Oracle® Retail High Availability Case Study - Retail Integration Suite Installation

Release 15.0
2151366.1

August 2016

Note: The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described for Oracle's products remains at the sole discretion of Oracle.

Contents

1 Installation Overview 1

2 Understanding Installation Directories 3

3 Installing RIB and RDMT 5

Install WebLogic Server 12c for RIB on Shared Mount of ribhost1 and ribhost2 5
Creating Database Repository for RIBDomain 11
Configure RIBDomain on ribhost1 17
Start WLS and Managed Servers - Create boot.properties for RIBDomain 31
Verification – Verify the Active/Passive VIP Configuration for RIBDomain 31
Configure weblogic.policy for RIB Application 35
Prepare for RIB and RDMT Installations 37
Verification – Verify the RIB Installation 60
Enable RIB Functional Artifacts Directory 62
Verification – Verify the RIB Functional Artifacts 66
Install RDMT 67
Verification – Verify RDMT Installation 67

4 Installing RTG Applications 71

Install WebLogic Server 12c with ADF on rtghost1 and rtghost2 71
Creating Database Repository for RTGDomain 77
Configure RTGDomain on rtghost1 83
Start WLS and Managed Servers - Create boot.properties for RTGDomain 97
Prepare RTG Application Installations for IGS and JMS 100
Verification - Verify IGS Installation 104
Verification - Verify RMS Web Services 141

5 Installing RSB and RIC Applications 143

Install WebLogic Server 12c with ADF on rtghost1 and rtghost2 143
Install OSB 12.1.3 on rtghost1 and rtghost2 149
Creating Database Repository for RSBDomain 153
Configure RSBDomain on rtghost1 160
Start WLS and Managed Servers - Create boot.properties for RSBDomain 173
Propagate RSBDomain from rthost1 to rtghost2 174
Configure the RSBDomain Memory Allocation Arguments 176
Configure weblogic.policy for RSB Application 177
Install RSB Application 179
Verification - Verify RSB Installation 181
Verification - Verify RIC Installation 183

6 OHS Configuration for RIB and RTG Applications 185

Enable Tunneling for Servers in RIB, RTG, and RSB Domains 185
Update Web Tier for RIB Applications 188
Update Web Tier for RTG and RSB Applications 191

Verification - Verify Web Tier Configuration for RSB 192
Verification - Verify Web Tier Configuration for RIC.....193
Verification - Verify Web Tier Configuration for JMS 195
Verification - Verify Web Tier Configuration for RSE 196
Verification - Verify Web Tier Configuration for IGS Web Services.....197
Verification - Verify Web Tier Configuration for RMS Web Services.....198

A Appendix: rsb-development-env-info.properties 199

B Appendix: ric-deployment-env-info.json..... 203

C Appendix: RIB mod_wl_ohs.conf..... 205

D Appendix: RSB and RTG mod_wl_ohs.conf 207

1

Installation Overview

Oracle Retail applications can be deployed and configured to operate in a High Availability environment utilizing Web Logic Server on hosts that are configured as active/active along with Oracle Real Application Cluser (RAC) databases to provide resilience, high availability and scalability as described in previous Part1 and Part2 docs. In order to facilitate this, the integration layer for Oracle Retail enterprise applications should be considered to be made highly available in order to make sure that messaging channels between the enterprise applications can remain open.

There are many options to configure an Oracle High Availability environment, and these options can have a significant impact on how the environment is configured and built. The intent of this case study is not to provide exhaustive documentation on all possible ways to configure a suitable High Availability environment for Oracle Retail, but rather to document a specific configuration that was used for verification purposes. Because of this, the specific installation steps documented in this case study only apply in their entirety to the exact hardware and software configuration described in this case study.

This case study details the steps needed to create key components of the integration layer for Oracle Retail applications. The components contained in this case study are as follows:

- Installed on hosts as Active/Passive (ribhost1 and ribhost2)
  - Retail Integration Bus (RIB)
- Installed on hosts as Active/Active (rtghost1 and rtghost2)
  - Retail Service Bus (RSB)
  - Retail Integration Console (RIC)
  - Java Messaging Service (JMS)
  - Integration Gateway Services (IGS)
  - Retail Service-Oriented Architecture Enabler (RSE)
  - RMS Web Services

2

Understanding Installation Directories

The following technical stack is installed from the locations described:

- Oracle Web Tier 11.1.1.9
  - WEBLOGIC_HOME=/u00/webadmin/products/wls_ohs
  - ORACLE_HOME= /u00/webadmin/products/wls_ohs/Oracle_WT1
  - Hosts:
    a. All hosts

    NOTE: Instructions for configuring Web Tier for the applications installations outlined in this doc are also included in this document. Instructions for the installation of Web Tier are located in the Oracle Retail 15.0.1 High Availability Case Study Fusion Middleware Cluster - Part 1 document.

- RIBDomain (RIB)
  - WebLogic 12.1.3
  - Installation directories are on shared mount between ribhost1 and ribhost2
  - WEBLOGIC_HOME=/u00/webadmin/products/wls_rib
  - DOMAIN_HOME=/u00/webadmin/config/domains/wls_rib
  - RIB_HOME=/u00/webadmin/RIB_INSTALL/Rib1500ForAll15xxApps/rib-home
  - Hosts:
    a. ribhost1
    b. ribhost2
- RTGDomain (JMS, IGS, RSE, RMS Web Services)
  - WebLogic 12.1.3
  - WEBLOGIC_HOME=/u00/webadmin/products/wls_rtg
  - DOMAIN_HOME=/u00/webadmin/config/domains/wls_rtg
  - RIB_HOME=/u00/webadmin/RIB_INSTALL/Rib1500ForAll15xxApps/rib-home
  - Hosts:
    a. rtghost1
    b. rtghost2
- RSBDomain (RSB, RIC)
  - WebLogic 12.1.3
  - WEBLOGIC_HOME=/u00/webadmin/products/wls_rib
  - DOMAIN_HOME=/u00/webadmin/config/domains/wls_rib
  - RIB_HOME=/u00/webadmin/RIB_INSTALL/Rib1500ForAll15xxApps/rib-home

  - RSB_HOME=/u00/webadmin/RSB_INSTALL/Rsb1500ForAll15xxApps/rsb-home
  - RIC_HOME=/u00/webadmin/RIC_INSTALL/ric-homeHosts:
  - Hosts:
    a. rtghost1
    b. rtghost2
- Java
  - JDK 1.7+ 64 bit (Using Java 7 update 85 for case stud)
  - JAVA_HOME=/u00/webadmin/products/jdk_java
  - Hosts:
    a. All hosts

3

Installing RIB and RDMT

The chapter contains instructions needed to install and configure the RIB. Included are instructions needed to setup the infrastructure and verify the installation.

Install WebLogic Server 12c for RIB on Shared Mount of ribhost1 and ribhost2

This section describes the steps required for installing WebLogic 12.1.3 on a shared mount between ribhost1 and ribhost2.

Perform the following steps to install WebLogic 12.1.3 on ribhost1 only. It will only need to be installed here since the installation is intended to be on a shared mount between the 2 hosts.

- Download software installation for WebLogic 12.1.3 that is bundled with ADF and place it in an installation staging directory on ribhost1.

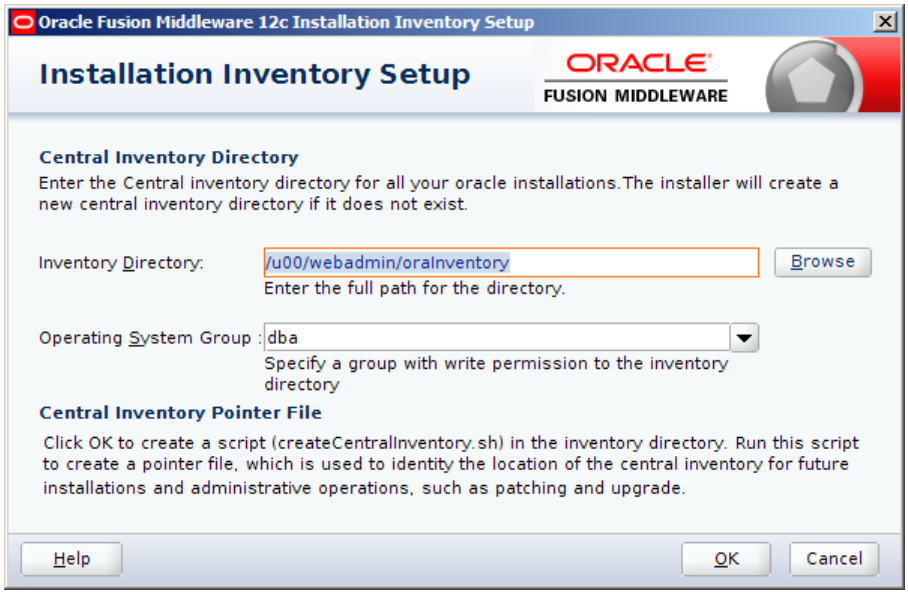
  Example:
  <INSTALL_DIR>/WLS-ADF12c_12.1.3.
- Set the following variables by exporting them before running the installer so that the JAVA_HOME is added to the PATH variable:
  - JAVA_HOME
  - PATH
  - DISPLAY

  Example:

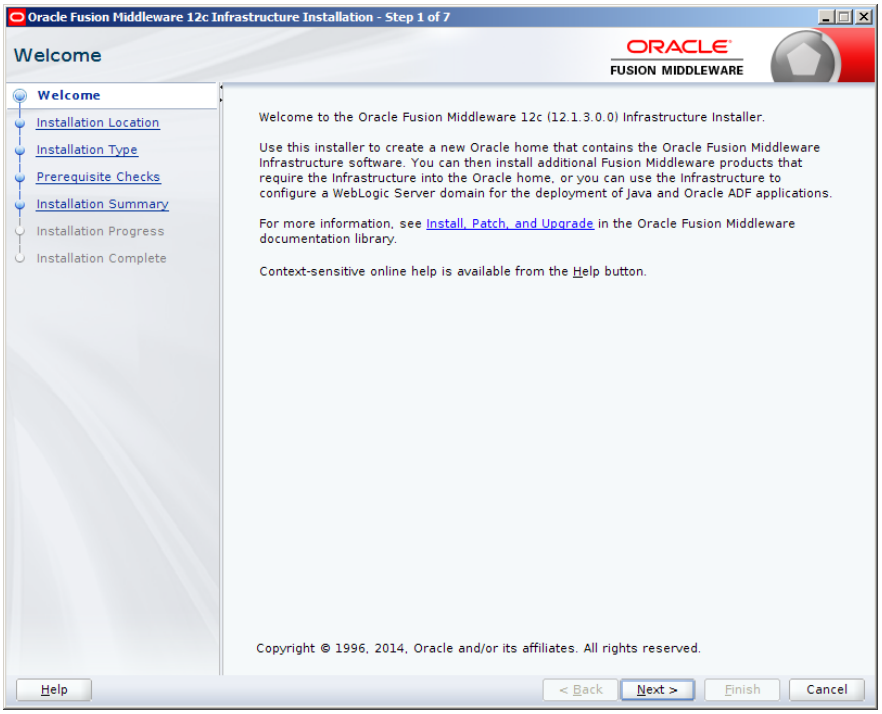
  ```
  $ export JAVA_HOME=/u00/webadmin/products/jdk_java
  $ export PATH=$JAVA_HOME/bin:$PATH
  $ export DISPLAY=<pc_ip_address:0.0>
  ```
- Run the following commands to run the installer:

  Example:

  ```
  $ cd <INSTALL_DIR>/ WLS-ADF12c_12.1.3
  $ java -jar ./fmw_12.1.3.0.0_infrastructure.jar
  ```
- The Installation Inventory Setup window displays. Update the Inventory Directory and Operation System Group fields according to the instructions on the screen and click **Next**. See screen shot for examples.



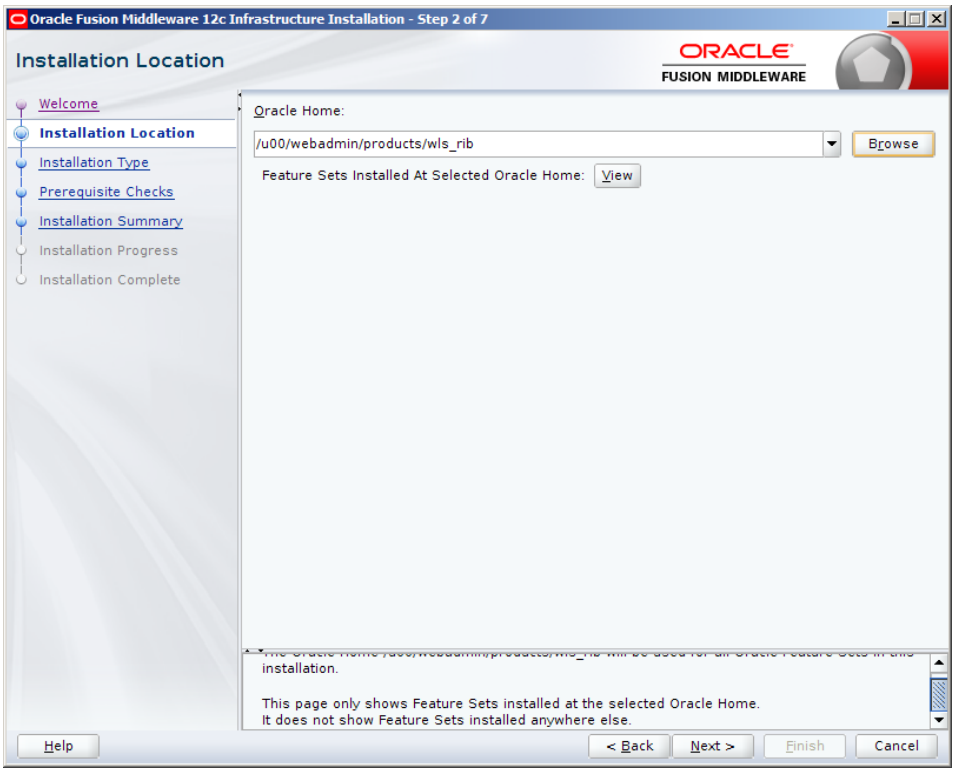
- The Welcome window of the WebLogic 12c installer displays. Click **Next**.



