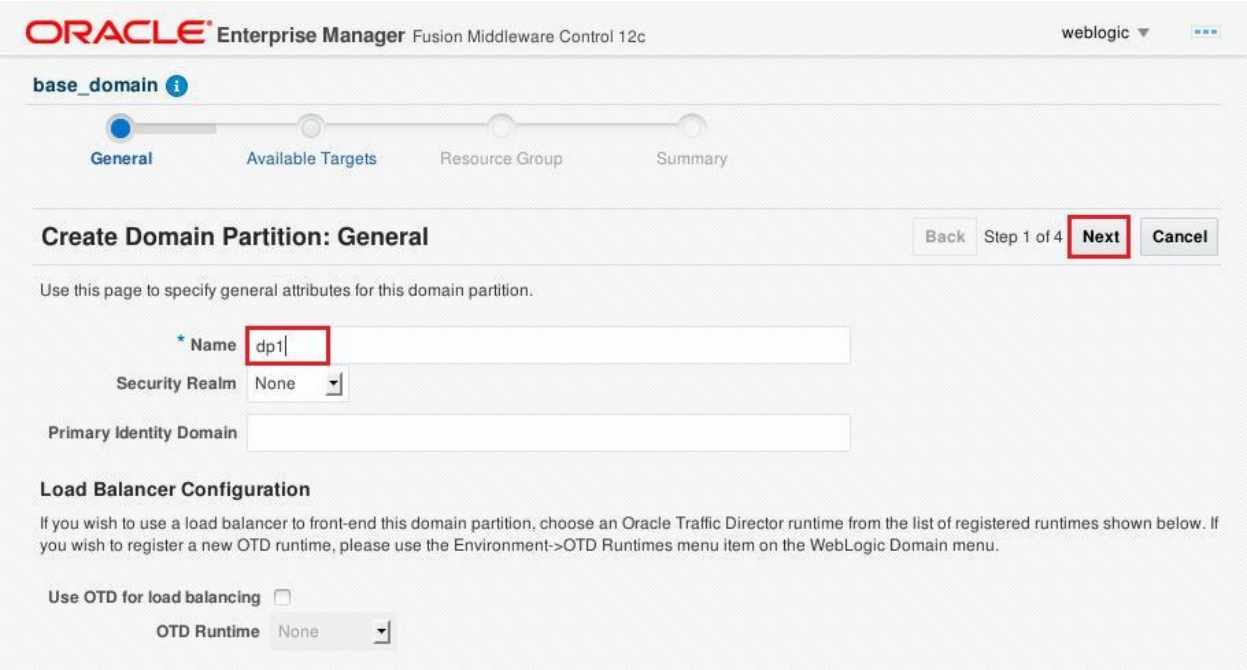
h. Select Cluster **app-cluster** as Target then click on **Create**.



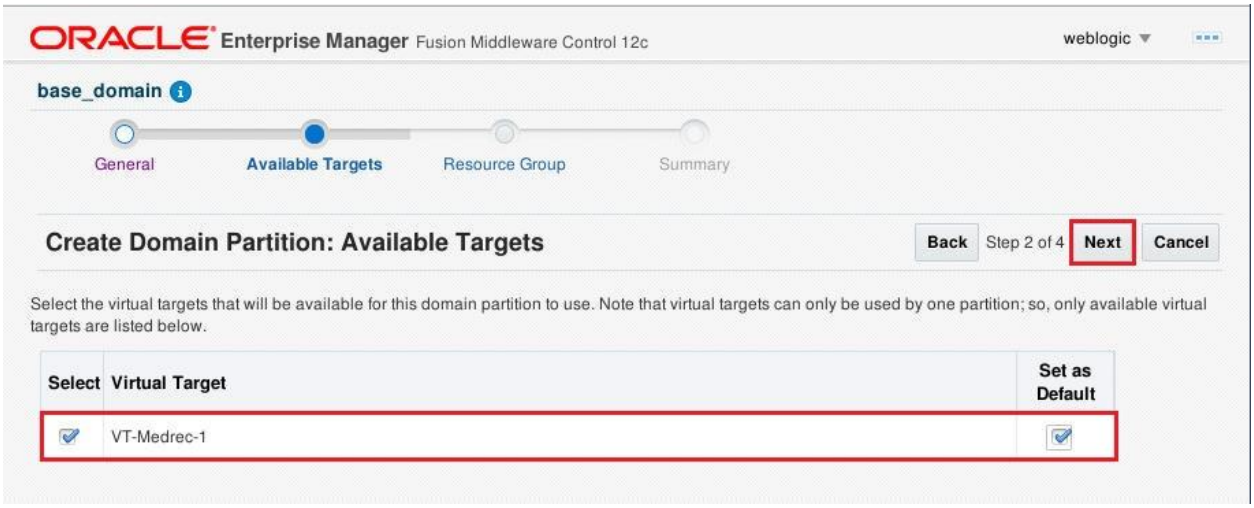
i. Click on **WebLogic Domain -> Environment -> Domain Partitions**.



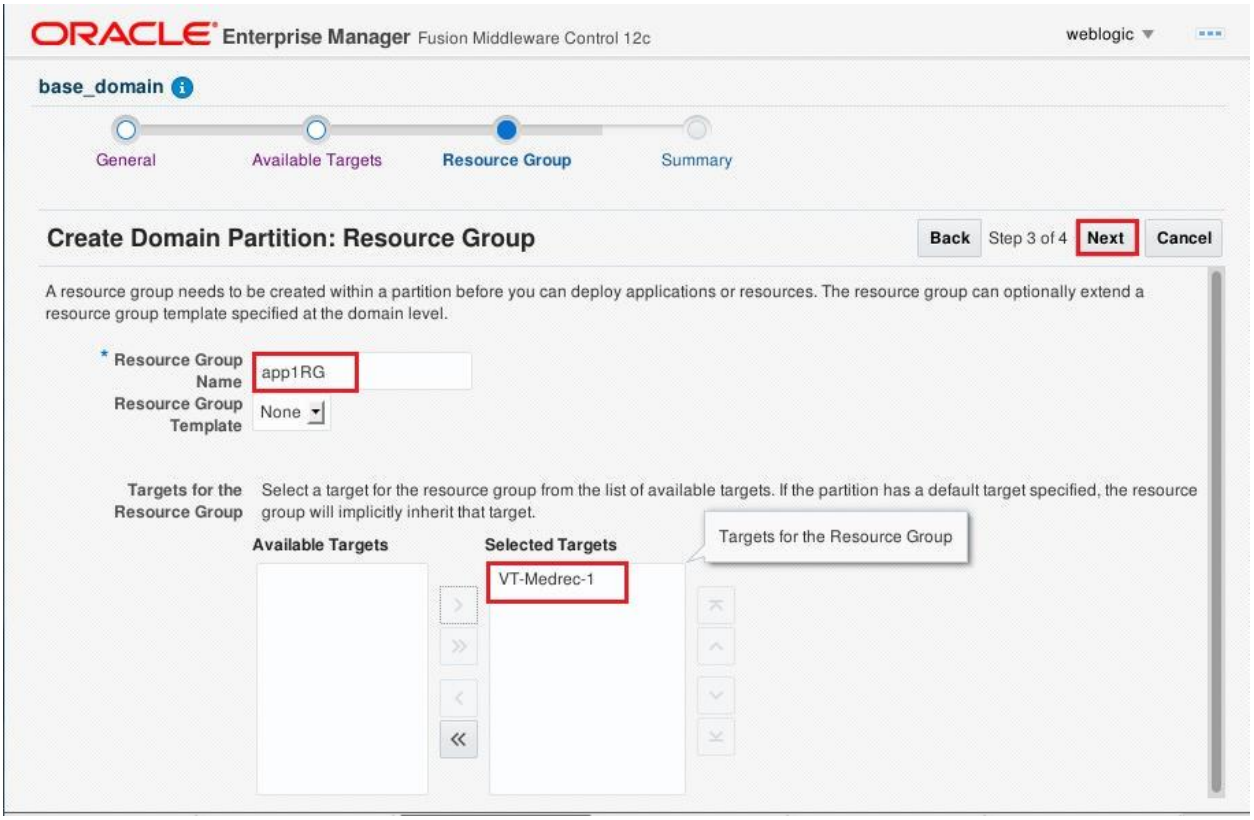
j. Click on "**Enable Lifecycle Manager**".
k. Go back to base_admin terminal, press CTRL+C, to stop the admin server.
l. ./startWebLogic.sh
m. Go back to Firefox, and type the Fusion Middleware Control Console URL http://localhost:7001/em .
n. Enter **weblogic/welcome1** as **User Name/Password** then Click on **Login**.
o. Click on **WebLogic Domain -> Environment –> Domain Partition**.
p. Click on **Create**.
q. Enter **dp1** as Name then click on Next.



r. Check the left box near **VT-Medrec-1** and also check the box for **Set as Default** then click on **Next**.

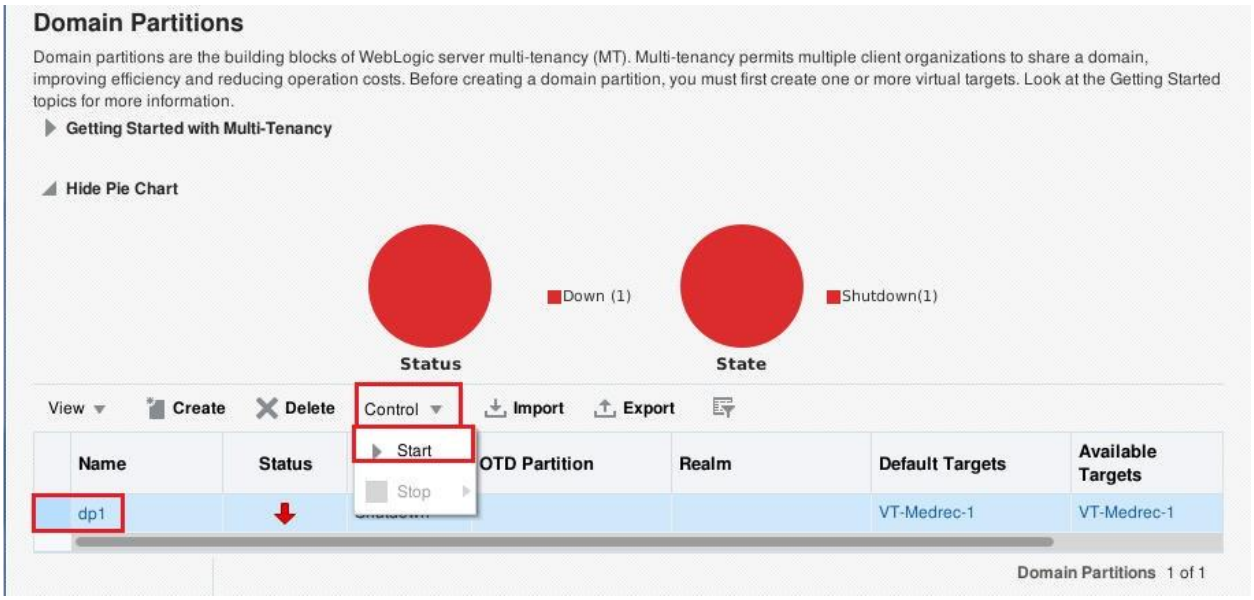


s. Enter **app1RG** as Resource Group name and **None** as Resource Group Template, Move the **VT-Medrec-1** virtual target to **Selected targets** then click on **Next**.



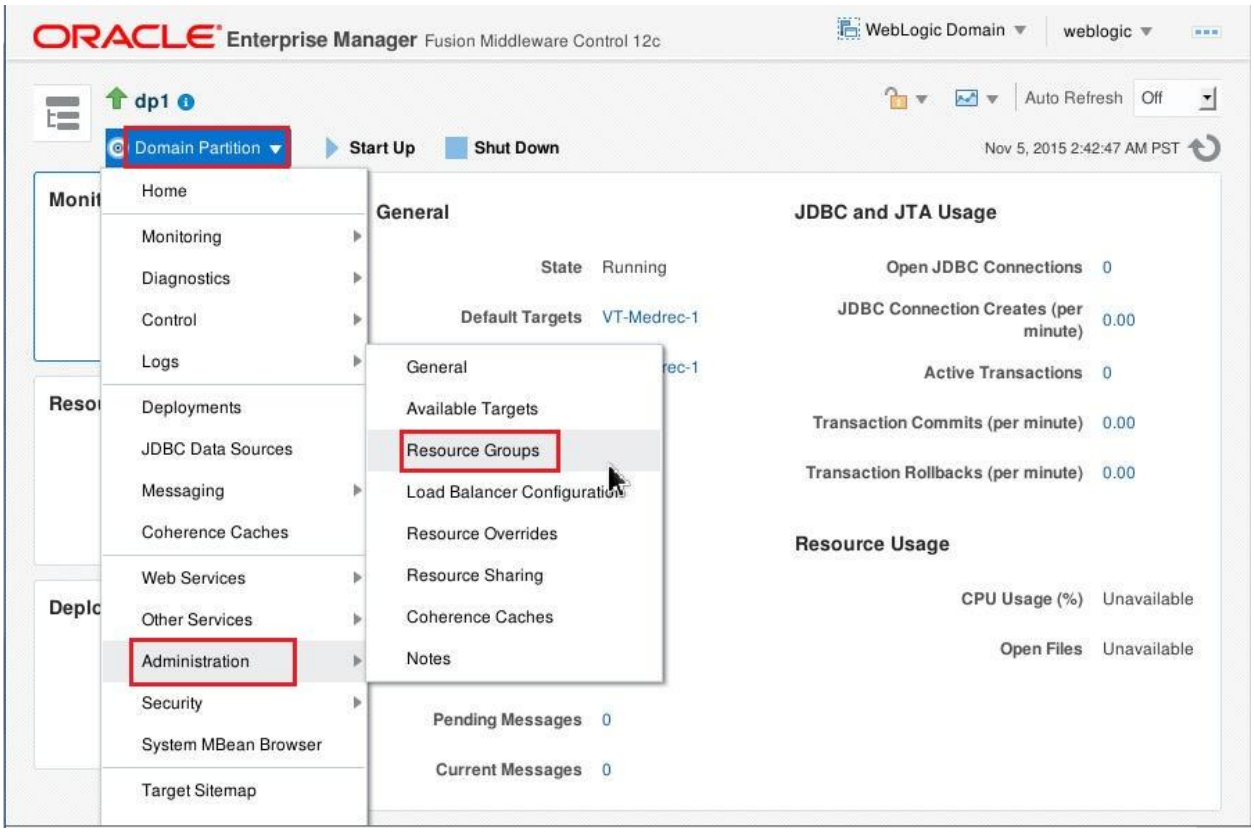
t. Verify the configuration and click on **Create**.

u. Check the box near **dp1** and click on **Control -> Start**. Once you notice the message "**Partition state after the operation is RUNNING**" then Click on Close.

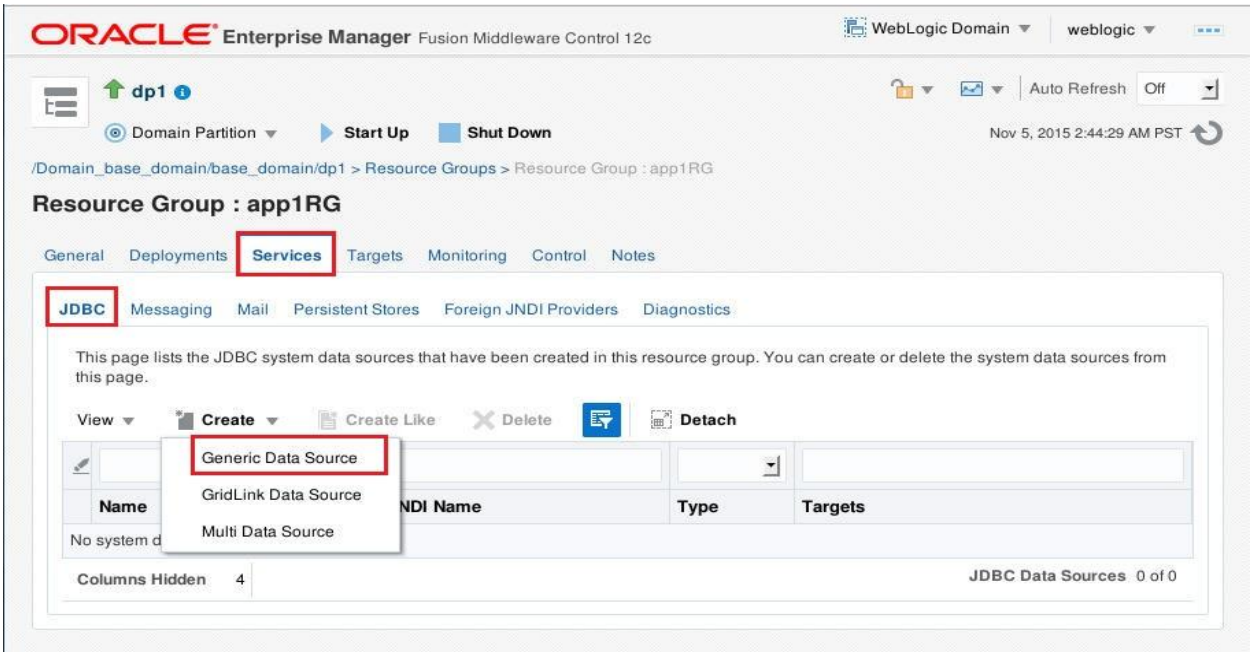


v. Click on the Domain Partition **dp1**.

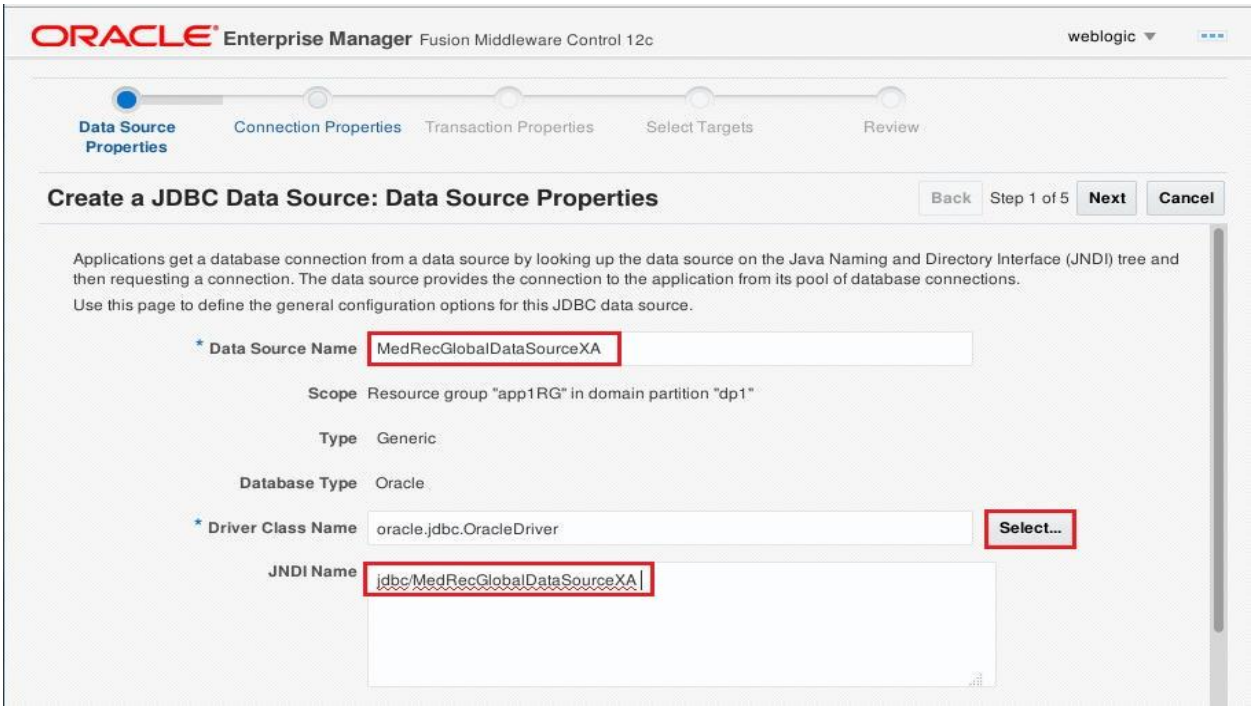
w. Click on **Domain Partition -> Administration -> Resource Group**.



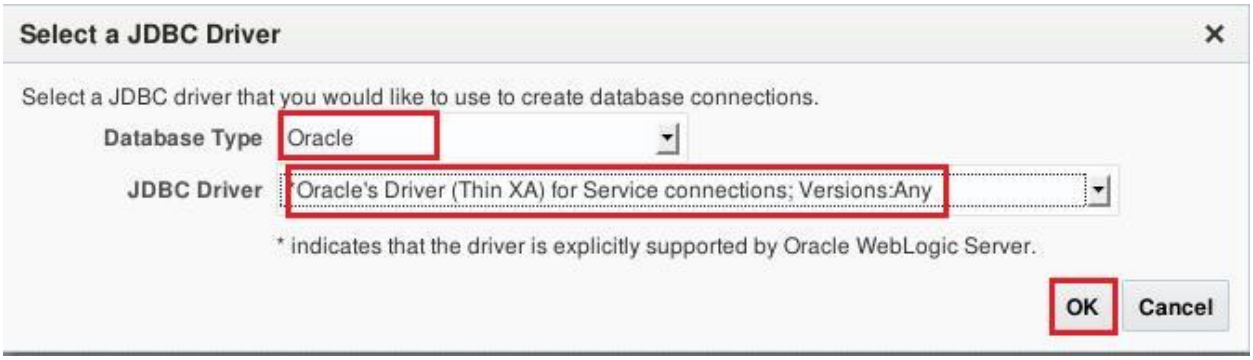
x. Click on Resource Group **app1RG**.
y. Creation of Datasource.
    i. Select the **Services** tab.
    ii. Choose JDBC tab, click on **Create -> Generic Data Source**.



iii. Enter **MedRecGlobalDataSourceXA** as Data Source Name and **jdbc/MedRecGlobalDataSourceXA** as JNDI Name, and then click on Select.



iv. Select **Oracle** as Database Type and "**Oracle's Driver (Thin XA) for service connections; Versions: Any**" as JDBC Driver then click on OK.

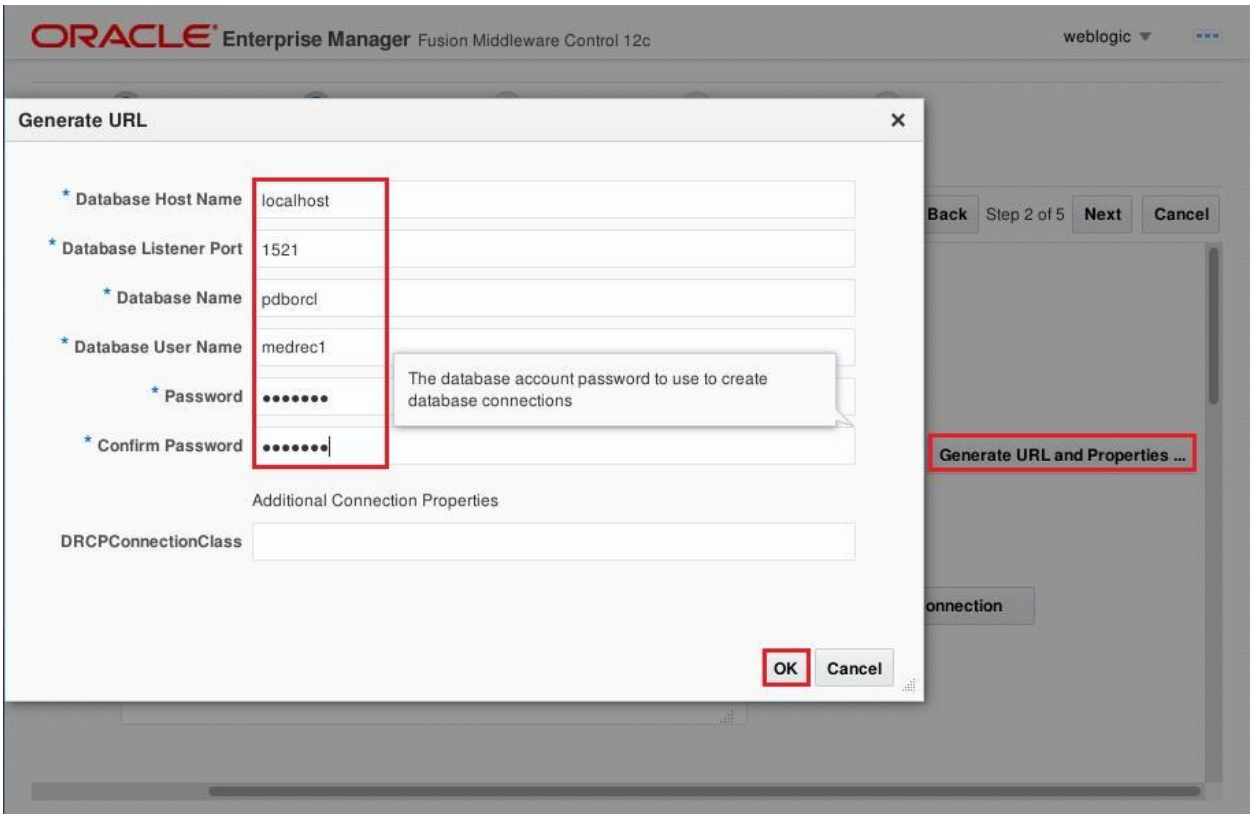


v. Click on **Next**.

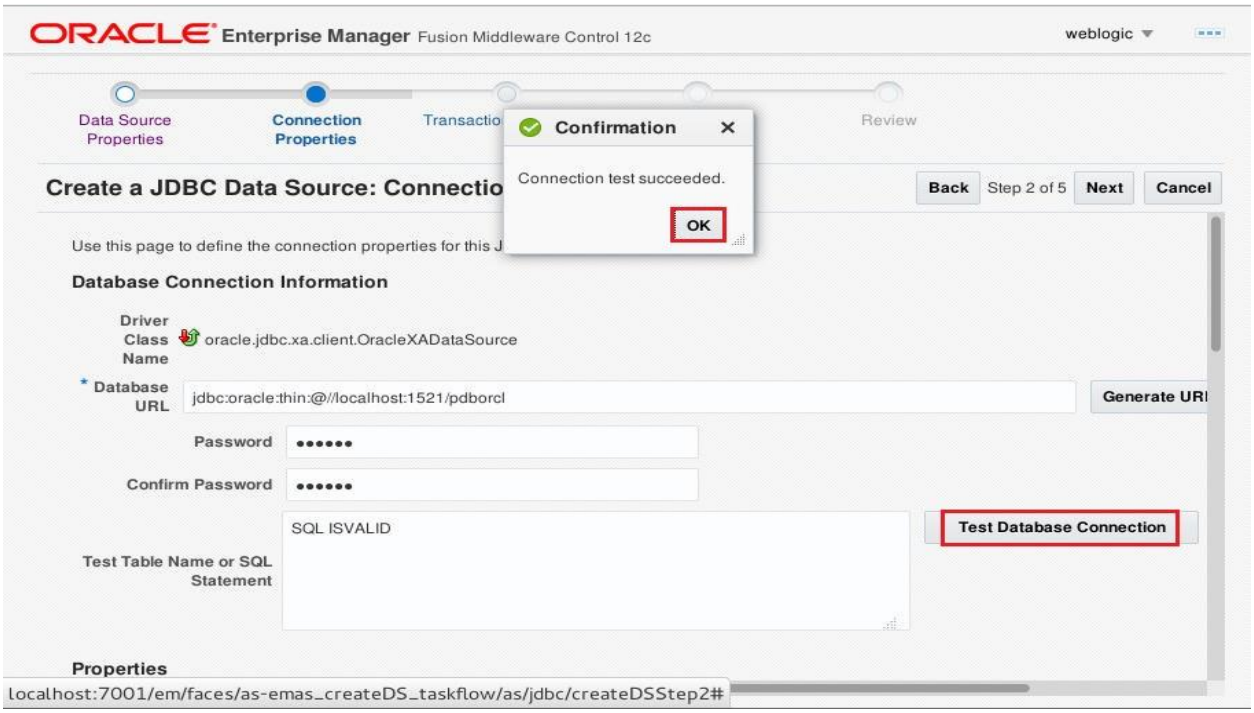
vi. Click on **Generate URL and Properties** and Enter the following:

Host Name:	localhost
Listen Port:	1521
Database Name:	pdborcl
User Name:	medrec1
Password:	medrec1
Confirm Password	medrec1

Click OK.



vii. Click on **Test Database Connection** to verify the connection. Click **Next**.

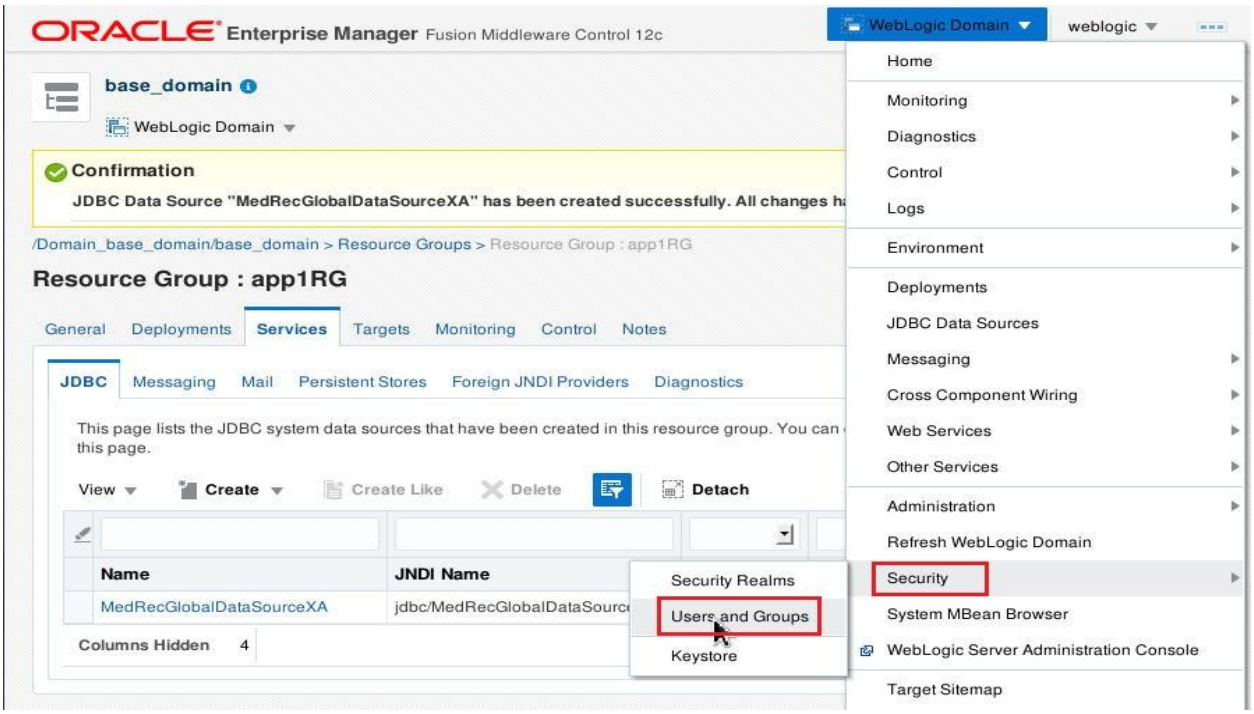


viii. Leave Default on **Transaction Options** and click on **Next**.

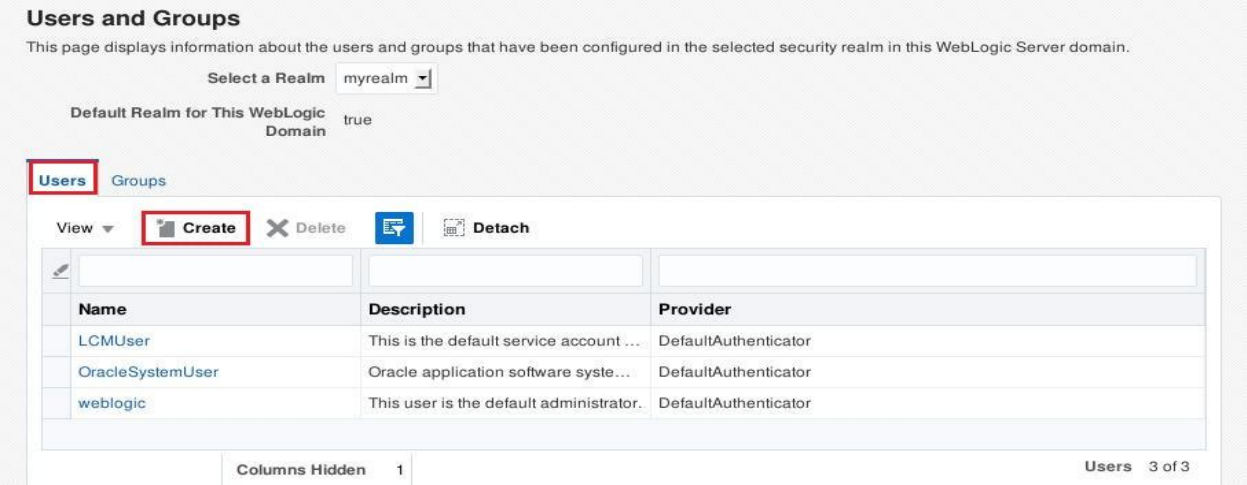
ix. Verify the configuration and click on **Create**.

z. Adding User to Default Realm.

i. Click on **Weblogic Domain -> Security -> Users and Groups**.



ii. In **Users** tab, click on **Create**.



iii. Enter the following then click on **Create**.

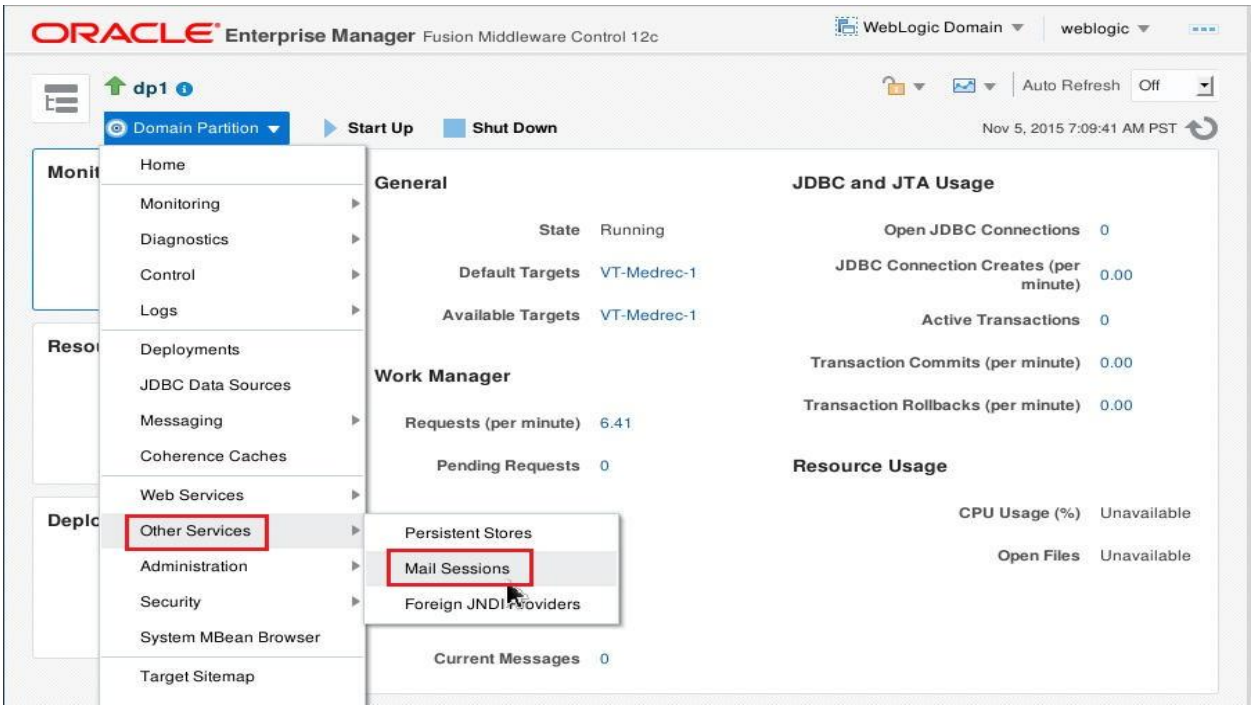
Name:	administrator
Description:	Medrec Admin
Provider:	Default Authenticator
Password:	administrator123
Confirm Password:	administrator123

aa. Configuring Mail Session.

i. Click on **Weblogic Domain ->Environment ->Domain Partition**.

ii. Click on Domain Partition **dp1**.

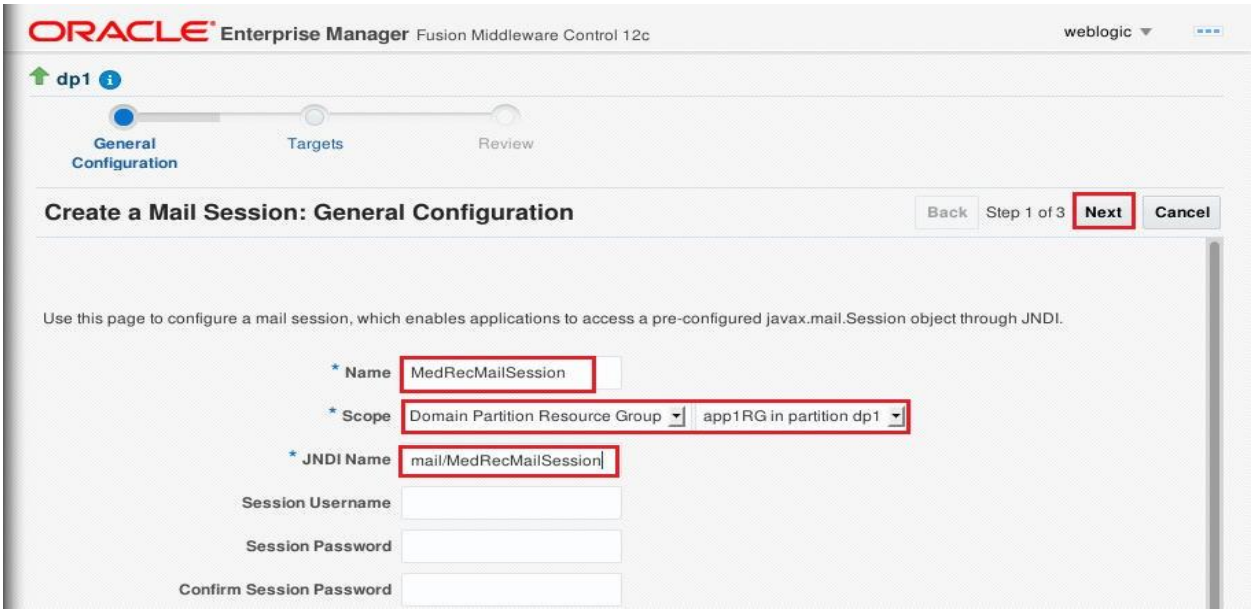
iii. Click on **Domain Partition -> Other Services -> Mail Sessions**.



iv. Click on **Create**.

v. Enter the following and click on **Next**.

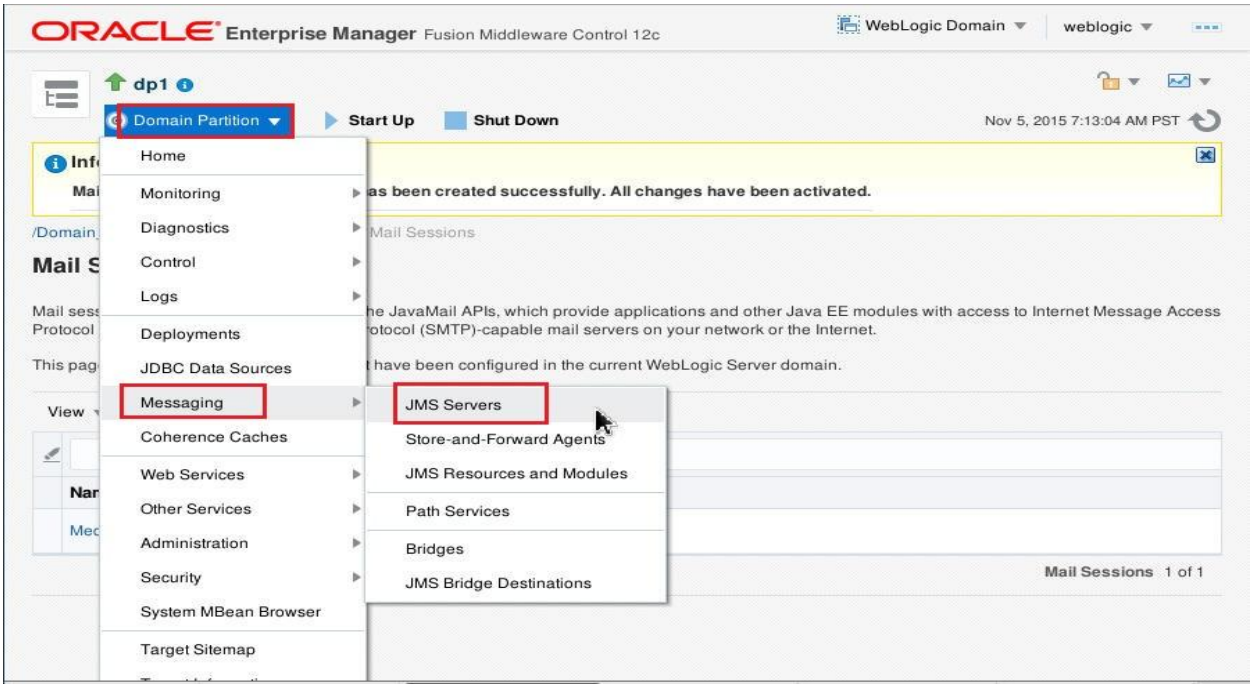
Name:	MedRecMailSession
Scope:	Leave as default
JNDI Name:	mail/MedRecMailSession



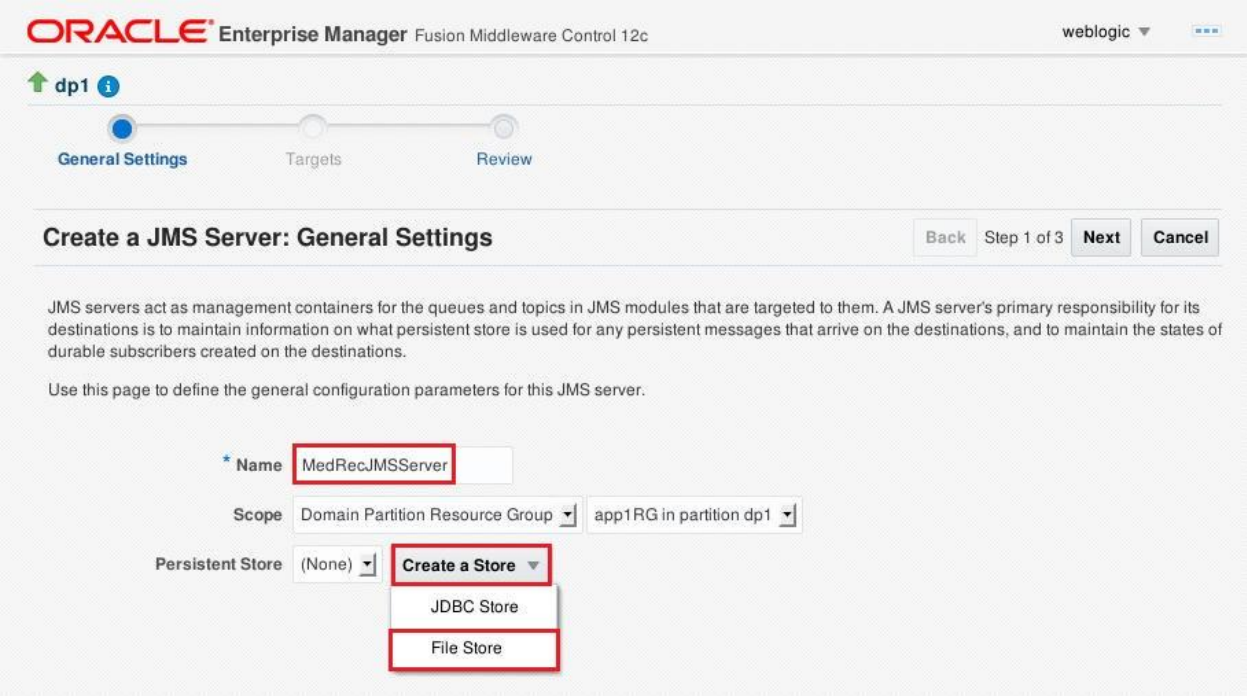
vi. Click on **Create**.

bb. Configuring JMS Server.

i. Click on **Domain Partition -> Messaging -> JMS Servers**.