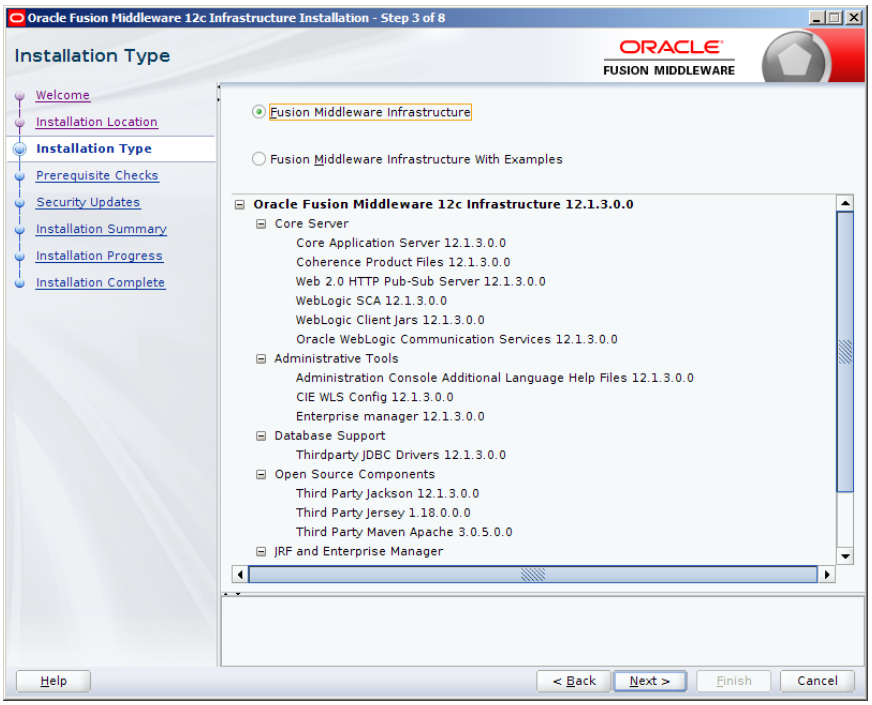
- The Installation Location window displays. Enter the path to where the ORACLE_HOME for RIB should be and click **Next**.

  Example:

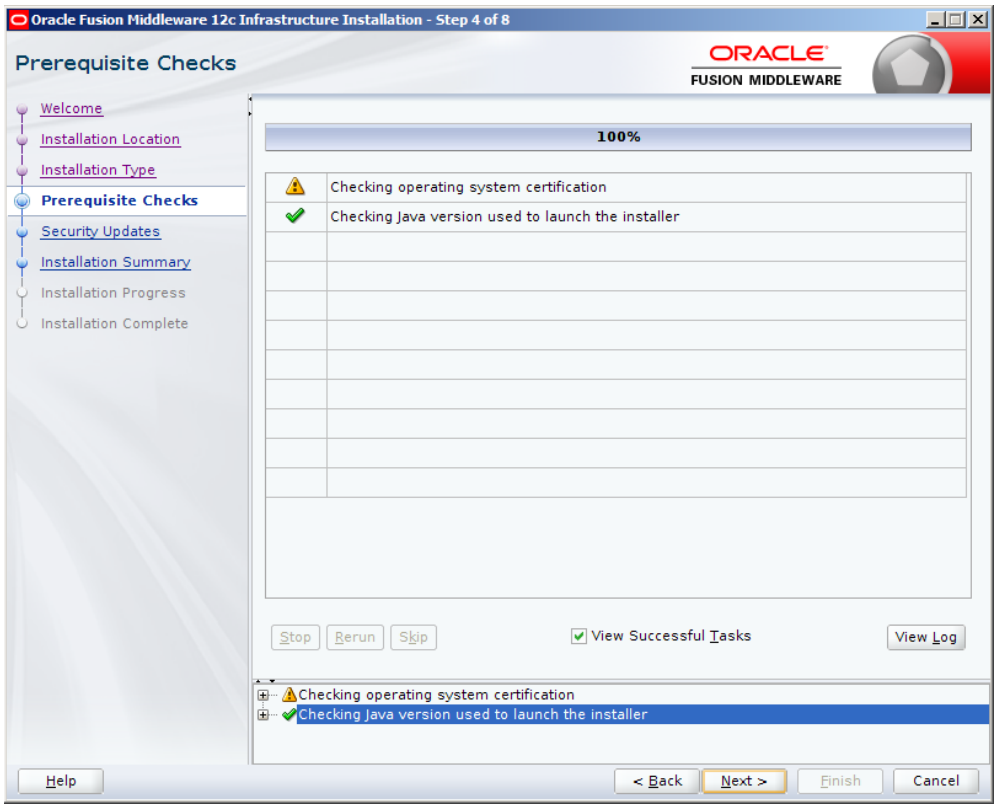
  `/u00/webadmin/products/wls_rib`



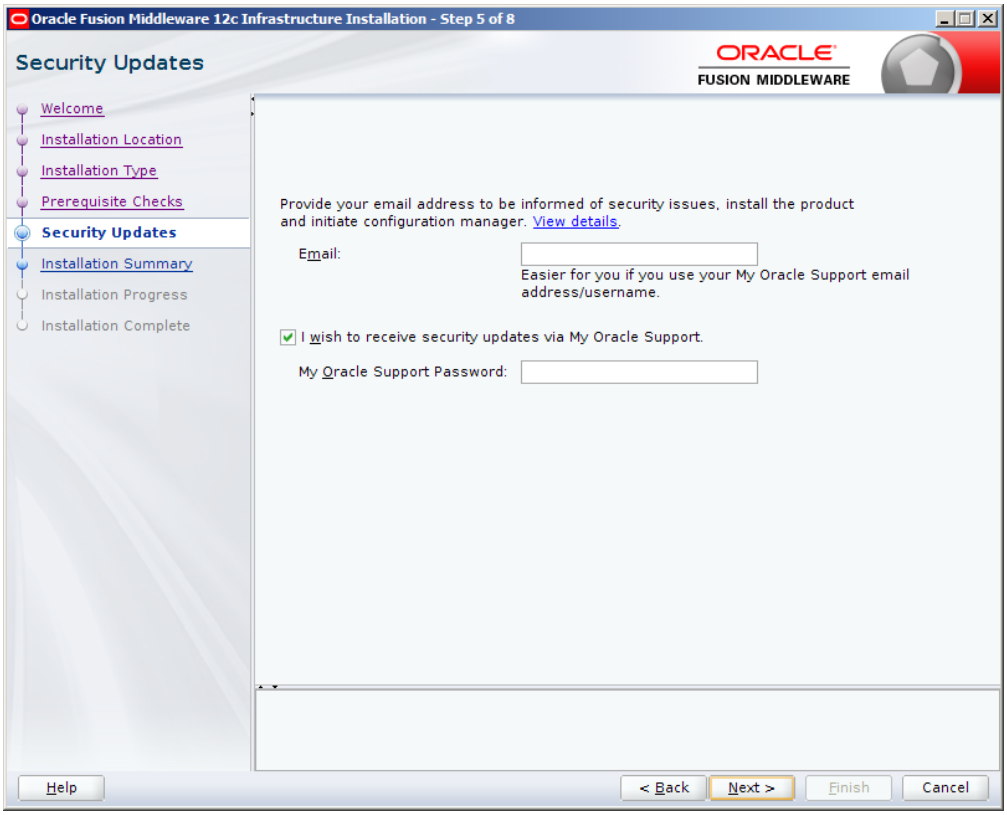
- The Installation Type window opens. Select Fusion Middleware Infrastructure and click **Next**.



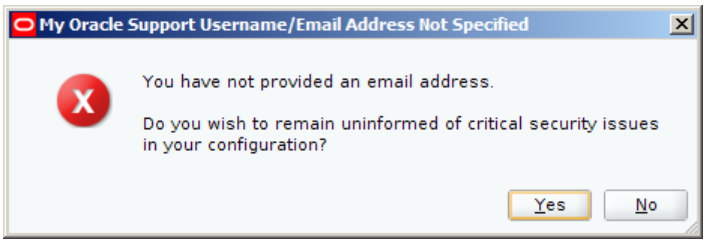
- The Prerequisite Checks window opens. After completing all the checks, click **Next**.



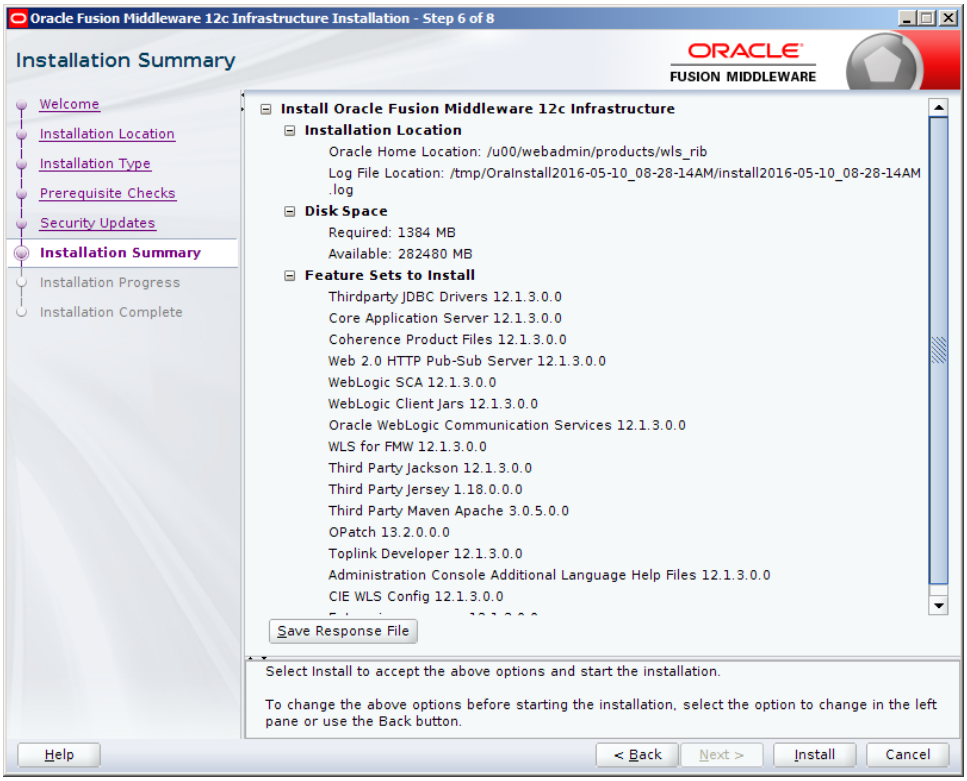
- The Security Updates window opens. Leave the screen blank, and click **Next**.



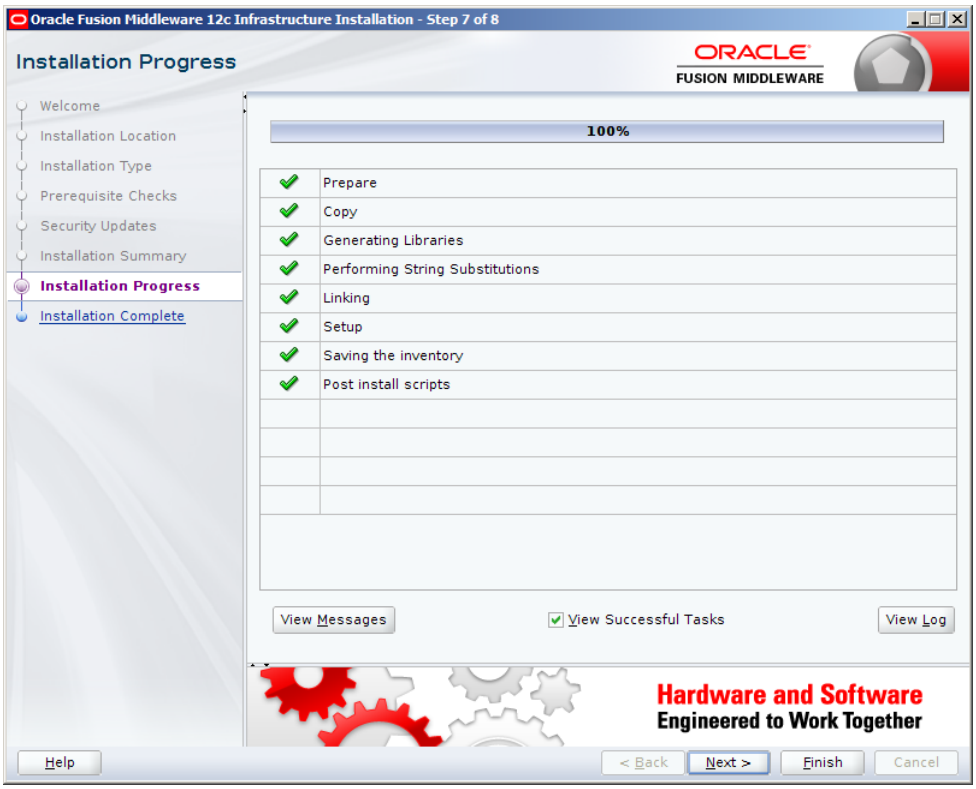
- A confirmation prompt is displayed. Click **Yes** to continue.



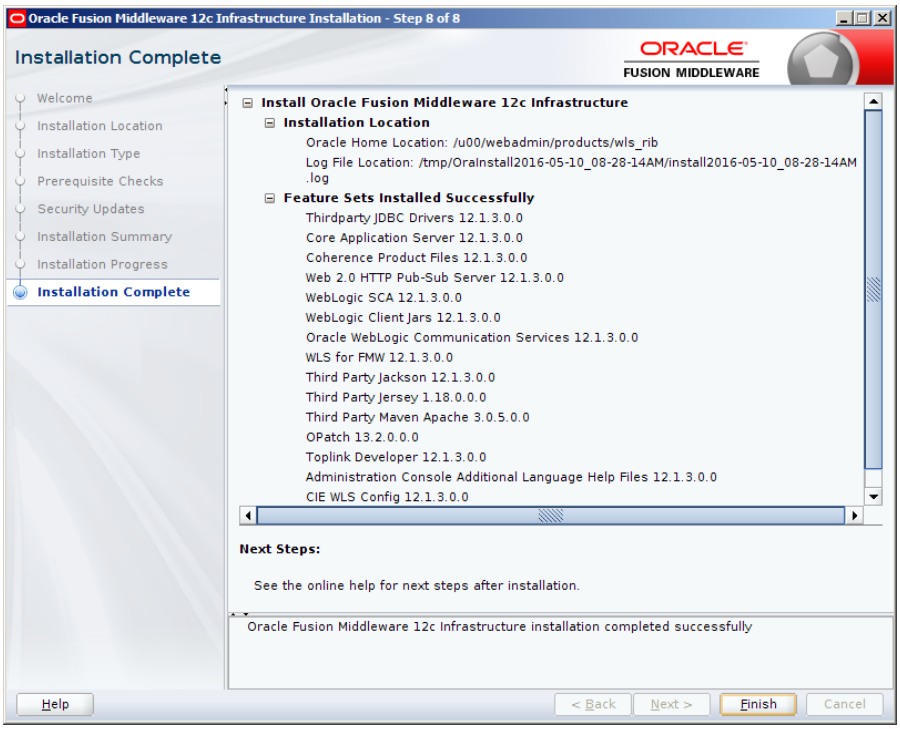
- The Installation Summary window opens. Confirm that Installation Location is correct and click **Next**.



- The Installation Progress window opens. Click **Next** once it has reached 100%.



- The Installation Complete window opens. Click **Finish**.



- Installation of WebLogic 12c is complete.

Creating Database Repository for RIBDomain

This chapter describes the steps to create the proper schemas required for the managed servers of the RIB installation. In order to run the Repository Creation Utility (RCU) - It is required to use the database credentials for the SYS user. Perform the following steps to create a database repository using Repository Creation Utility:

- Set the DISPLAY variable before running the installer:

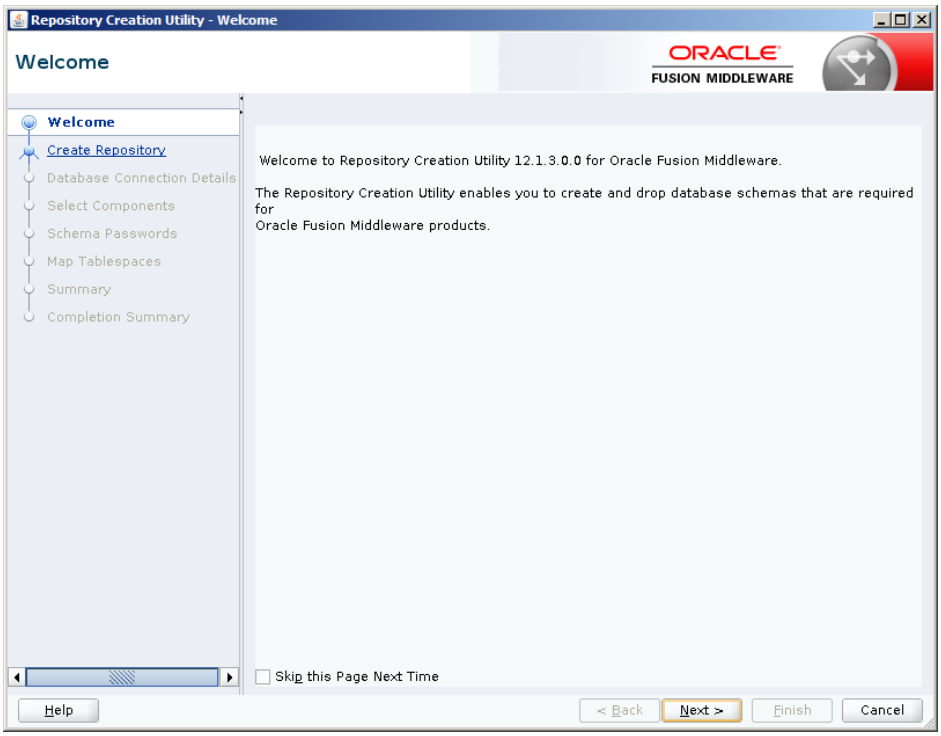
  Example:

  ```
  $ export DISPLAY=<pc_ip_address:0.0>
  ```

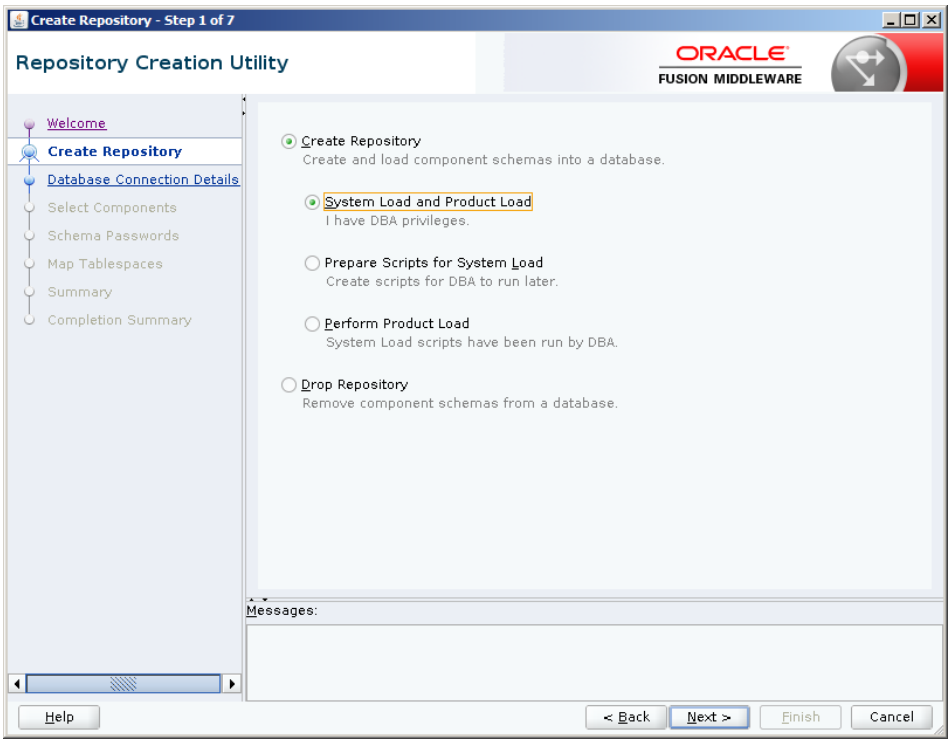
- RCU is packaged with 12c WebLogic installations so it is easy to make sure you are running the correct version of RCU. Navigate to the location of the rcu executable and run the installer by executing the following commands:

  ```
  $ cd /u00/webadmin/products/wls_rib/oracle_common/bin
  $ ./rcu
  ```

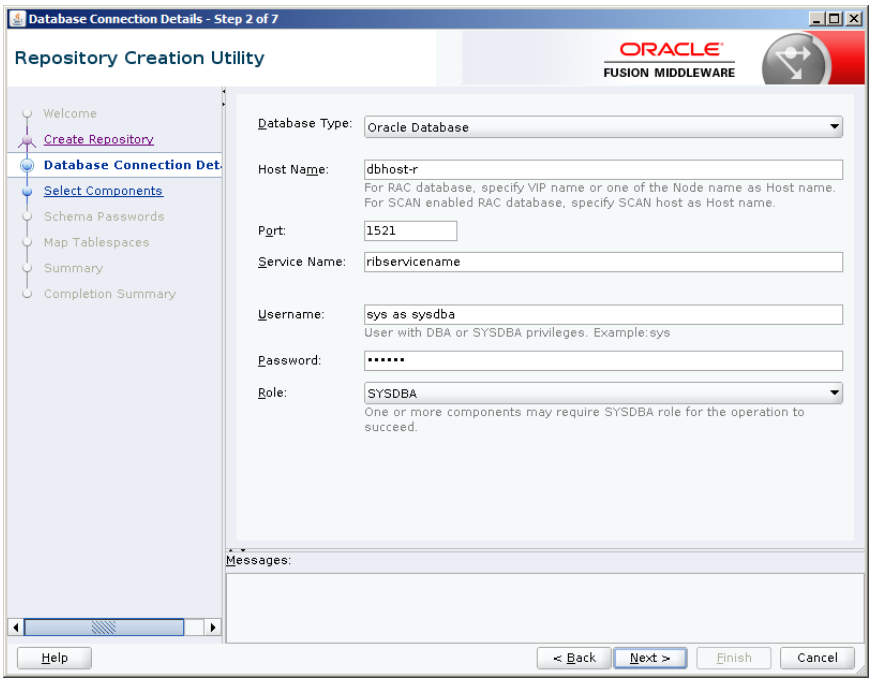
- The Welcome Window displays. Click **Next**.



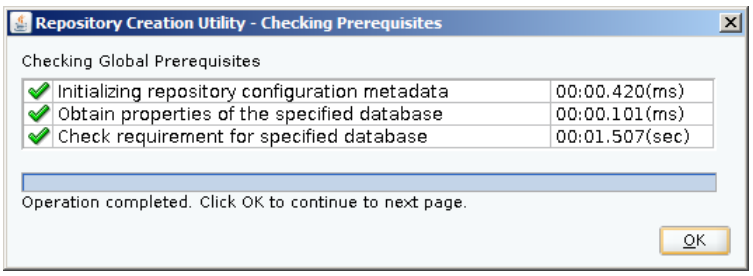
- The Create Repository window displays. Select 'Create' option and click **Next**.



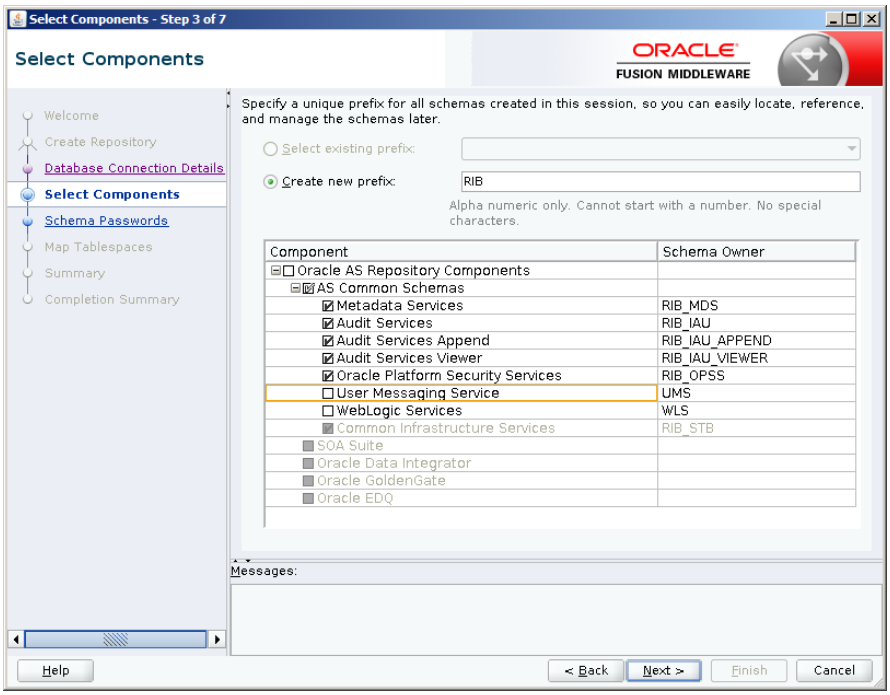
- The Database Connection Details window opens. Provide the database details and click **Next**.
  - Host Name: dbhost-r
  - Port: 1521
  - Service Name: ribservicename
  - Username (must have DBA or SYSDBA privileges)
  - Password



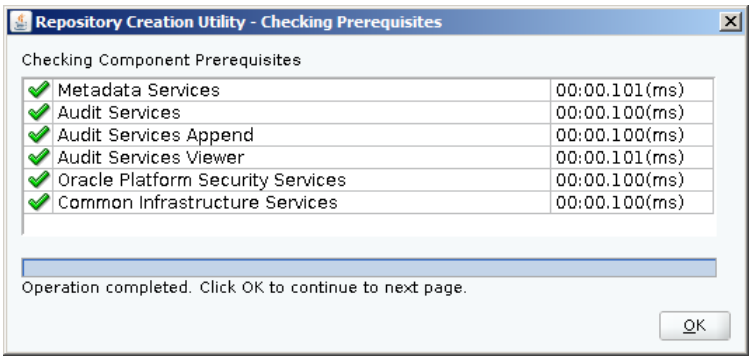
- RCU checks database prerequisites. Click **OK**.



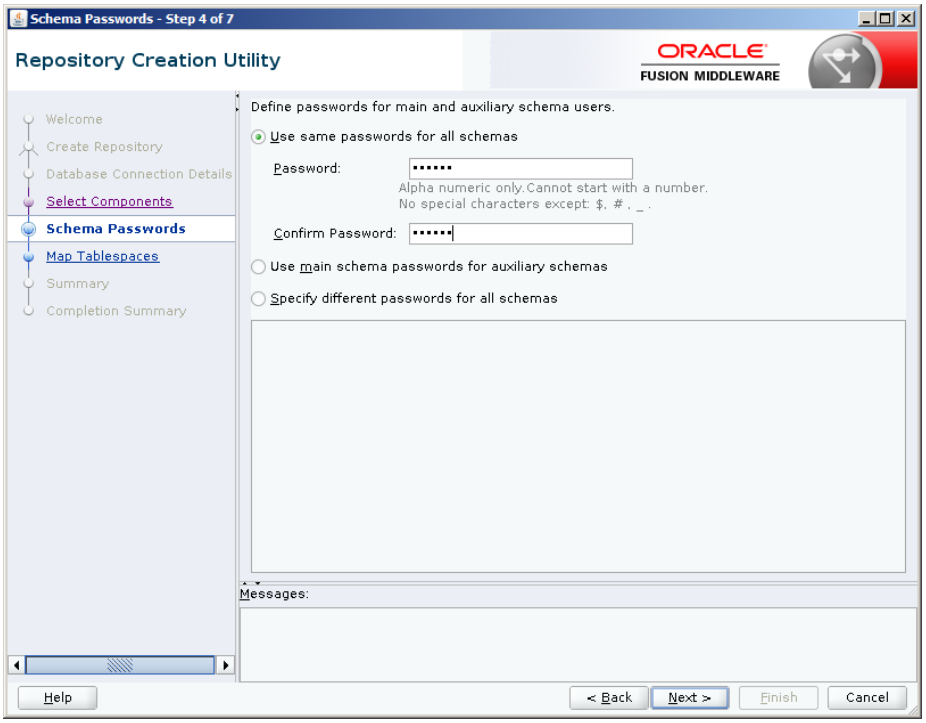
- The Select Components window opens. Select just the following items below and click **Next**.
  - Select Create a new Prefix: RIB
  - Select the following components:
    a. Metadata Services
    b. Audit Services
    c. Audit Services Append
    d. Audit Services Viewer
    e. Oracle Platform Security Services
    f. Common Infrastructure Services



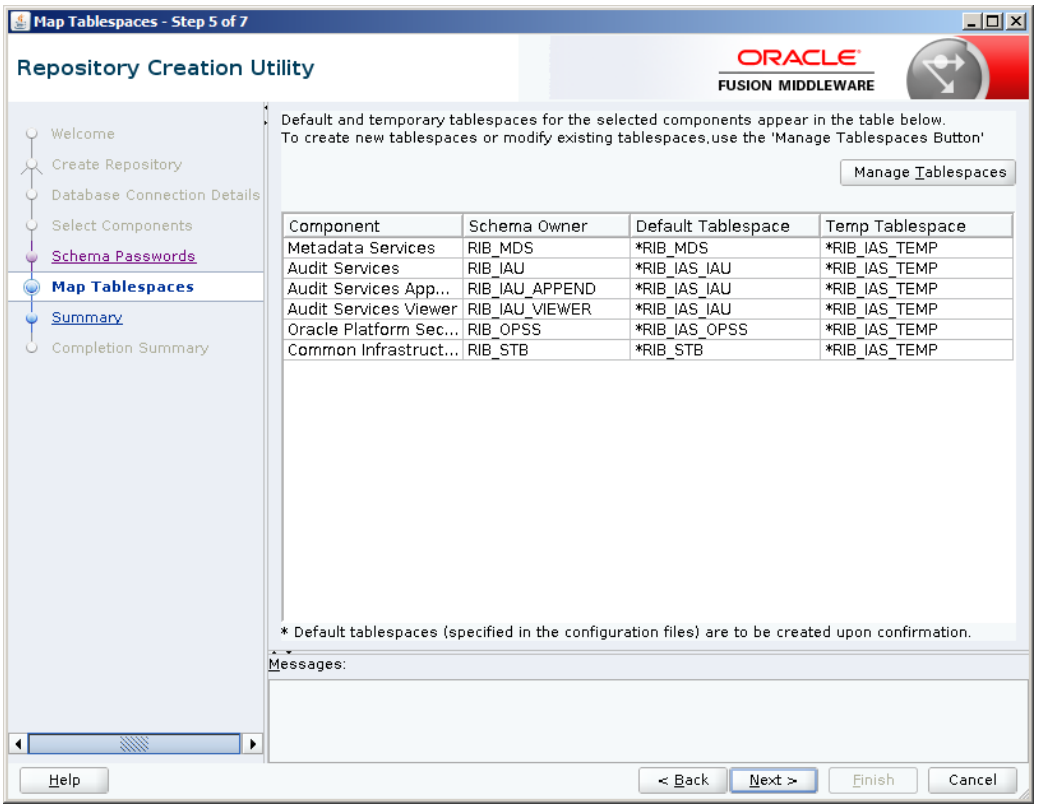
- RCU will check database prerequisites to make sure it can create the selected components. Click **OK**.