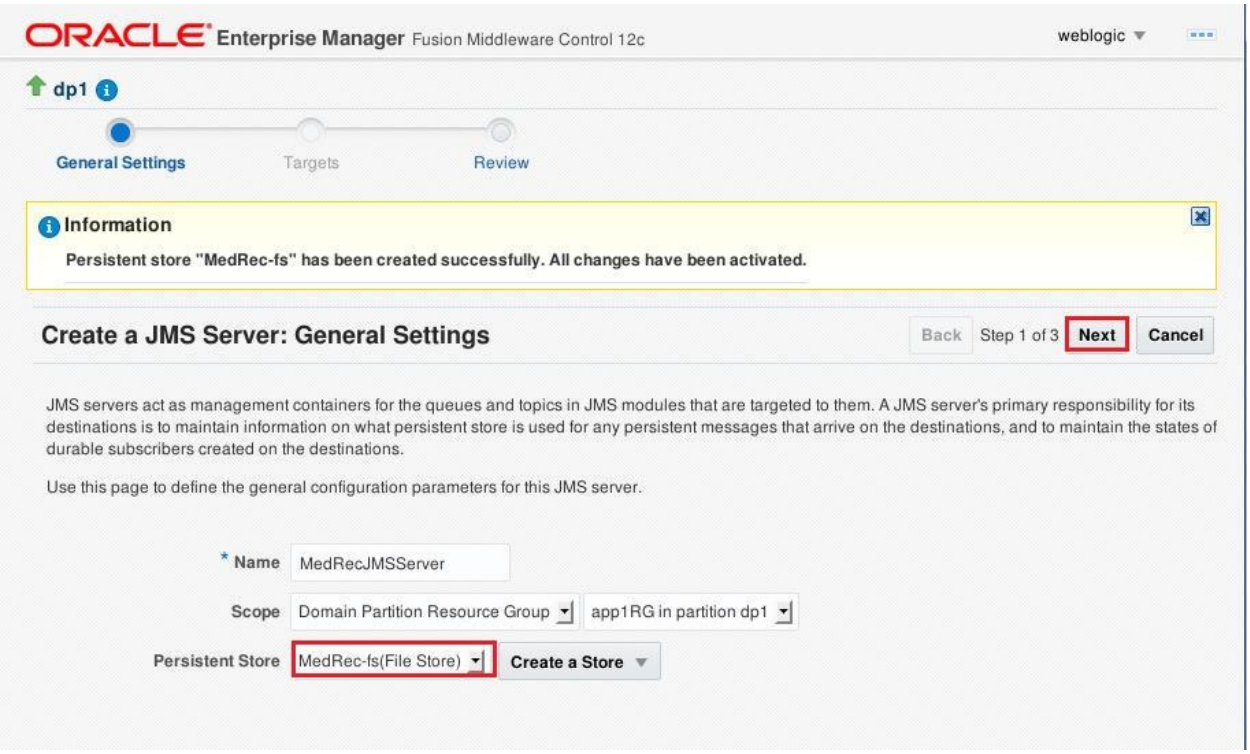


ii. Click on **Create**. Enter **MedRecJMSServer** as Name then click on **Create a Store** -> **File Store**.



iii. Enter **MedRec-fs** as Name then click on **Next**. Click on **Create**.

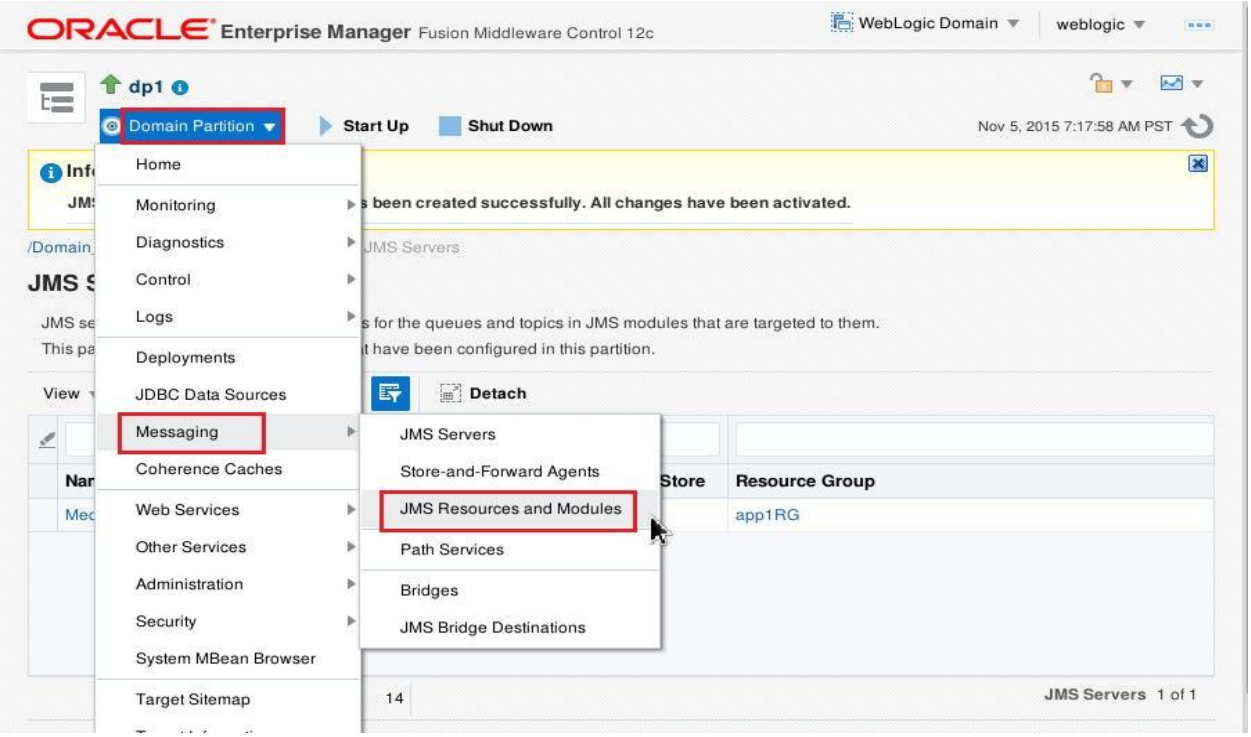
iv. In **JMS Server: General Setting**, Select the newly created file store as persistent store then click on **Next**.



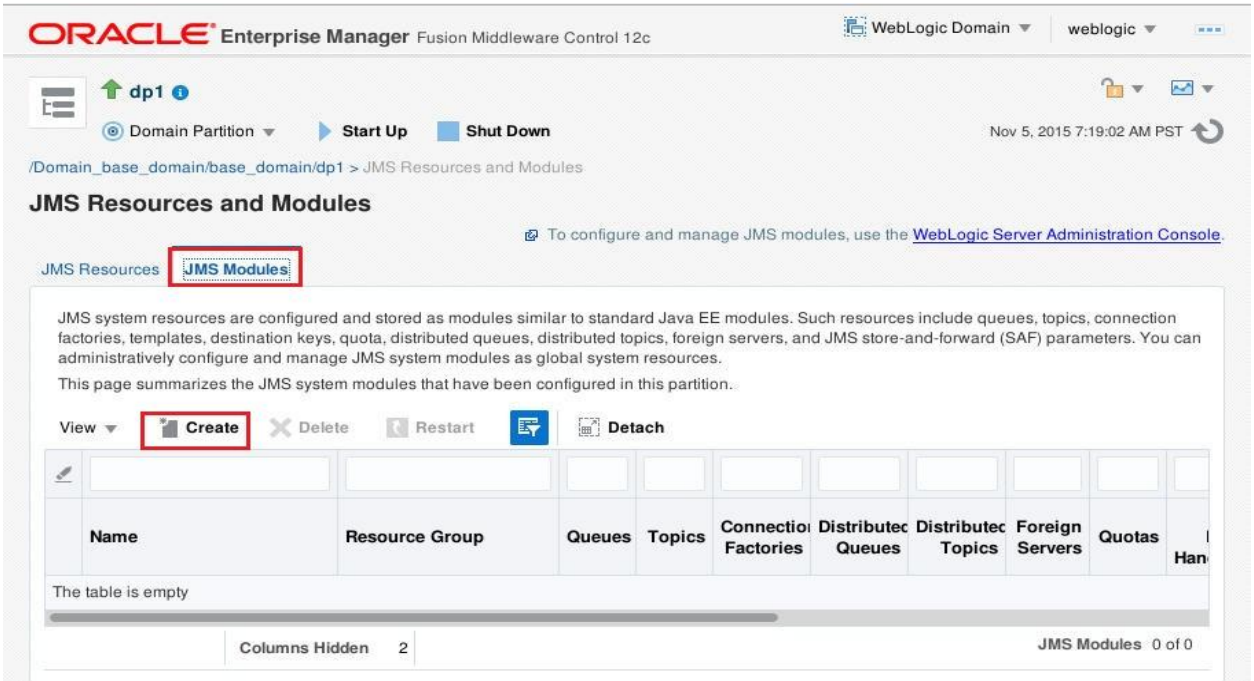
v. Click on **Create**.

cc. Configure JMS Module.

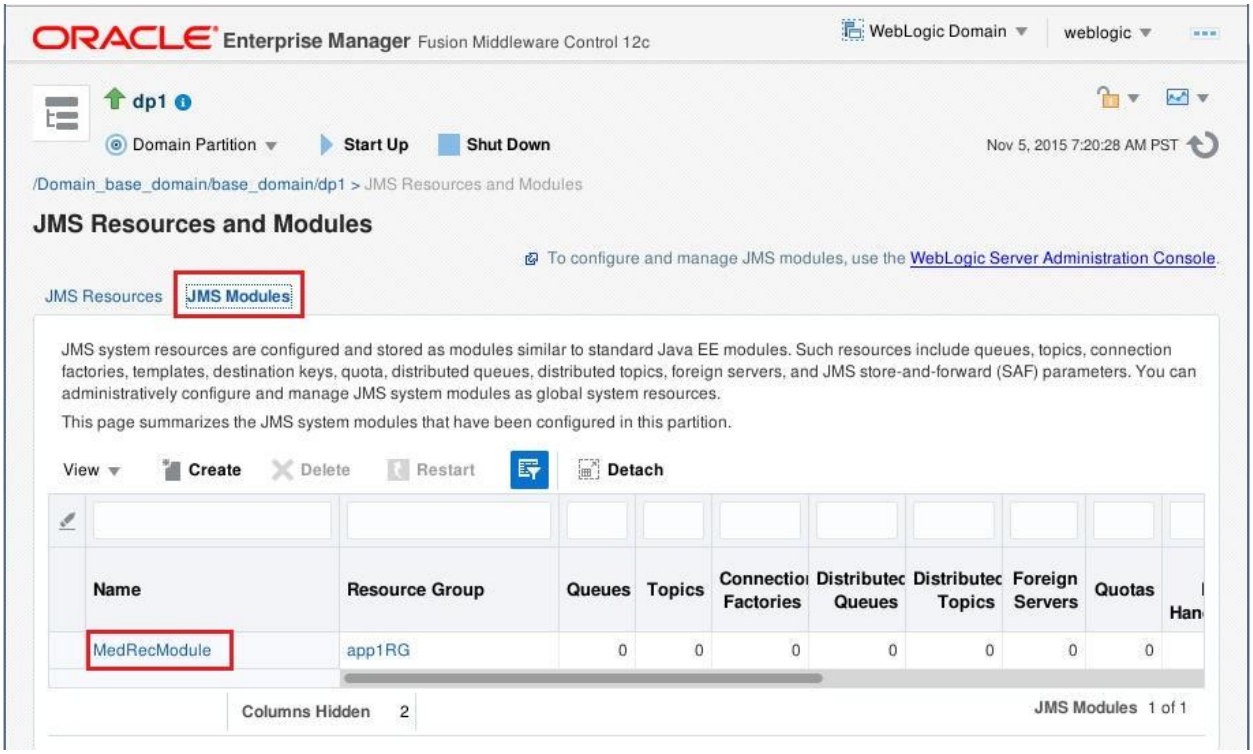
i. Click on **Domain Partition -> Messaging -> JMS Resources and Modules**.



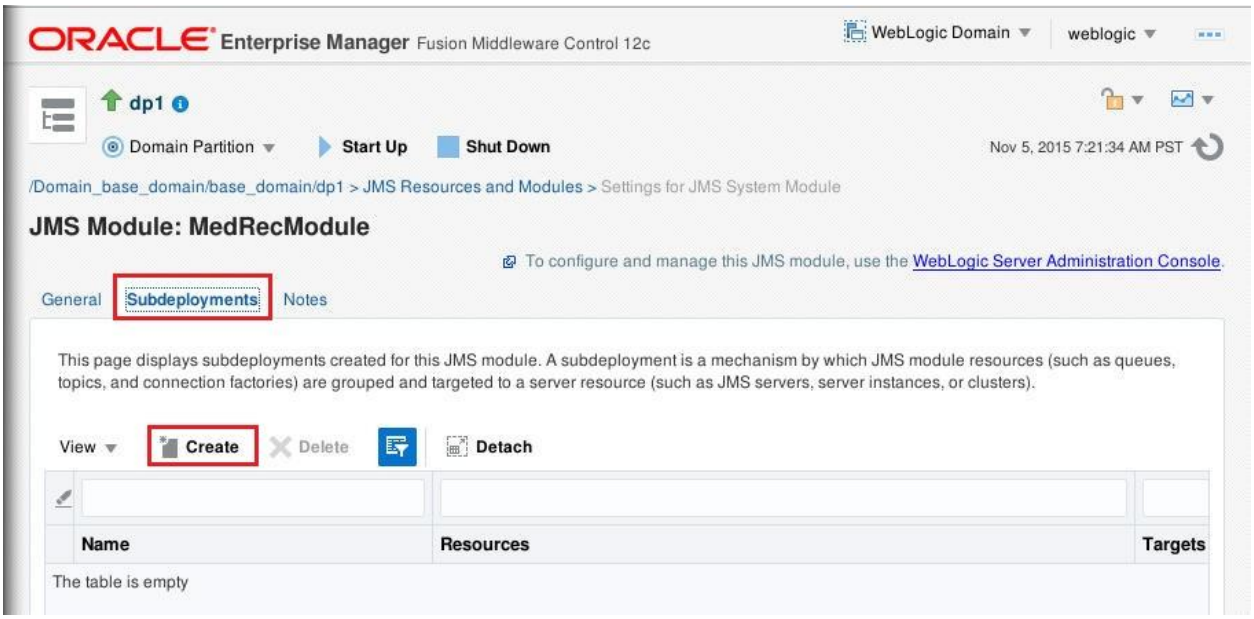
ii. Under the **JMS Modules** tab, click on **Create**.



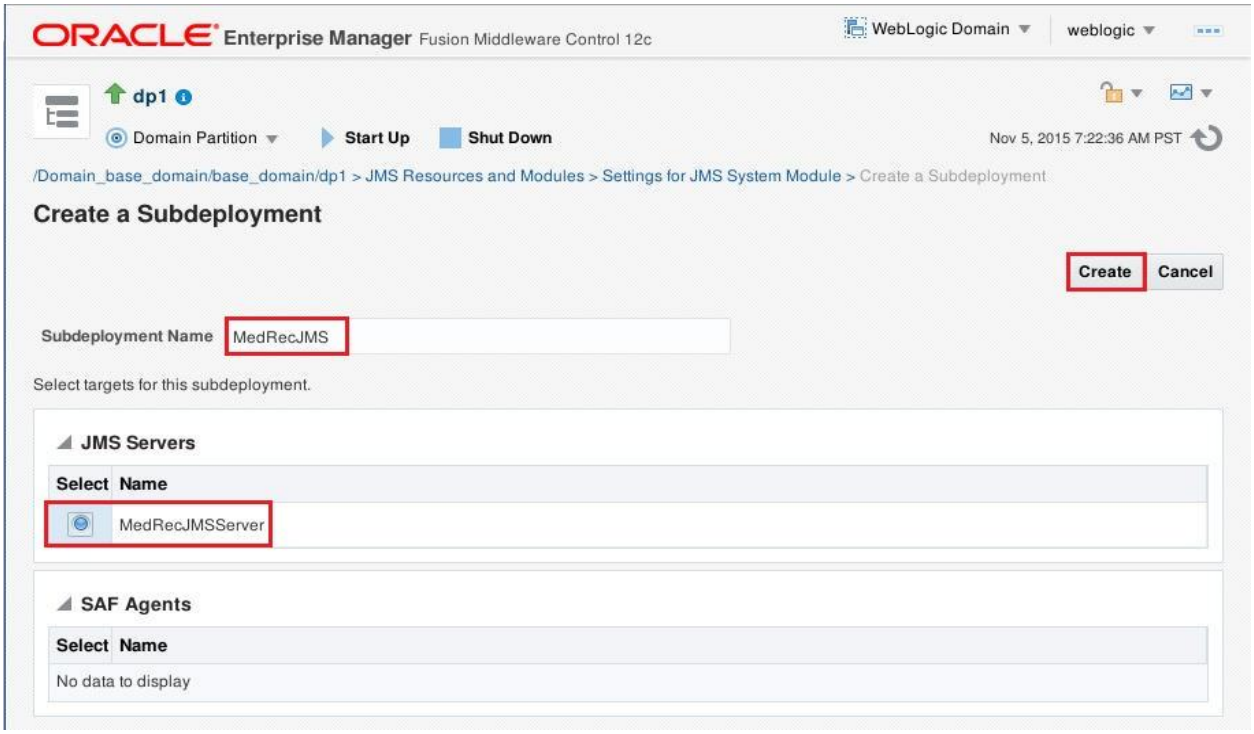
iii. Enter **MedRecModule** as Name then click on **Next**. Click on **Create**.
iv. Under **JMS Modules** tab, click on **MedRecModule**.



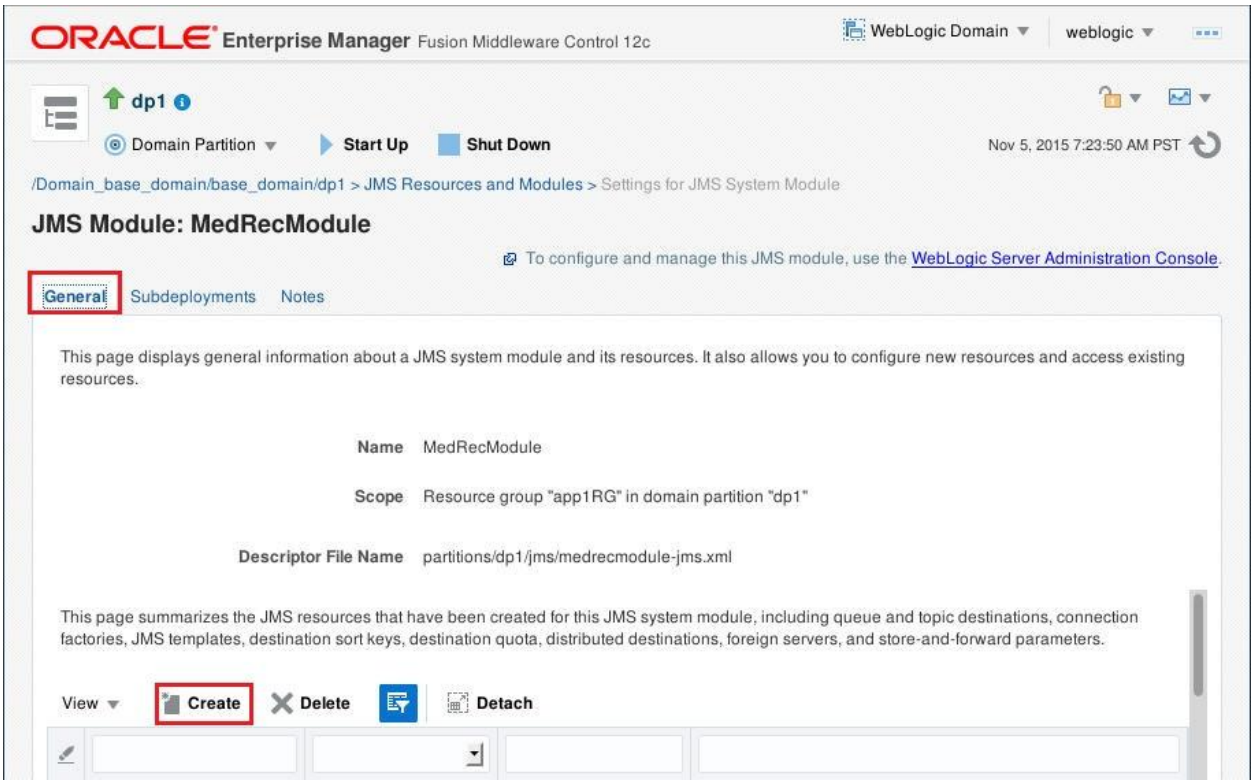
v. Under **Subdeployment** tab, click on **Create**.



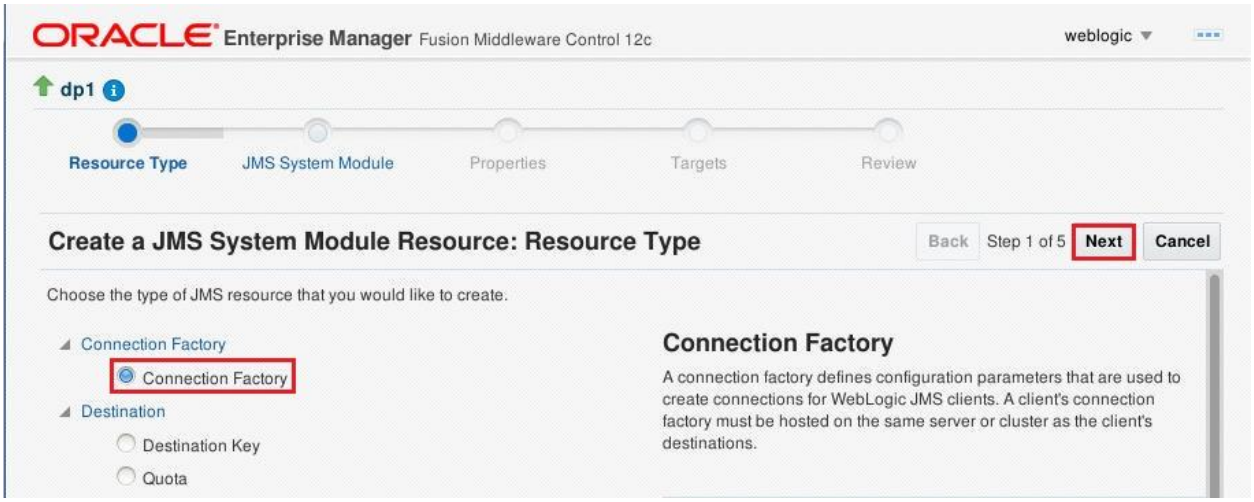
vi. Enter **MedRecJMS** as Name and Select the box near **MedRecJMSServer** then click on **Create**.



vii. Under **General** tab, click on **Create**.

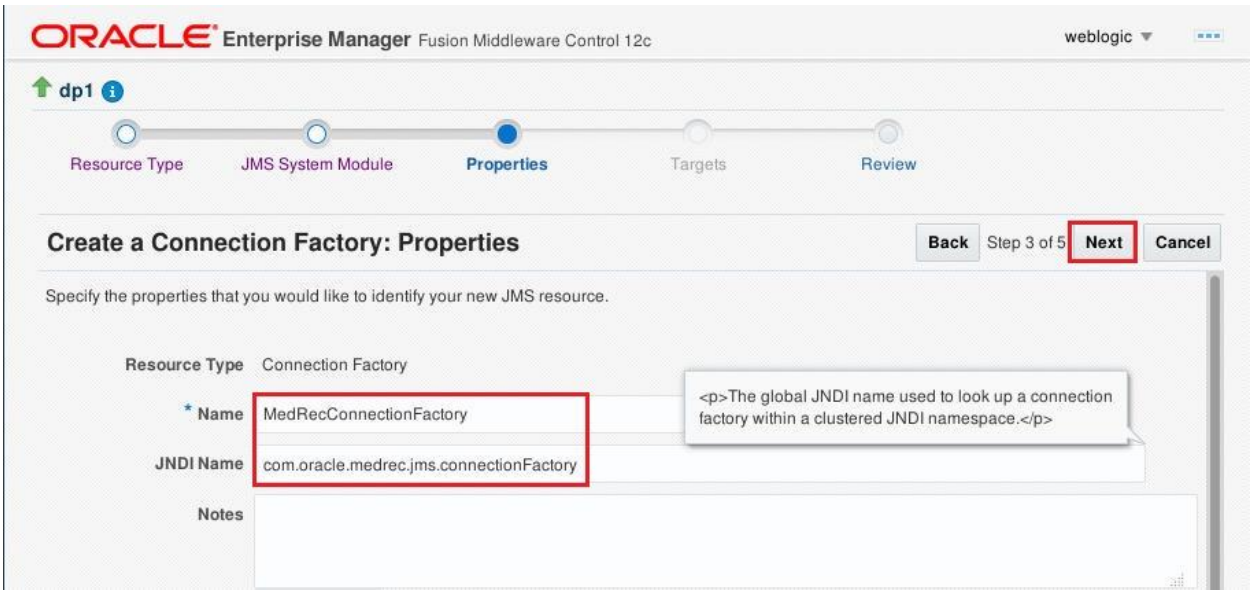


viii. Select box near **Connection Factory** then click on **Next**.



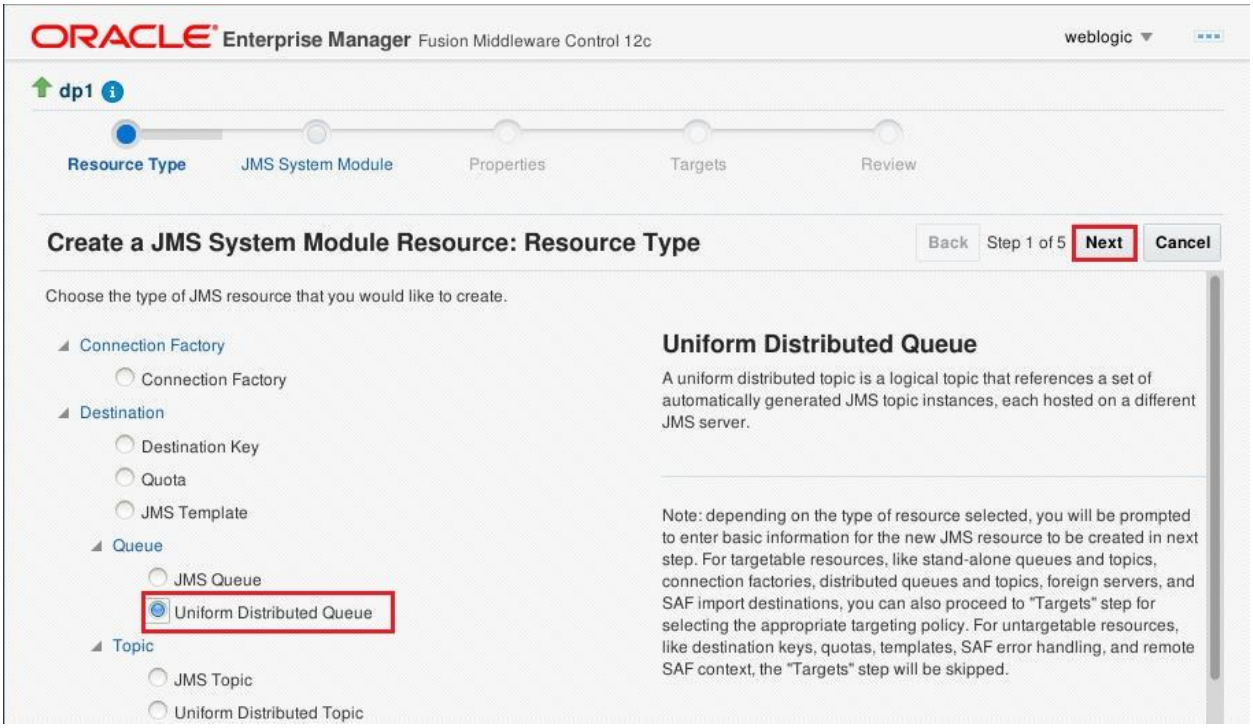
ix. Leave Default in **JMS System Module** Page then click on **Next**.

x. Enter **MedRecConnectionFactory** as Name and **com.oracle.medrec.jms.connectionFactory** as JNDI Name then click on **Next**. Click on **Create**.



xi. Under **General** tab, Click on **Create**.

xii. Select **Uniform Distributed Queue** then click on **Next**.

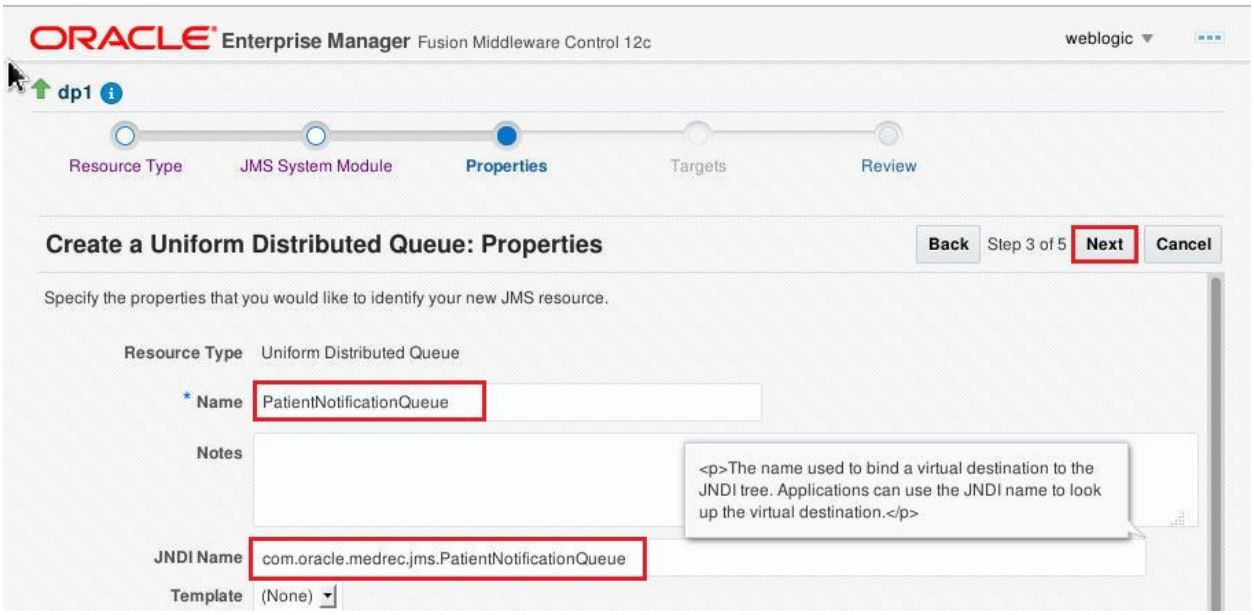


xiii. Click on **Next**.

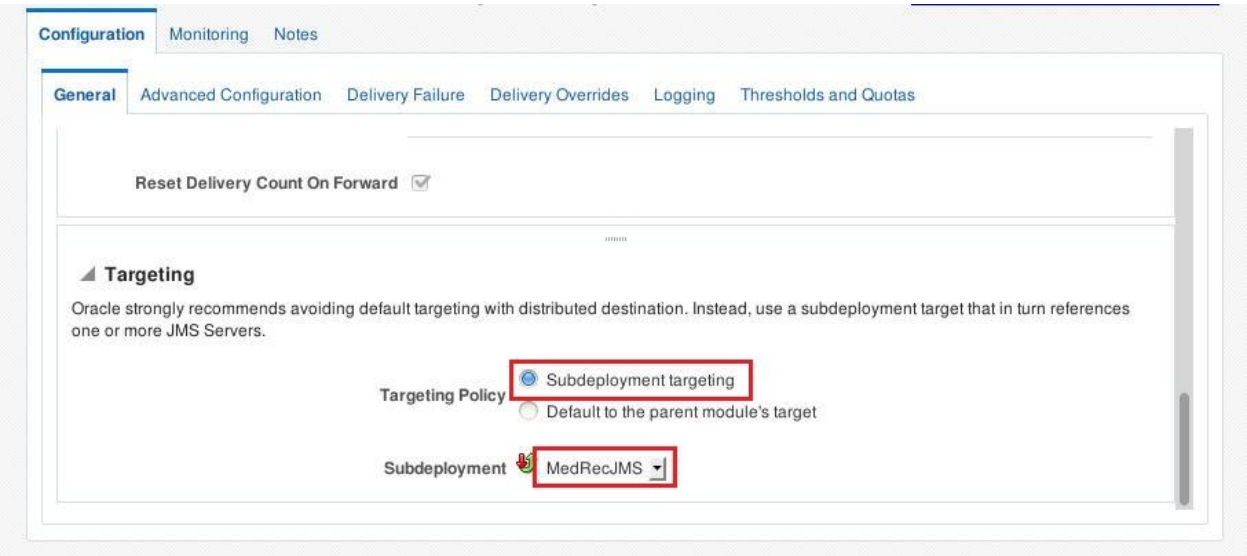
xiv. Enter the following and click on **Next**.

Name: PatientNotificationQueue

JNDI Name: com.oracle.medrec.jms.PatientNotificationQueue

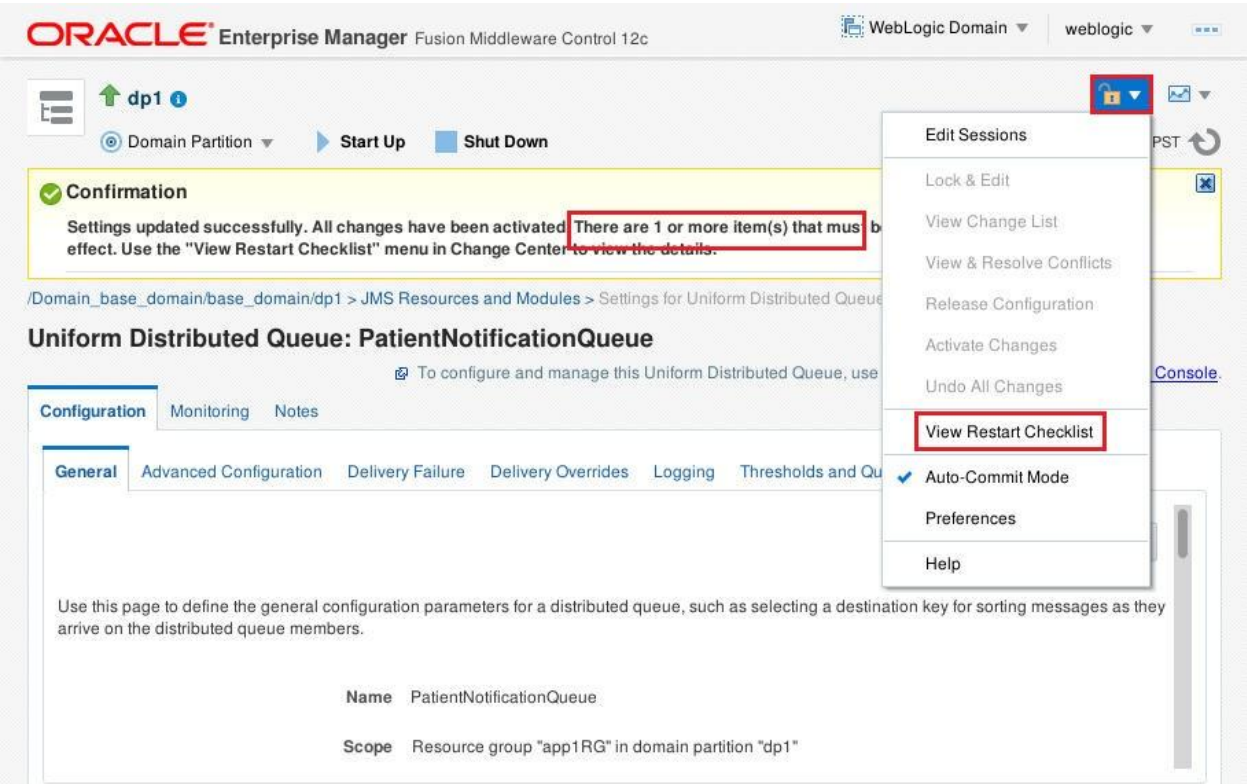


xv. Click on **Create.** Click on **PatientNotificationQueue**, Under **Targeting**, Select **Subdeployment targeting** as **Targeting Policy** and **MedRecJMS** as **Subdeployment** then click on Apply.



Note: You must restart domain partition before going to deploy Medrec Application.

a. Click on Change Center -> View Restart Checklist.

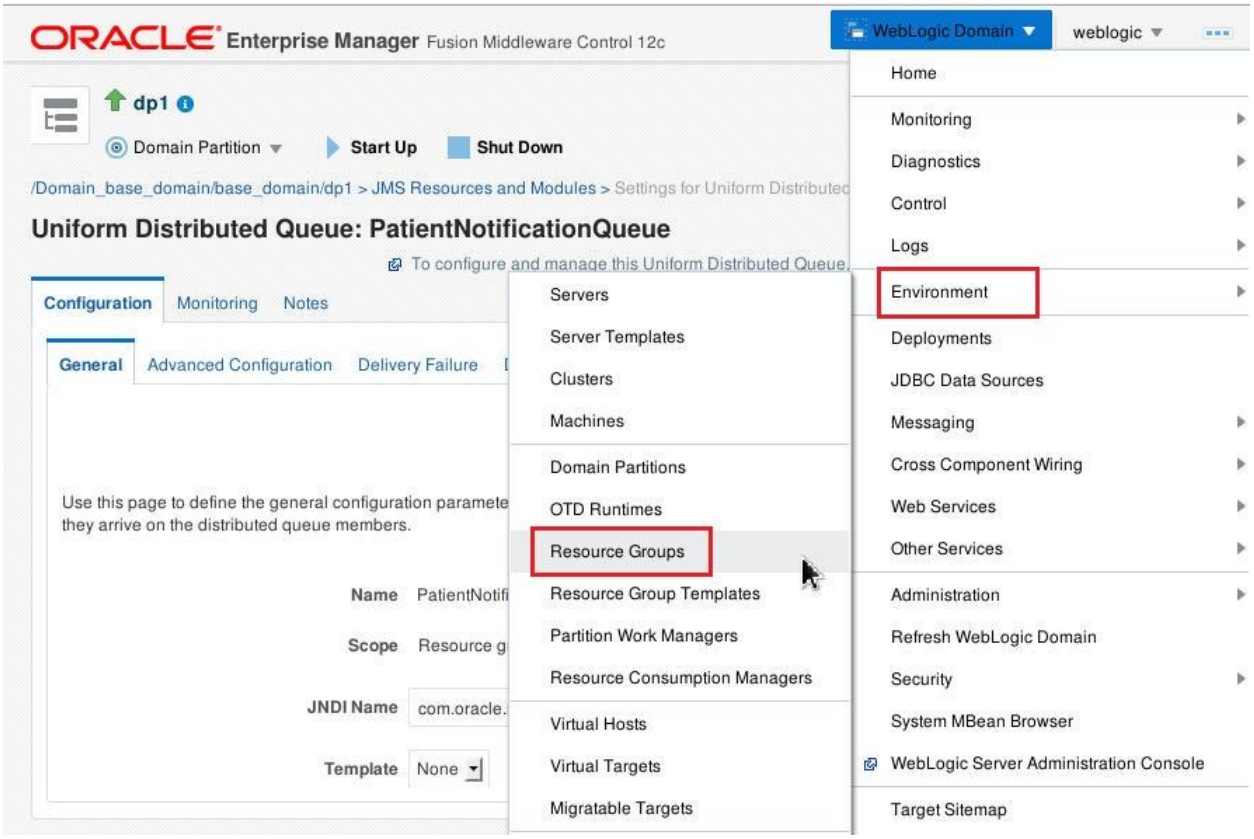


b. Select the box near to dp1 then click on Restart.