- The Schema Passwords window opens.
  - Select the Use same passwords for all schemas option.
  - Enter in the password.
  - Click **Next**.

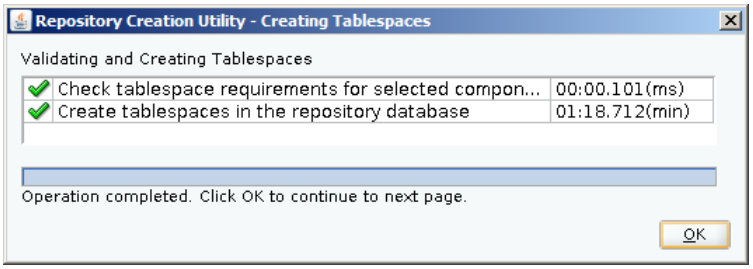


- The Map Tablespaces window opens. Click **Next**.

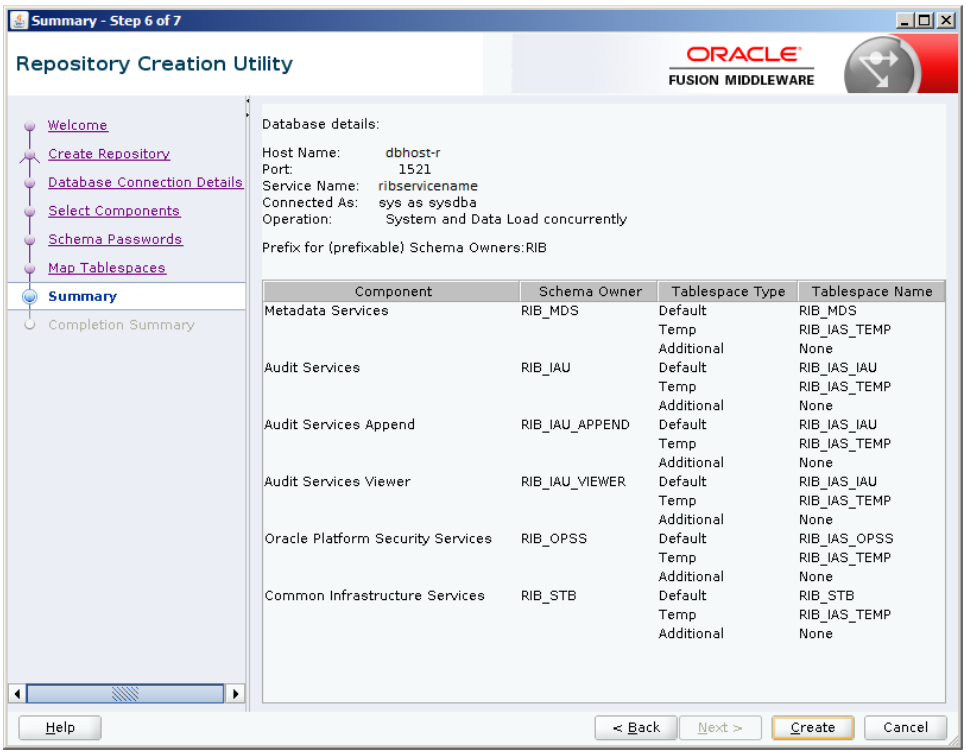


- The Repository Creation Utility - Confirmation window displays. Click **OK**.

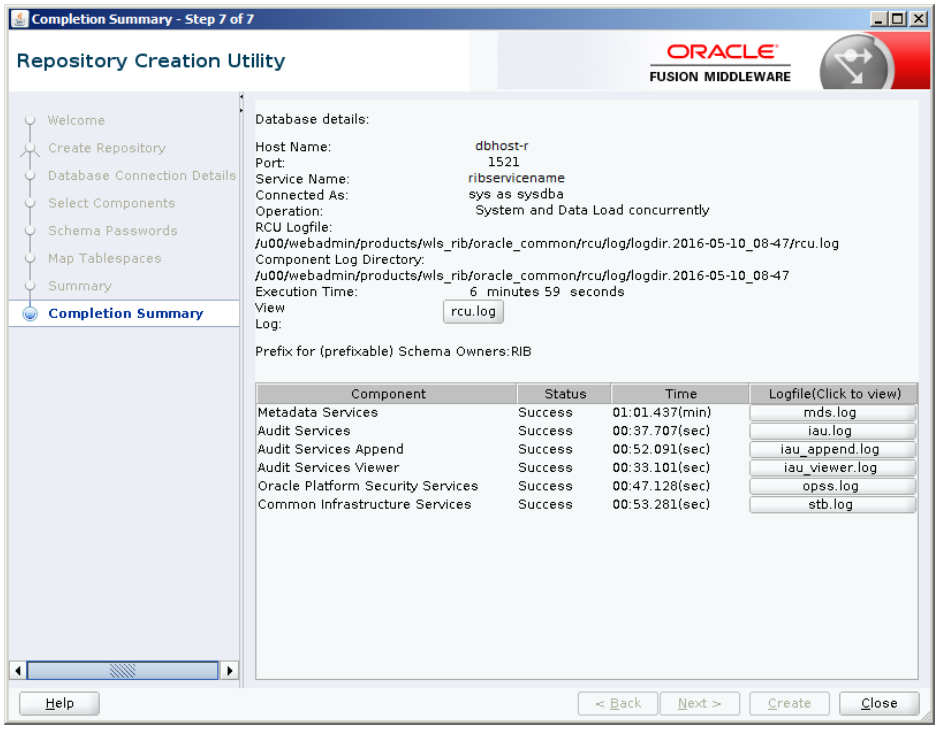
- The Repository Creation Utility – Creating Tablespaces window displays. Click **OK**.



- The Summary window opens. Click **Create**.



- The Completion Summary window opens. Click **Close**.



- The RCU schema creations are complete.

Configure RIBDomain on ribhost1

This section describes the steps to configure the admin and managed servers for the RIB. The steps should only be performed once on ribhost1. The configuration will target directories on the shared drive between ribhost1 and ribhost2.

Perform the following steps to configure RIB admin and managed servers on ribhost1 only.

- Set the following variables by exporting them before running the installer so that the JAVA_HOME is added to the PATH variable:
  - JAVA_HOME
  - PATH
  - DISPLAY

  Example:

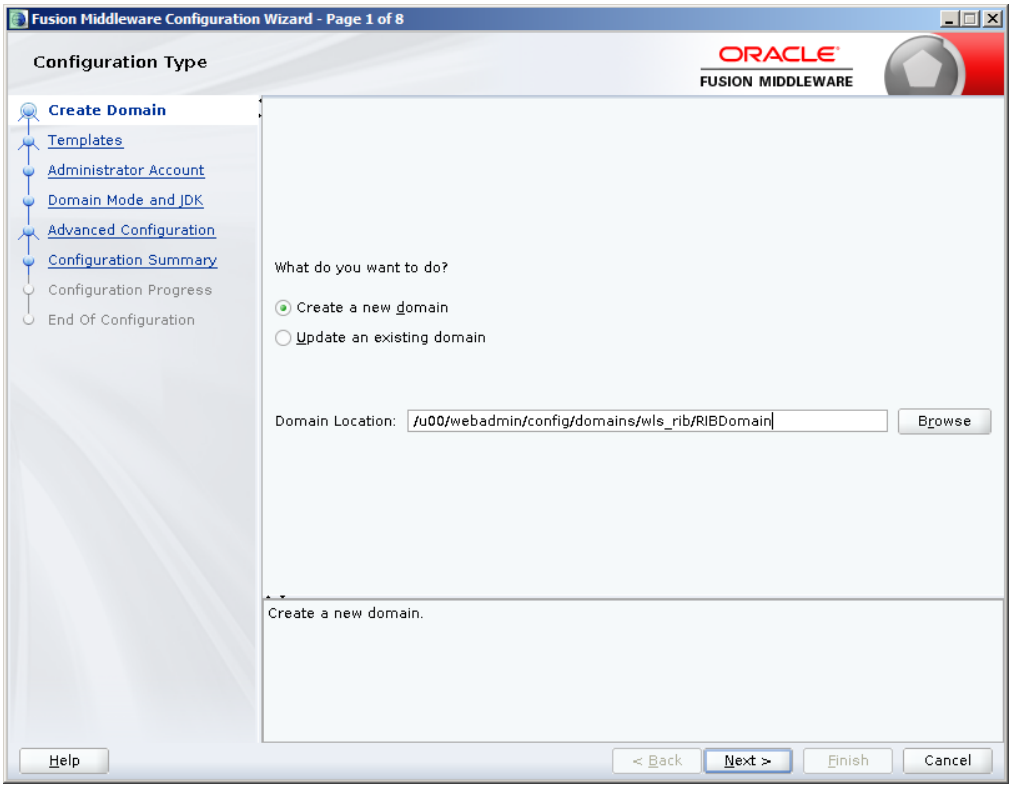
  ```
  $ export JAVA_HOME=/u00/webadmin/products/jdk_java
  $ export PATH=$JAVA_HOME/bin:$PATH
  $ export DISPLAY=<pc_ip_address:0.0>
  ```

- Navigate to the Navigate to <RIB_WEBLOGIC_HOME>/oracle_common/common/bin and run config.sh:

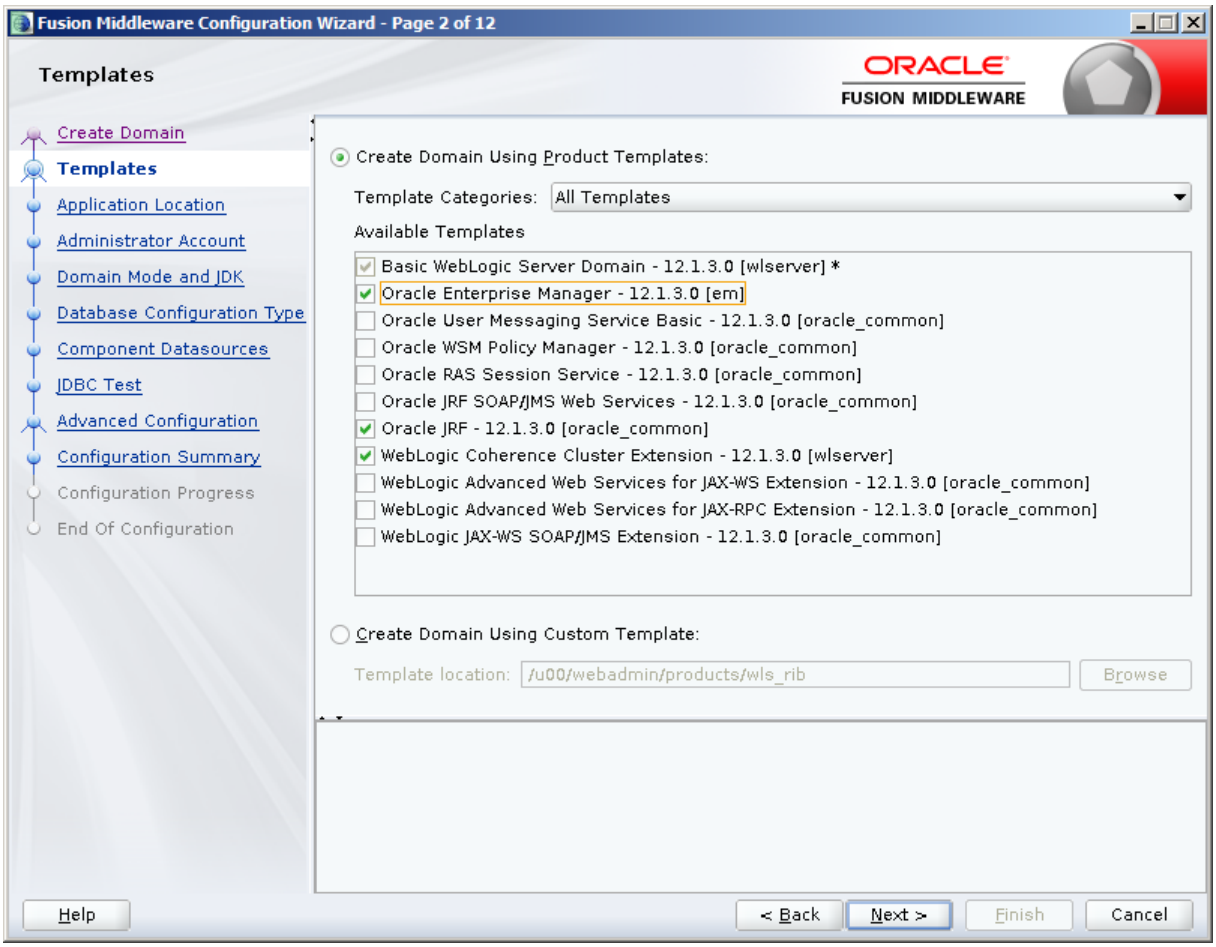
  Example:

  ```
  $ cd /u00/webadmin/products/wls_rib/wlserver/common/bin
   $ ./config.sh
  ```

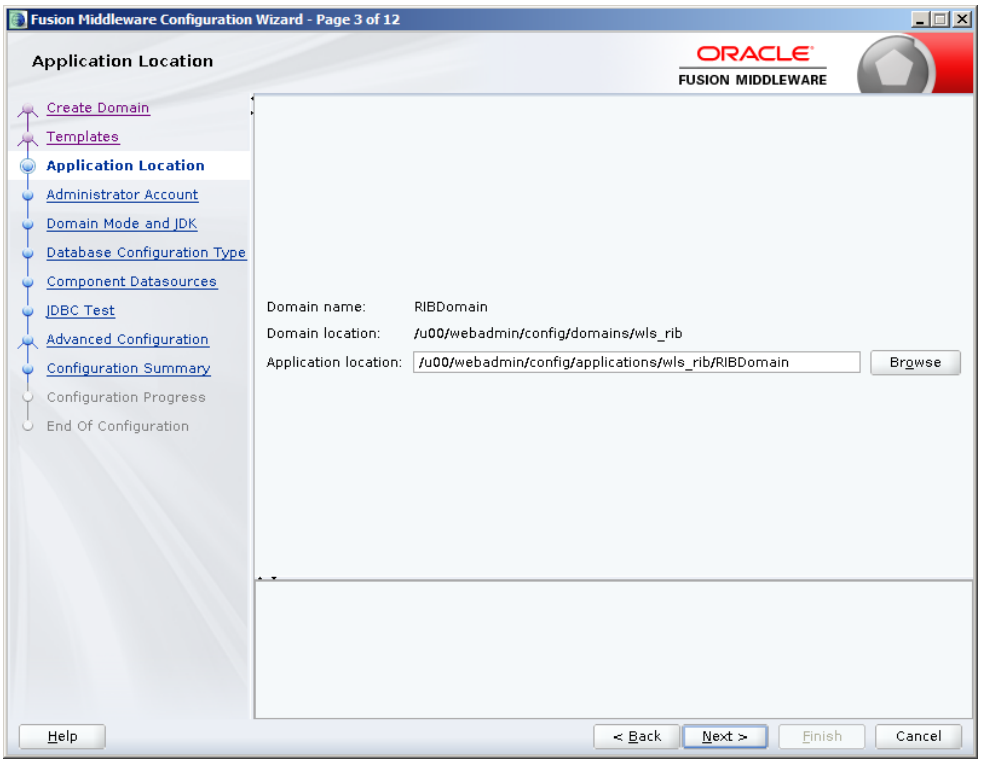
- The Configuration Type window opens. Update the screen as follows and click **Next**.
  - Select **Create a new domain.**
  - Domain Location: /u00/webadmin/config/domains/wls_rib/RIBDomain



- The Templates window opens. Select the following components and click **Next**.
  - Oracle Enterprise Manager – 12.1.3.0 [EM]
  - Oracle JRF – 12.1.3.0 [oracle_common]
  - WebLogic Coherence Cluster Extension – 12.1.3.0 [wlsserver]



- The Application Location window opens. Update the Application Location as follows and click **Next**.
  - Application Location: /u00/webadmin/config/applications/wls_rib/RIBDomain



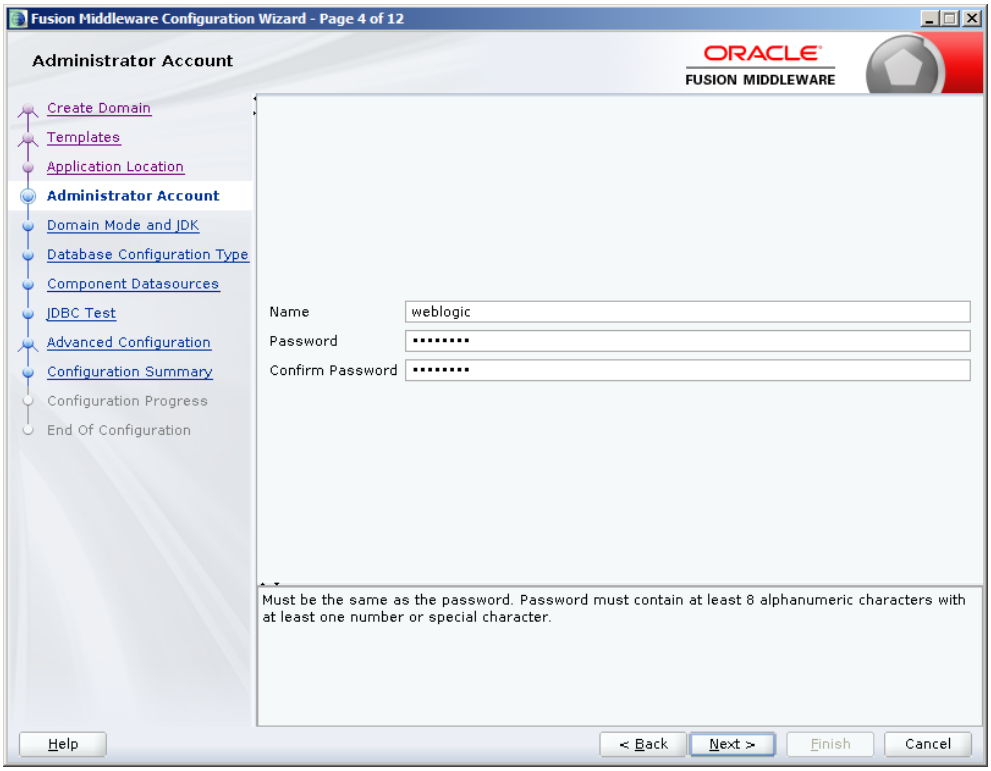
- The Administrator Account window opens. Provide the WebLogic administrator users credentials and click **Next**.

  Example:

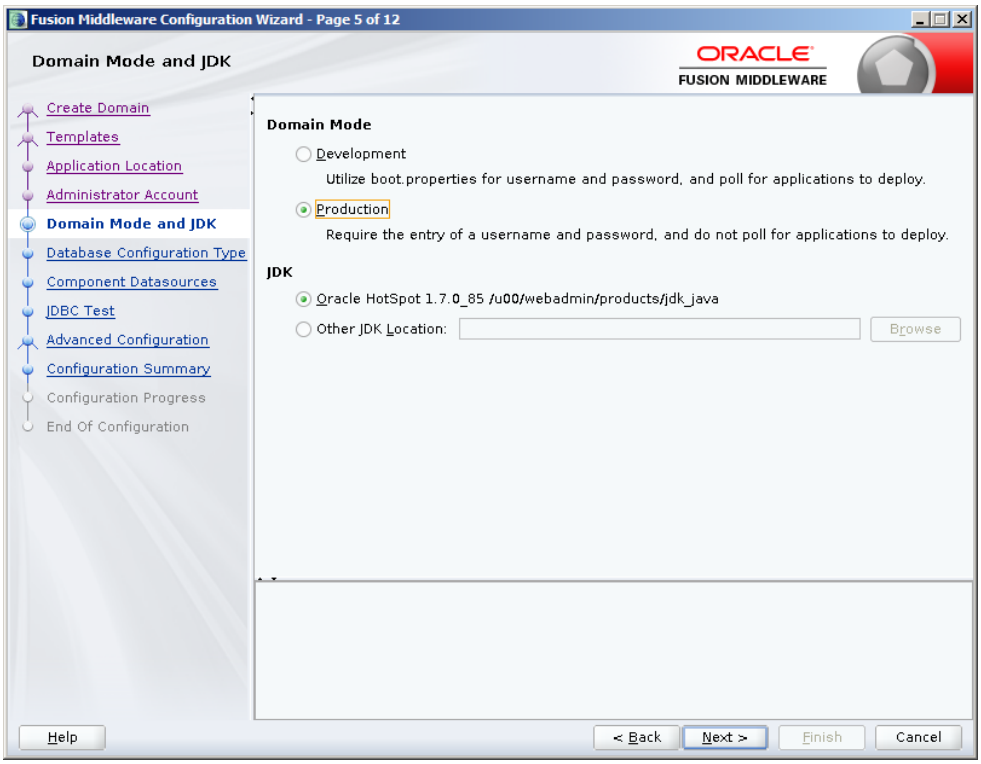
  – Name: weblogic

  – Password: <password>

  – Confirm Password: <password>

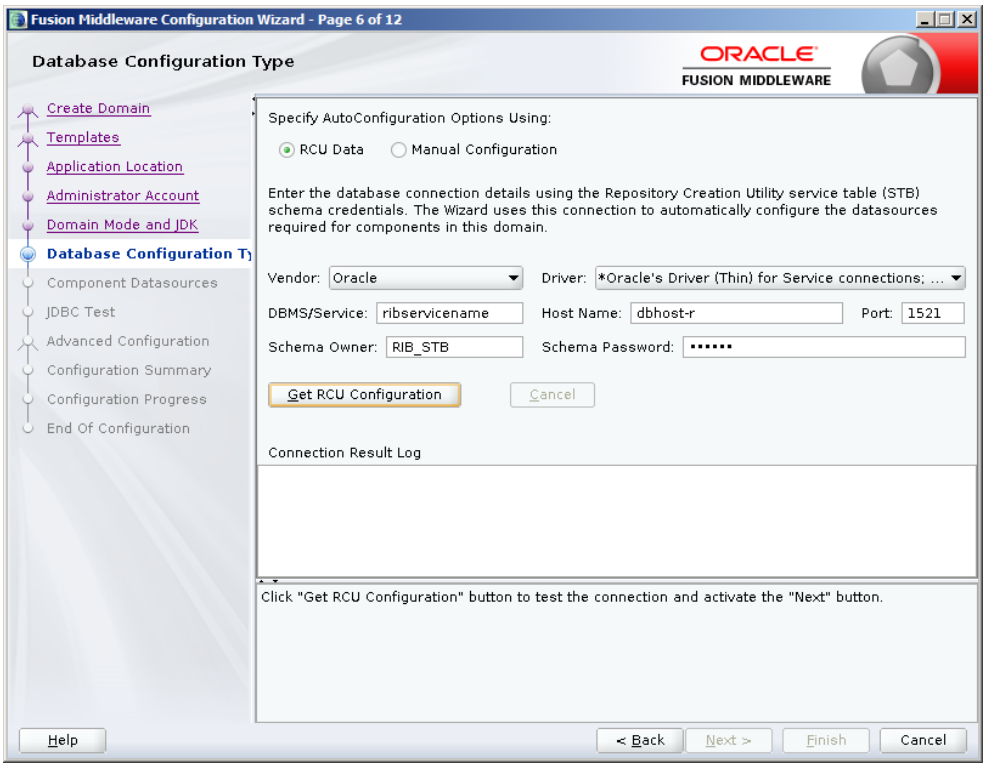


- The Domain Mode and JDK window opens. Select the Production Mode and the appropriate JDK options. Click **Next**.

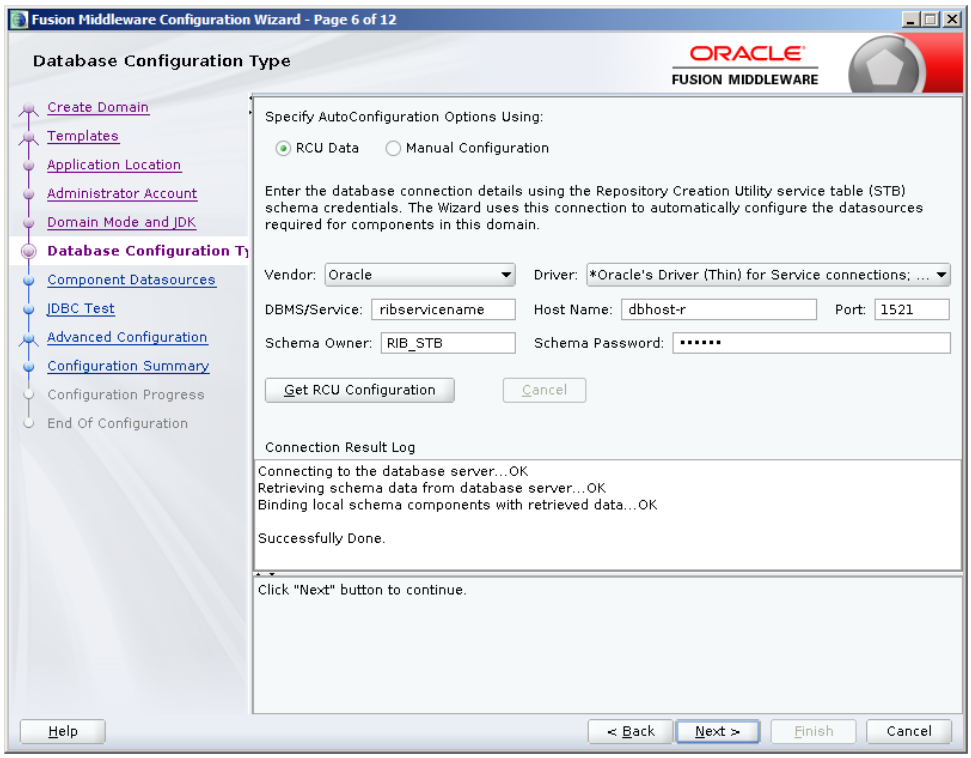


- The Database Configuration Type window opens. Update it as follows and click the Get RCU Configuration button.

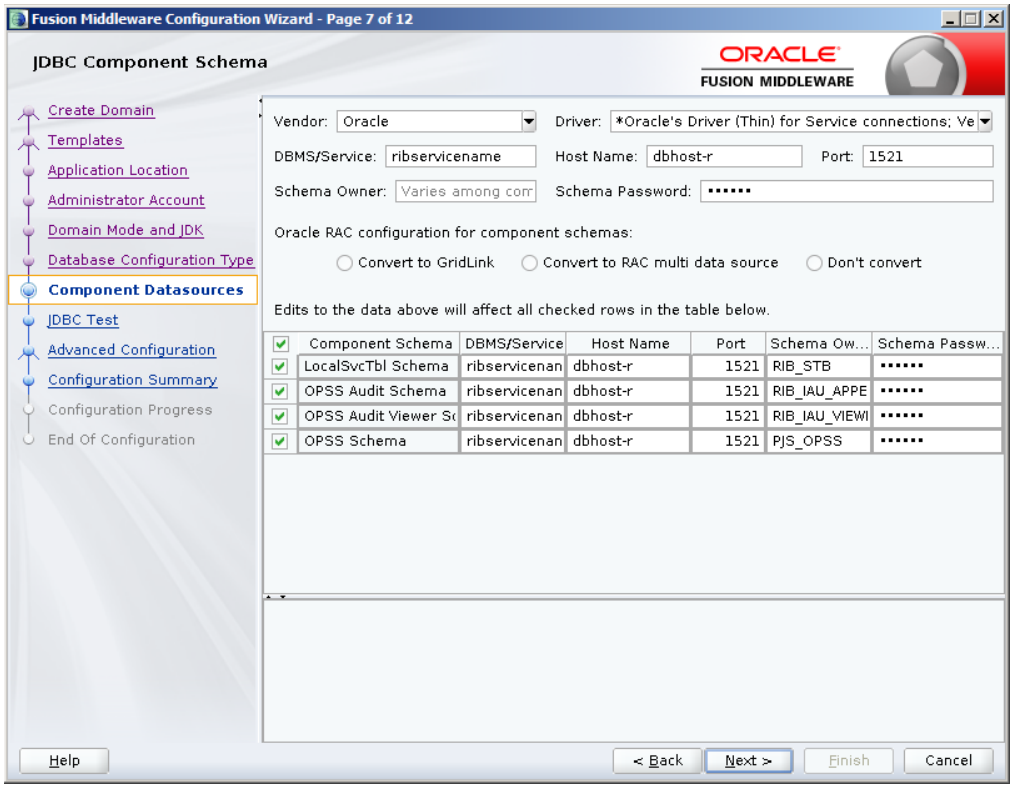
- Driver: *Oracle's Driver (Thin) for Service connections; Versions 9.0.1 and later
- DBMS/Service: ribservicename
- Host Name: dbhost-r
- Port: 1521
- Schema Owner: RIB_STB
- Schema Password: <password>