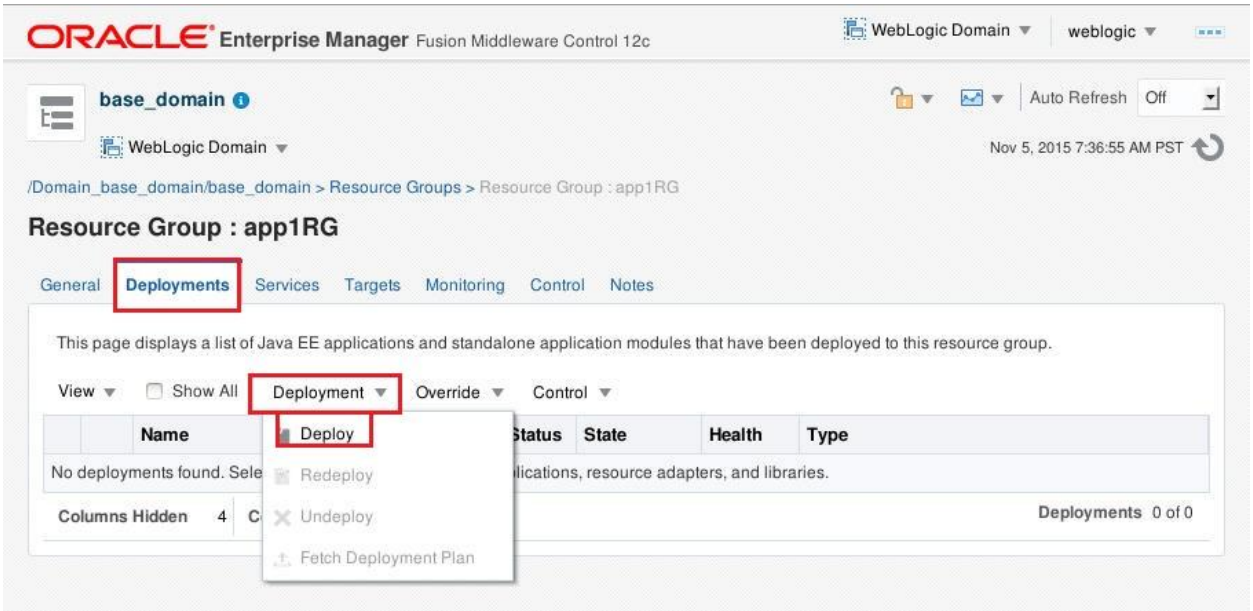
c. On Confirmation Screen, Click on Restart.

dd. Deployment of Medrec Application.

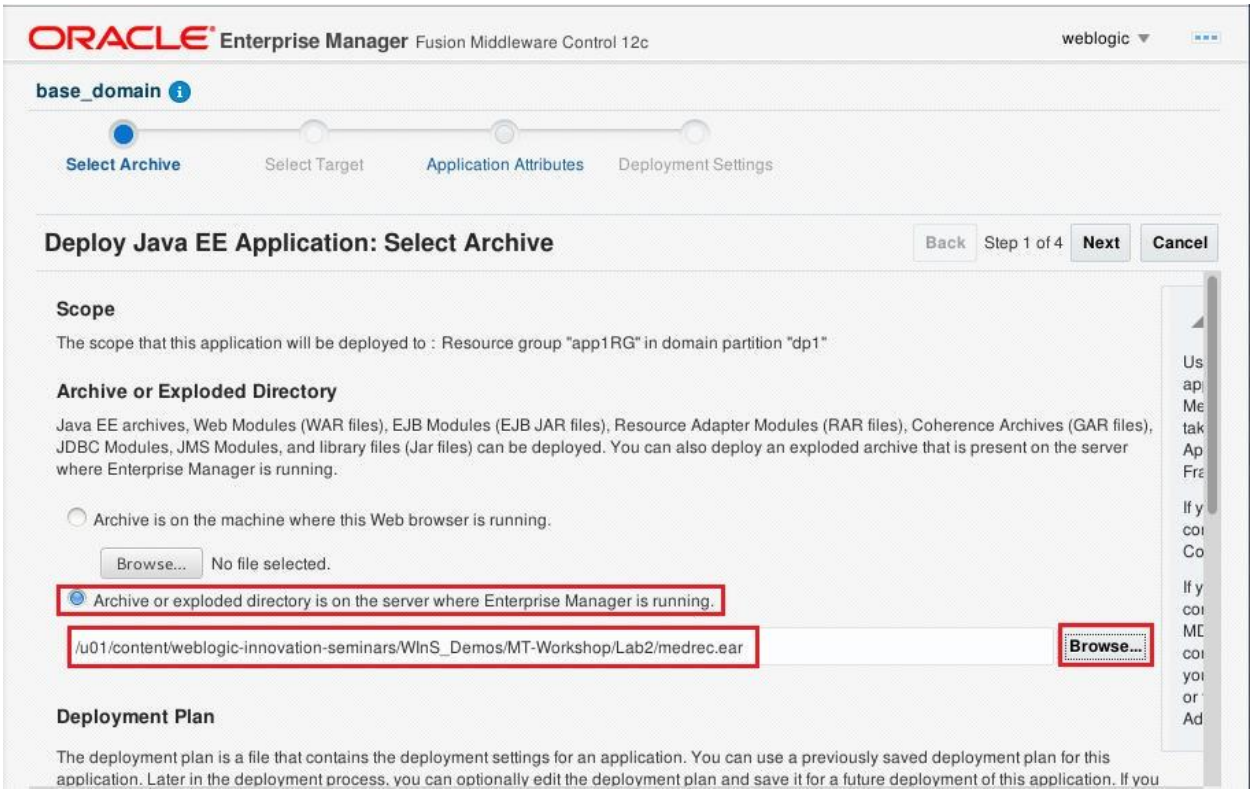
i. Click on **WebLogic Domain -> Environment -> Resource Groups.**



ii. Click on Resource Group **app1RG**, click on **Deployments** tab.
iii. Click on **Deployment -> Deploy**.



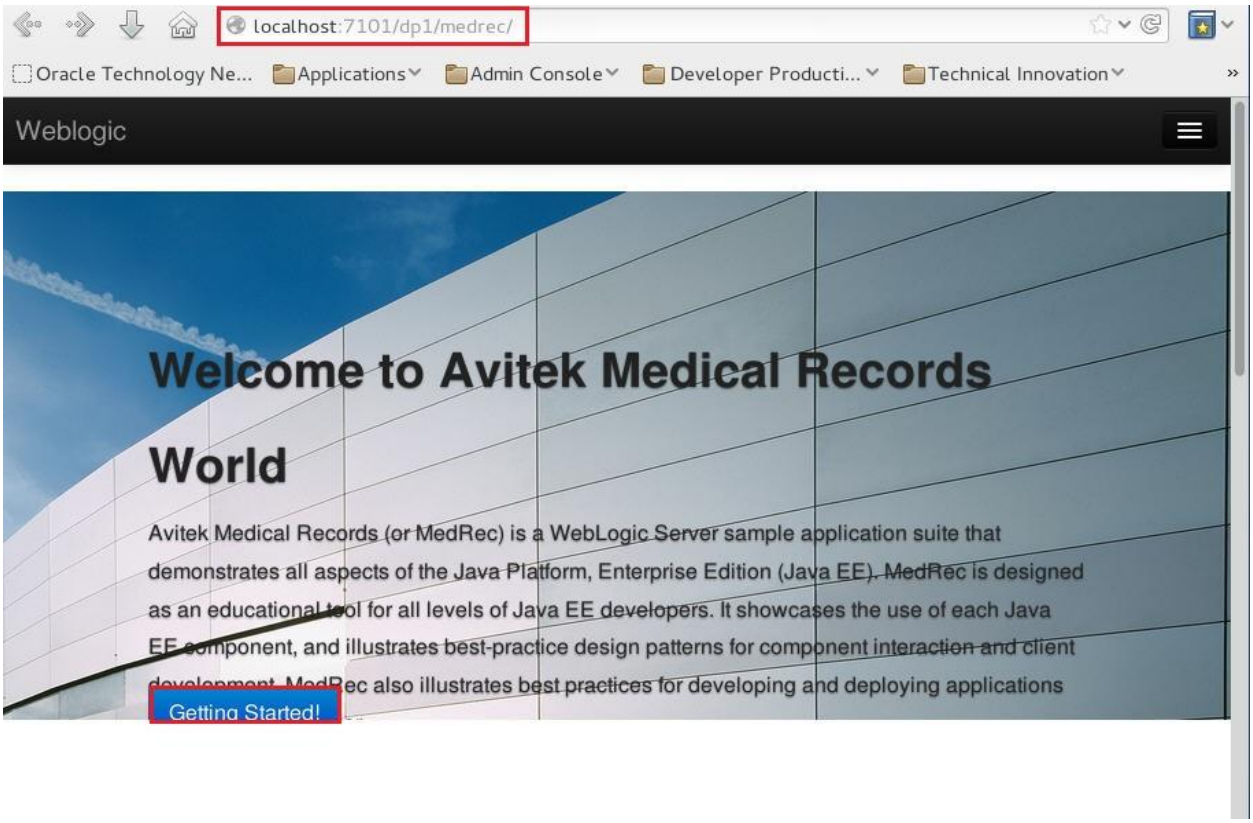
iv. Select "**Archive or exploded directory is on the server where Enterprise Manager is running**" then click on **Browse**. Specify the location of **medrec.ear** from **/u01/content/weblogic-innovation-seminars/WInS_Demos/MT-Workshop/Lab2** then click on **OK**.



v. Click on **Next** then click on **Deploy**. Click on Close.
vi. Deploy **physician.ear** file similarly from the **/u01/content/weblogic-innovation-seminars/WInS_Demos/MT-Workshop/Lab2** location.
vii. Deploy **chat.war** file similarly from the **/u01/content/weblogic-innovation-seminars/WInS_Demos/MT-Workshop/Lab2** location.

Accessing Medrec Application in Domain Partition dp1

i. Go to Firefox and type the URL: http://localhost:7101/**dp1**/medrec/
ii. Click on Getting Started.



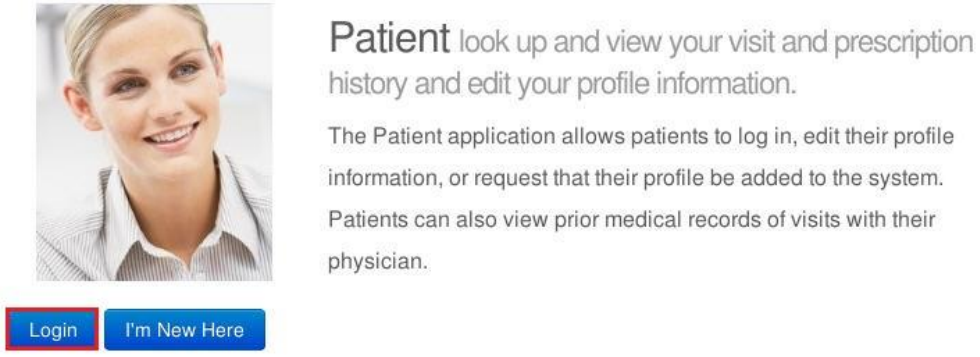
iii. Under Patient, Click on **I'm New Here**
iv. Enter the following or any other data then click on Submit.

Email:	weblogic@oracle.com
Password:	welcome1
Confirm Password:	welcome1
First Name:	Ankit
Last Name:	Pandey
Gender:	Male
DOB:	Jun 23, 1988
SSN:	123456788

Note: Make sure you not use 123456789 as SSN Number.
v. Click on **Getting Started** again on Medrec Home Page.
vi. Under Administrator, Click on **Login**.

vii. Enter **administrator/administrator123** as username and password then click on **Sign In**.

viii. Under **Pending Requests**, click on **Go**.
ix. Click on the Email Id, and then click on **Approve**.
x. Click on Logout. Click on Logout again.
xi. You can login as weblogic@oracle.com/welcome1 as username/password as Patient.
xii. You can view your record summary, and you can also have interaction with physician.



Configuration of Medrec Application in Domain Partition 2

In the next step we are creating the below configuration for Medrec application in domain partition dp2. We are using the same Medrec applications and same JNDI name for the Datasources, connection factory and Distributed queue but we will connect to different database. So there are two benefits of Multitenancy.

- In a domain, you can deploy the same application in two different domain partitions and there will be no JNDI conflict. You do not have to make any changes in application.
- In single domain, you can have same application deployed in two different domain partitions and connected to two databases. So both the application will have different Set of Users in our case or different set of Application Specific Data.

Virtual Target: VT-Medrec-2
Domain Partition: dp2
Resource Group app2RG
app1RG:
Datasource: MedRec2GlobalDataSourceXA, jdbc/MedRecGlobalDataSourceXA
Mail Session: MedRecMailSession, mail/MedRecMailSession
Persistence Store: MedRec2-fs
JMS Server: MedRec2JMSServer
JMS Module: MedRec2Module
MedRecModule:
Subdeployment: MedRec2JMS
Connection Factory: MedRec2ConnectionFactory, com.oracle.medrec.jms.connectionFactory
Distributed Queue: PatientNotificationQueue com.oracle.medrec.jms.PatientNotificationQueue
Applications: medrec.ear
physician.ear
chat.war

The Scripts MedrecInDP2.sh creates the Virtual Target, Domain Partition, Resource Group, JDBC Datasource, JMS Server, JMS Module, Distributed Queue and Connection Factory then deploys the medrec.ear, physician.ear and chat.war.

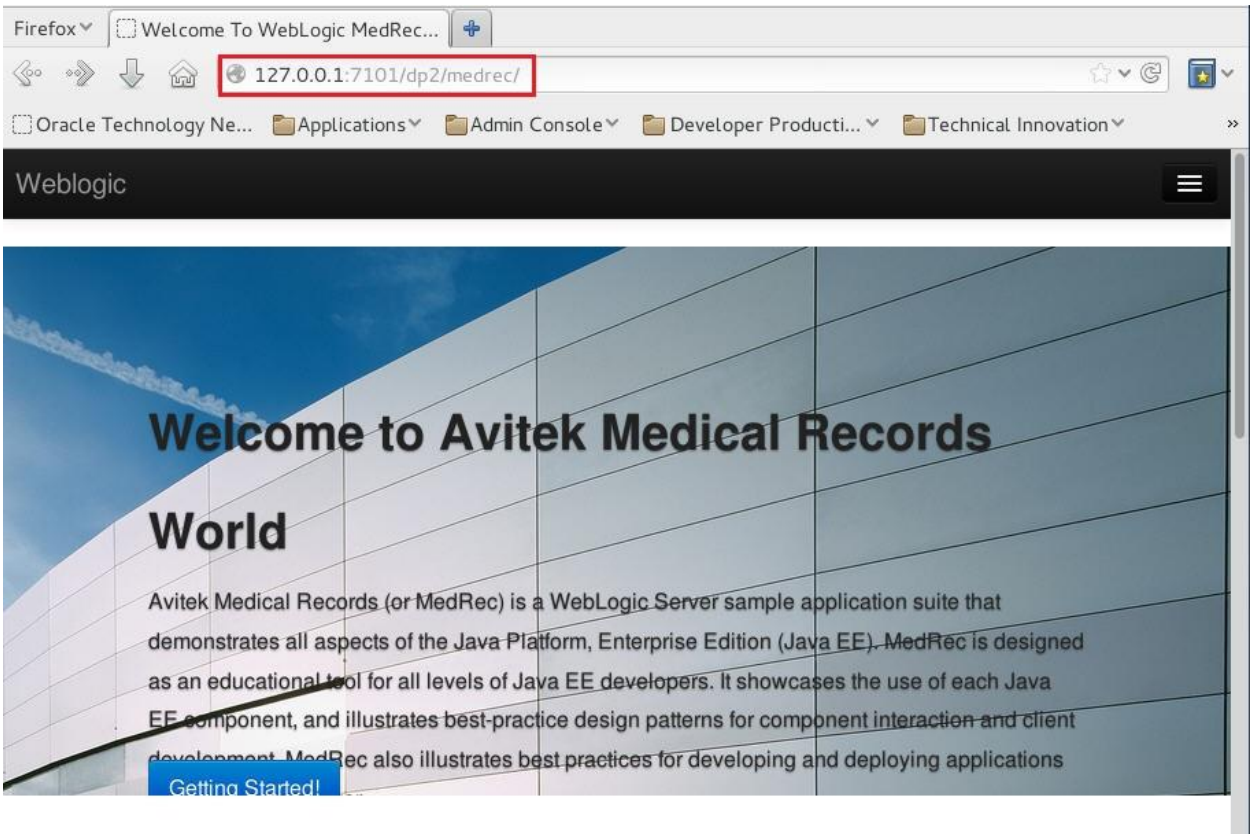
a. Open a new tab.
b. cd **/u01/content/weblogic-innovation-seminars/WInS_Demos/MT-Workshop/Lab2**
c. **./Medrec2DB.sh** (It creates the required user medrec2 in pdb2 and populates the sample data.)
d. **./MedrecInDP2.sh**
e. Go back to Fusion Middleware Control http://localhost:7001/em.
f. Verify the creation of the following resources.
    i. Click on **WebLogic Domain ->Environment ->Virtual Target**. Here we have **dp2** as **URI Prefix** for the Virtual Target **VT-Medrec-2**.
    ii. Click on **WebLogic Domain ->Environment ->Domain Partition**.
    iii. Click on Domain Partition **dp2** then Select **Domain Partition -> Administration -> Resource Groups**.
    iv. Click on Resource group **app2RG**.

v. In the "**Services**" and "**Deployments**" tab, you can verify the creation of above System Resources here.

Accessing Medrec Application in Domain Partition dp2

While accessing the application we need to use the Virtual Target URI. As domain partition dp2 is targeted to Virtual Target VT-Medrec-2, which has /dp2 as URI, we need to add it in URL for the accessing the application.

a. In Firefox, type the URL: http://localhost:7101/dp2/medrec/



b. As both this application has exactly same JNDI name used within application, JNDI name of Datasource, JMS connection factory, mail sessions, Distributed Queue.
c. Click on Getting Started!
d. Under **Patient**, Click on Login, Try to login with weblogic@oracle.com/welcome1. You will not be able to login. As both Medrec application is connected to different database. So in Multitenant WebLogic Server, you can deploy exactly same application with same configuration but with different database and there will be no JNDI conflict in domain.

Configuration of Day Trader application in domain partition 3

Here we will create the below configuration through WLST to Run Day Trader Application on Domain Partition dp3.

Virtual Target: VT-daytrader
Domain Partition: dp3
Resource Group app3RG
app1RG:
- Datasource:
  - Name: jdbc/datasources/TradeDataSource,
  - JNDI Name: **jdbc/datasources/TradeDataSource**
  - Name: jdbc/datasources/NoTxTradeDataSource
  - JNDI Name: **jdbc/datasources/NoTxTradeDataSource**
- Persistence Store: MyFileStore
- JMS Server: MyJMSServer
- JMS Module: MyJMSModule
- MyJMSModule:
  - Subdeployment: MySubdeployment
  - Connection Factory:
    - Name: jms/myQueueConnectionFactory,
    - JNDI Name: **jms/myQueueConnectionFactory**
    - Name: jms/myTopicConnectionFactory
    - JNDI Name: **jms/myTopicConnectionFactory**
  - Distributed Queue:
    - Name: jms/TradeBrokerQueue
    - JNDI Name: **jms/TradeBrokerQueue**
  - Distributed Topic:
    - Name: jms/TradeStreamerTopic
    - JNDI Name: **jms/TradeStreamerTopic**
- Applications: web-3.0.0.war

This DayTraderInDP3.sh creates the Virtual Target, Domain Partition, Resource Group, JDBC Datasource, JMS Server, JMS Module, Distributed Queue and Connection Factory then deploys the web-3.0.0.war.

a. cd/ **u01/content/weblogic-innovation-seminars/WInS_Demos/MT-Workshop/Lab2**
b. **./DayTrader3DB.sh** (It creates the required user for pdborcl database.)
c. ./DayTraderInDP3.sh
d. Go back to Fusion Middleware Control http://localhost:7001/em
e. Verify the creation of the following resources.
   i. Click on **WebLogic Domain ->Environment ->Virtual Target**. Here we have **dp3** as **URI Prefix** for the Virtual Target **VT-daytrader**.
   ii. Click on **WebLogic Domain ->Environment ->Domain Partition**.
   iii. Click on Domain Partition to **dp3** then Select **Domain Partition -> Administration -> Resource Groups**.
   iv. Click on Resource group **app3RG**.
   v. In **Services** and **Deployments** tab, you can verify the creation of above System Resources here.

Access Day Trader Application in Domain Partition dp3

a. Go to Firefox and type the URL: http://localhost:7101/dp3/daytrader/
b. We will not test here the application behavior. The application is loaded and is running inside partition.

You can create many domain partitions in WebLogic 12.2.1. Your application does not require any specific application development. Your application which is working in previous version of WebLogic 12c, you can deploy them in multitenant environment without modifying the application. Daytrader application is developed by IBM which we deployed in domain partition dp3. So you can also easily deployed application build on different platform to WebLogic Server.

LAB 3: SECURITY ISOLATION

Overview

A security realm comprises mechanism for protecting WebLogic resources. Each security realm consists of a set of configured security providers, users, groups, security roles, and security policies. You use realms to configure authentication, authorization, role mapping, credential mapping, auditing and other services.
WebLogic Server traditionally supports multiple realms in a domain configuration, but only one realm-typically referred to as the “default realm” or “admin realm” –can be active at any given time.
In contrast, WebLogic Server MT supports multiple active realms and allows each partition to execute against a different realm.
This means that a partition can have unique security providers, users, groups, security roles and security policies. Resources and applications in the domain partition are available to only to users within the domain partition’s security realm. Other tenants cannot see or access the resources or applications.

Note: Partition can share a security realm, with consequent loss of independence isolation. In particular if you do not specify a realm when you create a partition, the default realm is shared with the partition and there is no security isolation between the partition and the domain.

In this Lab, first we will create a new security realm then we will assign the security realms to a partition dp1. So we will have two security realms in our domain one which we created and one is default security realm. Domain partition dp1 will have new security realm and other domain partitions will have default security realm. Default security realm has a user administrator with password administrator123. And new security realm will have user administrator with password welcome1. So Medrec application deployed in dp1 can be used by users of new security realm. You will not be able to login as an Admin in Medrec Application with default security realm user (administrator/administrator123).

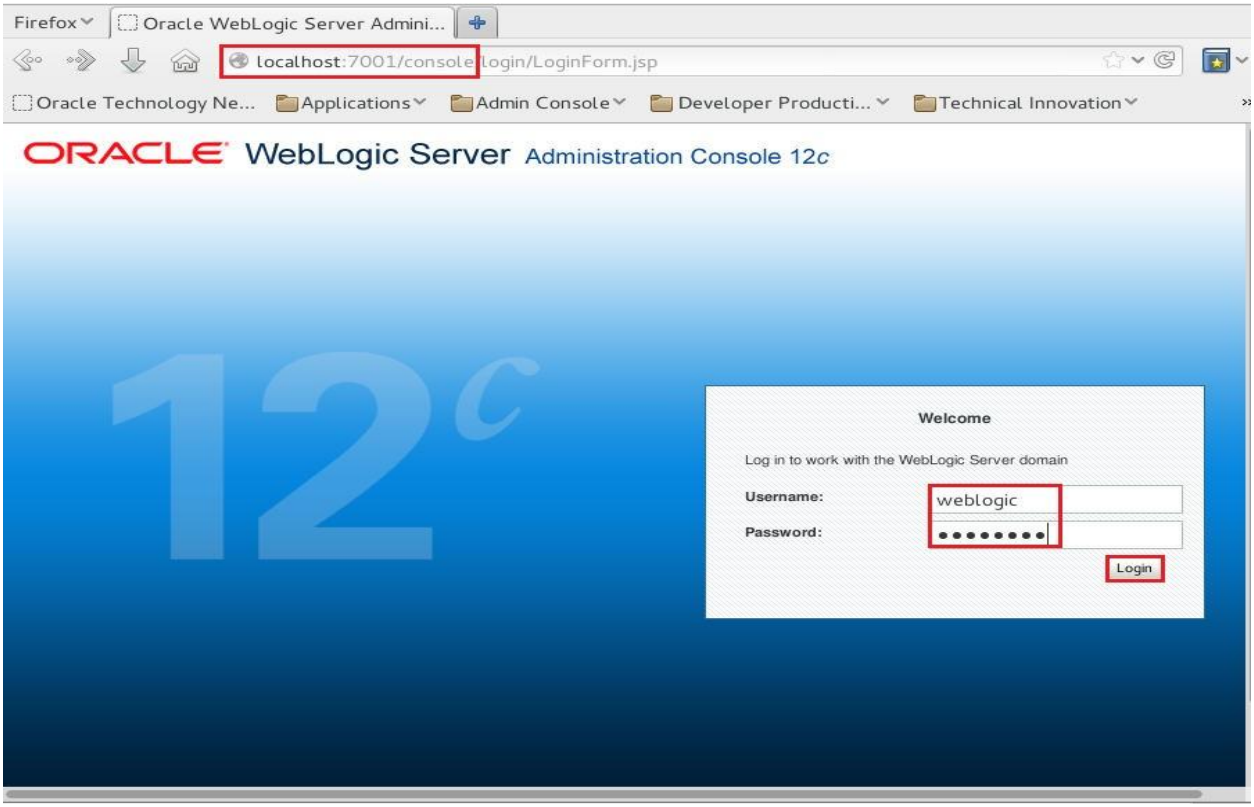
In this lab we are going to perform the following operations:

- Creation of New Security Realm.
- Assigning a new Security realm to Domain Partition.
- Medrec application deployed in two different domain partitions which are using two different security realms in single domain.

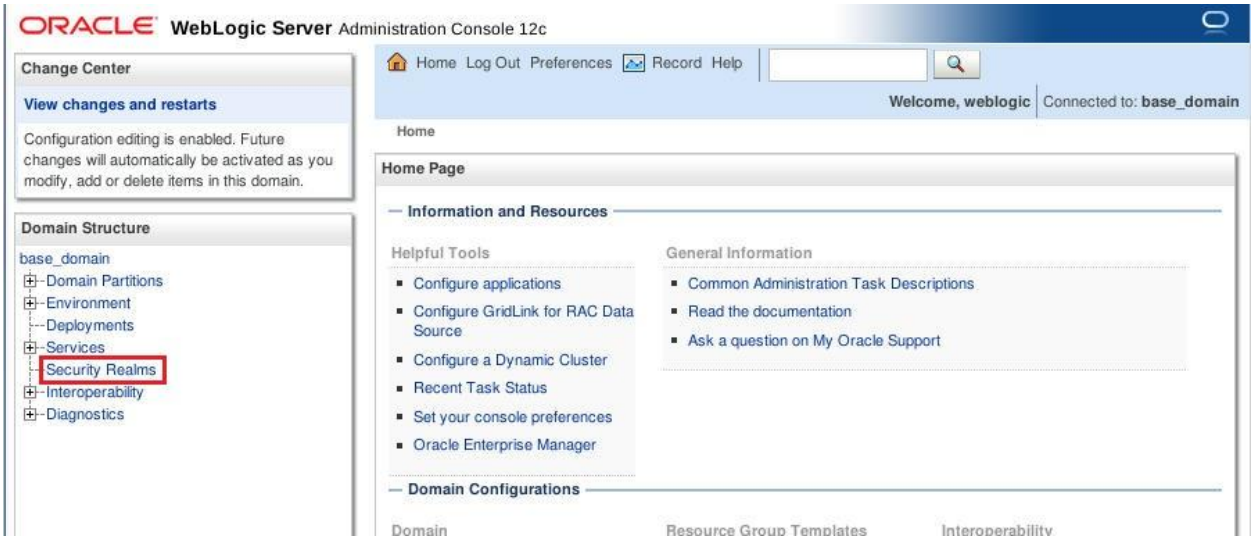
Creating a New Security Realm

i. The creation of Security Realm is more automated still inside WebLogic Console versus Fusion Middleware Control. This is why we will use WebLogic Console for that action. Go to Firefox and type the WebLogic Admin Console URL: http://localhost:7001/console.

ii. Enter **weblogic/welcome1** as username/password then click on Login.

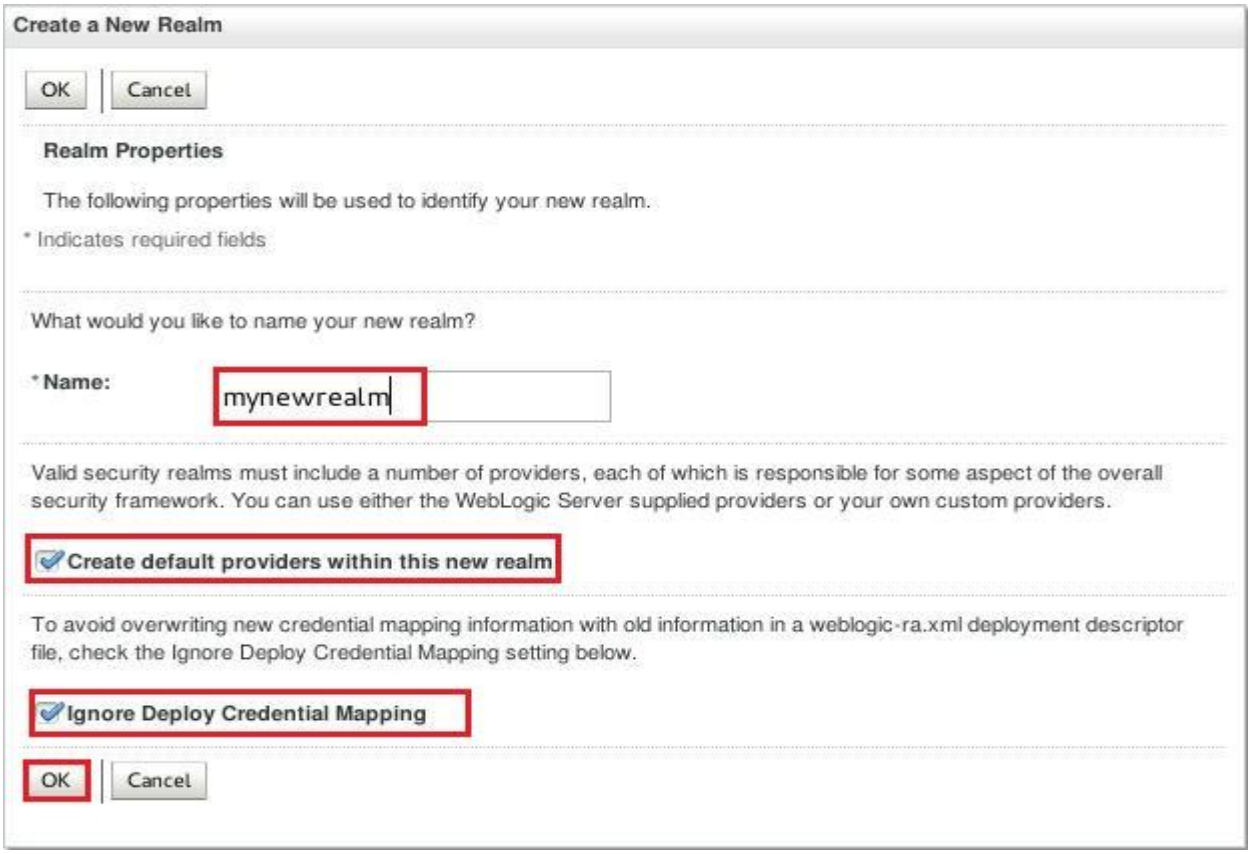


iii. Under Domain Structure, click on **Security Realms**.



iv. Click on **New**.

v. Enter **mynewrealm** as Name; check the box for "**Create default providers within new realm**" and "**Ignore Deploy Credential Mapping**" then click on **OK**.



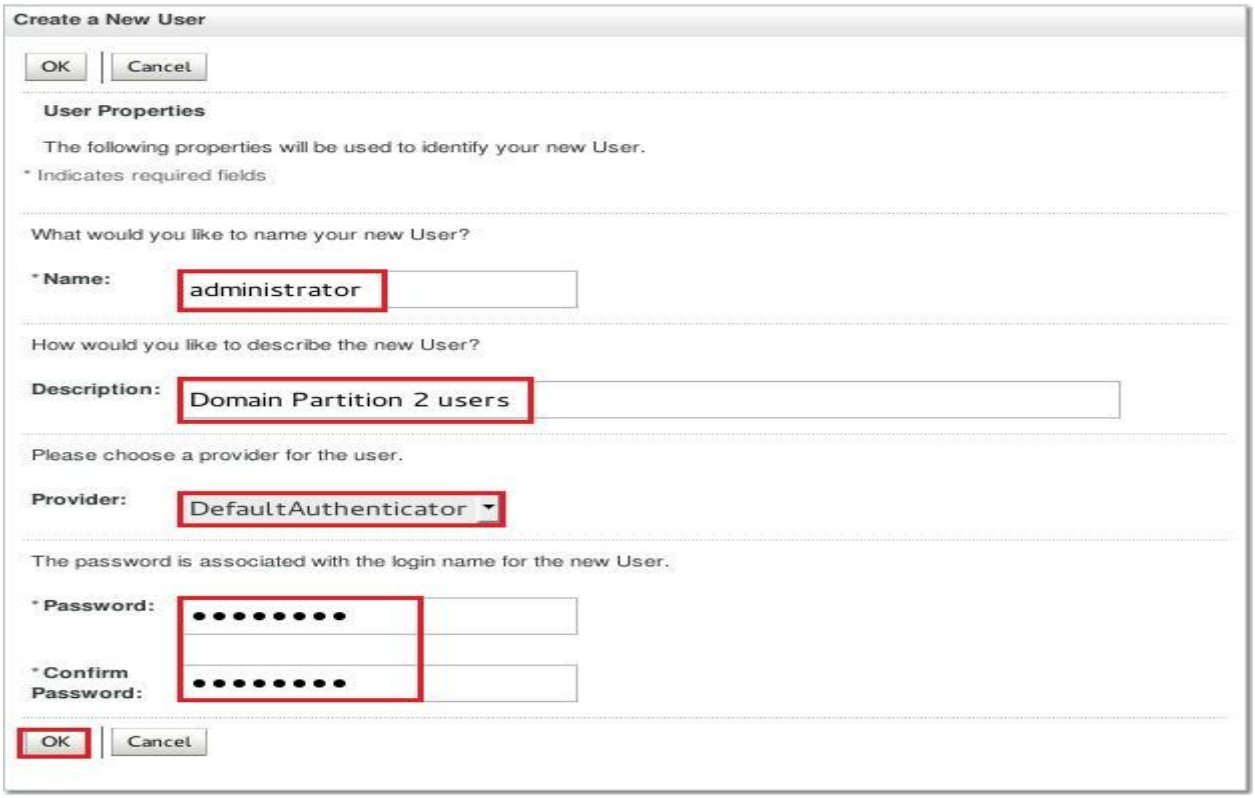
vi. Click on **mynewrealm**.

vii. Click on **Users and Groups -> Users** tab.

viii. Click on New.

ix. Enter the following then click on **OK**.

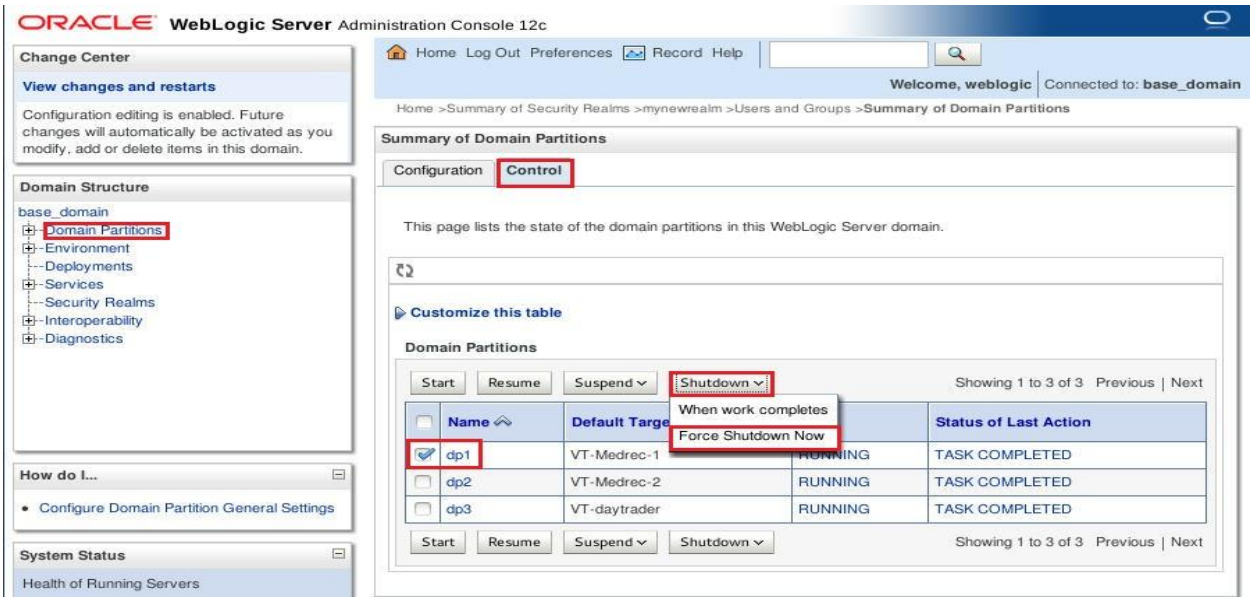
Name:	administrator
Description:	Domain Partition 2 users
Provider:	Default Authenticator
Password:	welcome1
Confirm Password:	welcome1



Assign the mynewrealm security realm to domain partition dp1.

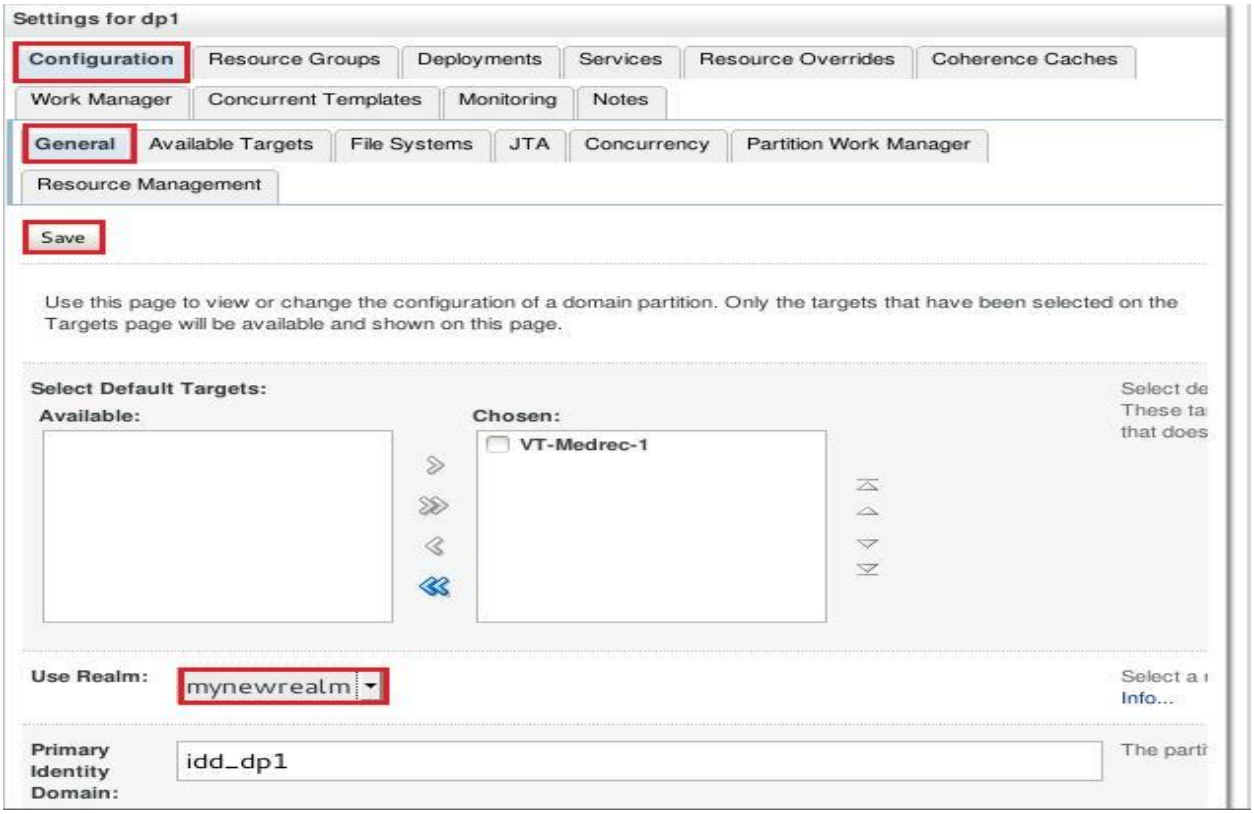
i. Click on **Domain Partitions**, then on **Control** tab.

ii. Check the box near dp1 and click on **Shutdown ->Force Shutdown Now**.

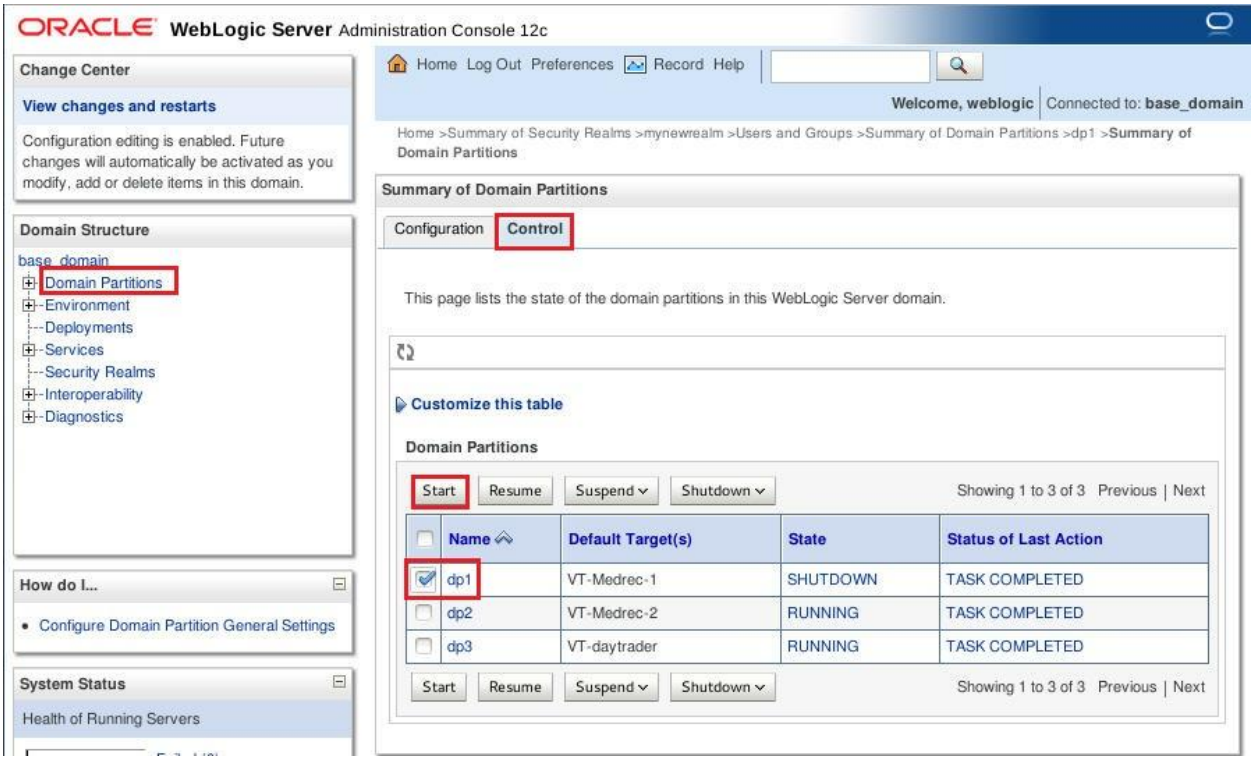


iii. Once domain partition shutdown, click on dp1.

iv. In **Configuration-> General** tab, Under Use Realm, select **mynewrealm** then click on **Save**.



v. Click on **Domain Partitions**, then on **Control** tab.
vi. Select the box near dp1 and click on **Start**.



Verified that we have two security realms in different domain partition in single domain

i. Go to Firefox and type the URL: http://localhost:7101/dp1/medrec/index.xhtml
ii. Under Administrator, click on Login.
iii. Login with old security realm credential that is administrator/administrator123.
iv. You must get “Incorrect username or password!”
v. Login with new security realm credential that is administrator/welcome1.
vi. Click on Logout. Click on Logout again.
vii. Go to Firefox and type the URL: http://localhost:7101/dp2/medrec/index.xhtml
viii. Under Administrator, click on Login.
ix. Login with new security realm credential that is administrator/welcome1.
x. You must get “Incorrect username or password!”
xi. Login with old security realm credential that is administrator/adminsitrator123.
xii. Click on Logout. Click on Logout again.

Here we learned how you can have two security realms in a domain. So one of your domain partitions can use default security realm and other can use custom security realm created by you in same domain.

LAB 4: EXPORT/IMPORT DOMAIN PARTITION

Overview

Exporting and importing partitions let WLS system administrators easily move partitions from one domain to another, including the applications that are deployed to the partition. This feature is useful for replicating partitions across the domains and for moving domains from a development to a testing then production environment.
Exporting a domain partition creates a partition backup and stores it in an archived format. You can export a domain partition from source domain with its entire configuration and data. With few configuration changes, you can then import the partition archive into another instance of multi-tenant WLS (the target domain). You might need to update any domain dependencies, such as targets and security realms, and optionally update other attributes in the partition configuration to make it valid.

When a partition is exported from the source domain it is packaged in a partition archive. Included in the partition archive is:

- The partition configuration.
- Any resource group contained in the partitions.
- Any resource group templates referred to by those resource groups.
- The contents of the file system, <partition-file-system>/config directory.
- Optionally, application binaries and configuration for applications deployed to the partition.

No application runtime state or application-specific runtime configuration is included in the partition archive. Examples of what would not be exported are JMS messages in queues, users in an embedded LDAP realm.
In this Lab, We will create a Non-JRF domain dev_domain and configure it with all required resources for Medrec application. You will remove domain partition dp1 from base_domain. As this domain partition is targeted to Virtual Target VT-Medrec-1 and we are going to import a new domain partition on this Virtual target. So we need to remove this domain partition dp1 for this Lab2.

We are going to learn the following:

- Exporting a domain partition from a Non-JRF domain dev_domain and importing it to a Restricted JRF domain base_domain.

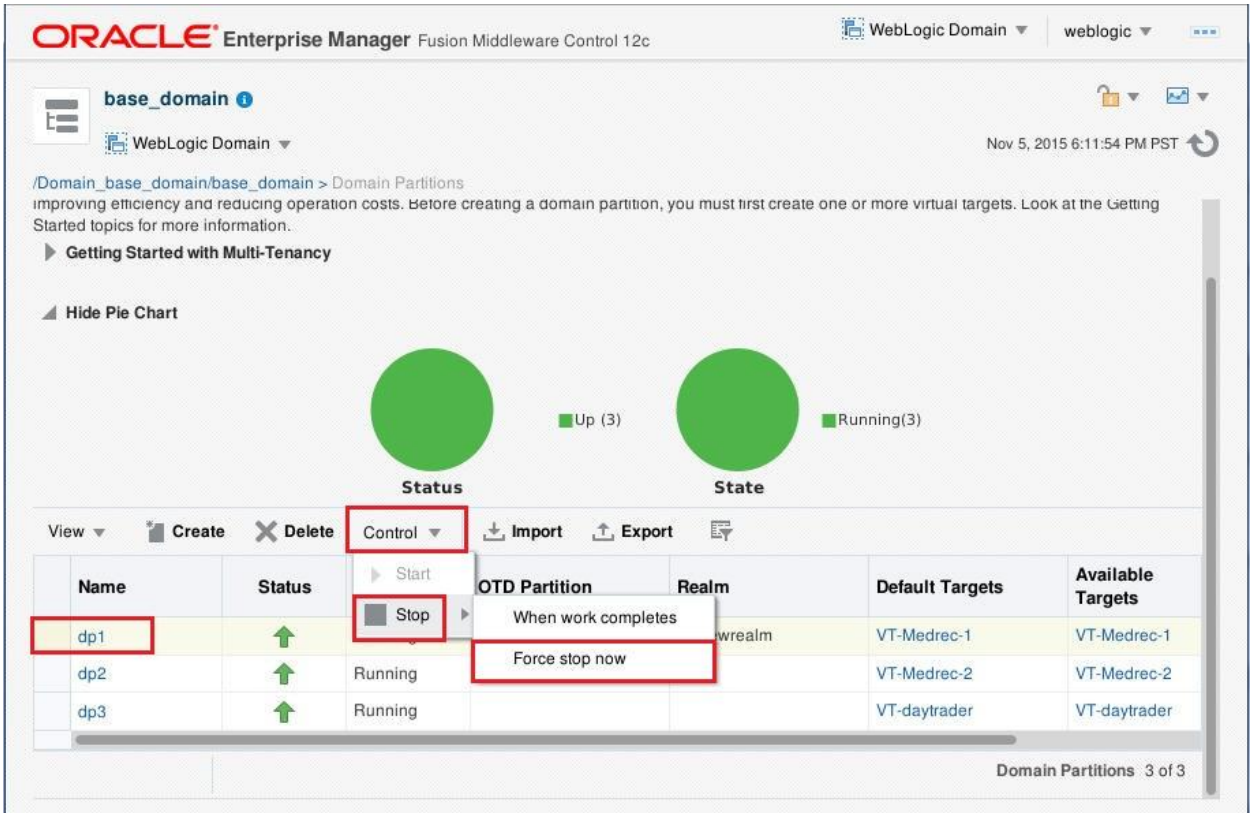
Note: To simplify import/export both domains contain Virtual Target with the same name. In our case, base_domain and dev_domain has VT-Medrec-1 as Virtual Target. It is possible to make import/export without that restriction. To import partition and bind it to virtual target of different name, you should modify settings inside associated “*-attributes.JSON” file

Please remember that Virtual Target may be assigned to the one and only one domain partition. This is why before importing please ensure that original domain partition “dp1” was completely removed from domain configuration (don’t miss step c on page 47!)

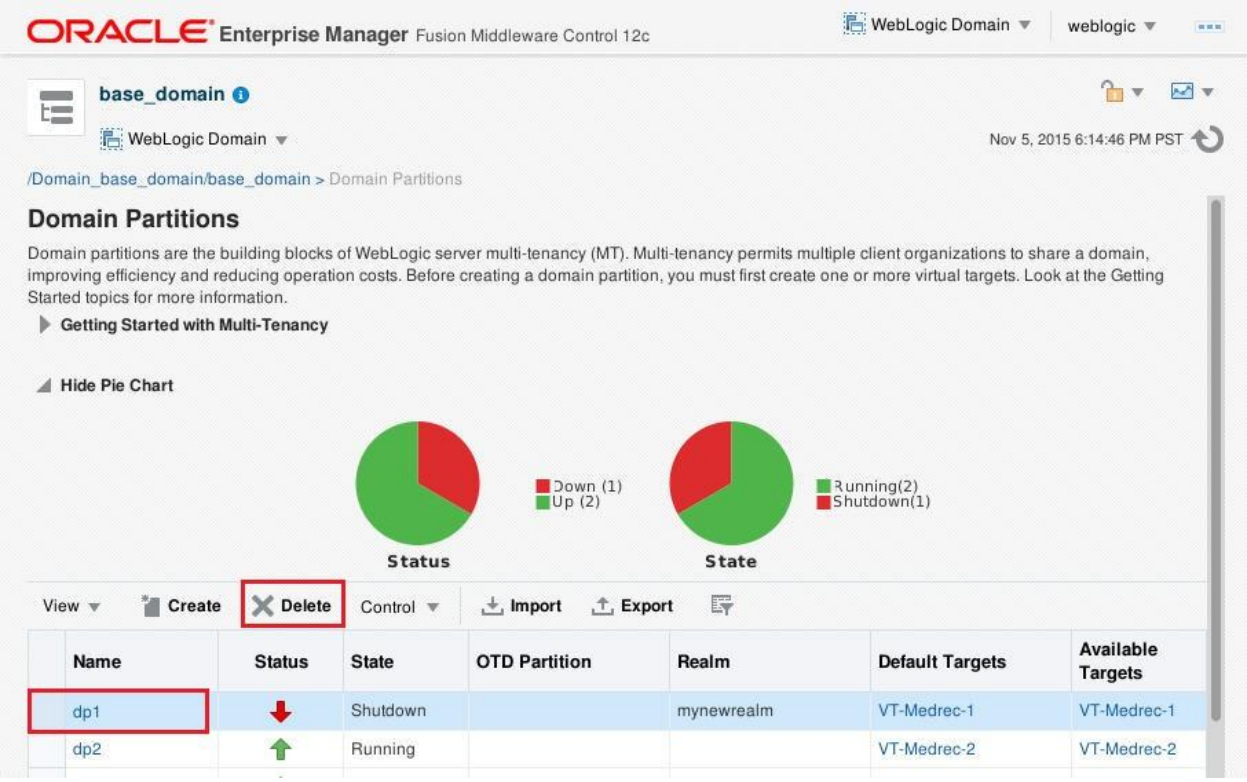
Stop and remove domain partition dp1 from base_domain.

a. In Fusion Middleware Control http://localhost:7001/em , Click on **WebLogic Domain -> Environment -> Domain Partition**.

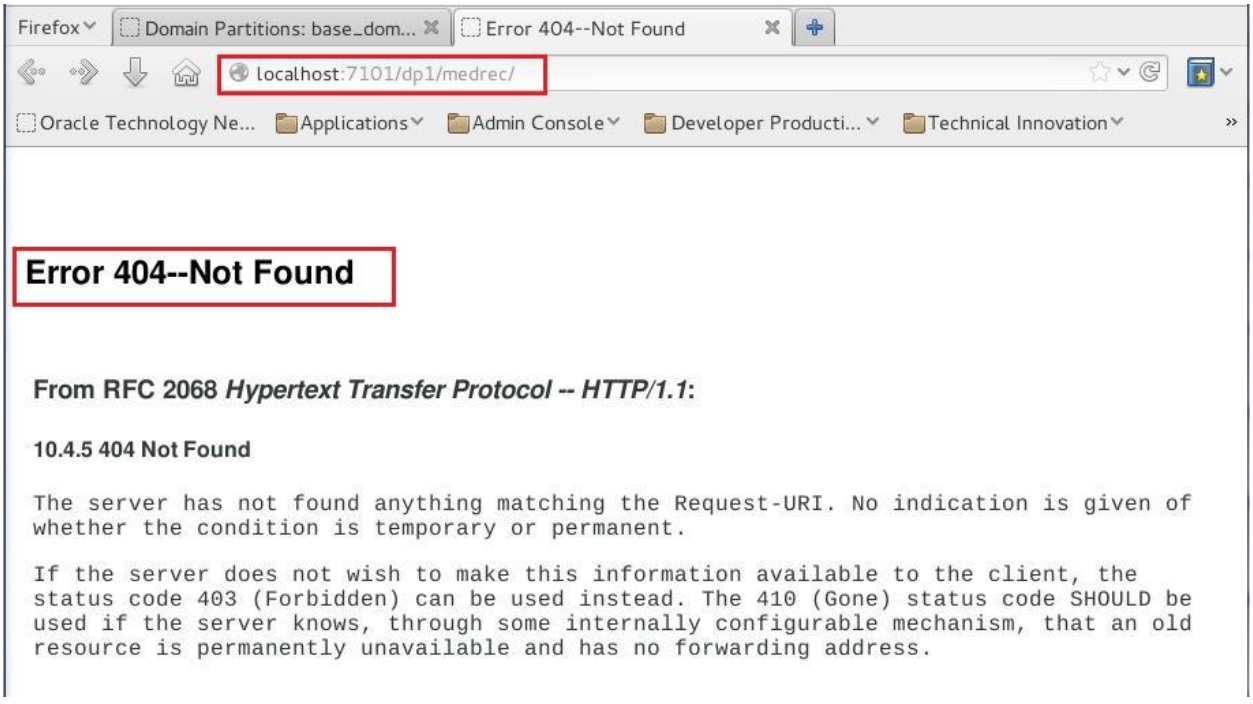
b. Check the **box** near **dp1** then click on **Control -> Stop->Force Stop Now.** On the Confirmation Screen Click on **OK**. Once you see the message "**Partition state after the operation is SHUTDOWN**" then click on **Close.**



c. Check the box near **dp1** and make it highlighted then click on **Delete**. In Delete Domain Partition Screen, click on OK.



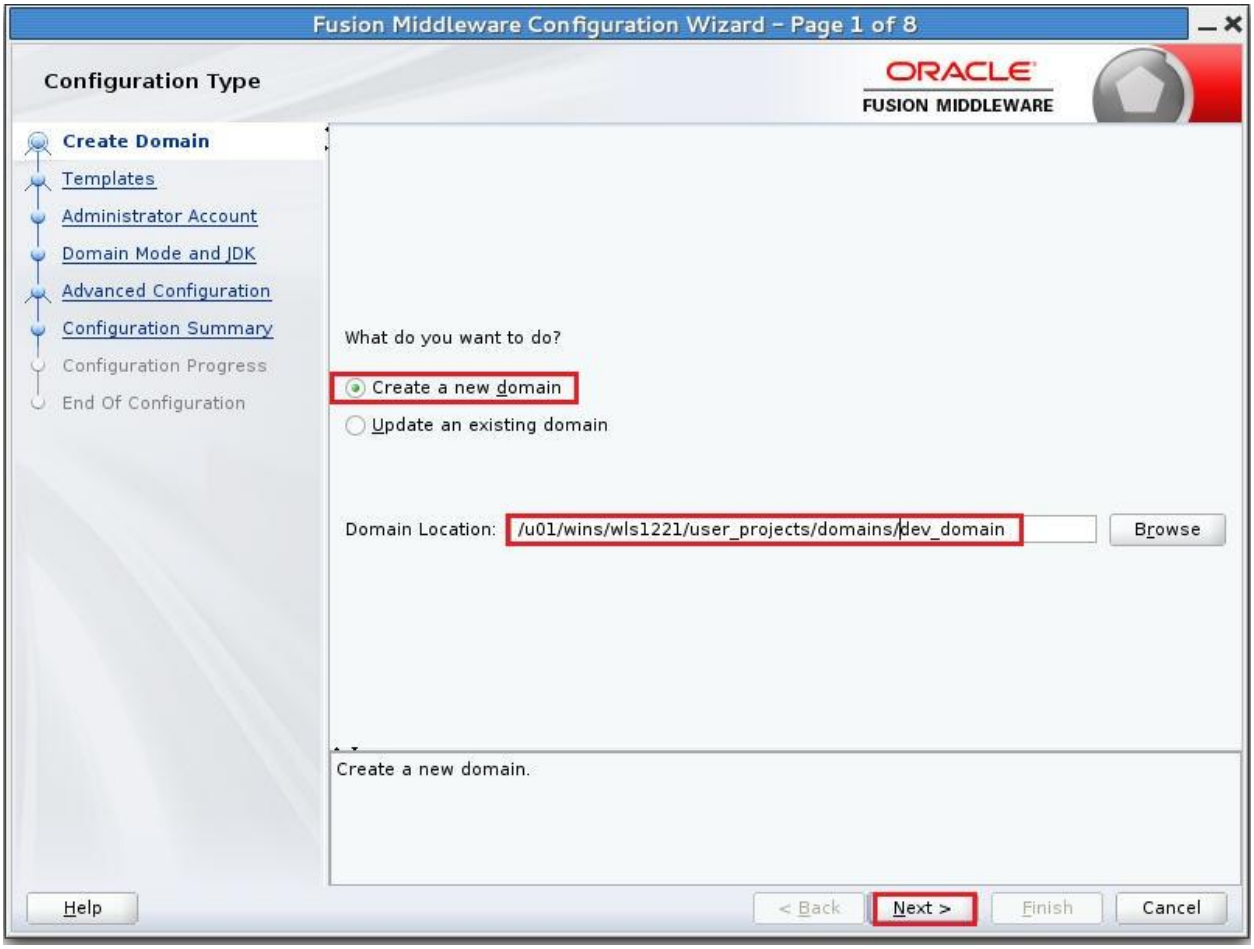
d. Go to Firefox and type the URL: http://localhost:7101/dp1/medrec/
e. Confirm that page return "**Error 404—Not Found**".



Create a new dev single server (Admin Server) domain

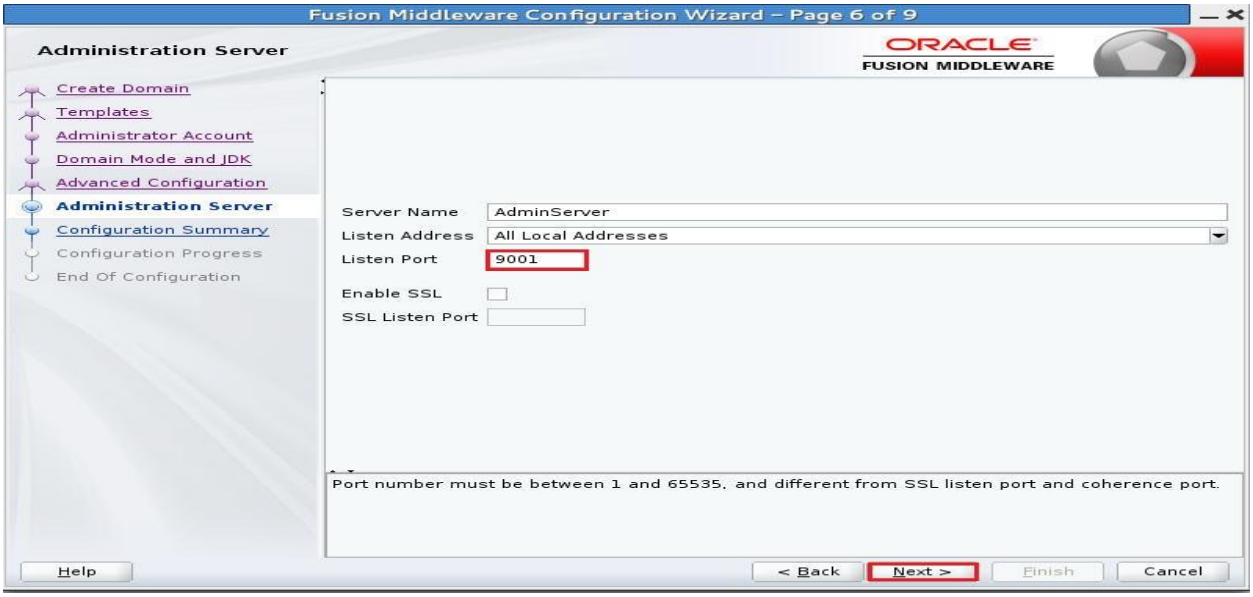
We are going to create a domain which contains only Admin Server, as the previous domain base_domain is created with RESTRICTED-JRF template. But this domain will be created by default template that is with basic template.

a. Open a new tab.
b. cd /u01/wins/wls1221/oracle_common/common/bin/
c. ./config.sh
d. Select "**Create a new domain**" and Enter **"/u01/wins/wls1221/user_projects/domains/dev_domain**" as Domain Location then click on **Next**.



e. Leave **Default** on Templates then click on **Next**.
f. Enter **weblogic/welcome1** as Name/Password in Administrator Account then click on **Next**.
g. Leave default on Domain Mode and JDK then click on **Next**.
h. Check the box near to Administration Server then click on **Next**.

i. In Administration Server, Enter 9001 as Listen Port then click on **Next**.

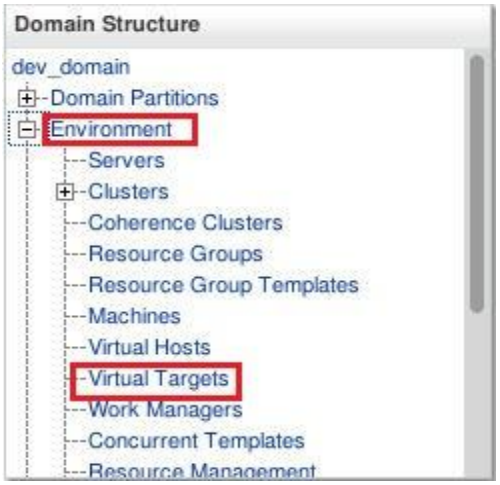


j. Click on **Create** then click on Next. Click on **Finish**.

Configure domain for Medrec Application.

In Lab 2, you configure the domain using Fusion Middleware console, but here we are going to configure domain using Weblogic Admin Console. The operation related to Multitenancy, we can also perform using WebLogic Admin Console.

a. cd /u01/wins/wls1221/user_projects/domains/dev_domain/
b. ./startWebLogic.sh
c. In Tab, Click on Terminal -> Set Title, Enter **dev_admin** as Title then Click on OK.
d. Go to Firefox and type the WebLogic Admin Console URL: http://localhost:9001/console.
e. Enter **weblogic/welcome1** as **Username/Password** then click on Login.
f. Creation of Virtual Target.
   i. Click on **Environment -> Virtual Targets**.



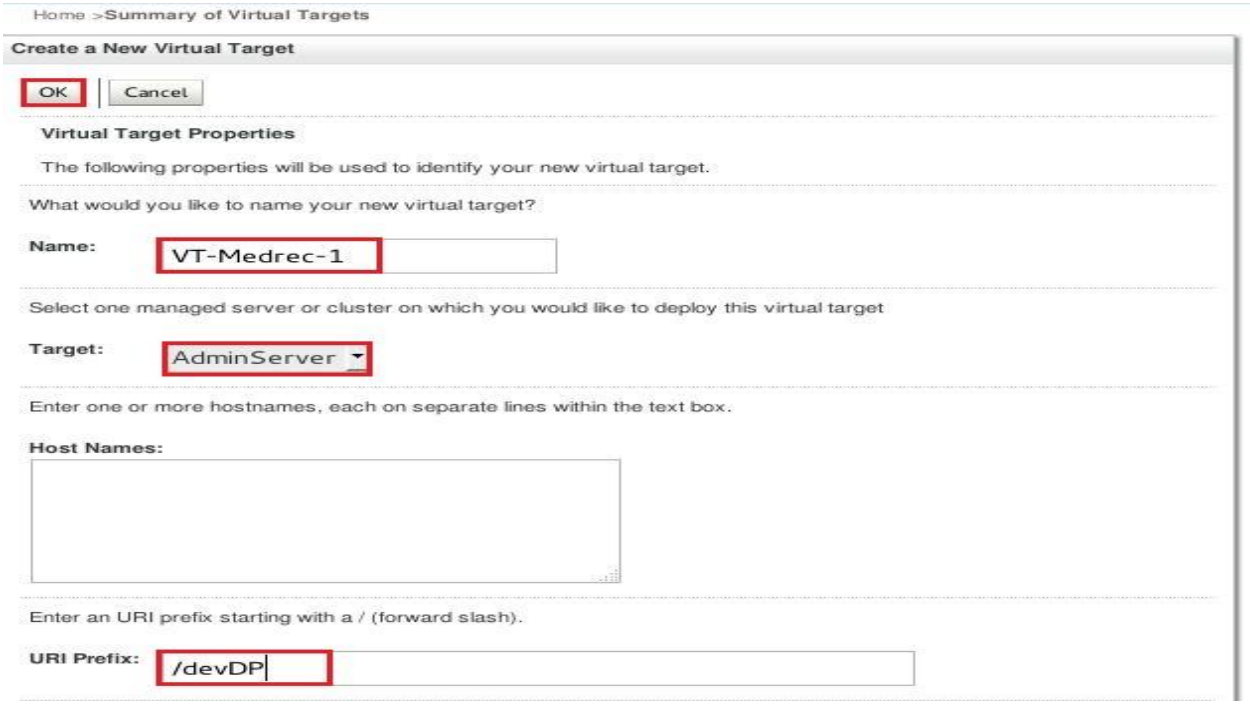
ii. Click on New.

iii. Enter the following then click on **OK**.

Name: VT-Medrec-1 (Both Virtual Target Name will be same)

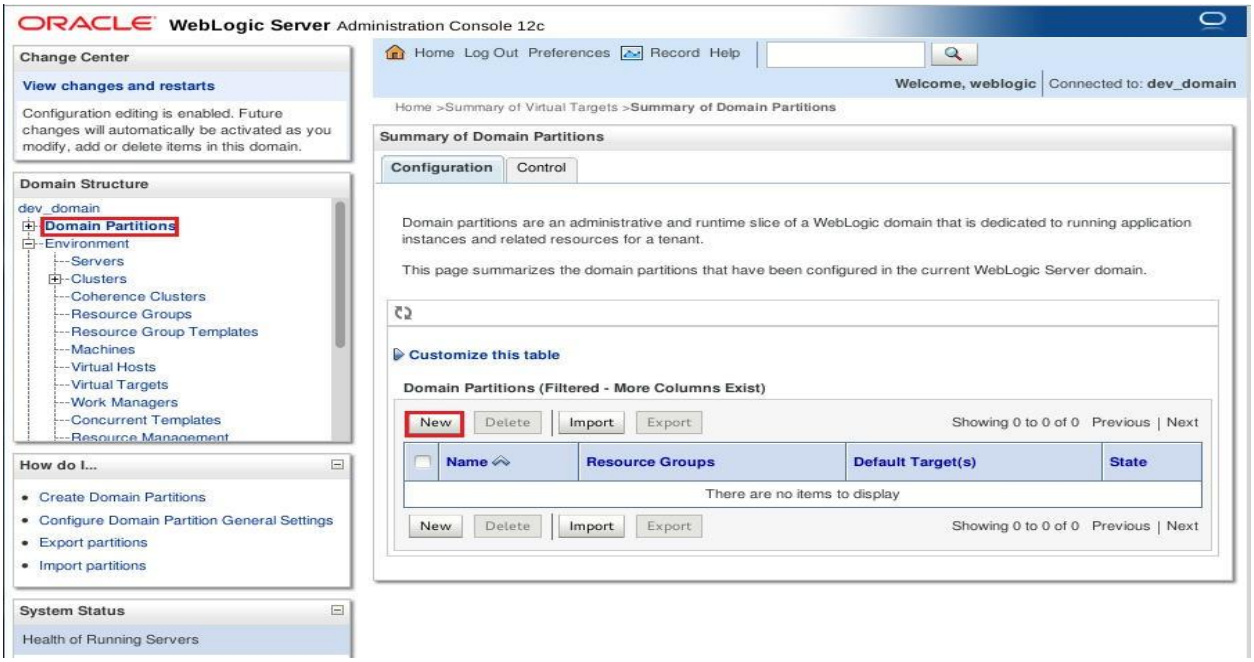
Target: Admin Server

URI Prefix: /devDP

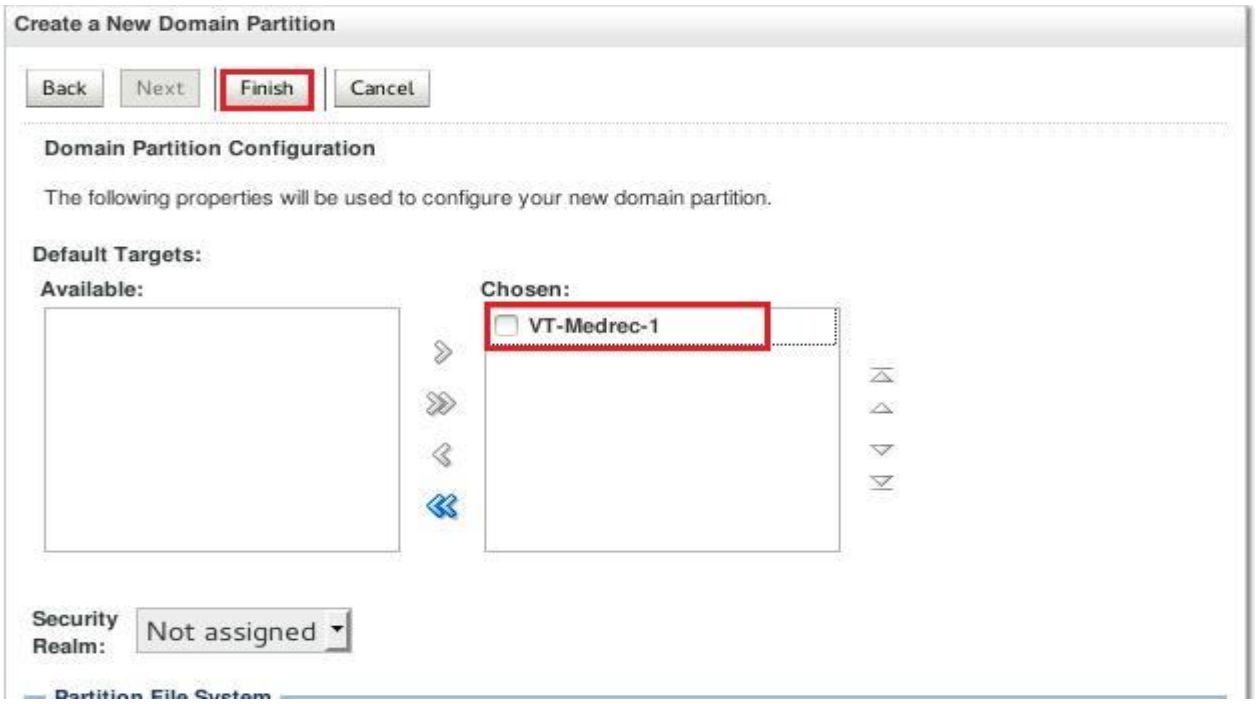


g. Creation of Domain Partition.

i. Click on **Domain Partition -> New**.

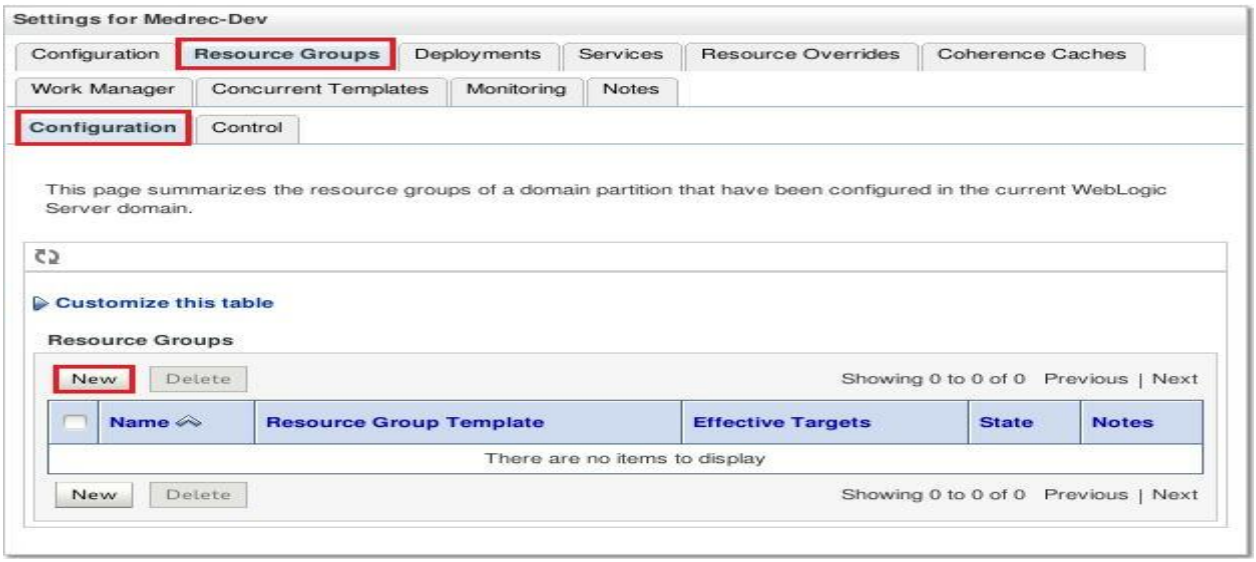


ii. Enter **Medrec-Dev** as Name and **uncheck** the box for **Default Resource Group**. Click on **Next**.
iii. Check the box for **VT-Medrec-1** then click on **Next**.
iv. In Default Targets, Move **VT-Medrec-1** to Chosen. Click on **Finish**.



v. In **Domain Partition -> Control tab**, check the box near Medrec-Dev and click on Start.

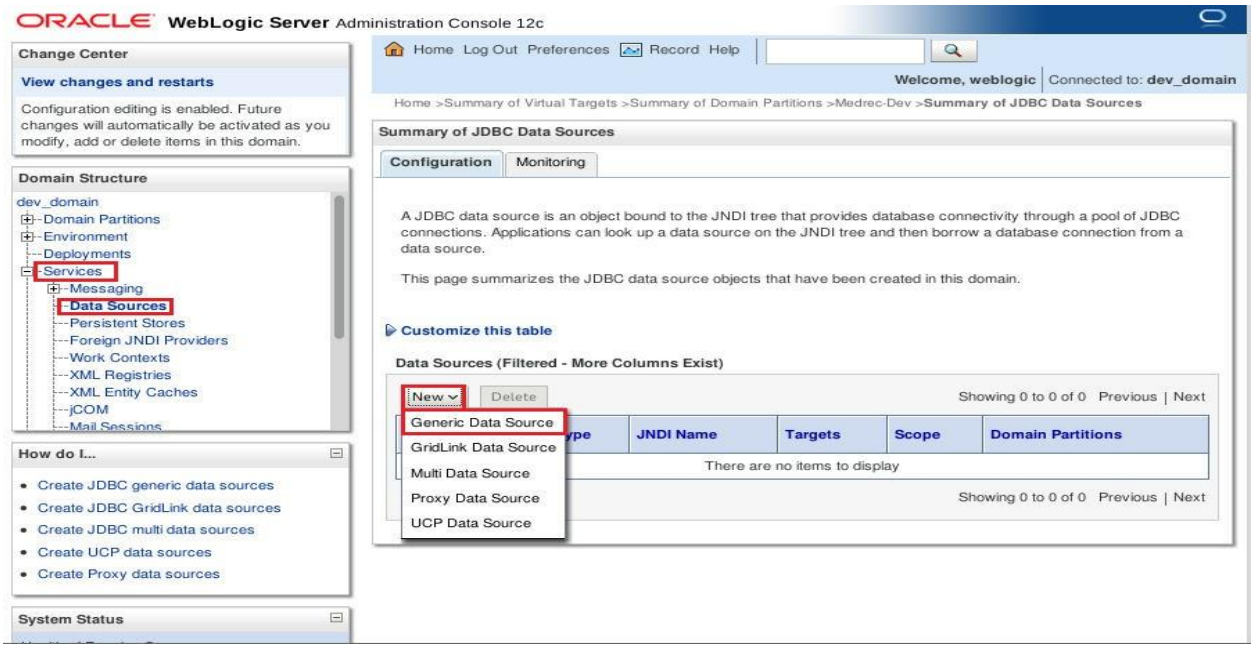
h. Creation of Resource Group
    i. Click on domain partition **Medrec-Dev**.
    ii. Click on **Resource Groups -> Configuration tab** then click on New.



    iii. Enter **appRG** as Name, Leave others as default then click on **OK**.

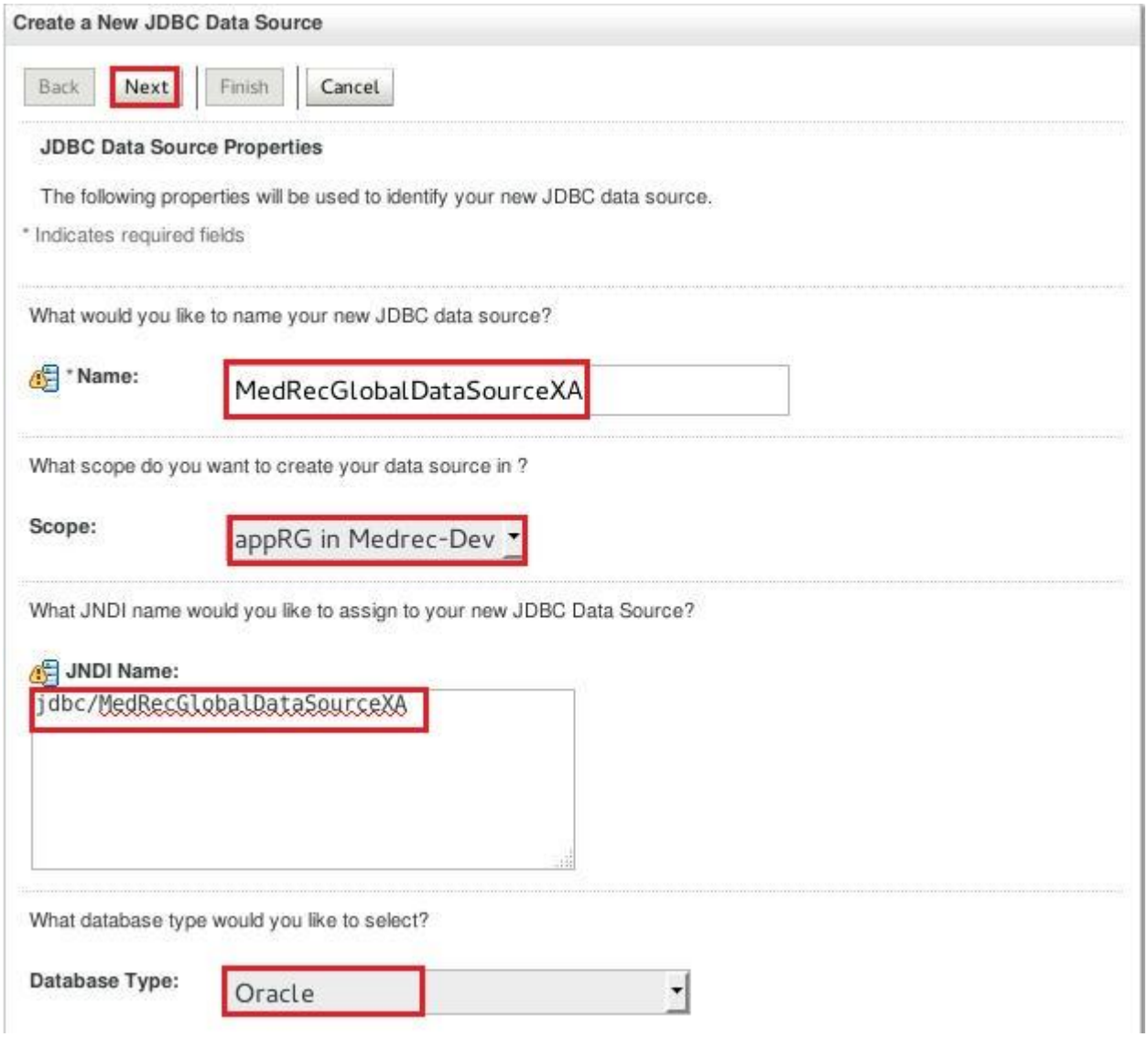
Note: We have created script through which you can create the remaining resources, you need to run the file MedrecInMedrec-Dev.sh inside /u01/content/weblogic-innovation-seminars/WInS_Demos/MT-Workshop/Lab4, and then you can directly go to **Exporting the domain partition** section.

i. Creation of Data Source.
    i. Click on **dev_domain -> Services -> Data Sources** then click on **New-> Generic Data Sources**.



ii. Enter the following and click on **Next**.