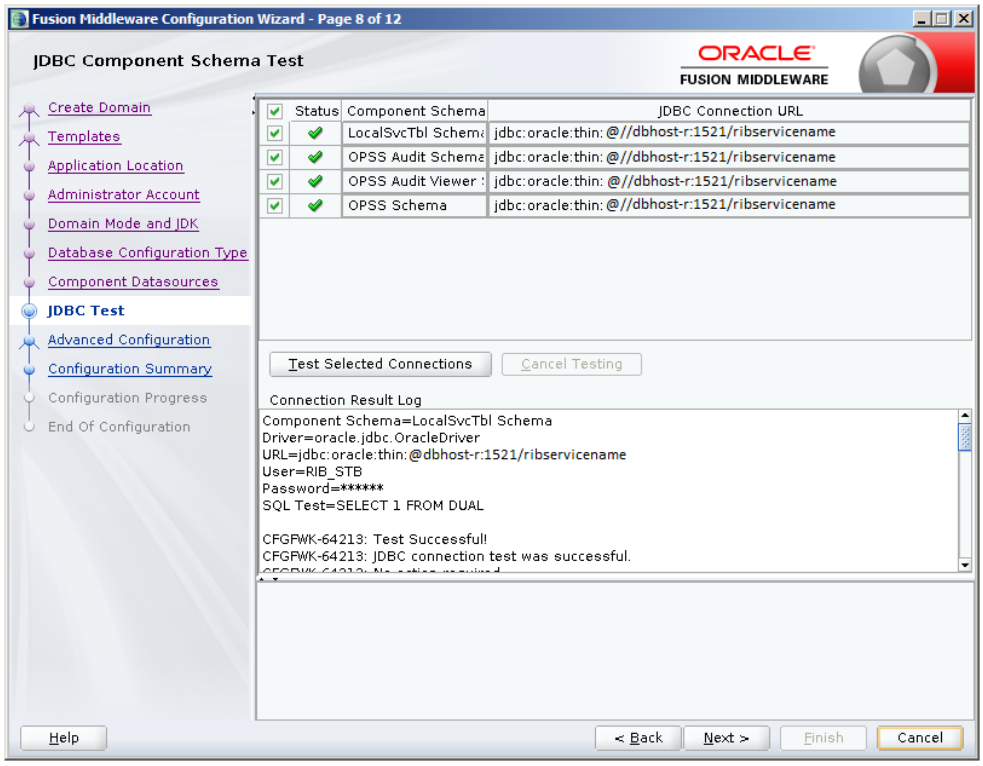
- After the screen validates the entries on the screen the **Next** button is enabled. Click **Next**.



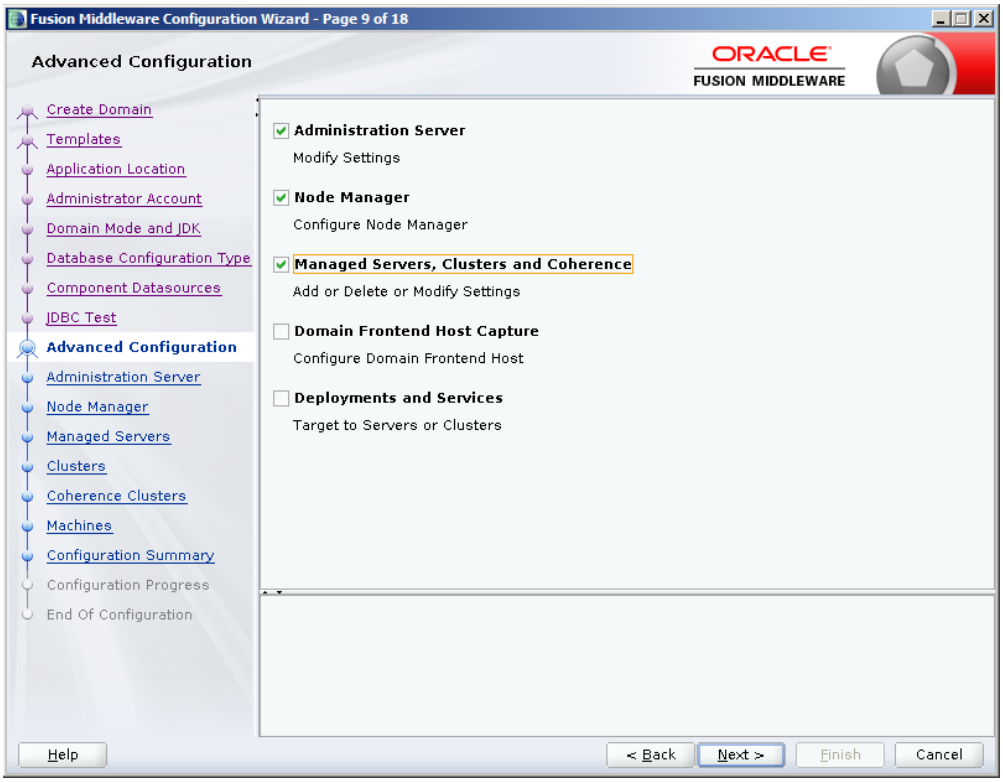
- The JDBC Component Schema is displayed. Review the entries and click **Next**.



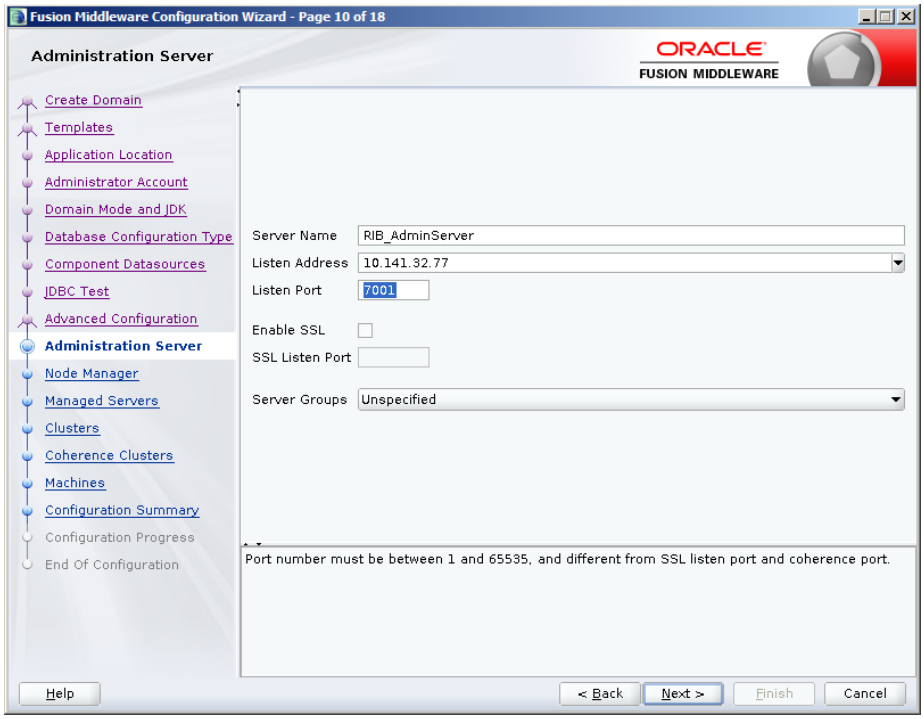
- The JDBC Component Schema Test window opens. Once all the connection tests are successful, click **Next**.



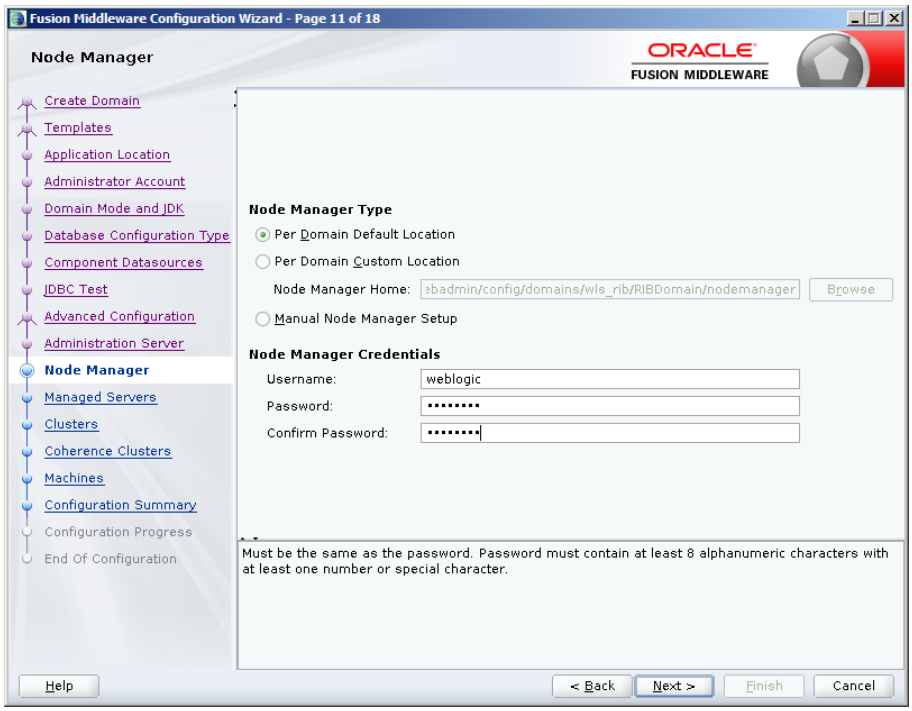
- The Advanced configuration window opens. Select the following and click **Next**.
  - Administration Server
  - Node Manager
  - Managed Servers, Clusters and Coherence



- The Administration Server window opens. Provide a Server Name, Listen Address, and port. The Listen address will not be the ribhost1 alias or IP address of ribhost1 – it has to be floating IP address that can be shared between ribhost1 and ribhost2. See screen shot for examples. Click **Next**.



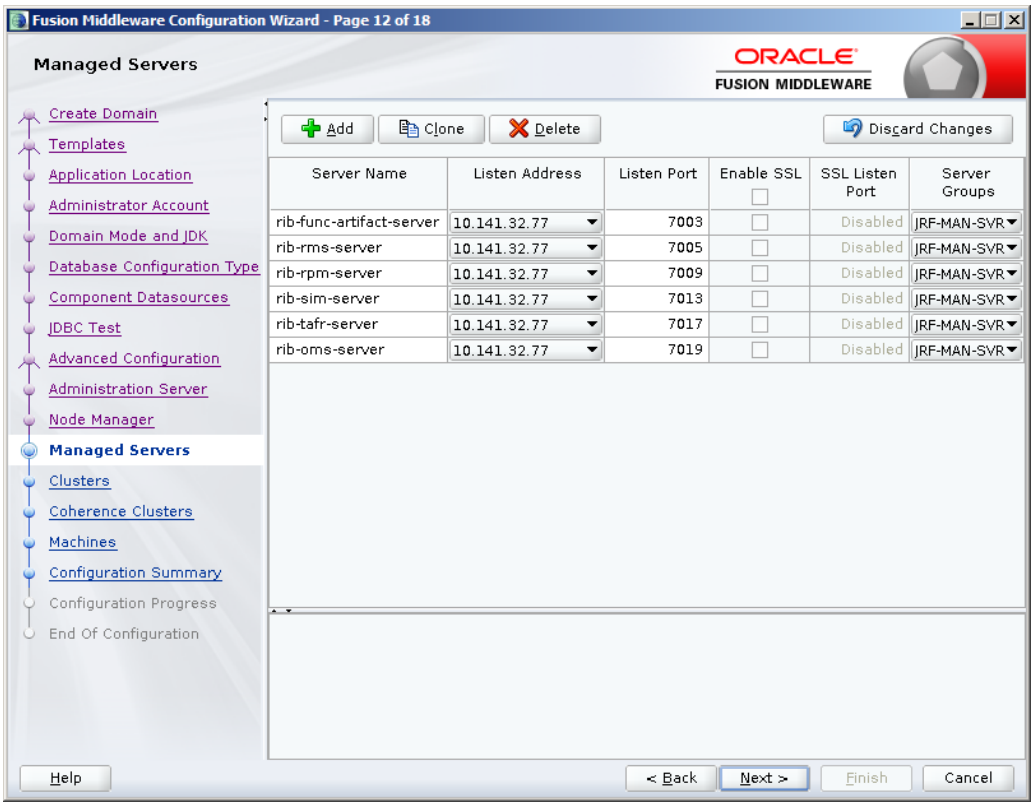
- The Node Manager window opens. Select **Per Domain Default Location** as a Node Manager Type and provide the Node Manager Credentials. Click **Next**.



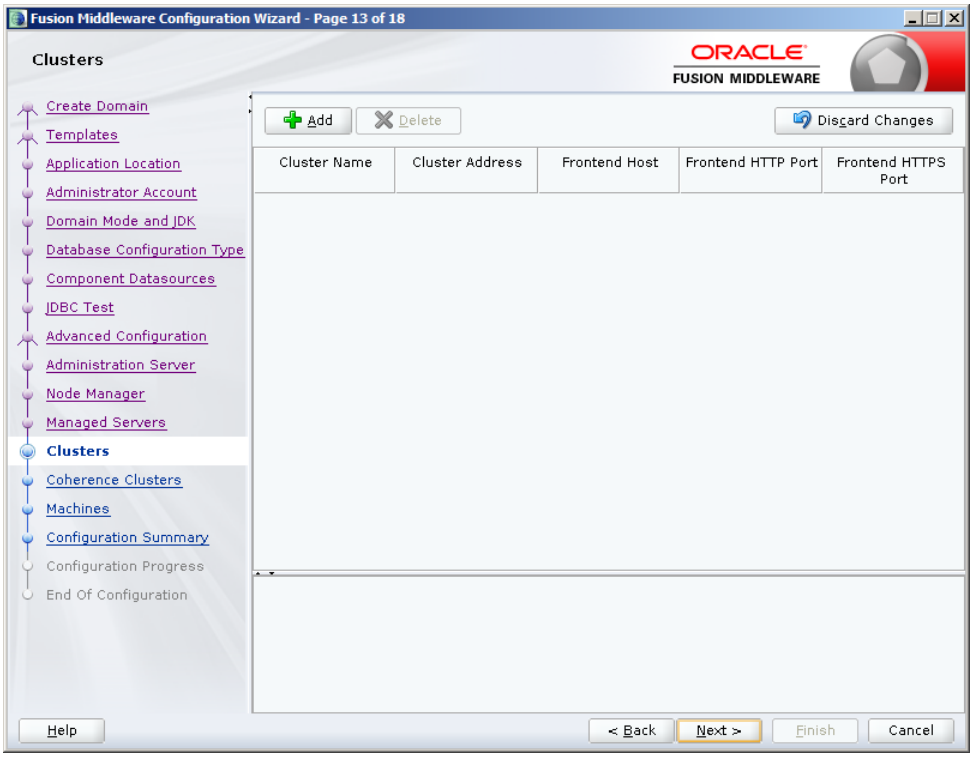
- The Managed Servers window opens. Using the Add button create an entry for each of the managed server and port specification shown in the screen shot

below. Make sure to specify the floating ip address for the Listen Address and Select JRF-MAN-SRV for the Server Groups. Click **Next**.