Name:	MedRecGlobalDataSourceXA
JNDI Name:	jdbc/MedRecGlobalDataSourceXA
Scope:	appRG in Medrec-Dev
Database Type:	Oracle

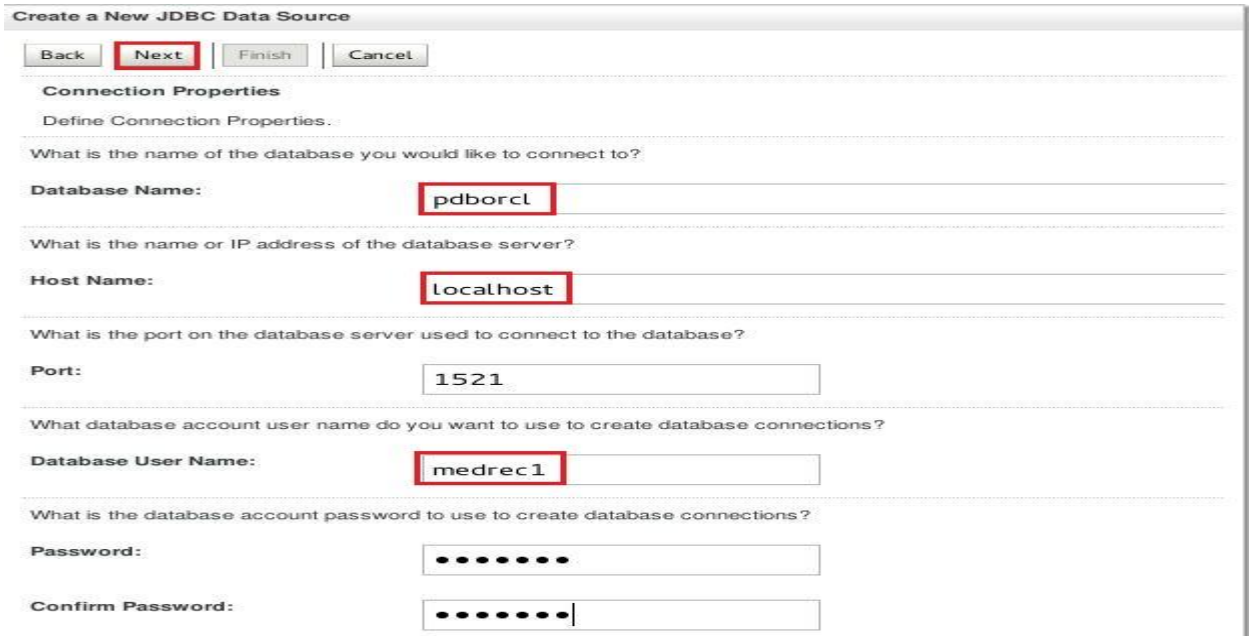


iii. Select "**Oracle's Driver (Thin XA) for Service connections; Version: Any** "then click on **Next**.

iv. In Transaction Options, click on **Next**.

v. Enter the following and click on **Next**.

Database Name:	pdborcl
Host Name:	localhost
Port:	1521
Database User Name:	medrec1
Password:	medrec1



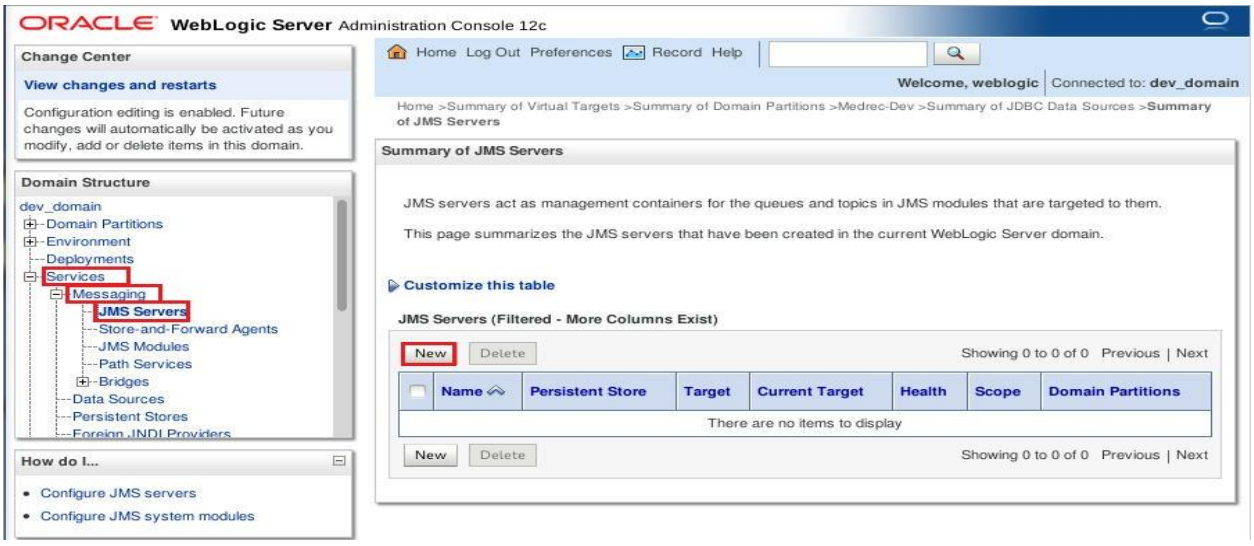
vi. Click on **Test Configuration** and verify the connectivity.

vii. Click on **Next** then Click on **Finish**.

j. Creation of JMS Server.

i. Click on **Services -> Messaging -> JMS Servers**.

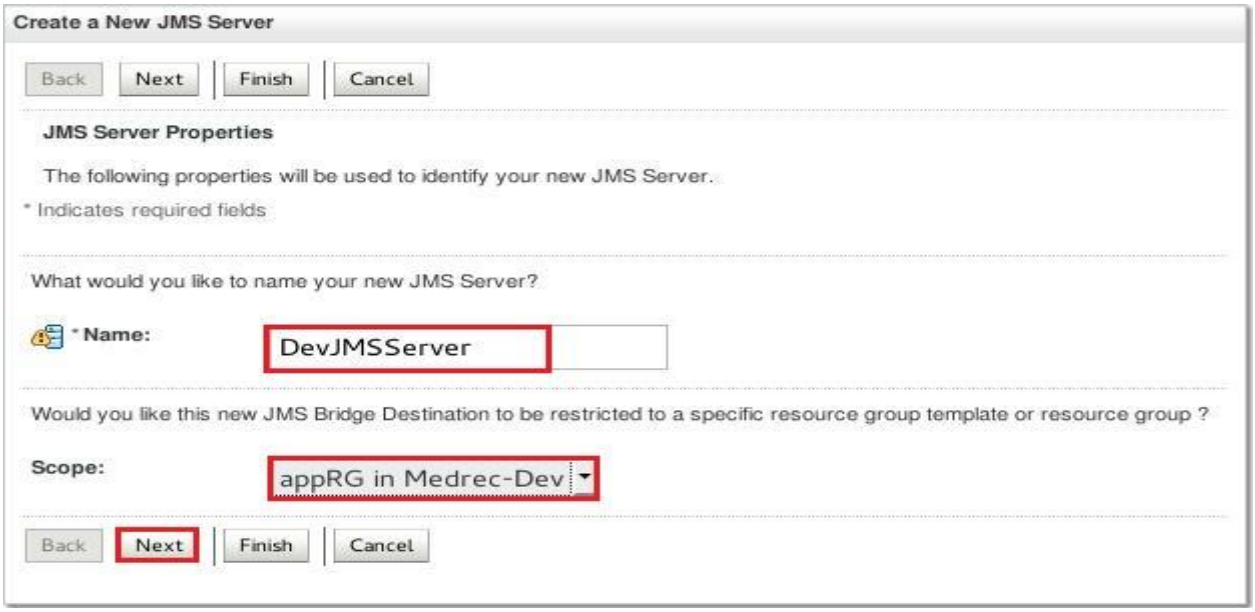
ii. Click on New.



iii. Enter the following and click on **Next**.

Name: DevJMSServer

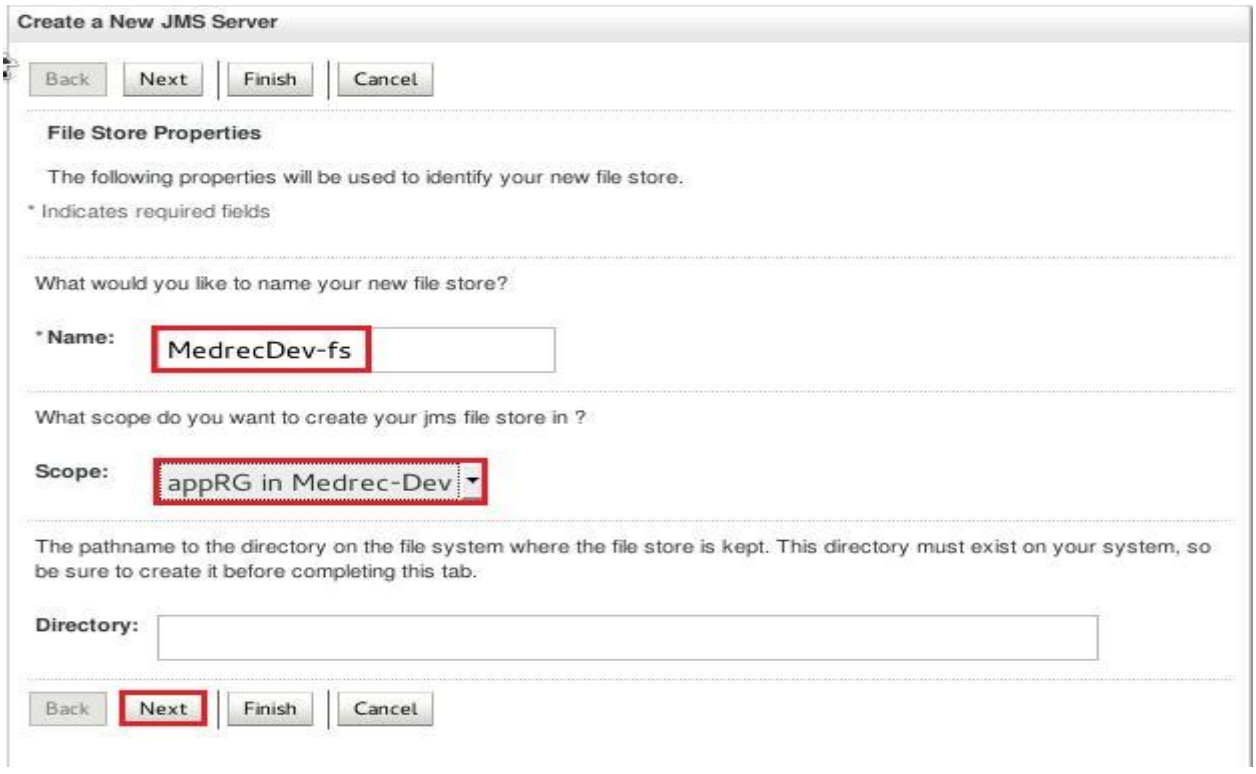
Scope: appRG in Medrec-Dev



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

v. Select "**File Store**" as Type then click on **Next**.

vi. Enter "**MedrecDev-fs**" as Name and "**appRG in Medrec-Dev**" as Scope then click on **Next**. Click on **Finish**.

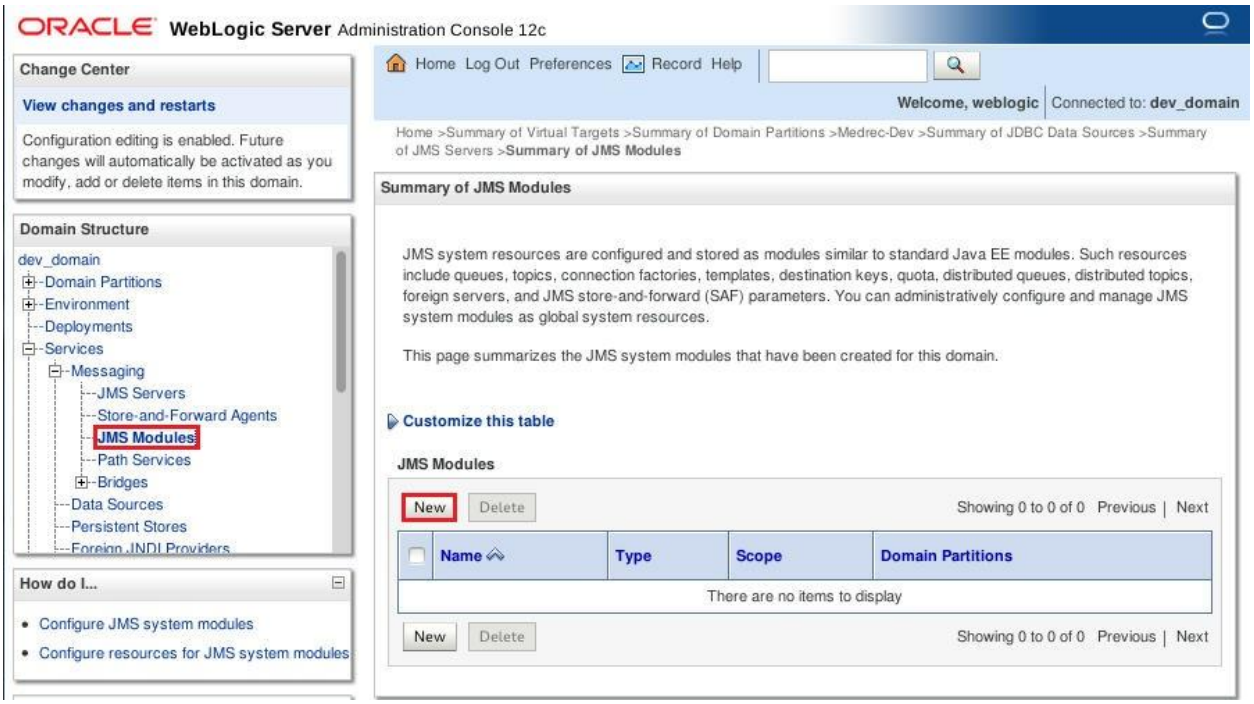


vii. Select "**MedrecDev-fs**" as Persistent Store then click on **Next**.

viii. Click on **Finish**.

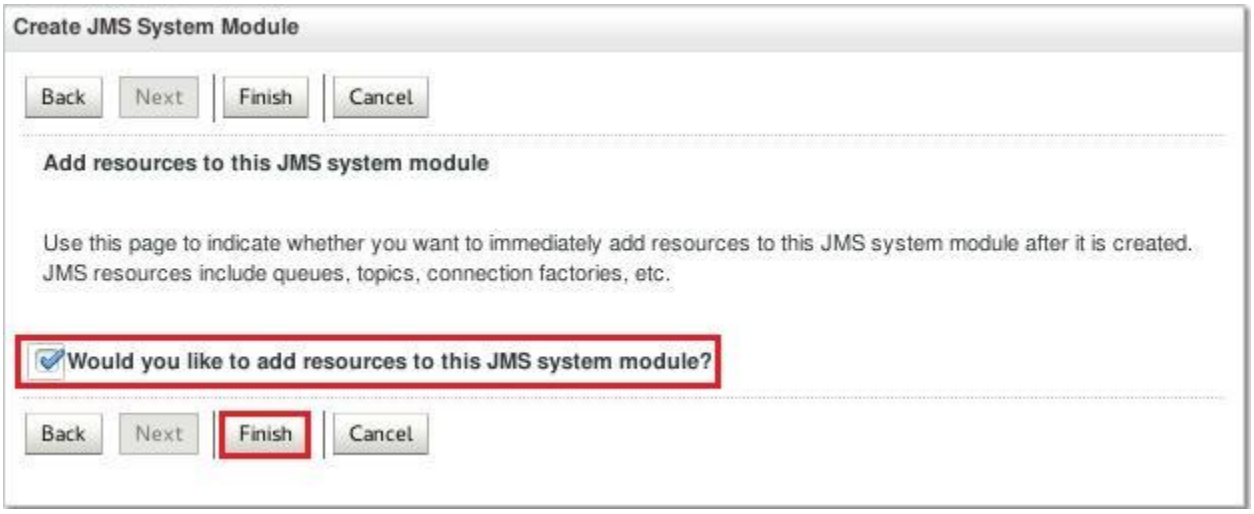
k. Creation of JMS Module.

i. Click on **Services -> Messaging -> JMS Modules** then click on New.



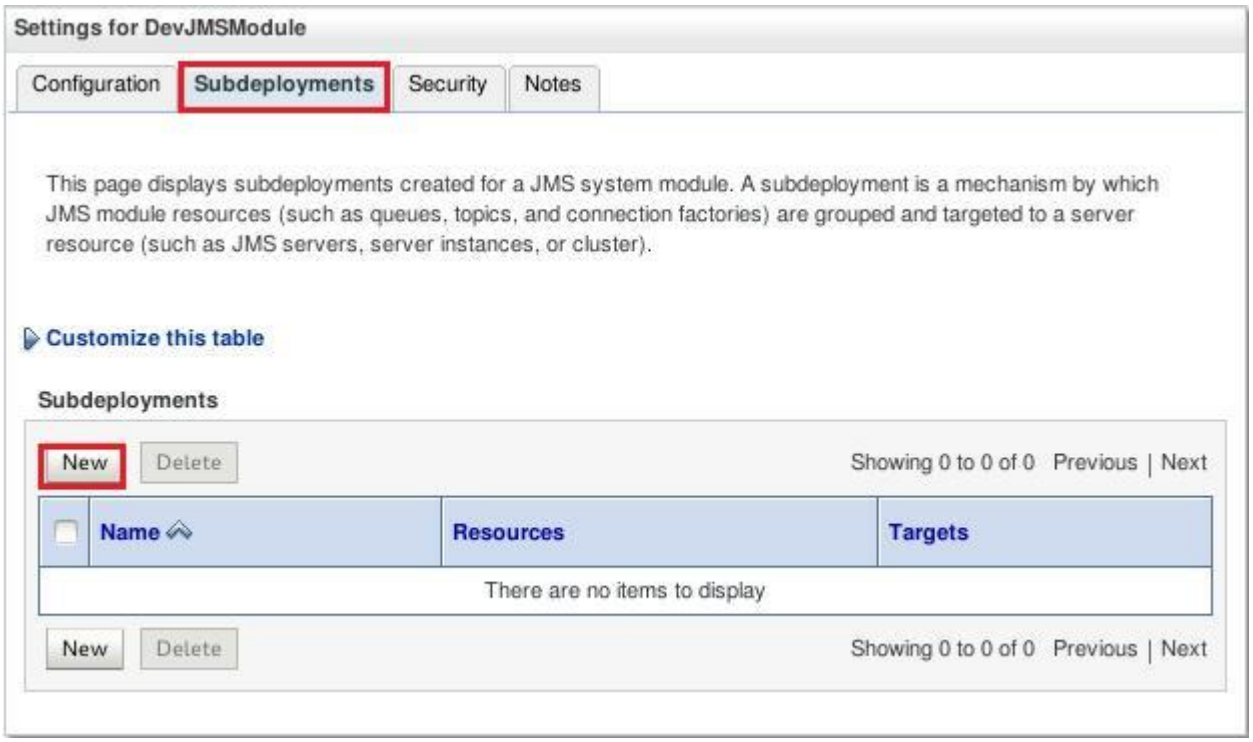
ii. Enter **DevJMSModule** as Name and "**appRG in Medrec-Dev**" then click on **Next**.

iii. Check the box for "**Would you like to add resources to this JMS system module**" then click on **Finish**.



l. Creation of Subdeployment.

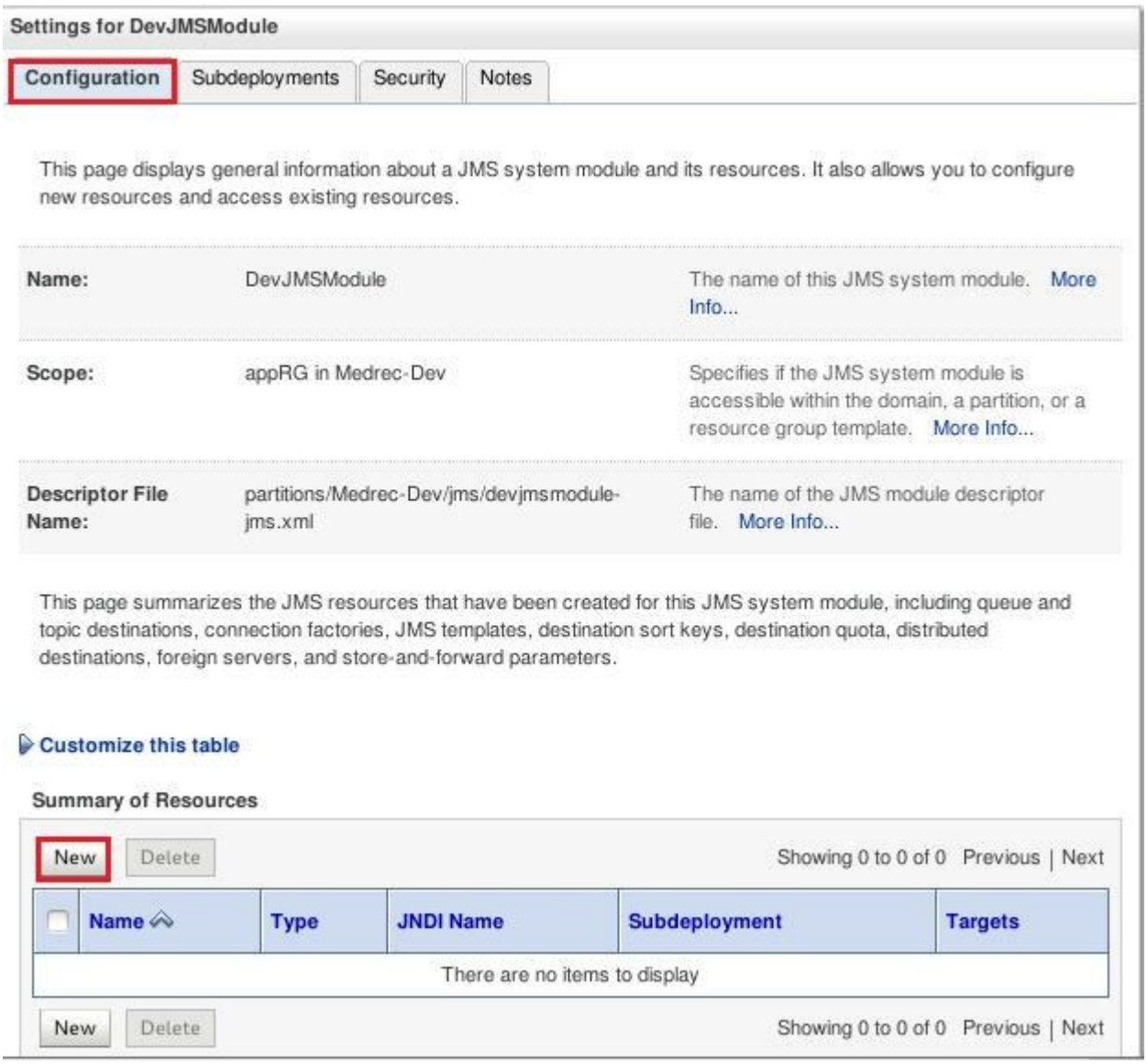
i. Select **Subdeployments** tab then click on New.



ii. Enter **DevMedrecJMS** as Name then click on **Next**.

iii. Select **DevJMSServer** as Target then click on **Finish**.

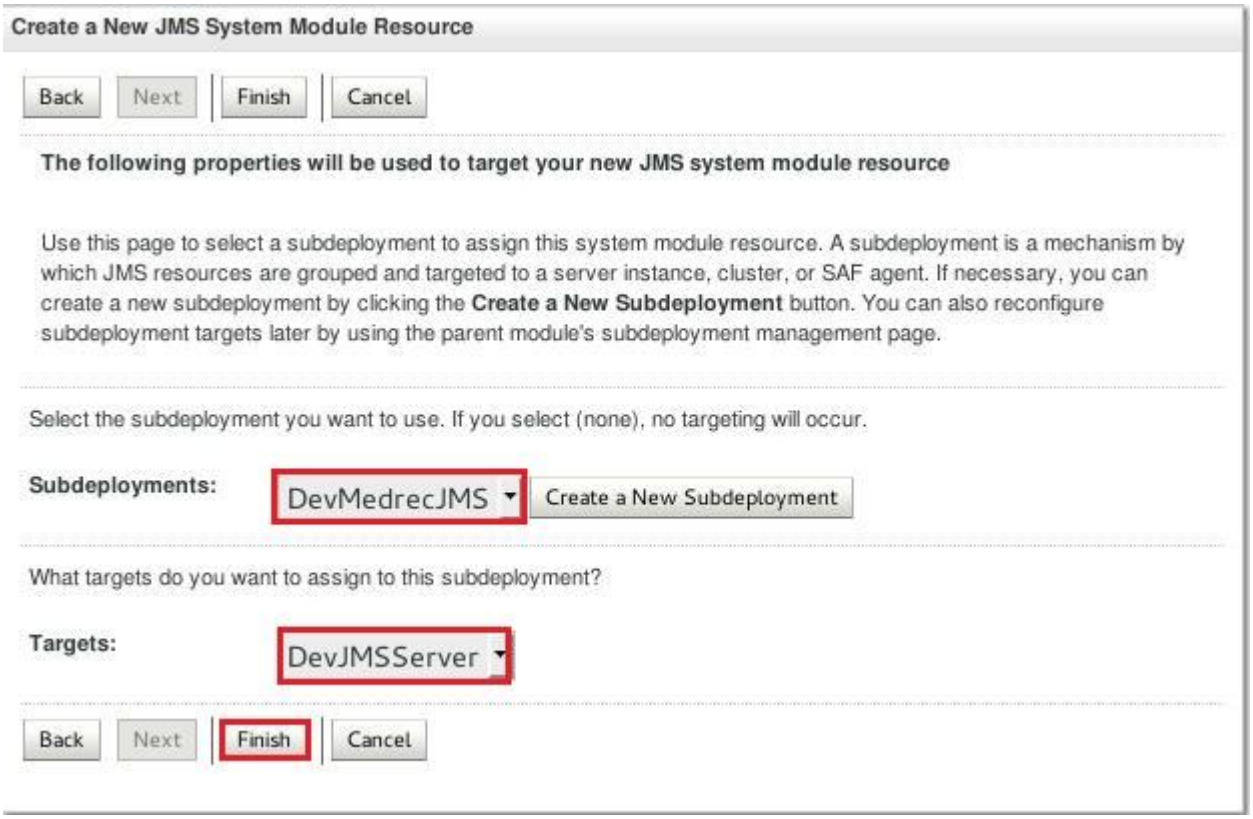
m. Creation of Connection Factory.
   i. Click on Configuration tab then click on New.



   ii. Select the box for Connection Factory then click on **Next**.
   iii. Enter **DevMedRecConnectionFactory** as Name and **com.oracle.medrec.jms.connectionFactory** and leave other default then click on **Next**.
   iv. Click on **Finish**.

n. Creation of Distributed Queue.
   i. Click on New.
   ii. Select the box for Distributed Queue then click on **Next**.
   iii. Enter **PatientNotificationQueue** as Name and **com.oracle.medrec.jms.PatientNotificationQueue** as JNDI Name and leave other default then click on **Next**.

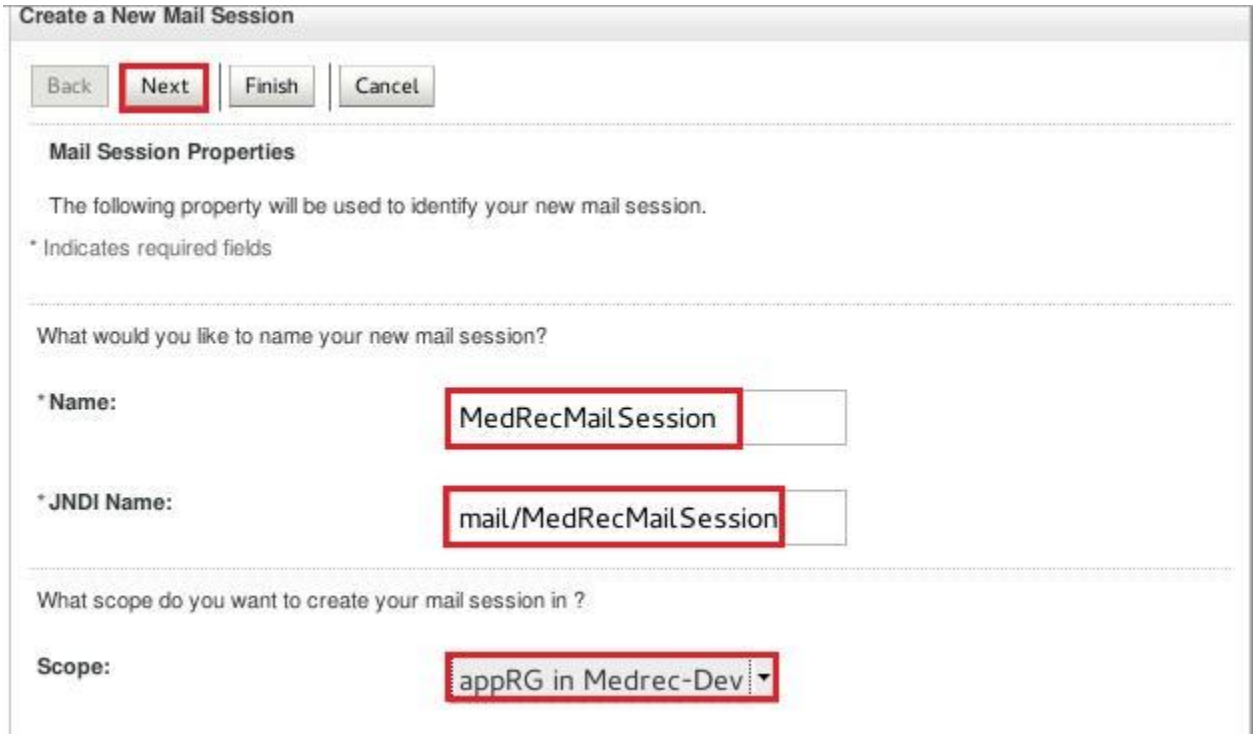
iv. Click on **Advanced Targeting**, Select DevMedrecJMS as Subdeployments and DevJMSServer as Targets then click on **Finish**.



o. Creation of Mail Session.
   i. Click on **Services -> Mail Sessions**.

ii. Click on New.
iii. Enter the following then click on **Next**.

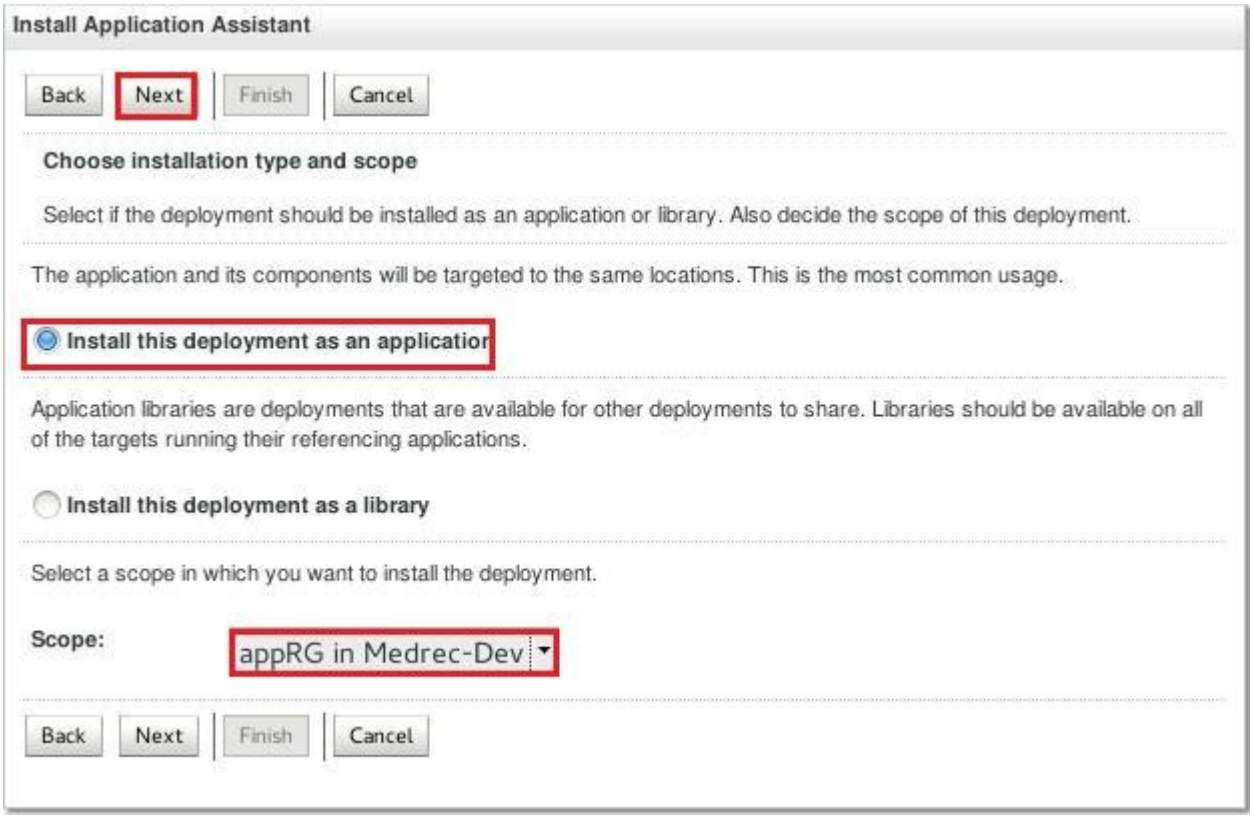
Name:	MedRecMailSession
JNDI Name:	mail/MedRecMailSession
Scope:	appRG in Medrec-Dev



iv. Click on **Finish**.

p. Deployments of Application
   i. Click on **Deployments** then on **Install**.
   ii. Select the medrec.ear application from the Path /u01/content/weblogic-innovation-seminars/WInS_Demos/MT-Workshop/Lab4/ then click on **Next**.

iii. Select "**Install this deployment as an application**" and "**appRG in Medrec-Dev**" as Scope then click on **Next**. Click on **Finish**.



iv. Click on Install then Select the physician.ear application from the Path /u01/content/weblogic-innovation-seminars/WInS_Demos/MT-Workshop/Lab4/ then click on **Next**.

v. Select "**Install this deployment as an application**" and "**appRG in Medrec-Dev**" as Scope then click on **Next**. Click on **Finish**.

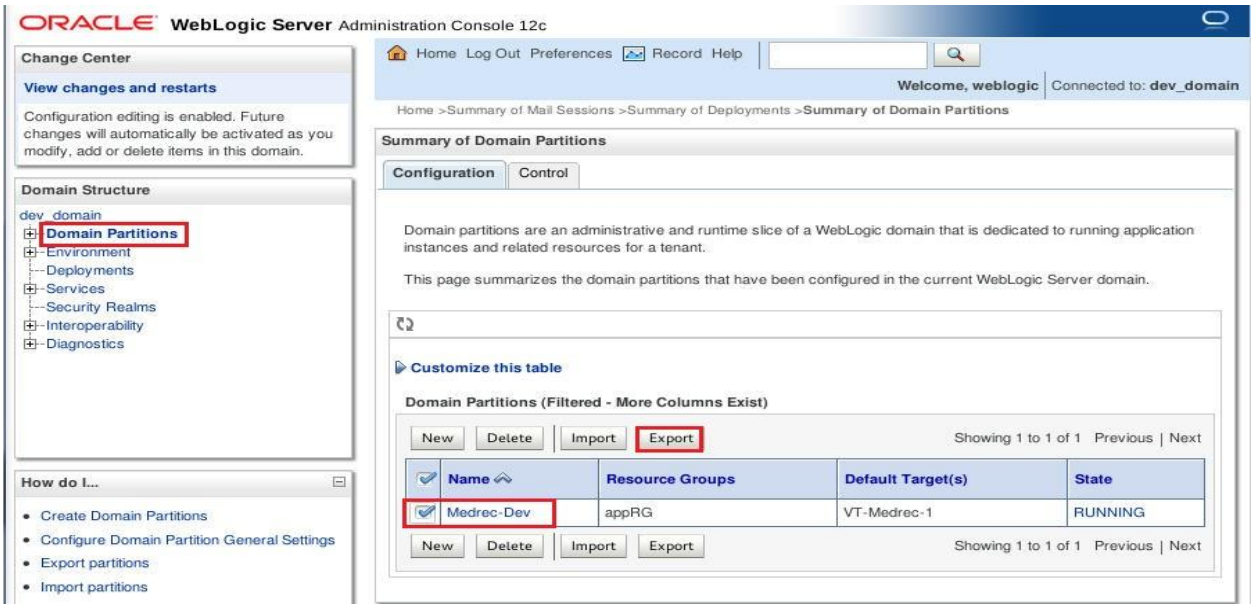
vi. Click on Install then Select the chat.war application from the Path /u01/content/weblogic-innovation-seminars/WInS_Demos/MT-Workshop/Lab4/ then click on **Next**.

vii. Select "**Install this deployment as an application**" and "**appRG in Medrec-Dev**" as Scope then click on **Next**. Click on **Finish**.

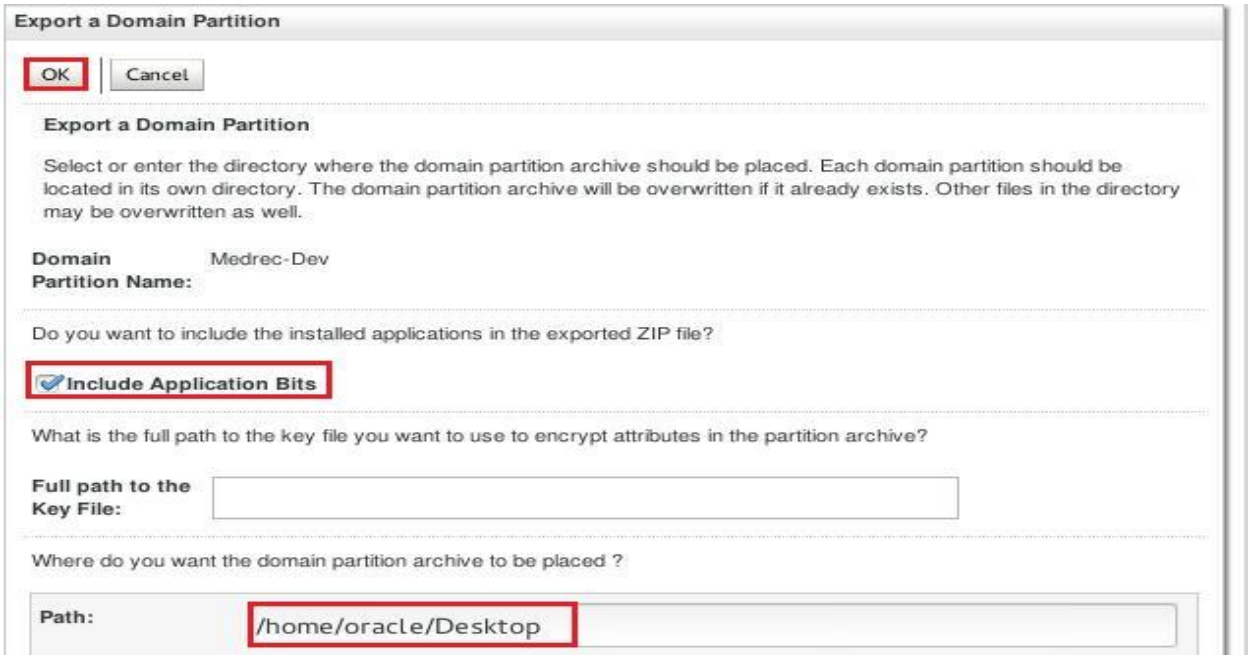
viii. Go to Firefox and type the URL: http://localhost:9001/devDP/medrec/ and confirm the execution of application.

Exporting the domain partition

a. Go back to admin console http://localhost:9001/console/ of dev_domain.
b. Click on **Domain Partition**, and then check the box near to "**Medrec-Dev** then click on Export.



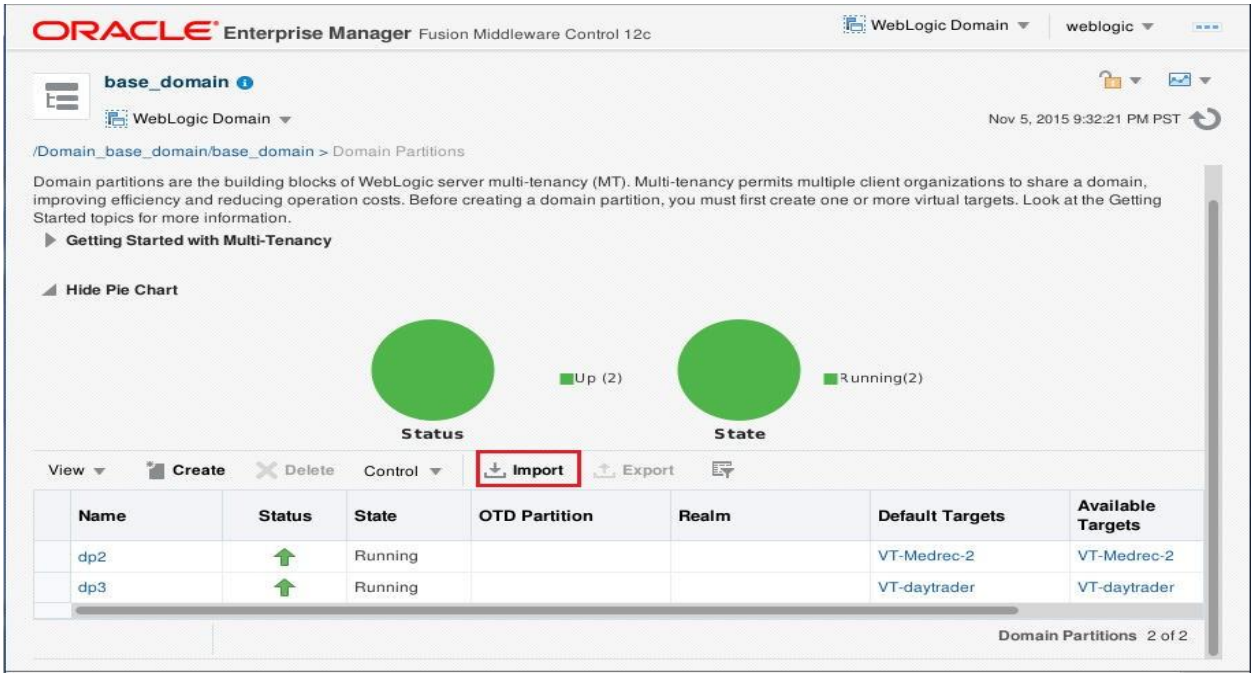
c. Select the box for "**Include Application Bits**" and enter **/home/oracle/Desktop** as Path then click on **OK**.

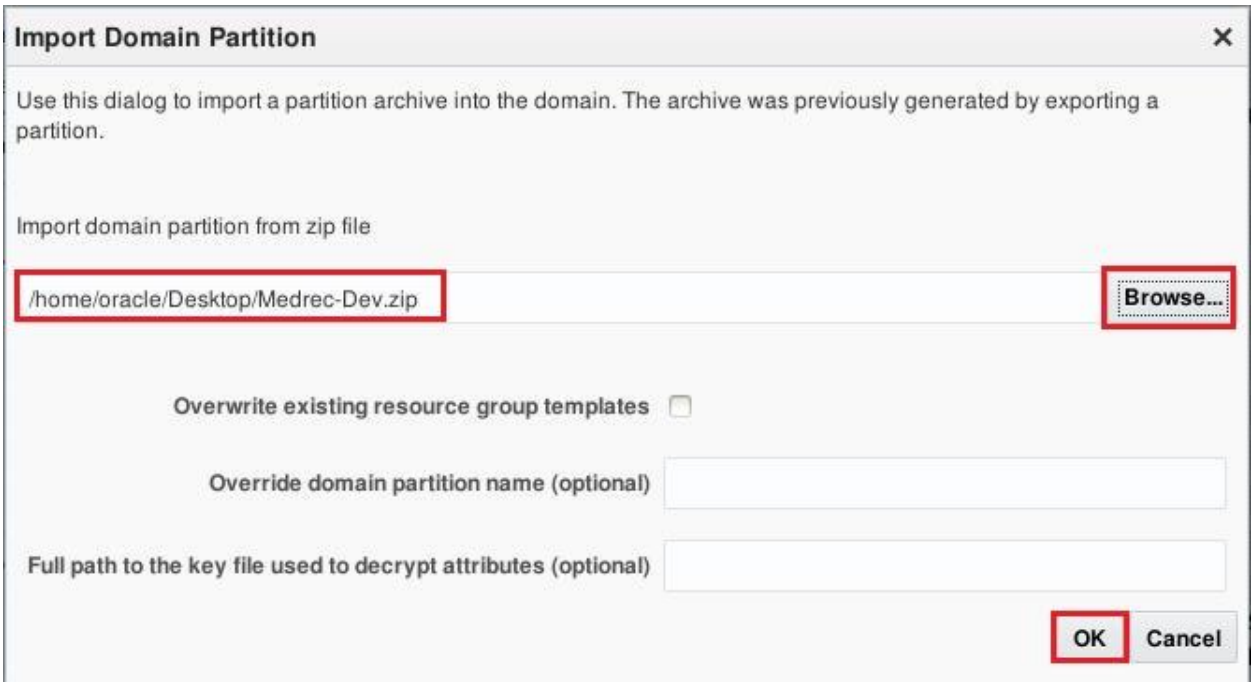


d. Go to Desktop and Verify the Creation of **Medrec-Dev-attributes.json** and **Medrec-Dev.zip** file.

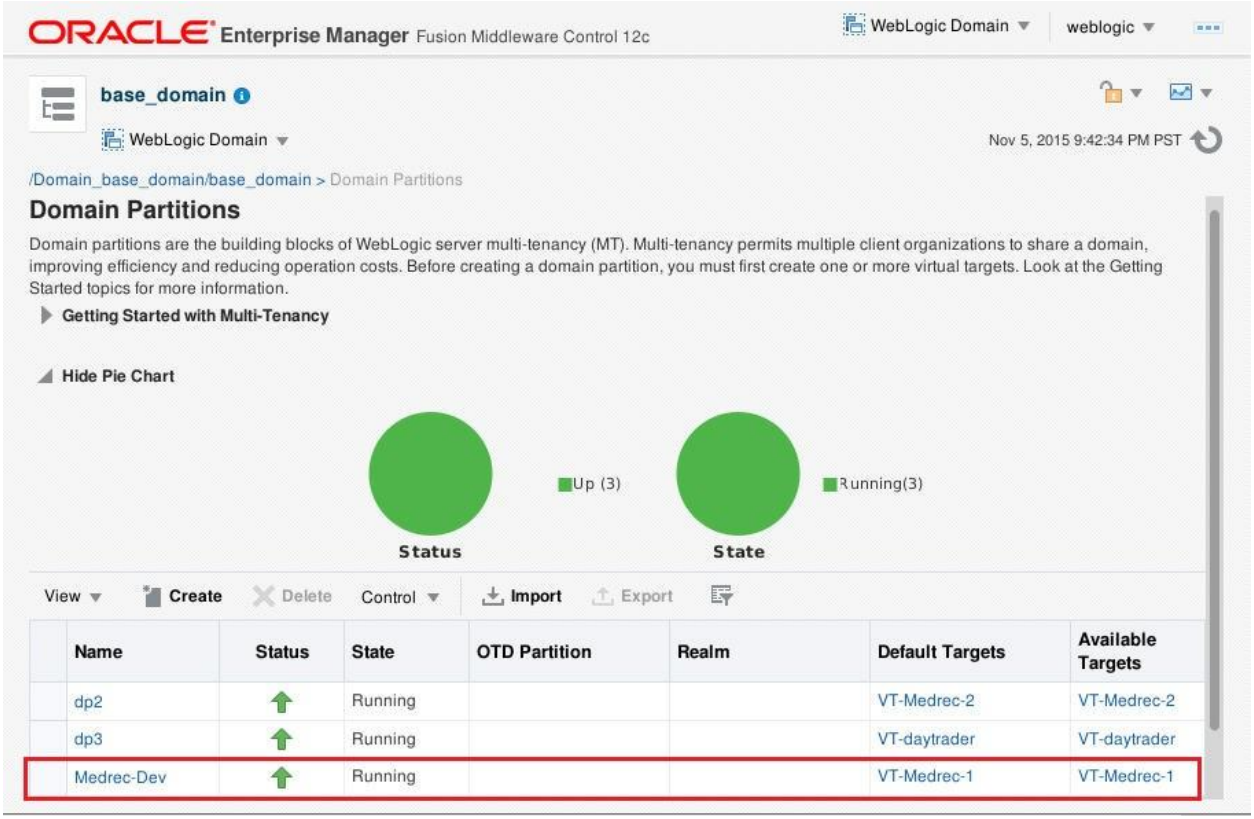
Importing the domain partition

a. Go back to EM console of base_domain. Go to Firefox and type the WebLogic Admin Console URL: http://localhost:7001/em.
b. Enter **weblogic/welcome1** as Username/Password and click on Login.
c. Click on **WebLogic Domain-> Environment ->Domain Partition**.
d. Click on Import. Click on **Browse** button, Select the file **Medrec-Dev.zip** from **/home/oracle/Desktop** directory then click on **OK**.





e. Initially it will have State "**Unknown**". Wait for 1 or 2 minute, click on Refresh icon to get the current state.
f. Once the status for **Medrec-Dev** domain partition is **Shutdown**, check the box, near Medrec-Dev then click on **Control -> Start**. Click on **Close**. Click on the Refresh icon to get the current state.



g. Go to Firefox and type the URL: http://localhost:7101/dp1/medrec/
h. Click on "**Getting Started!"** Under Administrator, click on Login.
i. Enter **administrator/administrator123** as Username/Password then click on Sign in.
j. Click on Logout. Click on Logout again.

Note: As we have VT-Medrec-1 as Virtual target in both the domains base_domain and dev_domain. In base_domain, we have added administrator user to default security realm. So as this domain partition becomes part of this domain. It also uses the default security realm. Here you are accessing the application using /dp1 in the URL because, we have Virtual target VT-Medrec-1 has /dp1 as URI, So Virtual target definition do not change during import.

k. Come back to tab **dev_admin**.
l. Stop the Weblogic Server running in dev_domain, by pressing Ctrl +C in tab in which Admin Server is running. Close the Tab.

LAB 5: RESOURCE CONSUMPTION MANAGEMENT

Overview

When applications that are deployed to multiple collocated Domain Partitions, access shared resources (low level resources such as CPU, network, storage) two key problems are likely to be faced:

- Contention and unfairness during allocation: Multiple request for a Shared resources results in contention and interference. Abnormal resource consumption requests may happen due to benign reasons (high traffic-genuine or DDoS), misbehaving, buggy applications or malicious code. These requests could overload the capacity of shared resources, thereby preventing another consumer's access to the resource.
- Variable performance leading to potential Service Level Agreement (SLA) violations: From a cloud operations perspective, performance for different collocated consumers.

It is therefore critical to manage and isolate access to shared resources in the WebLogic application Server by domain partition to ensure fairness in allocation, prevent contention/interferences of access to shared resources and to provide consistent performance for multiple co resident tenants. The Resource Consumption Management (RCM) feature in WebLogic 12.2.1 Multitenancy allows WebLogic System administrator to specify resource consumption management policies (allows the specification of constraints, recourse actions and notification) on shared resources such as CPU, Heap, File and Network. You can create resource manager based on the following parameter:

- Retained Heap
- CPU Time
- Open File Descriptor

You need to provide the value for below action, as it reaches to that value. It will trigger the following action:

Notify: Inform administrator that a threshold has been crossed.
Slow: Reduce Partition ability to consume more resources.
Stop: Initiate the shutdown sequence for the offending partition.

In this lab first we create a resource manager based on the Heap size then we specify the values with respective actions. So as following values reaches that action get triggered. We will use sample application through which we can modify the value of Heap size and we will see the action associated.

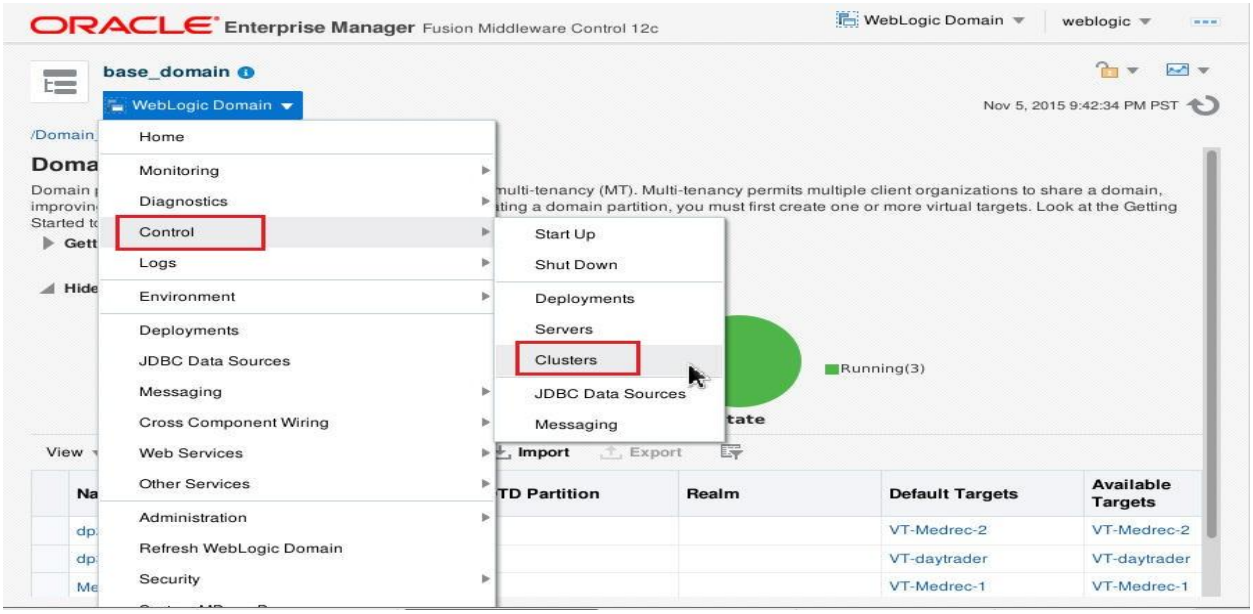
In this lab we are going to perform the following operations:

- Enabling RCM by adding extra arguments in Server Java Arguments.
- Creating Resource manager on the basis of Heap Size.
- Assign Resource manager to a Domain Partition.
- Running an example to understand the functioning of RCM.

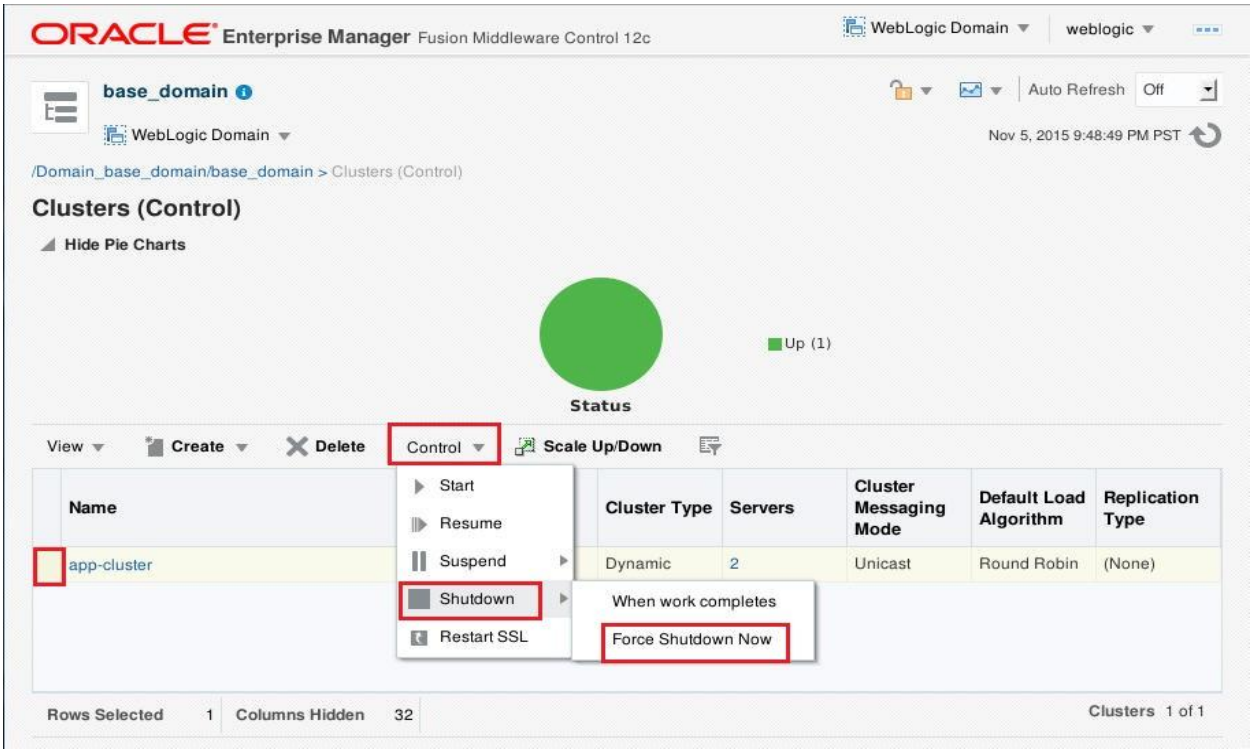
Enabling RCM by adding extra arguments in Server JAVA_OPTION Arguments

a. Go to Firefox and type the Fusion Middleware Control Console URL: http://localhost:7001/em

b. Click on **WebLogic Domain -> Control -> Clusters**.



c. Check the boxes near app-cluster then click on **Control -> Shutdown ->Force Shutdown Now**. Click on **Forcibly Shutdown Servers**.

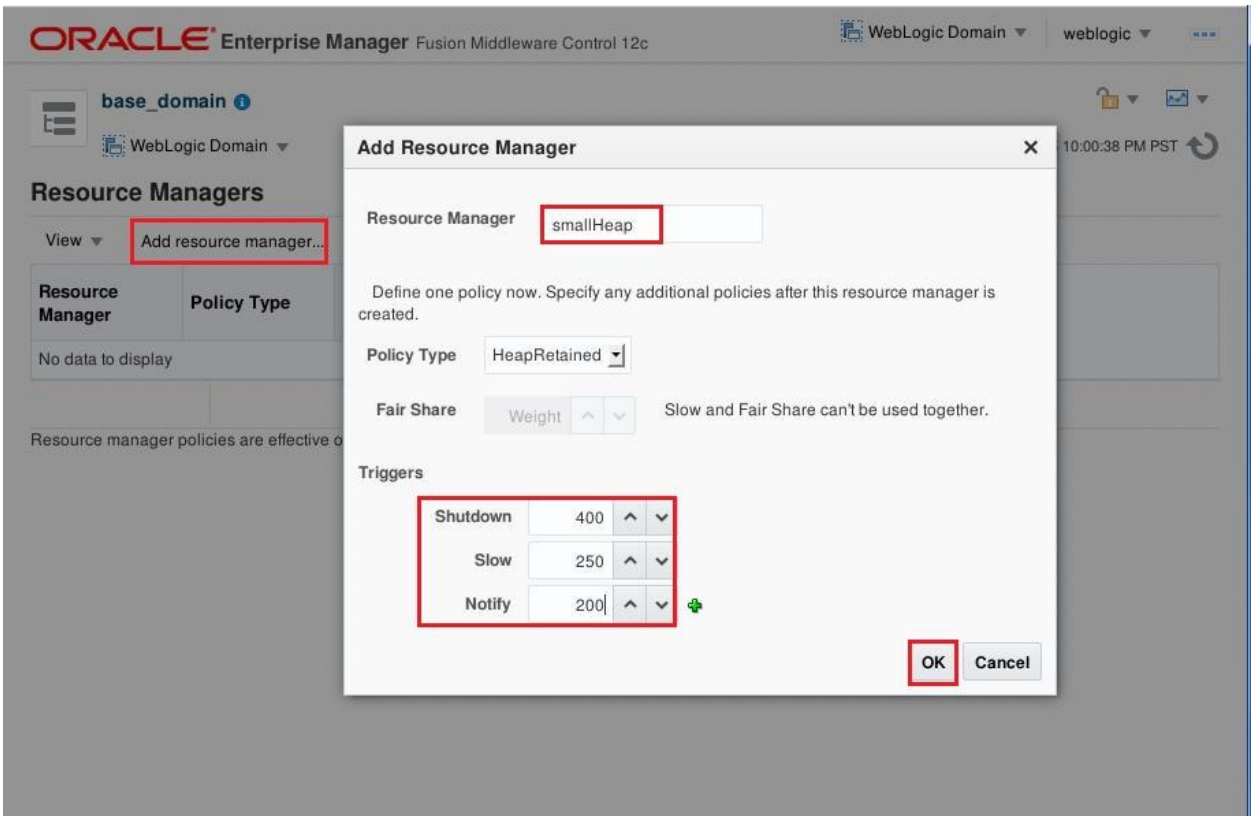


d. Open a new tab.
e. cd /u01/content/weblogic-innovation-seminars/WInS_Demos/MT-Workshop/Lab5
f. cp setDomainEnv.sh /u01/wins/wls1221/user_projects/domains/base_domain/bin/
g. In above command we have modified the JAVA_OPTIONS as specified below. JAVA_OPTIONS="**-XX:+UnlockCommercialFeatures -XX:+ResourceManagement -XX:+UseG1GC** ${JAVA_OPTIONS} ${JAVA_PROPERTIES}"
h. Go back to Fusion middleware control then click on **WebLogic Domain -> Control -> Clusters**.
i. Check the box near to **app-cluster** to make it highlighted and then click on **Control -> Start -> Start Servers**.
j. tail -f /u01/wins/wls1221/user_projects/domains/base_domain/servers/app-cluster-1/logs/app-cluster-1.log
k. In this tab, Click on Enter Terminal -> Set Title and app-cluster-1 then click on **OK**. We will use these logs to monitor resource consumption manager lab.

Creating a Resource Manager and Configuring Resource Manager for a domain partition

a. Go to FMW control http://localhost:7001/em
b. Enter weblogic/welcome1 as Username/Password then click on Login.
c. Click on **WebLogic Domain->Environment -> Resource Consumption Managers**.
d. Click on **Add Resource Manager** and Enter the following value then click on **OK**

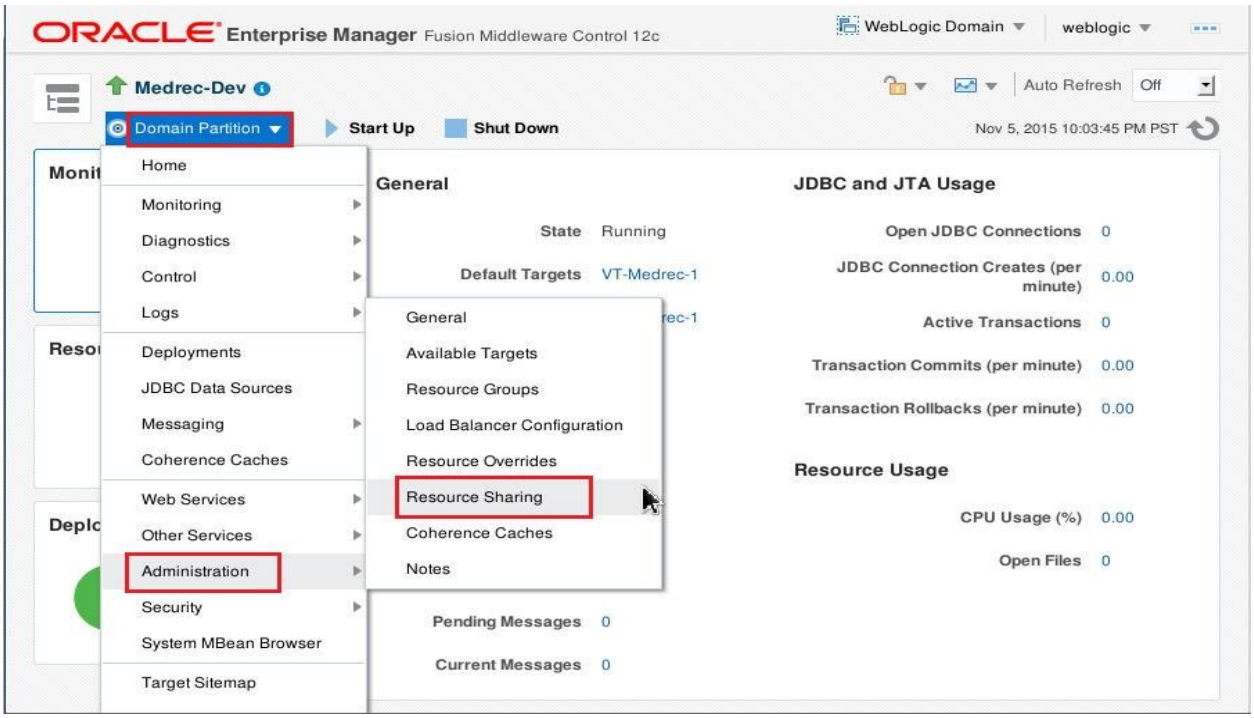
Resource Manager:	smallHeap
Policy Type:	HeapRetained
Shutdown:	400
Slow:	250
Notify:	200



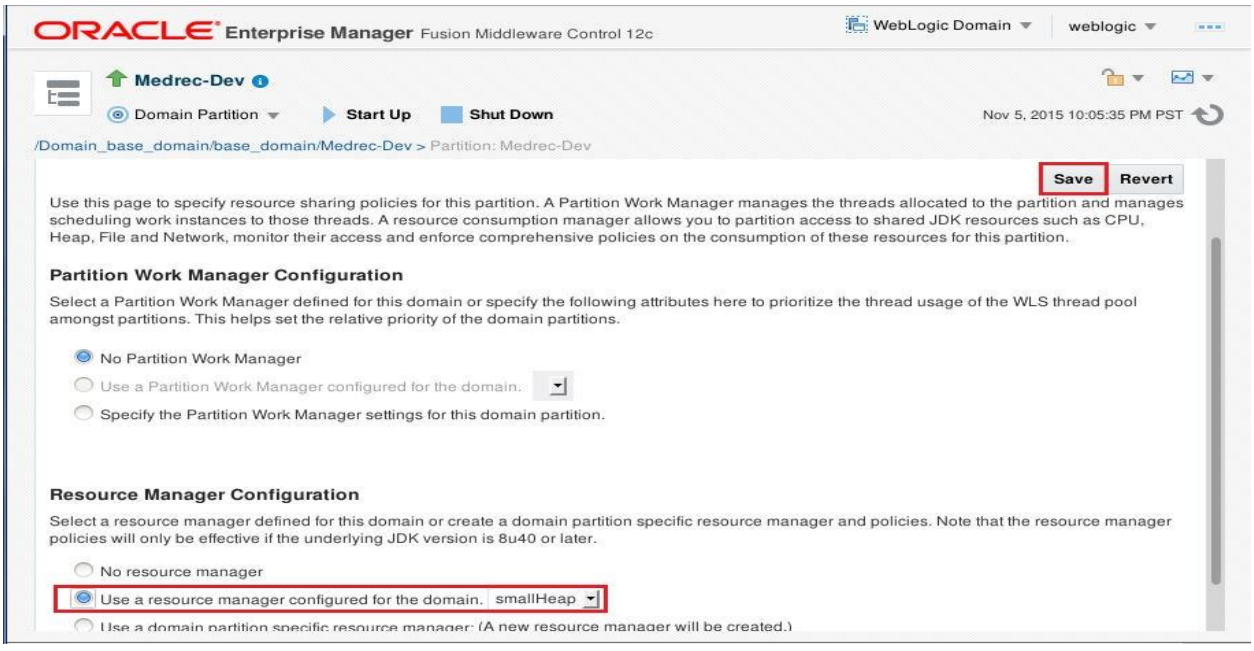
Associate the Resource Manager with Medrec-Dev domain partition.

a. Click on **WebLogic Domain -> Environment->Domain Partition** then click on **Medrec-Dev**.

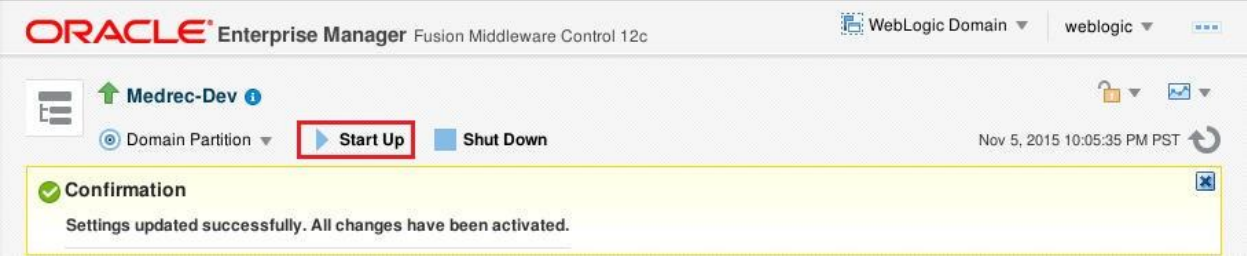
b. Click on **Domain Partition ->Administration -> Resource Sharing**.



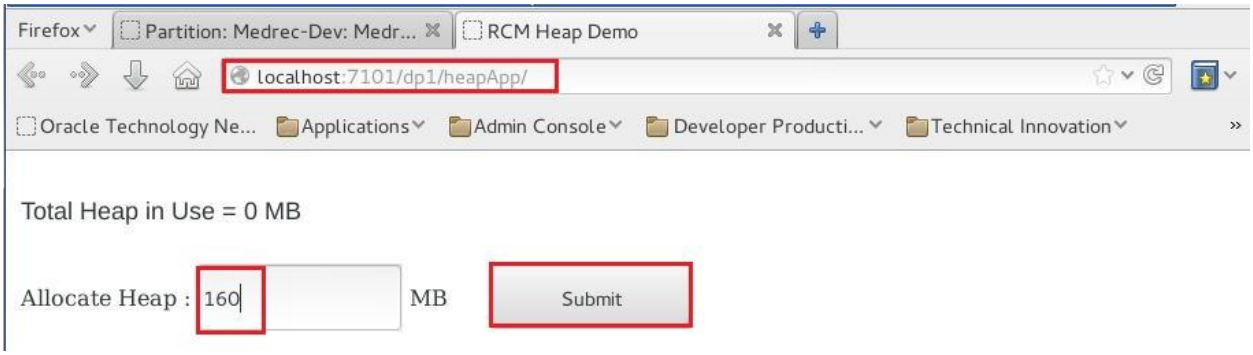
c. Under **Resource Manager Configuration**, and Select "**Use a Resource Manager configured for the domain**" and choose "**smallHeap**" then click on **Save**.



d. Click on **Start Up** near Domain partition. Click on Close.



e. Open a new tab.
f. cd /u01/content/weblogic-innovation-seminars/WInS_Demos/MT-Workshop/Lab5
g. ./DeployHeap.sh
h. Close the tab.
i. Go back to Firefox and type the URL: http://localhost:7101/dp1/heapApp/
j. Enter 160 in **Allocate Heap** then click on Submit then observe the logs of app-cluster-1 managed server. After that no Warning message should be observed in the log as we didn't cross the boundary of any RCM action.



k. Enter 50 in **Allocate Heap** then click on Submit then observe the logs of app-cluster-1 managed server. We will cross the first ("Notify") boundary of RCM actions. So we should see associated log message.

```
####<Nov 5, 2015 10:12:23 PM PST> <Notice> <RCM> <localhost.localdomain> <app-cluster-1> <Thread-73
> <weblogic> <> <f11a4b64-f694-4af6-8b28-7f32a9f96d47-0000003c> <1446790343249> <[Notifying Quota R
eached For Partition: Medrec-Dev] [severity-value: 32] [rid: 0] [Previous Usage: 207153216] [partit
ion-id: cd3147a2-5e49-4731-bad1-94689b69cd5b] [Resource Name: com.oracle.weblogic.rcm.framework.bas
e.HeapRetainedResourceAttributes] [Current Usage: 215825984] [partition-name: Medrec-Dev] > <BEA-21
65799> <Resource Consumption Management Notification Message: Given quota has been reached for the
partition and a notify action has been executed.>
```

l. Enter 50 in **Allocate Heap** then click on Submit then observe the logs of app-cluster-1 managed server. We crossed the second limit ("Slow") so the associated message should be seen in the log.

```
####<Nov 5, 2015 10:14:04 PM PST> <Notice> <RCM> <localhost.localdomain> <app-cluster-1> <Thread-72
> <weblogic> <> <f11a4b64-f694-4af6-8b28-7f32a9f96d47-0000003d> <1446790444011> <[Slow Action Quota
 Reached For Partition: Medrec-Dev] [Current Usage: 299741448] [severity-value: 32] [Previous Usage
: 298576032] [Was Required action to Slow the Partition is executed?: true] [Resource Name: com.ora
cle.weblogic.rcm.framework.base.HeapRetainedResourceAttributes] [partition-id: cd3147a2-5e49-4731-b
ad1-94689b69cd5b] [partition-name: Medrec-Dev] [rid: 0] > <BEA-2165800> <Resource Consumption Manag
ement Slow Message: Given quota has been reached for the partition and a slow action has been execu
ted.>
```

m. Enter 150 in **Allocate Heap** then click on Submit then observe the logs of app-cluster-1 managed server. This will exceed the limit of memory allowed to be used by that partition. So to prevent other partitions from suffering of lack of memory WebLogic will shutdown the partition.

```
####<Nov 5, 2015 10:15:15 PM PST> <Notice> <RCM> <localhost.localdomain> <app-cluster-1> <Thread-74
> <weblogic> <> <f11a4b64-f694-4af6-8b28-7f32a9f96d47-0000003e> <1446790515227> <[severity-value: 3
2] [Proposed Usage: 459764600] [rid: 0] [partition-id: cd3147a2-5e49-4731-bad1-94689b69cd5b] [Shutd
own Action Quota Reached For Partition: Medrec-Dev] [Resource Name: com.oracle.weblogic.rcm.framewo
rk.base.HeapRetainedResourceAttributes] [Current Usage: 442240104] [partition-name: Medrec-Dev] > <
BEA-2165801> <Resource Consumption Management Shutdown Message: Given quota has been reached for th
e partition and a shutdown action has been executed.>
```

l. Refresh the page, http://localhost:7101/dp1/heapApp/ which return 404 and confirm shutdown of the domain partition Medrec-dev in managed server 1.

Error 404--Not Found

From RFC 2068 *Hypertext Transfer Protocol -- HTTP/1.1*:

10.4.5 404 Not Found

The server has not found anything matching the Request-URI. No indication is given of whether the condition is temporary or permanent.

If the server does not wish to make this information available to the client, the status code 403 (Forbidden) can be used instead. The 410 (Gone) status code SHOULD be used if the server knows, through some internally configurable mechanism, that an old resource is permanently unavailable and has no forwarding address.

Note: As this domain partition is target to virtual target which is target at cluster which consists of two managed servers. So this domain partition stopped working on managed server 1, but if you access the application on managed server 2, you still will be able to access the application in this domain partition. If similar things happen in managed server 2 and domain partition shutdown on managed server 2 as well, then domain partition will be shutdown.

LAB 6: OTD INTEGRATION AND RESOURCE MIGRATION

Overview

This lab describes how Oracle Traffic Director can front end a WLS MT Deployment topology for a large enterprise, providing an integrated, end to end administration experience for the server life cycle, and partition management.
For this lab, we installed OTD in Collocated mode. We create one domain otd_domain for OTD instance then we show how to register an OTD instance in WebLogic Domain base_domain. Then we show how you can create partition front ended by OTD.
Oracle Traffic Director has an administration plug in which is responsible for handling life cycle events and automatically configuring Oracle Traffic Director with the corresponding configuration
When you create a WLS MT partition, a corresponding Oracle Traffic Director partition is created for you. The Oracle Traffic Director partition is simply a grouping with the same name as the partition and the resource group. The life cycle of an Oracle Traffic Director partition and its corresponding artifacts are linked to the life cycle of the partition.
The Oracle Traffic Director console provided a partition table with the list of Oracle Traffic Director Partitions to identify the Oracle Traffic director artifacts that are mapped to partitions and resource groups.
Oracle Traffic Director Artifacts map to WebLogic Server MT artifacts as follows:

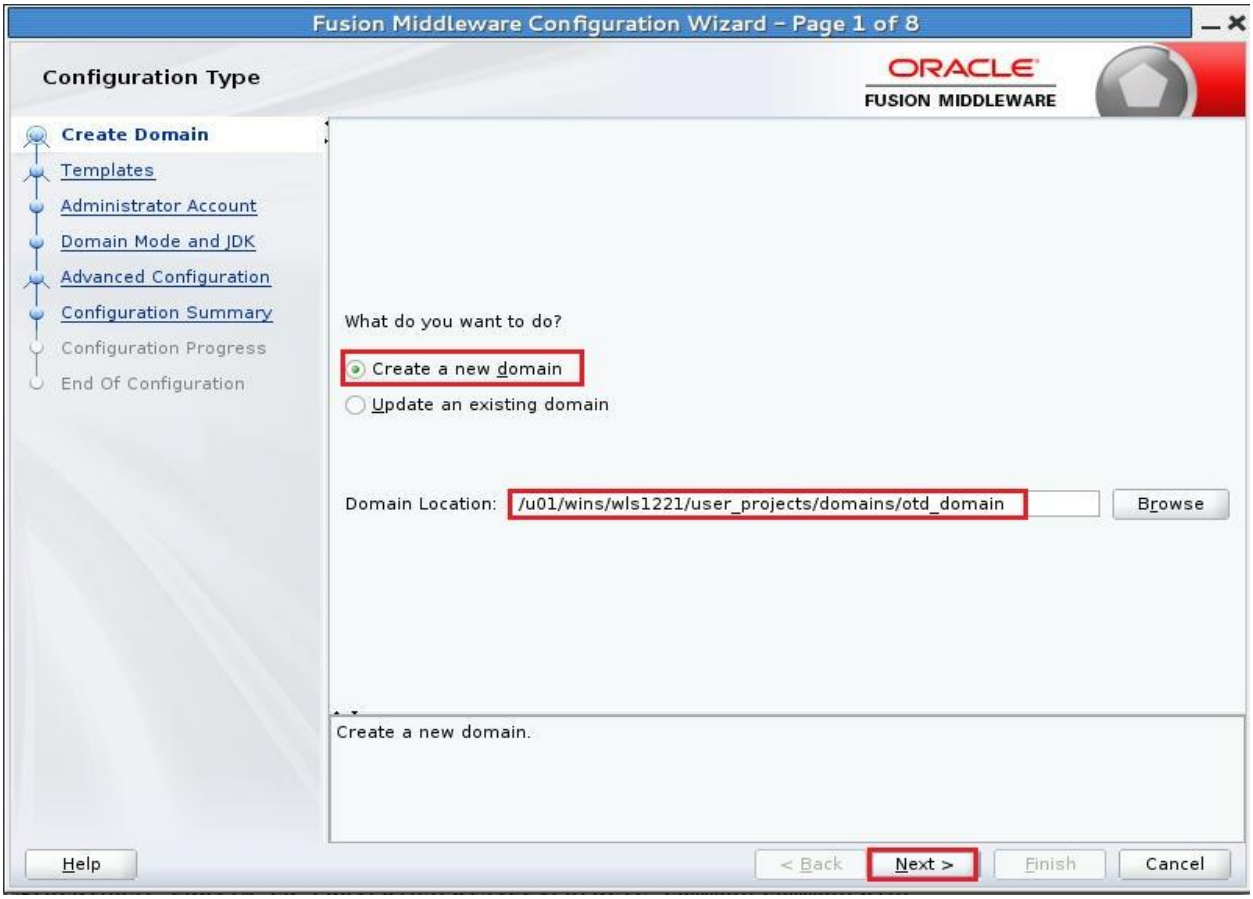
- Each cluster maps to an origin server pool.
- The hostnames of a virtual target that is associated with the partitions and/or resource groups maps to a virtual server.
- Each partition or resource group maps to a route within the virtual server corresponding to the hostname of the virtual target.

In this lab, we are going to perform the below operations.

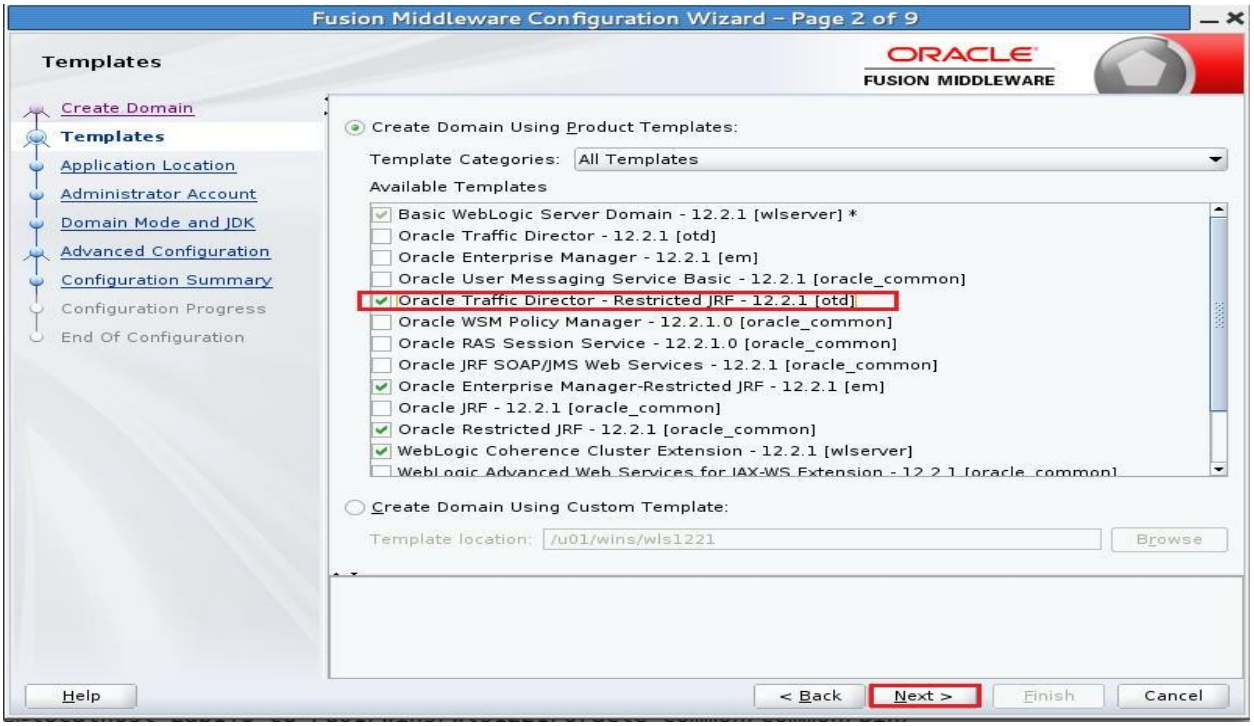
- We create an OTD domain; in that domain we create a machine and OTD configuration.
- We register that OTD Runtime instance inside the WebLogic domain.
- We create a domain partition front ended by OTD.
- We deploy a simple application to verify the OTD integration with domain partition.
- We migrate a resource group from one cluster to other cluster.