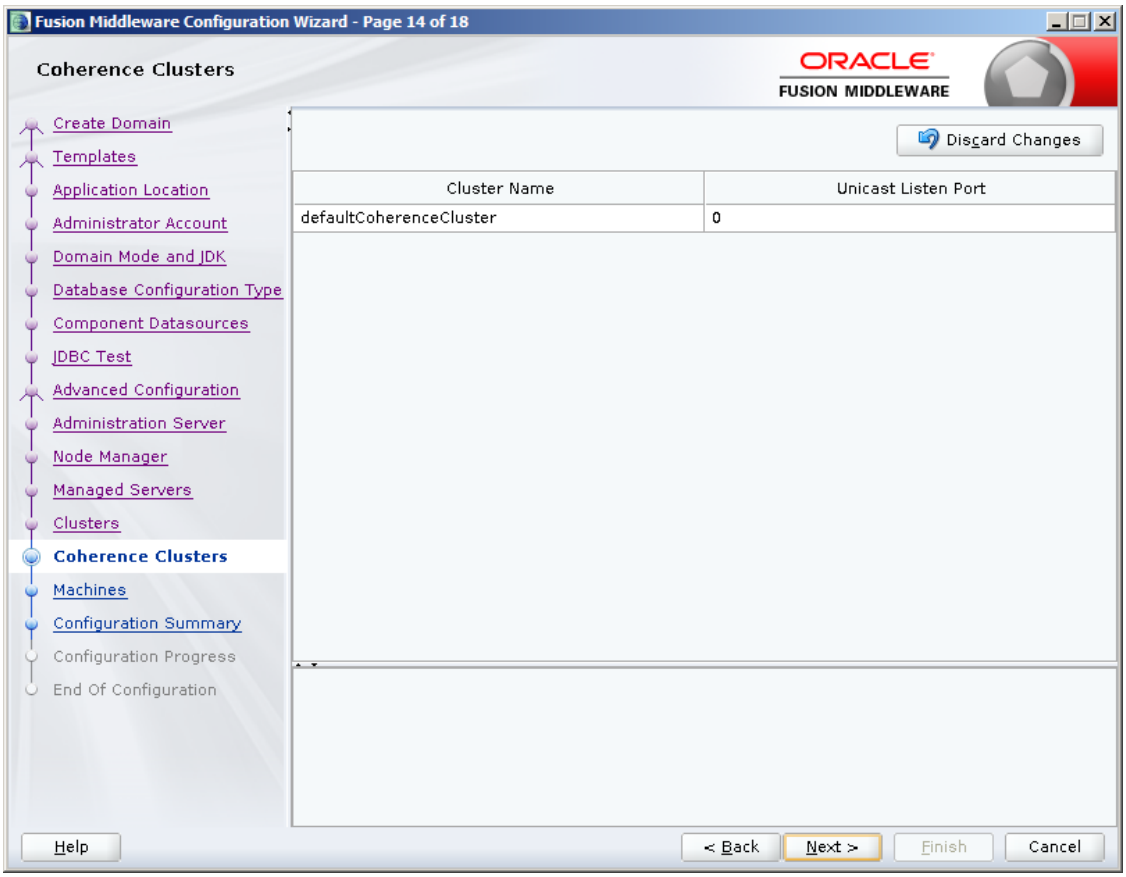
> **NOTE**: Server ports can be anything but they do have dependencies in other applications so be careful to consider that when creating them. The ports used in this document are in alignment with RSB and RGBU Application ports used in examples from other 15.0.1 HA documents.



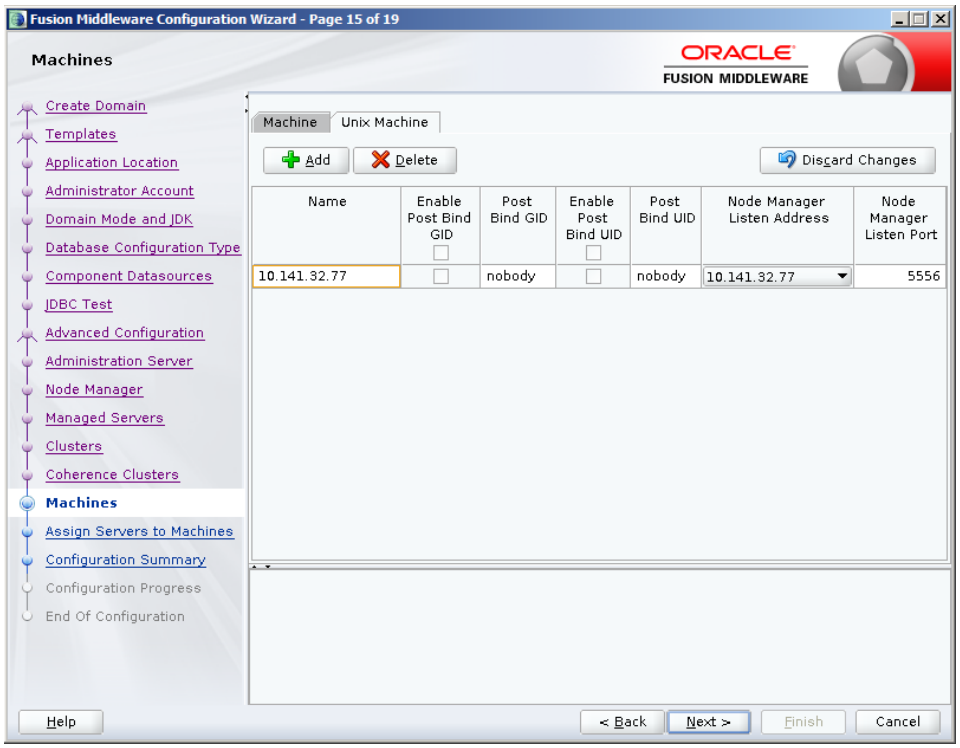
- The Clusters window opens. Click **Next**.



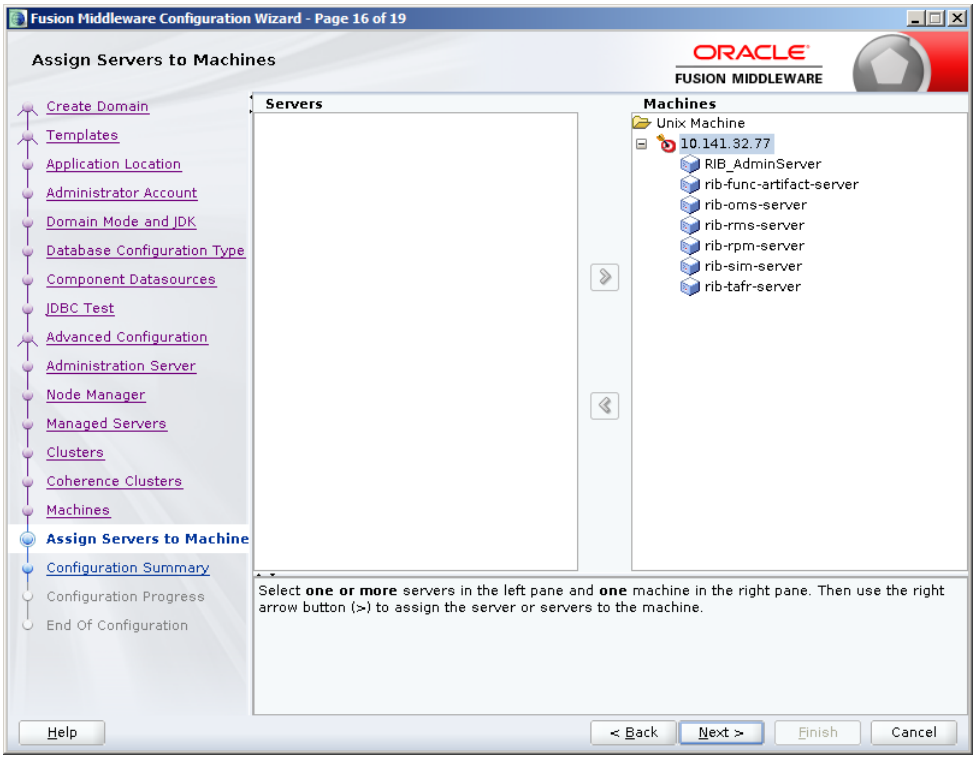
- The Coherence Clusters window opens. Click **Next**.



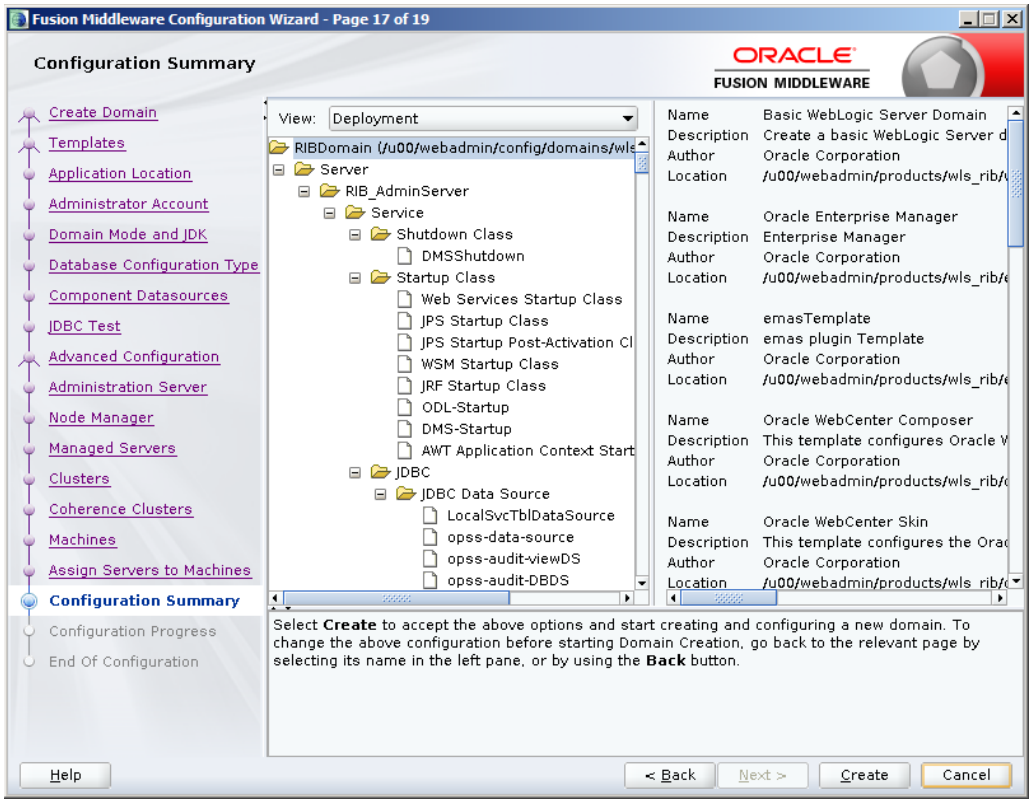
- The Machines window opens. Make sure to click the **Unix Machine** tab so the screen looks like the one shown below. Provide the following details and click **Next**.
  - Name: This is usually named after the host – Use floating IP address.
  - Node Manager Listen Address: This is usually named after the host. Use floating IP address.
  - Node Manager Listen Port.



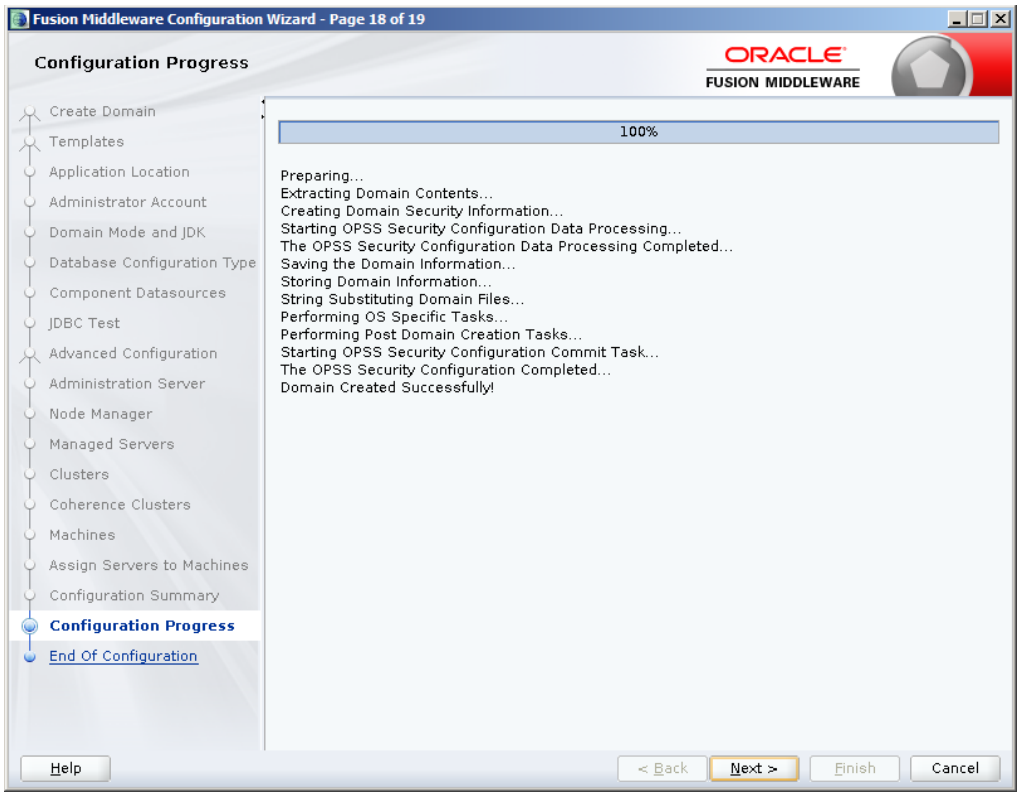
- The Assign Servers to Machines window opens. Assign all of the servers to the Unix Machine, and click **Next**.