Create OTD Restricted JRF Domain

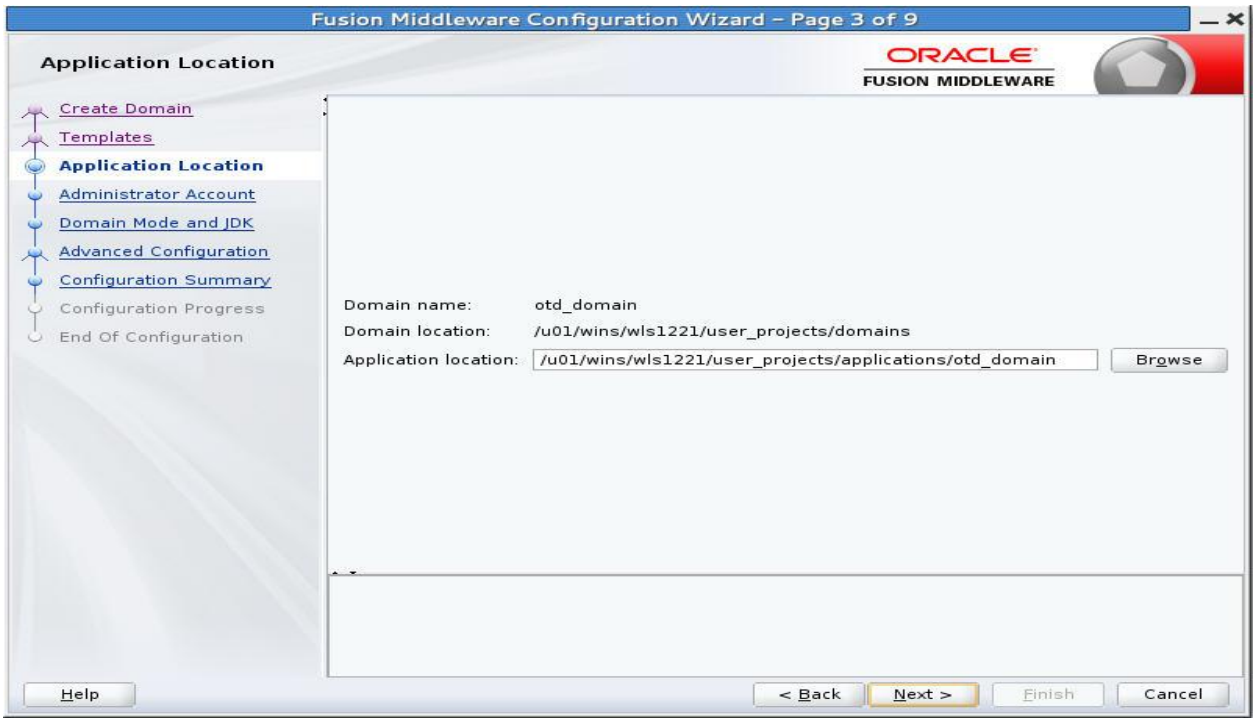
a. Open a new tab.
b. cd /u01/wins/wls1221/oracle_common/common/bin/
c. ./config.sh
d. Select “Create a new domain” and enter “**/u01/wins/wls1221/user_projects/domains/otd_domain**” as Domain Location then Click on Next.



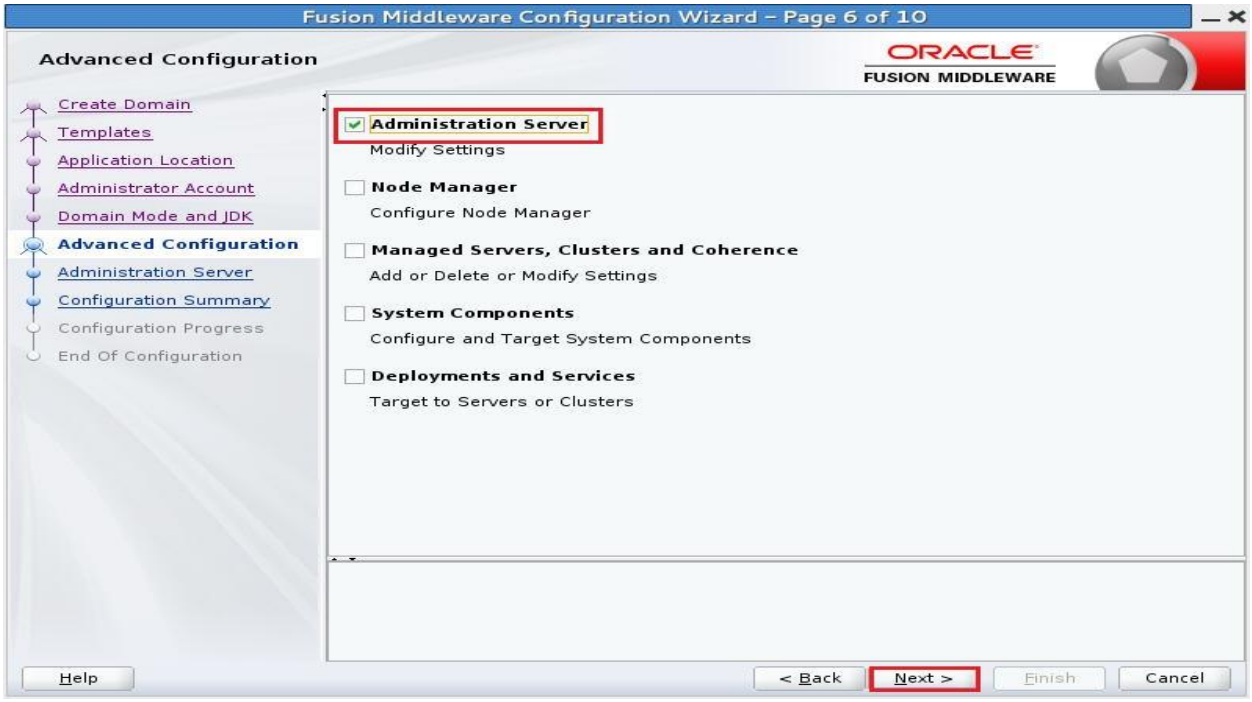
e. Enter "**Oracle Traffic Director- Restricted JRF**" Template as it also selects other required template then Click on Next.



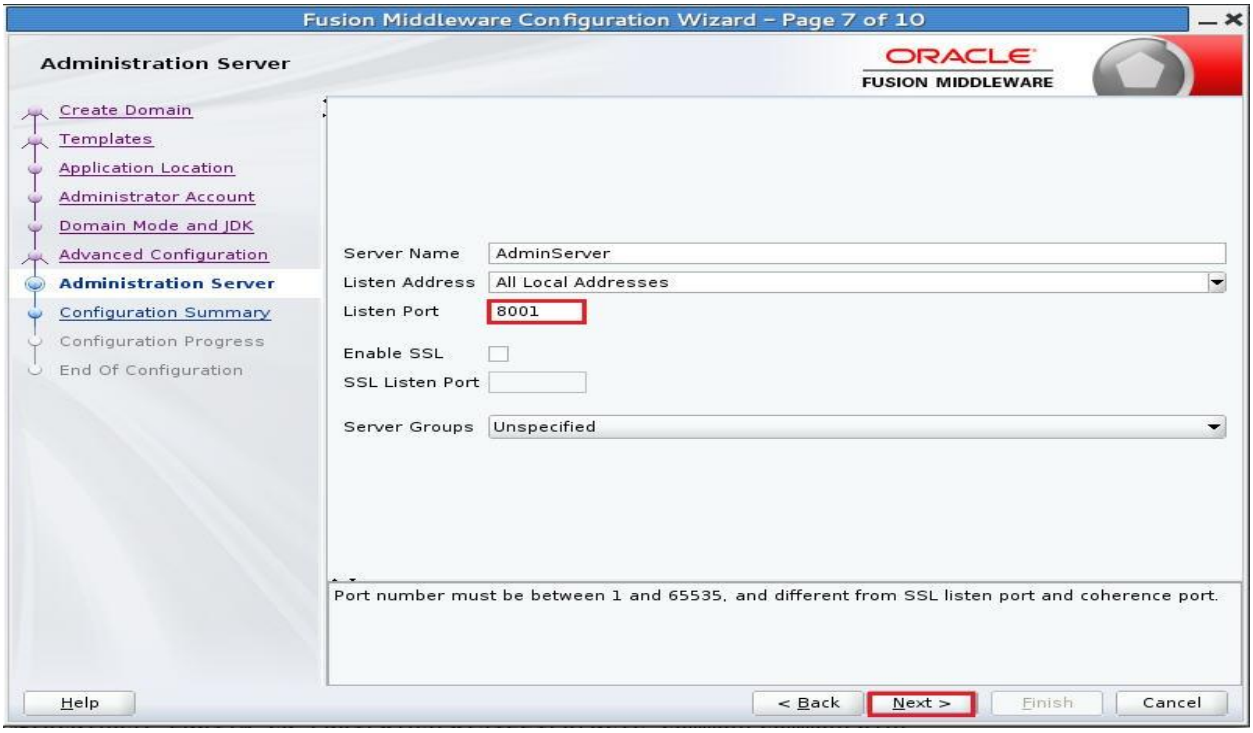
f. Enter "**/u01/wins/wls1221/user_projects/applications/otd_domain**" as Application Location then Click on Next.



g. Enter "**weblogic/welcome1**" as **Username/Password** then Click on Next.
h. Leave Default in **Domain Mode and JDK** then Click on Next.
i. In **Advanced Configuration**, Select the box near **Administration Server** then Click on Next.



j. Change Listen port to **8001** then click on Next.



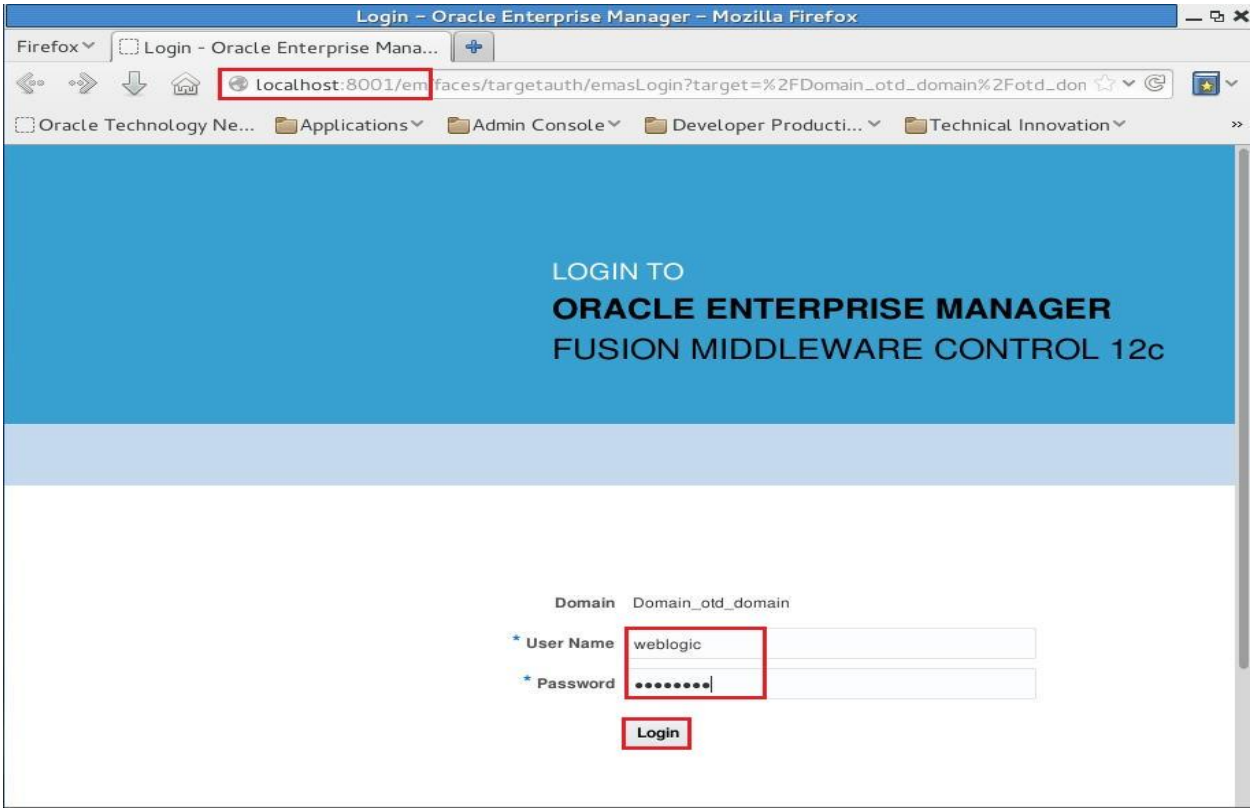
k. Click on **Create**.
l. Click on **Next** then Click on **Finish**.
m. cd /u01/wins/wls1221/user_projects/domains/otd_domain/
n. ./startWebLogic.sh
o. In terminal, Click on **Terminal -> Set Title**. Enter **otd_admin** as Title then click on OK.

p. Open a New tab.
q. cd /u01/wins/wls1221/user_projects/domains/otd_domain/
r. vi nodemanager/nodemanager.properties
s. Change Listen Port to 5557 then save the file and close it.

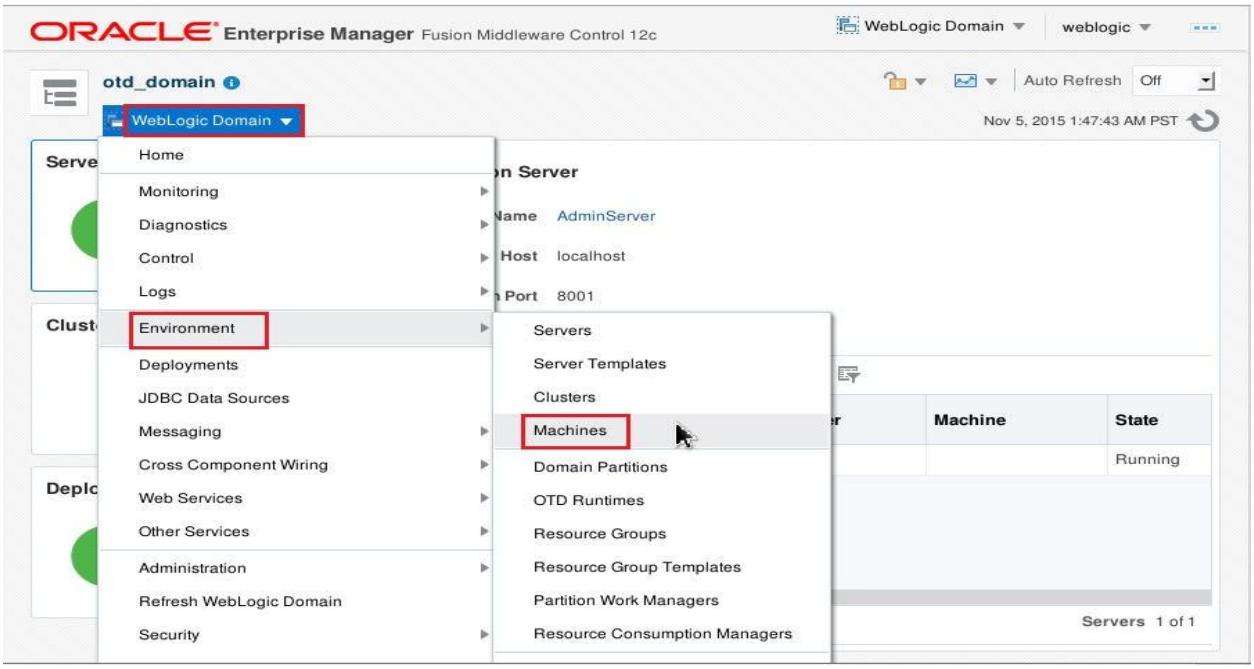
```
#Node manager properties
#Thu Nov 05 01:39:36 PST 2015
DomainsFile=/u01/wins/wls1221/user_projects/domains/otd_domain/nodemanager/nodemanager.domains
LogLimit=0
PropertiesVersion=12.2.1
AuthenticationEnabled=true
NodeManagerHome=/u01/wins/wls1221/user_projects/domains/otd_domain/nodemanager
JavaHome=/u01/java/jdk1.8.0_60
LogLevel=INFO
DomainsFileEnabled=true
StartScriptName=startWebLogic.sh
ListenAddress=localhost
NativeVersionEnabled=true
ListenPort=5557
LogToStderr=true
SecureListener=true
LogCount=1
StopScriptEnabled=false
QuitEnabled=false
LogAppend=true
StateCheckInterval=500
CrashRecoveryEnabled=false
StartScriptEnabled=true
LogFile=/u01/wins/wls1221/user_projects/domains/otd_domain/nodemanager/nodemanager.log
LogFormatter=weblogic.nodemanager.server.LogFormatter
ListenBacklog=50
```

t. bin/startNodeManager.sh
u. In tab, Click on **Terminal -> Set Title**. Enter **otd_nm** as Title then click on OK.

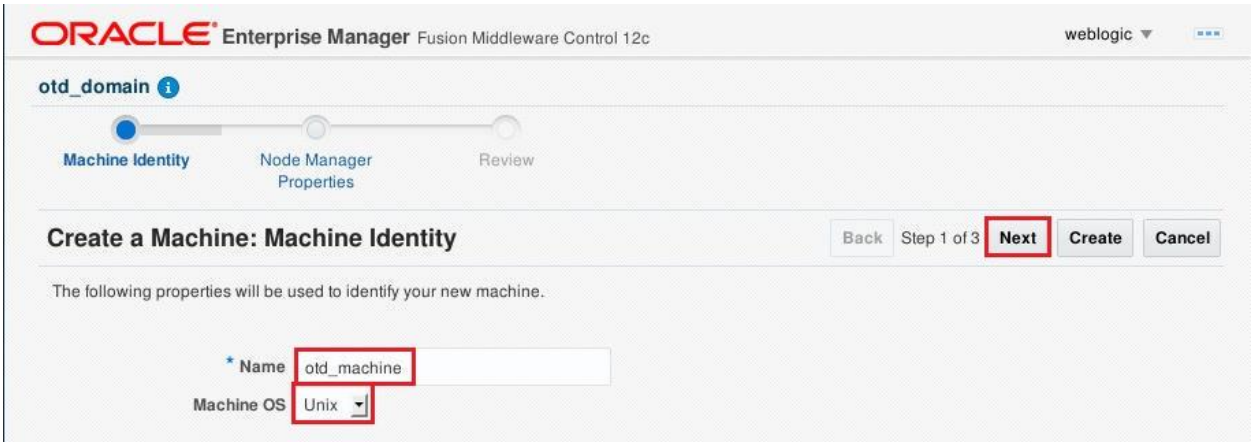
v. Go to Firefox and type the Fusion Middleware Control URL http://localhost:8001/em .
w. Enter "**weblogic/welcome1**" as **Username/Password** then click on Login.



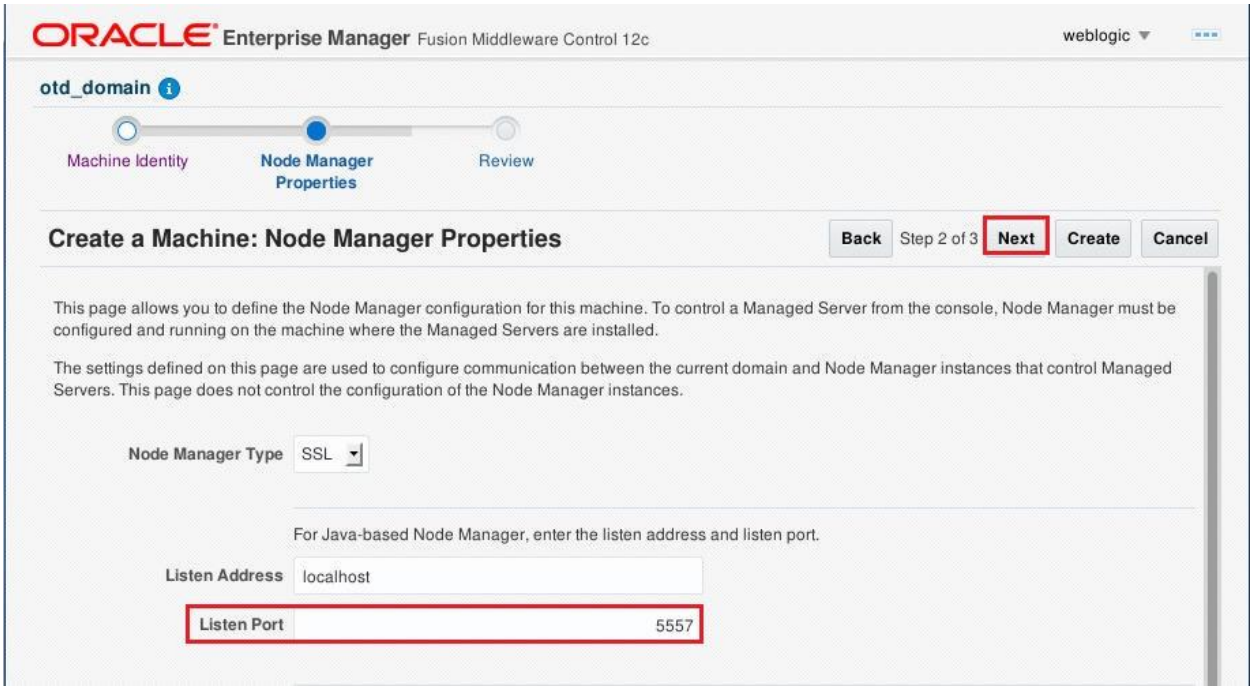
x. Create a Machine.
   i. Click on **WebLogic Domain ->Environment ->Machines.**



ii. Click on Create, Enter "**otd_machine**" as Name and "**Unix**" as Machine OS then Click on Next.

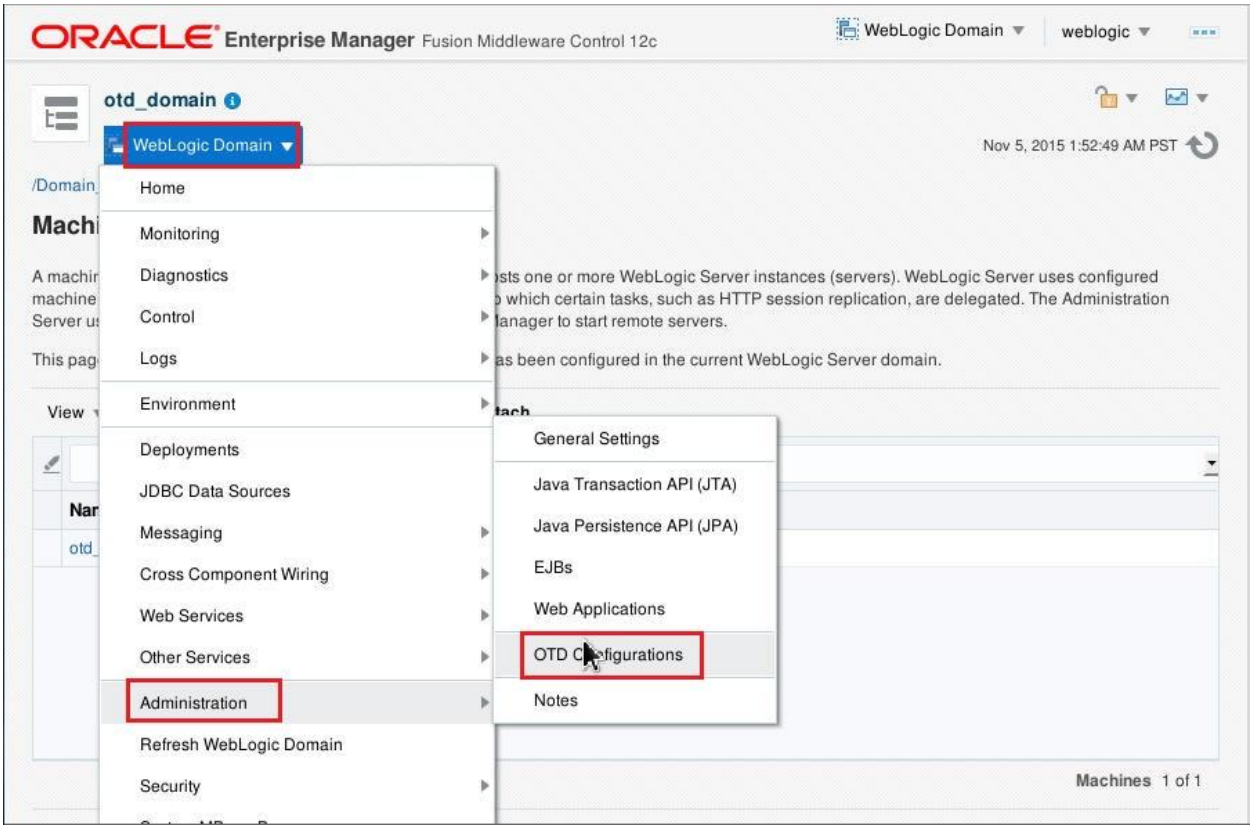


iii. Change Listen Port to **5557** then Click on Next.

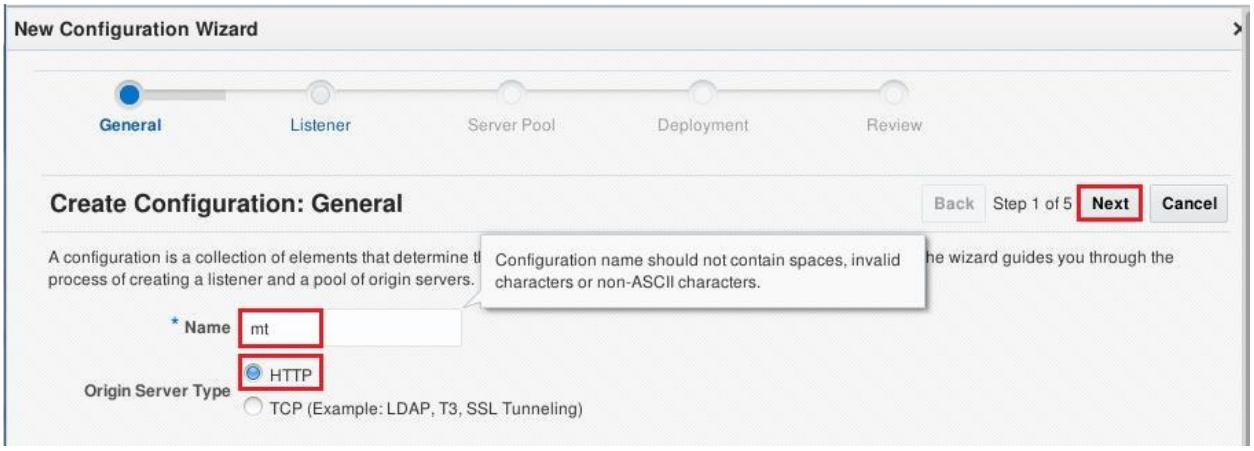


iv. Click on **Create**.

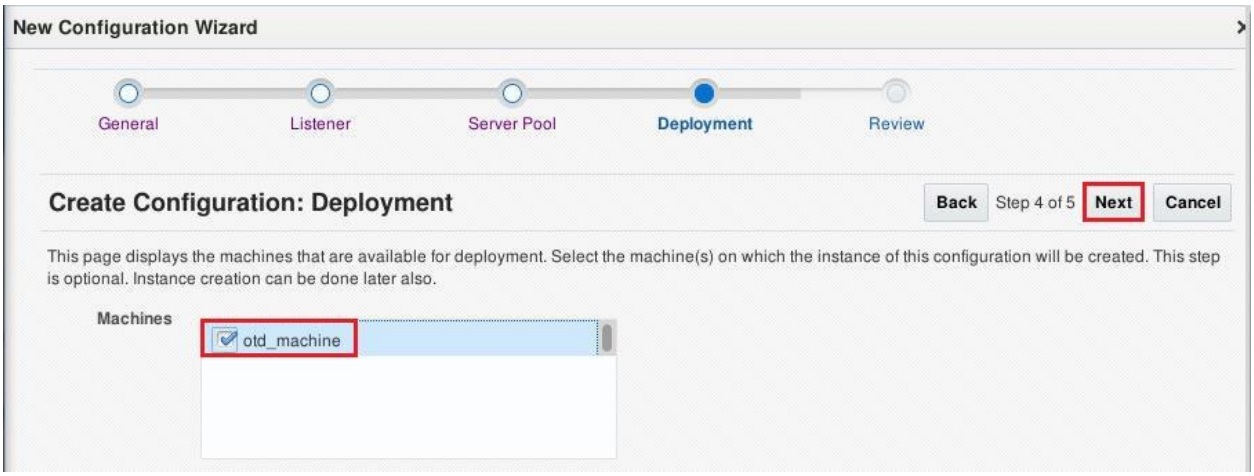
y. Create an OTD Configuration.
   i. Click on **WebLogic Domain -> Administration -> OTD Configurations**.



   ii. Click on Create.
   iii. Enter "**mt**" as Name and "**HTTP**" as **Origin Server Type** then click on Next.

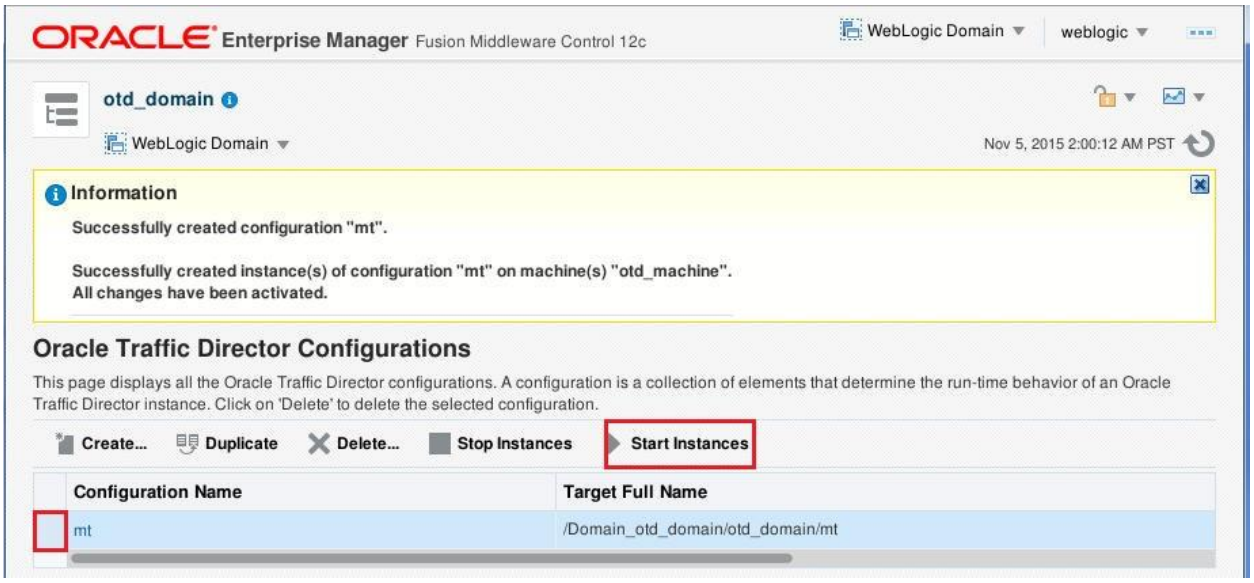


   iv. Leave Default in "**Create Configuration: Listener**" then click on Next.
   v. Leave Default in "**Create Configuration: Server Pool**" then click on Next.
   vi. Select the "**otd_machine**" then click on Next.

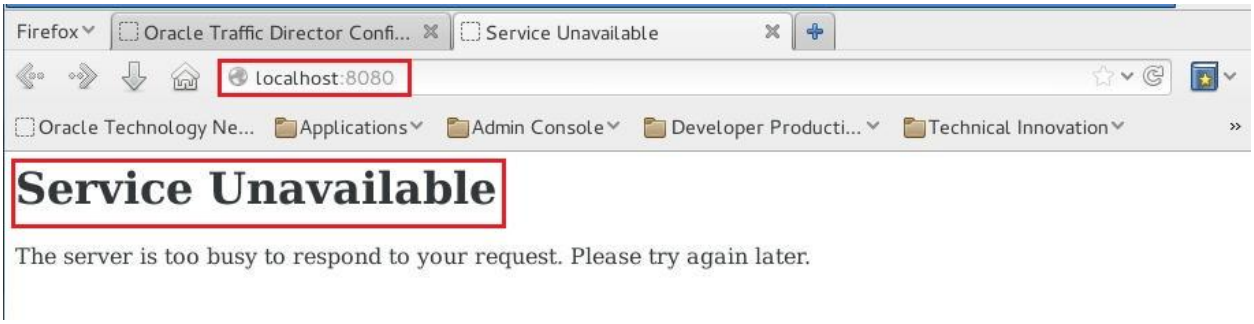


vii. Click on "**Create Configuration**".

viii. Check the box near "**mt**" to make it highlighted and then click on **Start Instances**. Click on Close.

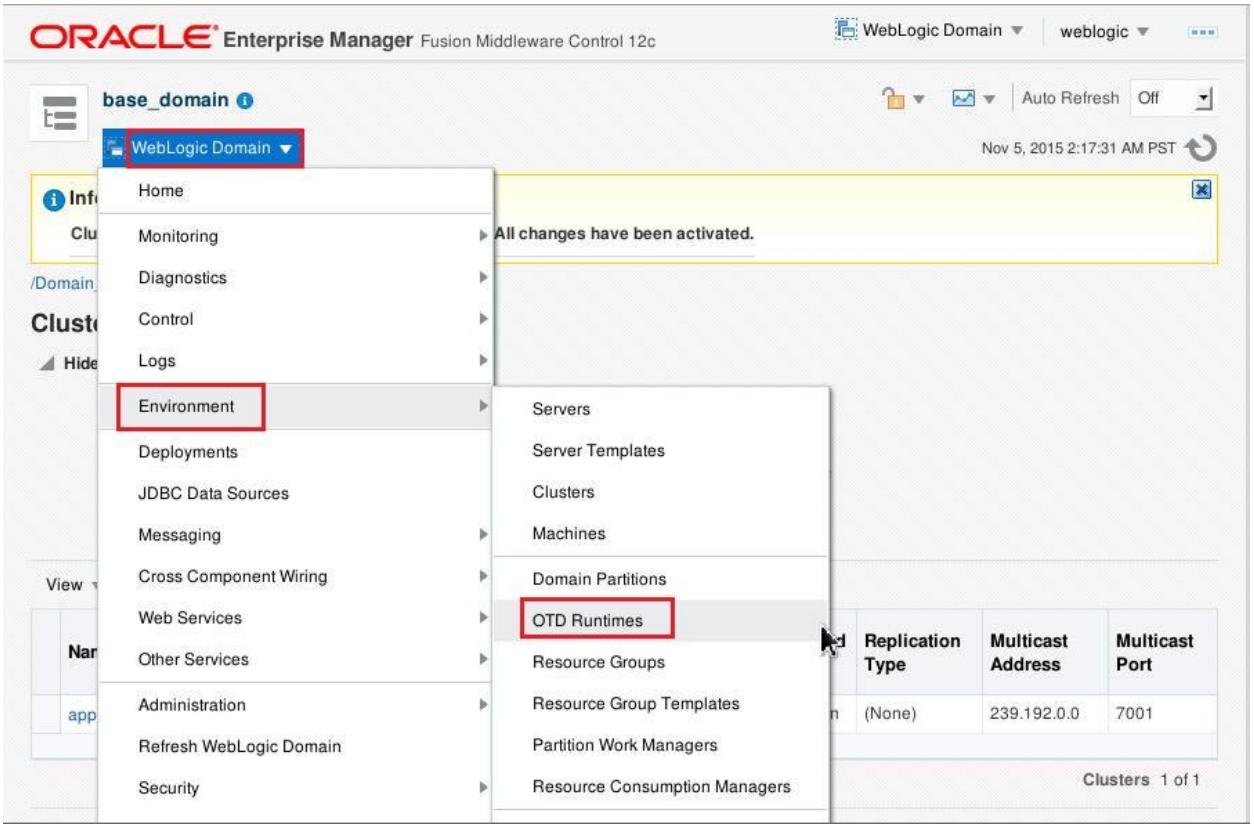


Note: Go to Firefox and type the URL http://localhost:8080/ to verify that server is listening.

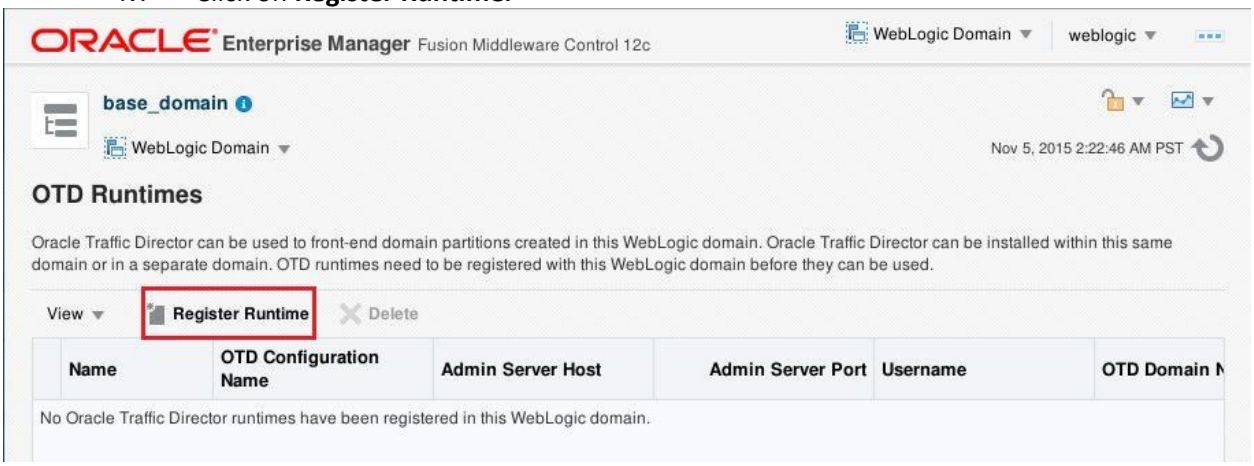


Registering an OTD Runtime Instance

i. Go to base domain Fusion Middleware Control Console http://localhost:7001/em .
ii. Enter weblogic/welcome1 as User Name/Password then click on Login.
iii. Click on **Weblogic Domain -> Environment -> OTD Runtimes.**

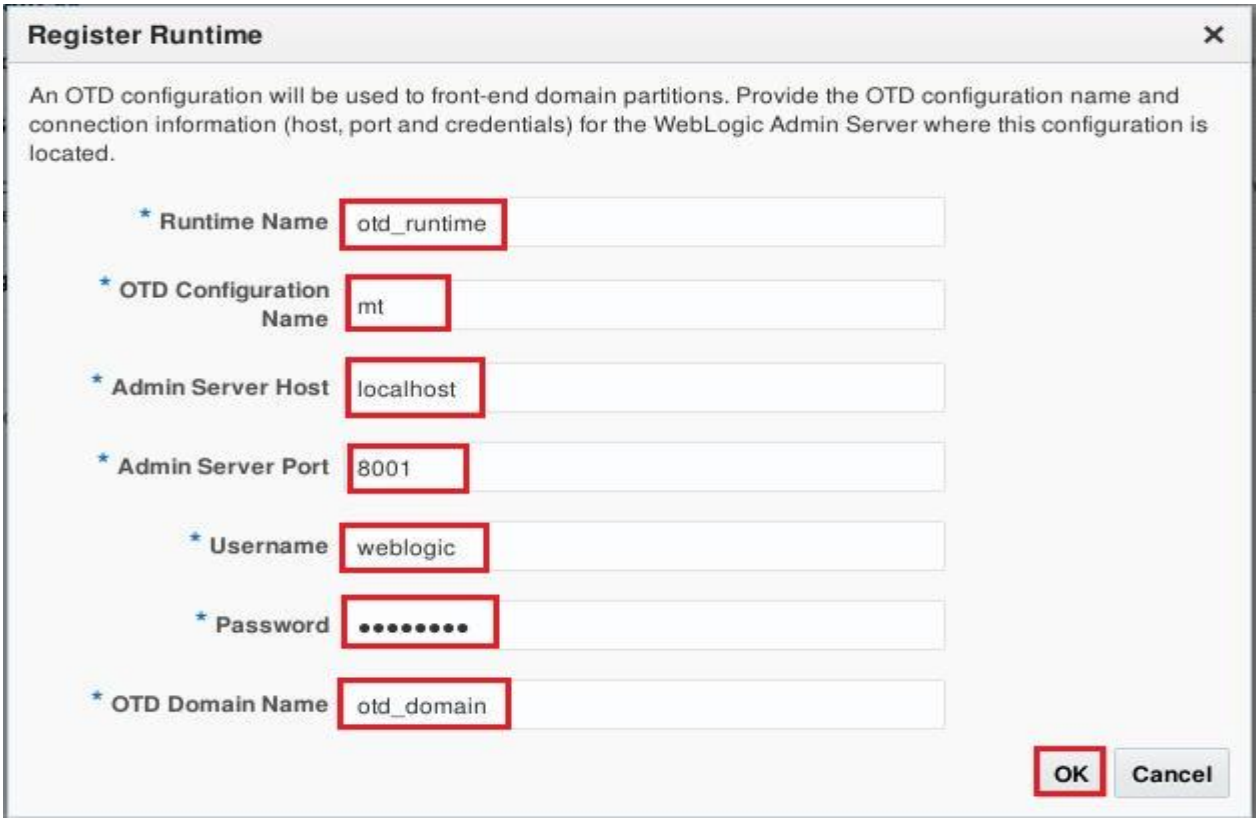


iv. Click on **Register Runtime.**



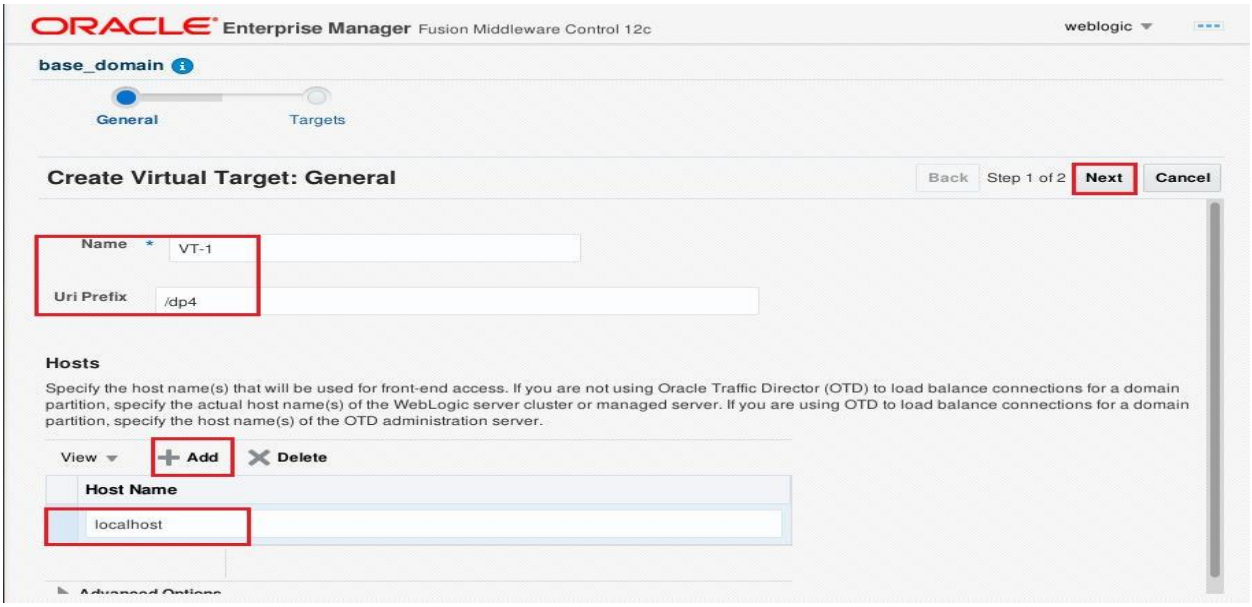
v. Enter the following and then click on OK.

Runtime Name:	otd_runtime
OTD Configuration Name:	mt
Admin Server Host:	localhost
Admin Server Port:	8001
Username:	weblogic
Password:	welcome1
OTD Domain Name:	otd_domain

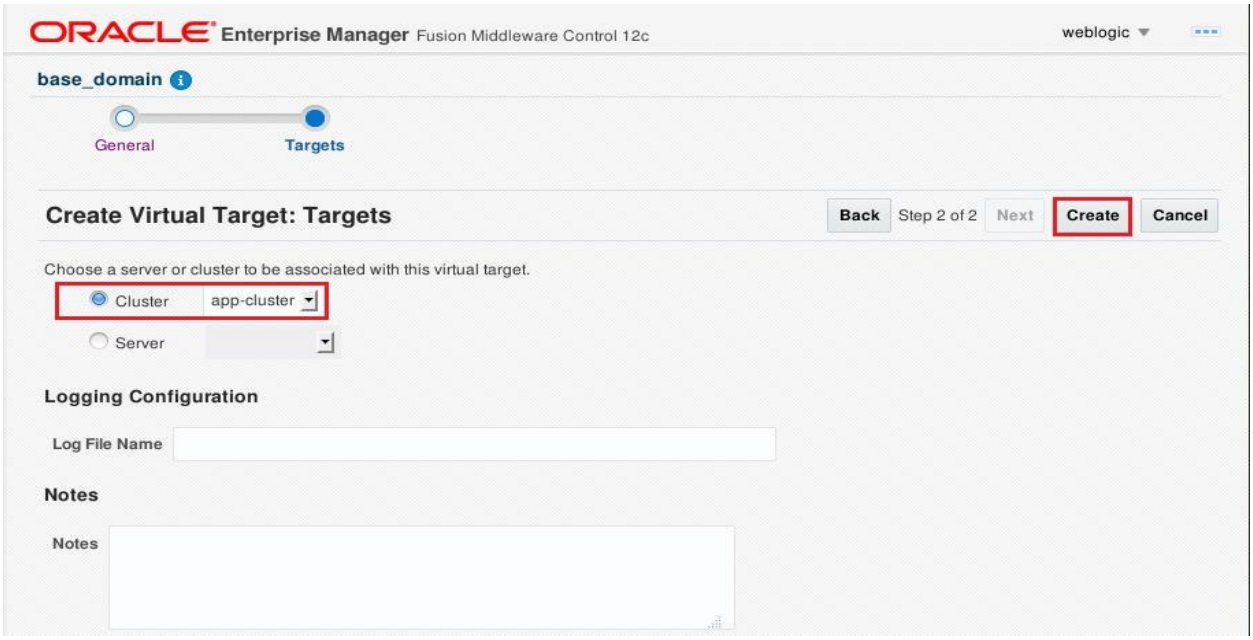


Creating Domain Partition front ended by OTD

a. Click on **WebLogic Domain -> Environment ->Virtual Targets**.
b. Click on Create.
c. Enter **VT-1** as Name and **/dp4** as Uri Prefix then click on Add to Enter **localhost** as Host Name then click on Next.

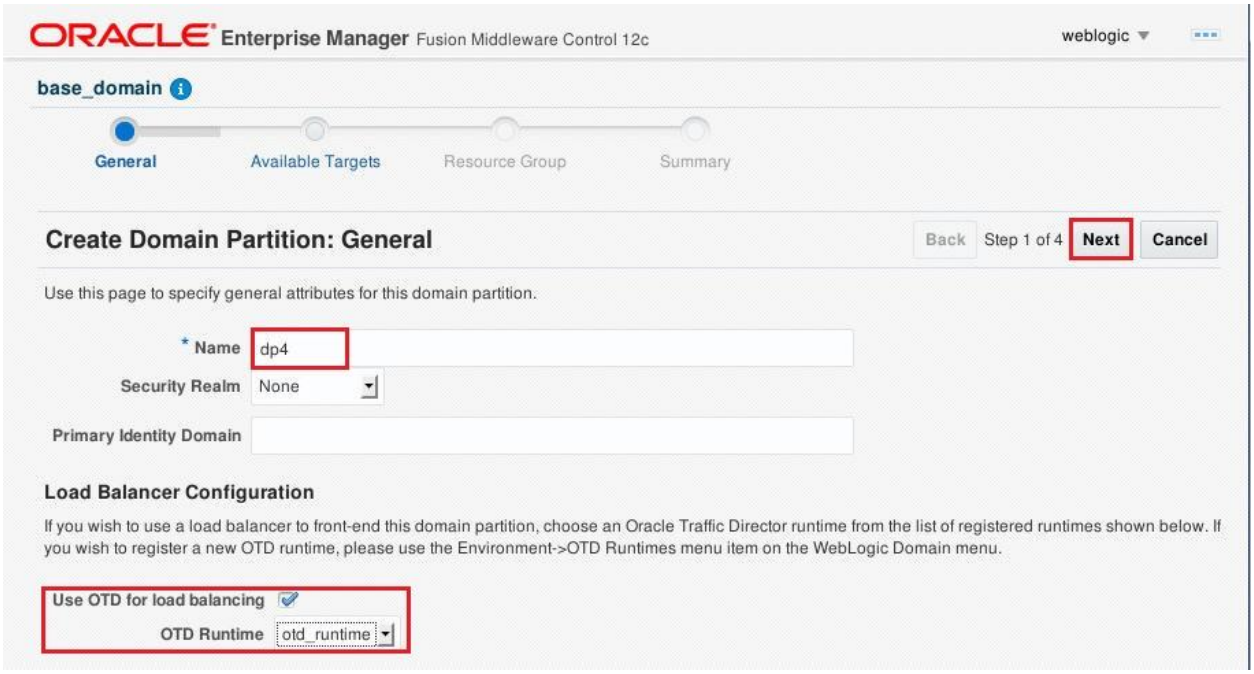


d. Select "**app-cluster**" as Cluster then click on **Create**.

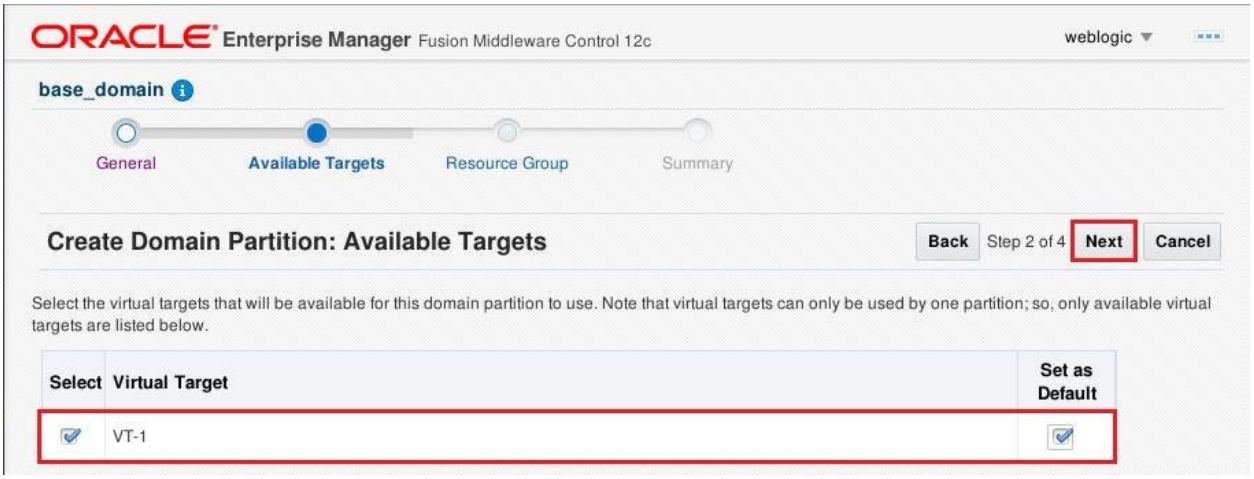


e. Click on **WebLogic Domain -> Environment -> Domain Partition**.
f. Click on **Create**.

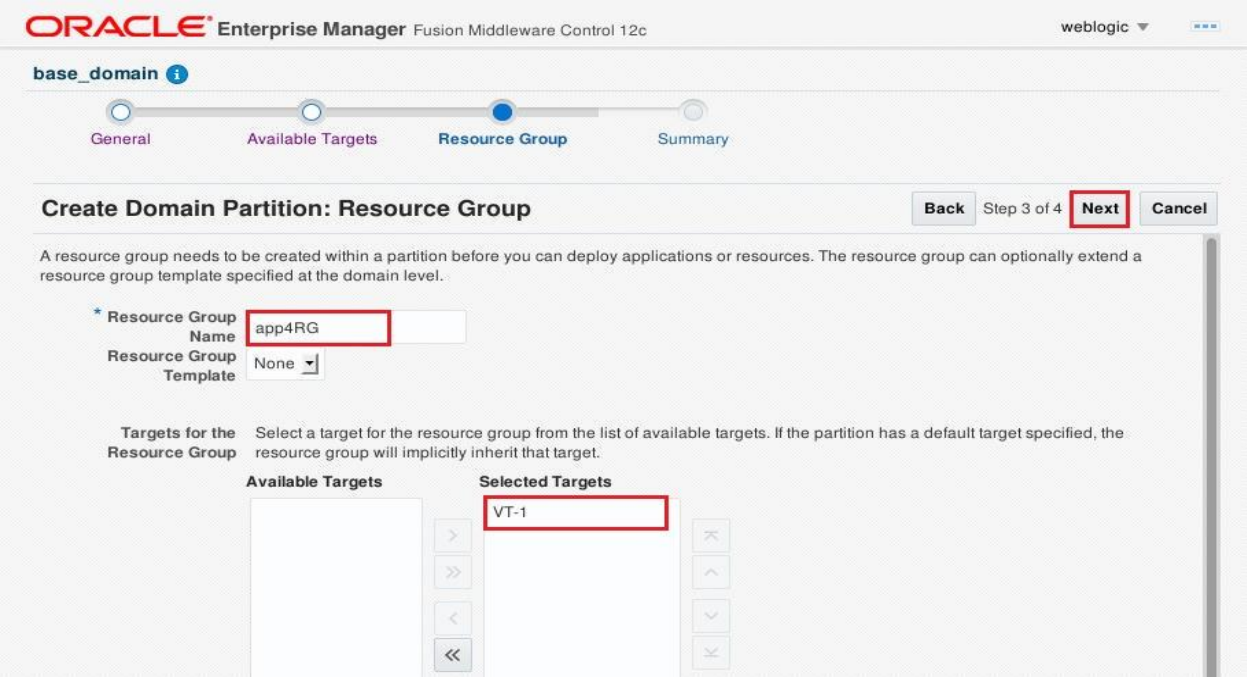
g. Enter **dp4** as name and Check the box for "**Use OTD for load balancing**" and select "**otd_runtime**" then click on Next.



h. Check the box for "VT-1" as shown below then click on Next.

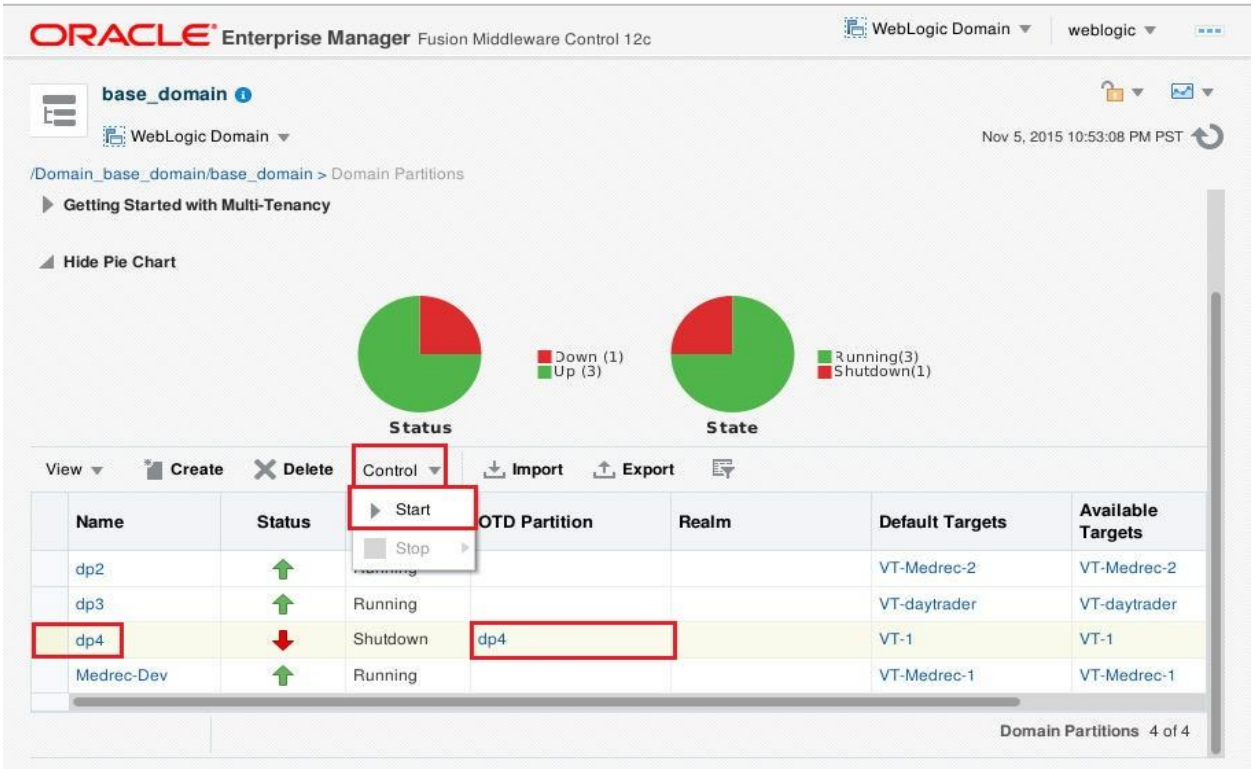


i. Enter **app4RG** as Resource Group Name and Move **VT-1** to **Selected Targets** then click on Next.



j. Review the Configuration then click on **Create**.

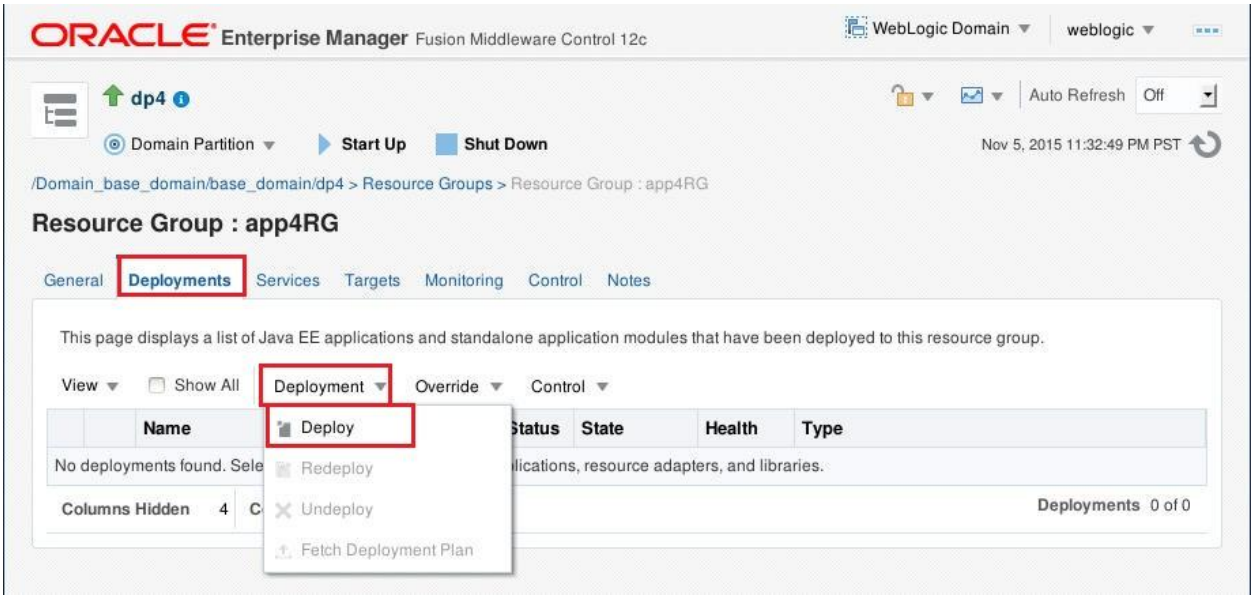
k. Check box near to **dp4** and click on **Control -> Start**. Click on Close.



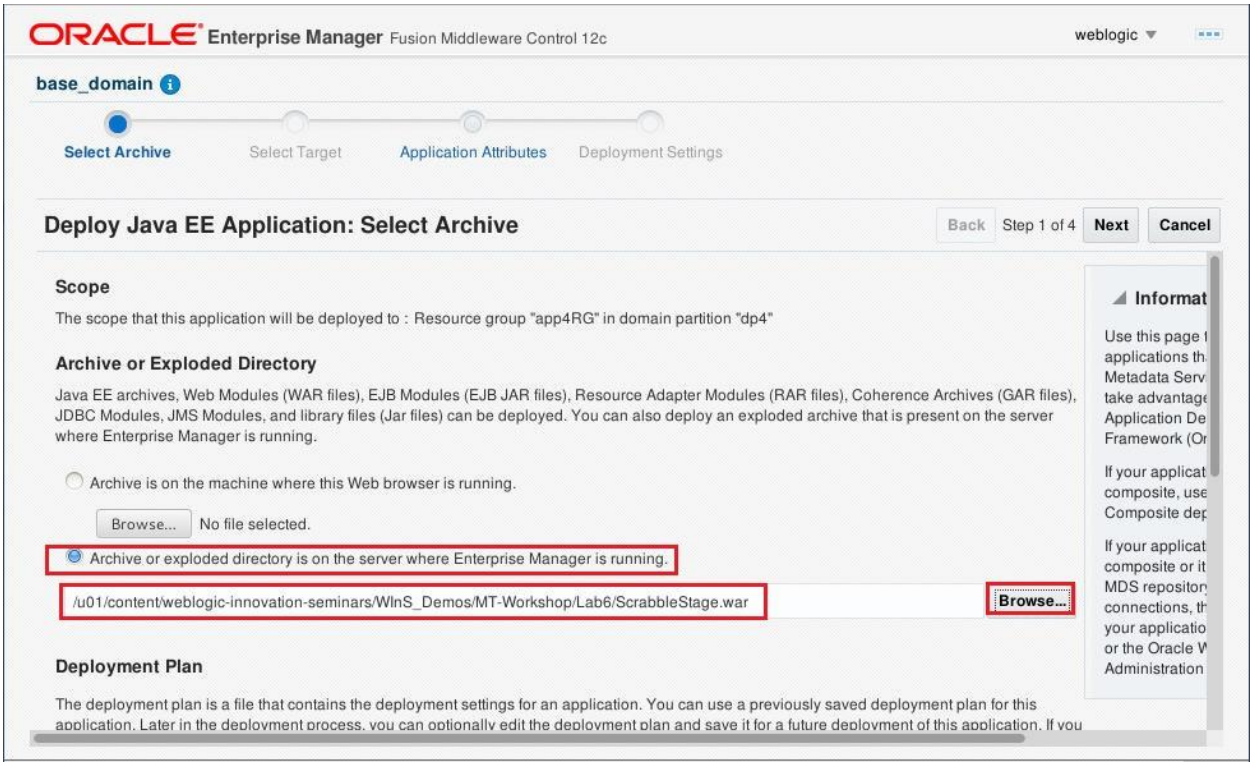
Deploying simple application to test OTD integration with weblogic

Here we are going to deploy heapApp.war which we used in Lab5. We will access that application through **load balancer** now.

a. Click on dp4.
b. Click on **Domain Partition -> Administration -> Resource Groups**.
c. Click on Resource Group **app4RG**.
d. Click on **Deployments** tab, and then click on **Deployment -> Deploy**.



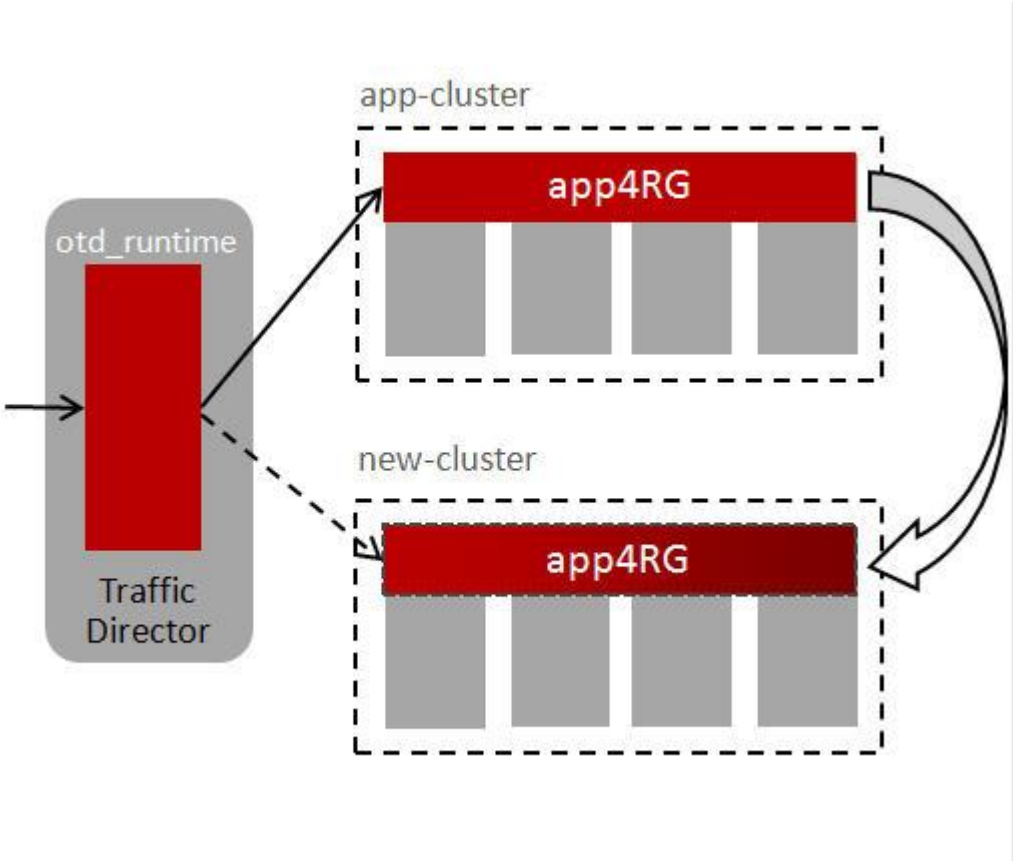
e. Select "**Archive or exploded directory is on the server where Enterprise Manager is running**" then click on **Browse**. Select the file **ScrabbleStage.war** from /u01/content/weblogic-innovation-seminars/WInS_Demos/MT-Workshop/Lab6/ location then click on OK.



f. Click on **Next** then click on **Deploy**. Click on **Close**.
g. Go to Firefox and type the URL http://localhost:8080/dp4/ScrabbleStage/Scrabble.jsp .

Migration Resource Group from one Cluster to other Cluster

Now we are going to migrate Resource Group app4RG from app-cluster to another cluster. As the application deployed in app4RG are front ended by OTD, when you migrate your resource group on other cluster. WebLogic automatically detect the server setting and you can access the application continuously without doing any manual configuration.



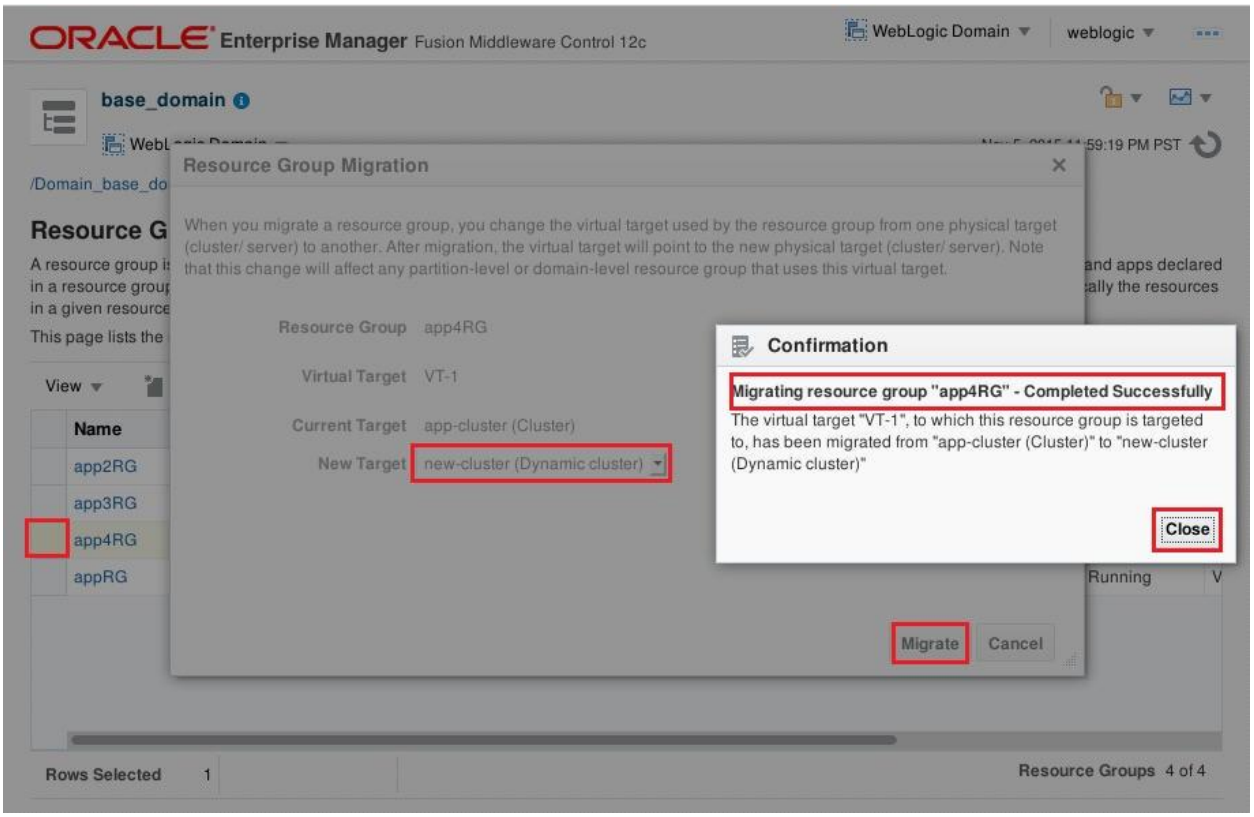
- When you click on **migrate** the following actions performs:
- Migrate API is called
- Resource Group starts on new hosts
- Sessions are replicated to new hosts.
- New origin server pool added to the OTD Configuration
    - Old pool used for sticky request to old pool only
    - New pool used for all new requests.
- Graceful shutdown called on resource group on original hosts.
- Virtual target configuration updated with new cluster only.

So using the same URL, you can access the application.

a. Go back to Fusion middleware control console http://localhost:7001/em
b. Create a new Cluster.
    i. Click on **WebLogic Domain -> Environment -> Clusters**.
    ii. Click on **Create -> Dynamic Cluster**.
    iii. Enter **new-cluster** as Name then click on Next.
    iv. Enter 1 as Dynamic Cluster Size then click on Next.
    v. Select the box for "**Use a single machine for all dynamic servers**" and choose "**machine**" as Selected Machine then click on Next.
    vi. Enter 9100 and 10101 as **Base Listen Port** and **SSL Base Listen Port** Respectively then click on Next.
    vii. Review the Configuration then click on **Create**.
    viii. Click on **WebLogic Domain -> Control -> Clusters**.
    ix. Check the box near **new-cluster** to make it highlighted and then click on **Control -> Start -> Start Servers**.

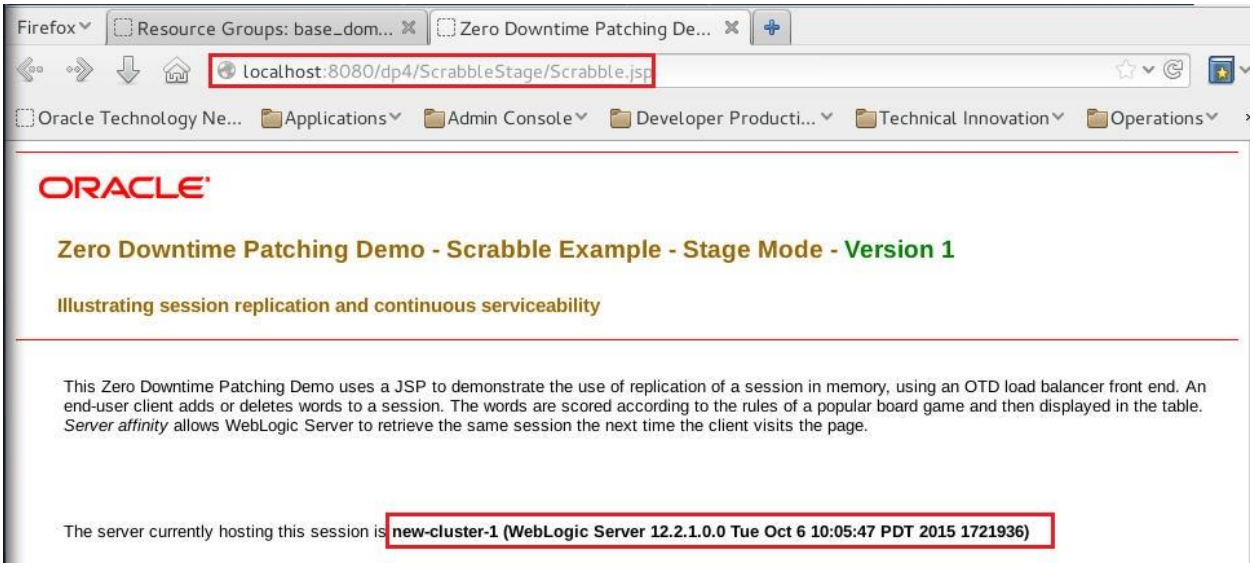
Note: Before going to next step, wait till managed server in new-cluster get started, you can monitor the progress in base_nm tab.

c. Migration of Resource Group.
    i. Click on **WebLogic Domain -> Environment -> Resource Groups.**.
    ii. Check the box near to app4RG to make it highlighted and then click on Migrate. Select **new-cluster** as New Target then click on Migrate. In Confirmation window, click on Migrate.
    iii. Once you notice "**Migrating resource group "app4RG" – Completed successfully**" message then click on Close.

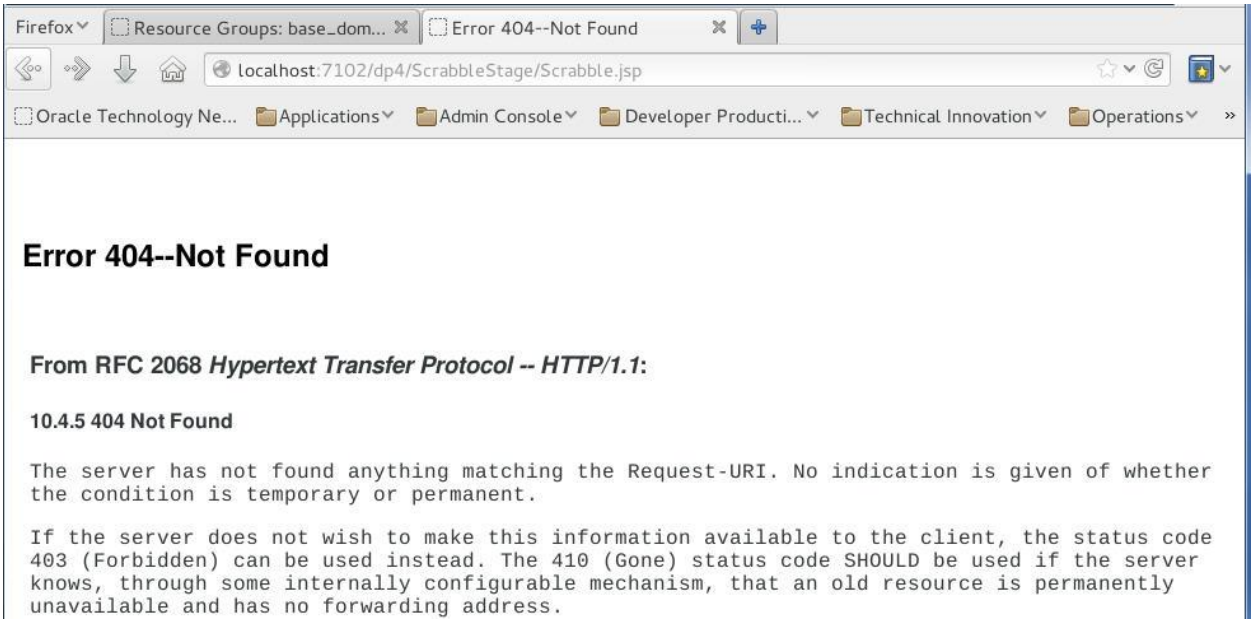


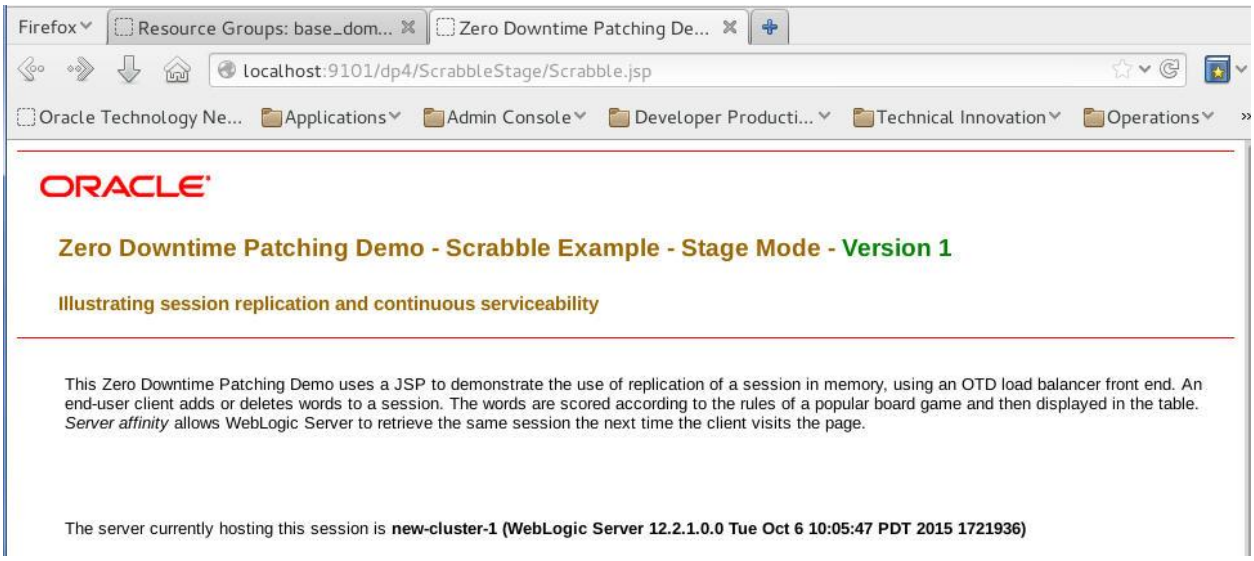
Access Application through OTD

a. Go to Firefox and type the URL http://localhost:8080/dp4/ScrabbleStage/Scrabble.jsp .



b. As the Resource group migrated from app-cluster to new-cluster. But still you are able to access the application through same URL. Load balancer is redirecting all requests to new cluster. new-cluster-1 is running on 9101 port and app-cluster-2 is running on 7102.



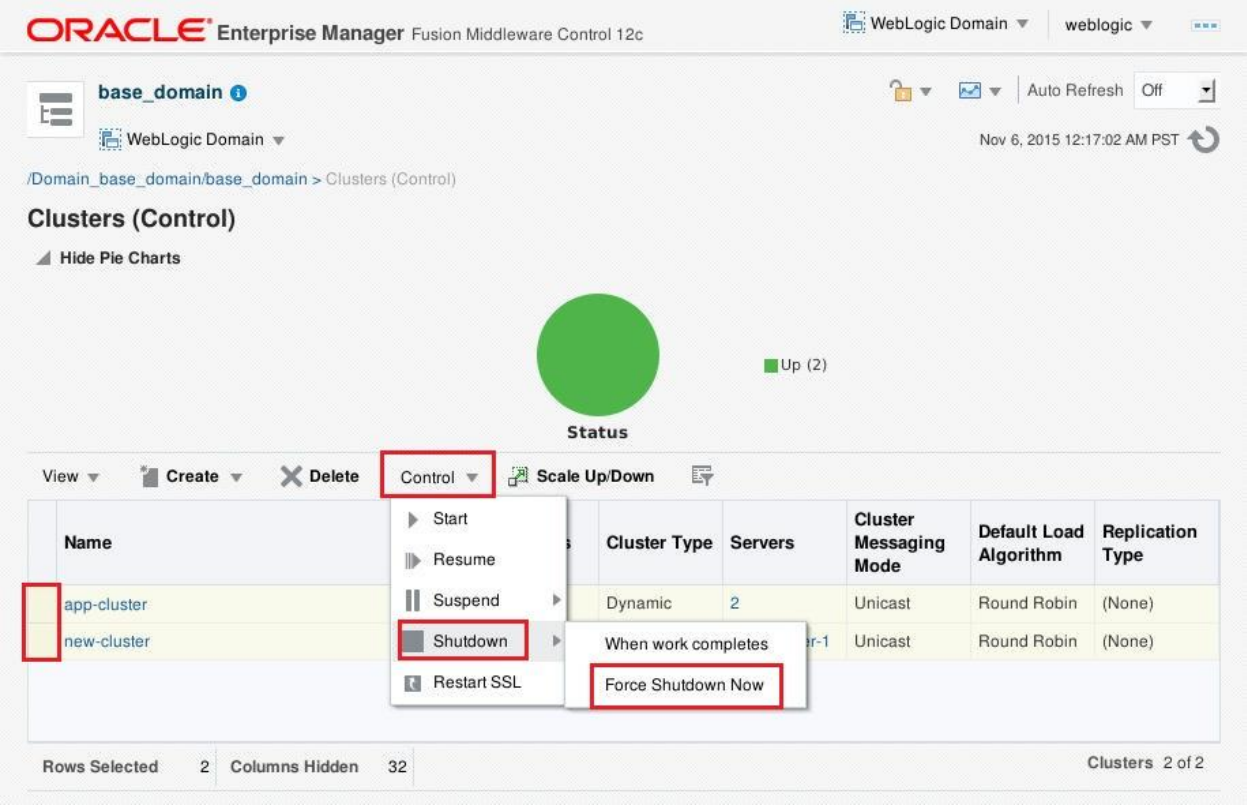


So you can successfully migrate the resource group from one cluster to another cluster without affecting the front end of your application.

CLEANING AND RESETING

Cleaning up Environment

a. Go back to Fusion Middleware Control http://localhost:7001/em
b. Click on WebLogic Domain -> Control -> Cluster.
c. Check the boxes near to both the cluster then click on Control -> Shutdown -> Force Shutdown now. Click on Forcibly Shutdown Servers.



d. Press CTRL + C on Tab title with otd_admin, otd_nm, base_nm, base_admin, dev_admin,app-cluster-1.
e. Open a new terminal.
f. cd /u01/content/weblogic-innovation-seminars/WInS_Demos/MT-Workshop/CleanUp/
g. ./CleanEnvironment.sh

Appendix:

This Section is to provide you help, if you already run this workshop once, and only interested to showcase any specific features or any specific lab. We created few scripts about which the description is given below.

Lab1-2.sh: It starts the DB, and create user and populates the DB for all domain partition we need. Then it create the base_domain, in that domain we create the machine and dynamic cluster. After that we create three domain partition dp1, dp2 and dp3. Basically after execution of this script, you will be at the end of Lab 2.

Note: After executing Lab1-2.sh, if you want to see proper metrics/state of cluster/servers you need to manually apply JRF template to cluster. You need to restart the cluster as well.

Lab3.sh: This script stop the domain partition dp1 and create new security realm newrealm and assign this security realm to dp1, and start the domain partition dp1. It perform the same task what Lab 3 does.

Lab4.sh: This script creates dev_domain, it configures the Medrec application in Medrec-Dev domain partition in dev_domain, and then it removes the dp1 in base_domain. So you can login to admin console of dev_domain and export the partition and later import it in base_domain through admin console.

Lab5.sh: It sets the configuration parameter for RCM in SetDomainEnv.sh script then it restart the cluster and also deploys heapApp.war in dp2 partition in base_domain.

Lab6.sh: This script creates otd_domain first, inside otd_domain; it creates otd_machine and OTD Configuration. Then it creates load balancer instance and start it. Then in base_domain, it creates the Virtual target VT-1, and registers OTD Runtime, after that it creates domain partition dp4 with otd_runtime and then deploy the heapApp.war in app4RG resource group inside dp4 partition. It also create new-cluster dynamic cluster, and start domain partition dp4 and server in new-cluster. After running this script, you can move to resource group app4RG in dp4 in FMW console, and migrate the resource group.

CleanEnvironment.sh: This script stops the servers and Node managers, remove the otd_domain, base_domain, dev_domain. It also removes the user from pluggable database, and then it stops the DB.