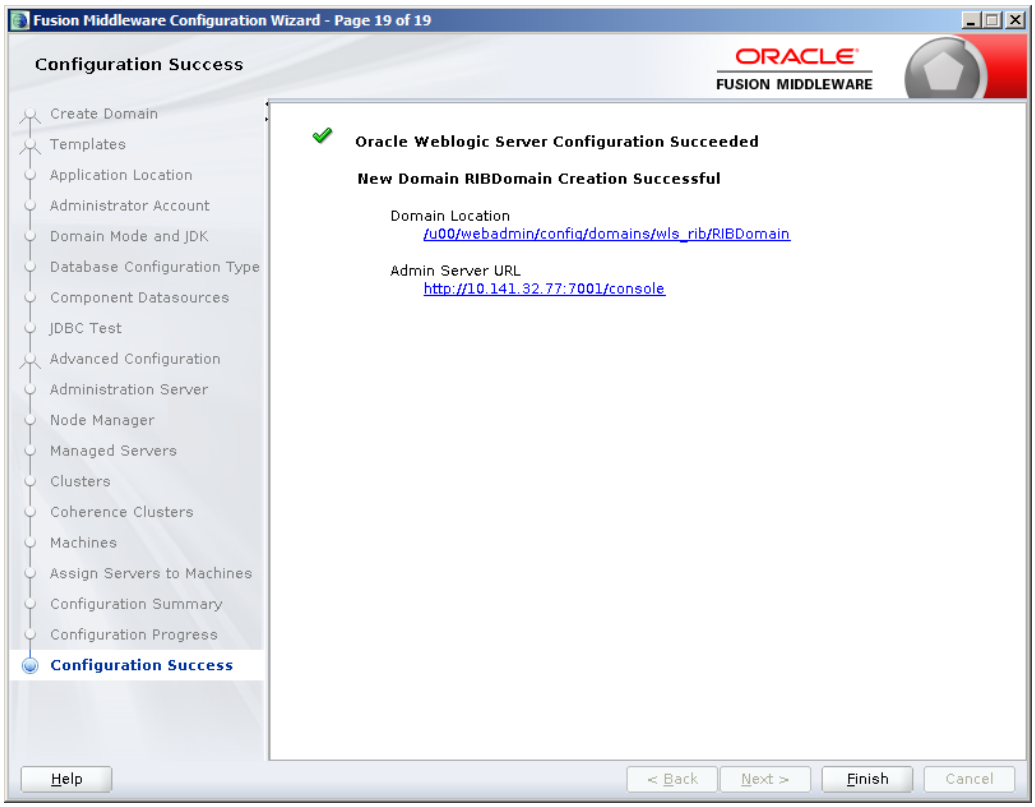
- The Configuration Summary window opens. Click **Create**.



- The Configuration Progress window opens. Click **Next**.



- The Configuration Success window opens. Note down the Admin Server URL, and click **Finish**.



- RIBDomain configuration setup is complete.

Start WLS and Managed Servers - Create boot.properties for RIBDomain

Start the Node Manager, WebLogic admin server and create WebLogic boot.properties file with username and password so that the startup script does not prompt or require user input to start the admin server.

- Navigate to <RIB_DOMAIN_HOME>/bin and run the startNodeManager.sh script to start the Node Manager.

  Example:

  ```
  $ cd /u00/webadmin/config/domains/wls_rib/RIBDomain/bin
  $ nohup ./startNodeManager.sh &
  ```

  Create WebLogic boot.properties as follows:

- On ribhost1 - navigate to <RIB_DOMAIN_HOME>/bin and run the startWebLogic.sh script. Enter in the admin server name user and password when prompted

  Example:

  ```
  $ cd /u00/webadmin/config/domains/wls_rib/RIBDomain/bin
  $ ./startWebLogic.sh
  username=weblogic
  password=<password>
  ```

- Shut down the admin server.
- Navigate to <RIB_DOMAIN_HOME>/servers/AdminServer/security. If the security directory does not yet exist – create it.
- Create a new file called boot.properties and add the WLS admin username and password credentials.

  Example:

  ```
  $ cd
  /u00/webadmin/config/domains/wls_rib/RIBDomain/servers/AdminServer/securit
  y
  $ vi boot.properties
  username=weblogic
  password=<password>
  ```

- Start the WebLogic admin server.

  Example:

  ```
  $ cd /u00/webadmin/config/domains/wls_rib/RIBDomain/bin
  $ nohup ./startWebLogic.sh &
  ```

- Log onto the WebLogic admin server and start the managed servers using the following URL with your floating IP address: http://<floating_ip>:7001/console

Verification – Verify the Active/Passive VIP Configuration for RIBDomain

The active/passive configuration is setup during host creation along with the configuration of a shared mount between the hosts. Testing it here against the RIB application will ensure that it is setup correctly at the host level and that the RIBDomain configuration was setup correctly so that that the application will be reachable when either ribhost1 or ribhost2 is down.

- Get current configuration of ribhost1 and ribhost2 by running the following command on both hosts to make sure you can identify the floating IP is on one of the hosts. The floating IP is identifiable by highlighted text in the example output as eth0:1

  Example:

  ```
  $ /sbin/ifconfig -a
  ```

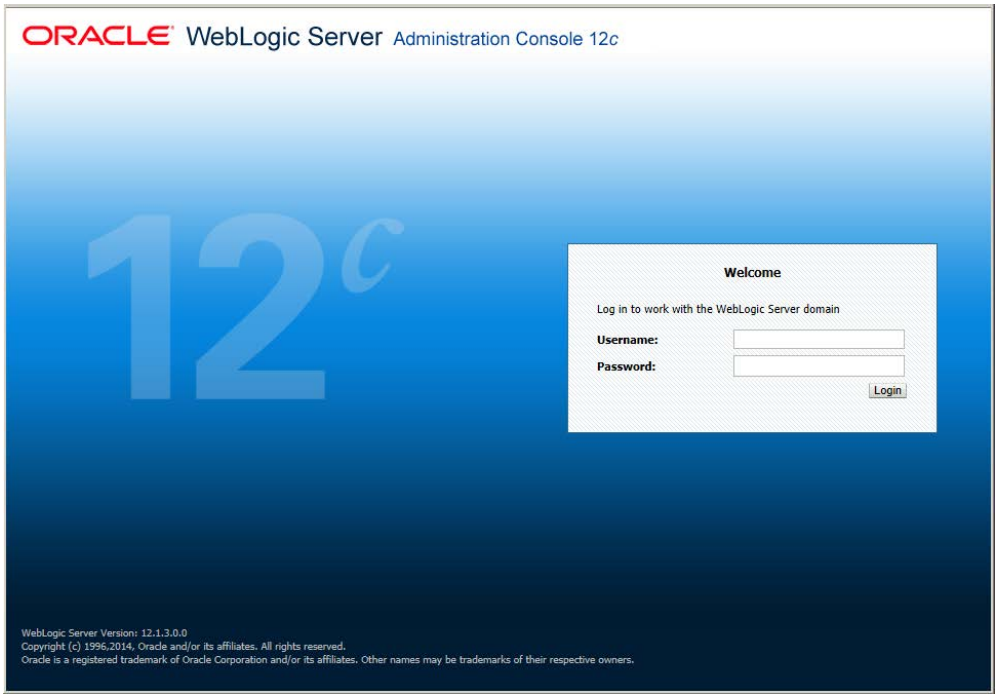
Output ribhost1: (IP addresses masked in example output.)

```
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet <ribhost1_ip_address>  netmask <netmask_ip>  broadcast
<broadcast_ip>
        inet6 fe80::221:f6ff:fe70:1956  prefixlen 64  scopeid 0x20<link>
        ether 00:21:f6:70:19:56  txqueuelen 1000  (Ethernet)
        RX packets 88204520  bytes 51234066890 (47.7 GiB)
        RX errors 0  dropped 2  overruns 0  frame 0
        TX packets 25448143  bytes 56771207745 (52.8 GiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

eth0:1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet <floating_ip_address>  netmask <netmask_ip>  broadcast
<broadcast_ip>        ether 00:21:f6:70:19:56  txqueuelen 1000  (Ethernet)

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet <network_ip>  netmask <netmask_ip>
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 0  (Local Loopback)
        RX packets 439105  bytes 68721422 (65.5 MiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 439105  bytes 68721422 (65.5 MiB)
```

Output ribhost2:

```
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet <ribhost2_ip_address>  netmask <netmask_ip>  broadcast
<broadcast_ip>
        inet6 fe80::221:f6ff:fee1:2dee  prefixlen 64  scopeid 0x20<link>
        ether 00:21:f6:e1:2d:ee  txqueuelen 1000  (Ethernet)
        RX packets 33291601  bytes 4805964369 (4.4 GiB)
        RX errors 0  dropped 2  overruns 0  frame 0
        TX packets 2556429  bytes 257047493 (245.1 MiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet <network_ip>  netmask <netmask_ip>
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 0  (Local Loopback)
        RX packets 4642  bytes 426547 (416.5 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 4642  bytes 426547 (416.5 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

- Using the virtual host alias for the RIB hosts launch the Weblogic admin console. All that is required is making sure that the URL works. No need to login. This verifies that the virtual host alias was able to access the application on ribhost1.



- Shutdown the Weblogic admin and managed servers for RIB.
- Locate and shut down all remaining processes that may be still running on ribhost1 for the applications that have been installed. Do not shut down system processes – just the ones that were created for the applications installed. They can be located by executing the following commands.

  Example:

  ```
  $ ps  -ef --cols 1000|grep /u00/webadmin/config
  $ ps  -ef --cols 1000|grep /u00/webadmin/products
  ```

- Unmount the floating IP address from ribhost1 by running the following command.

  Example:

  ```
  $ /sbin/ifconfig eth0:1 down
  ```

- Shut down ribhost1.
- On ribhost2 enable the floating IP by executing the following commands:

  Example:

  ```
  $ /sbin/ifconfig eth0:1 10.141.32.77 netmask 255.255.252.0
  $ /sbin/arping -q -U -c 3 -I eth0 10.141.32.77
  ```

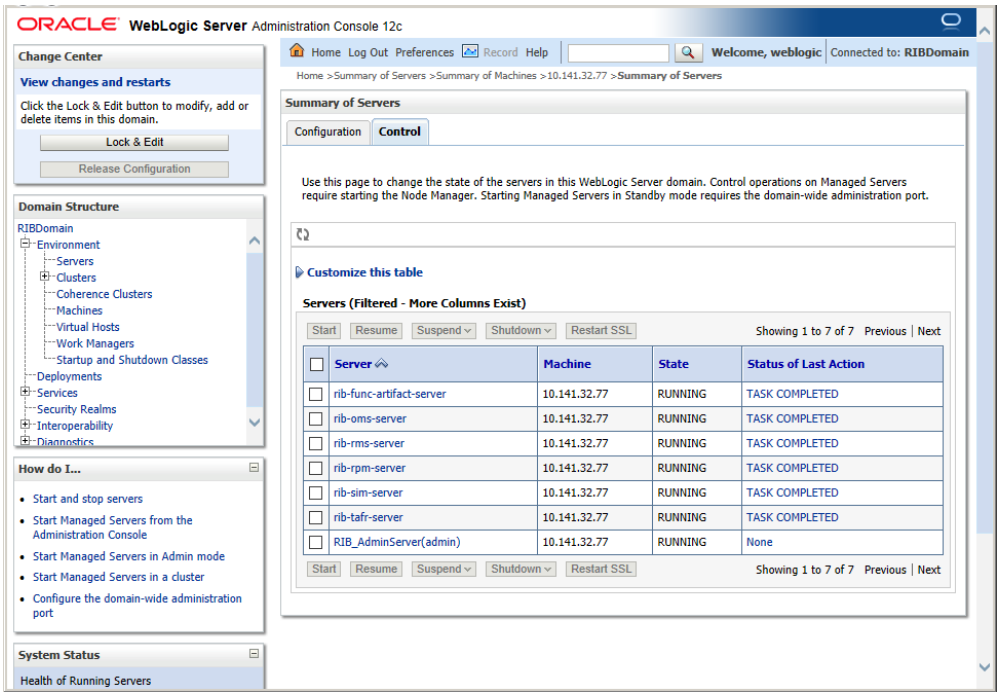
- On ribhost2 restart the Node Manager and WebLogic admin server. This validates that the application was installed on a shared mount between ribhost1 and ribhost2 correctly by being able to navigate on the same directory structure and then start applications.

  Example:

  ```
  $ cd /u00/webadmin/config/domains/wls_rib/RIBDomain/bin
  $ nohup ./startNodeManager.sh &
  $ nohup ./startWebLogic.sh &
  ```

- Using the virtual host alias for the RIB hosts launch the Weblogic admin console and restart the managed servers. This verifies that the virtual host alias was able to access the application on ribhost2 and that the WebLogic admin console

created on ribhost1 can restart the managed servers with the Node Manager using the floating IP address as a listen address.



- Revert the changes by performing the following steps:
- Shutdown the Weblogic admin and managed servers for RIB on ribhost2.
- Locate and shut down all remaining processes that may be still running on ribhost2 for the applications that have been installed. Do not shut down system processes – just the ones that were created for the applications installed. They can be located by executing the following commands.

  Example:

  ```
  $ ps  -ef --cols 1000|grep /u00/webadmin/config
  $ ps  -ef --cols 1000|grep /u00/webadmin/products
  ```

- Unmount the floating IP address from ribhost2 by running the following command.

  Example:

  ```
  $ /sbin/ifconfig eth0:1 down
  ```

- On ribhost1 enable the floating IP by executing the following commands:

  Example:

  ```
  $ /sbin/ifconfig eth0:1 10.141.32.77 netmask 255.255.252.0
  $ /sbin/arping -q -U -c 3 -I eth0 10.141.32.77
  ```

- Do not restart the Node Manager or RIB WebLogic admin and managed servers – the **Next** 2 sections require them to be shut down.

Configure the rib-<app>-server for Memory Allocation

This section explains how to configure the memory allocation for the managed servers in the RIBDomain. For these changes to take effect the server will have to be restarted after the changes have been made.

- Shutdown the WebLogic admin and managed servers if they are not already down.

- Take a back up of the <RIB_DOMAIN_HOME>/bin/startWebLogic.sh script.

  Example:

  ```
  $ cd /u00/webadmin/config/domains/wls_rib/RIBDomain/bin
  $ cp startWebLogic.sh startWebLogic.shbk1
  ```

- Edit the <RIB_DOMAIN_HOME>/bin/startWebLogic.sh script to add the following attributes just before the line that reads: # START WEBLOGIC:

  ```
  CLASSPATH=$DOMAIN_HOME/servers/$SERVER_NAME:$CLASSPATH
  JAVA_OPTIONS="-Dweblogic.ejb.container.MDBMessageWaitTime=2
  ${JAVA_OPTIONS}"
  JAVA_VM="-server"
  USER_MEM_ARGS="-Xms1024m -Xmx2048m "
  ```

  Example with changes highlighted as it is embedded in code:

  ```
  ...

  # CLASS CACHING

  if [ "${CLASS_CACHE}" = "true" ] ; then
          echo "Class caching enabled..."
          JAVA_OPTIONS="${JAVA_OPTIONS} -
  Dlaunch.main.class=${SERVER_CLASS}
  -Dlaunch.class.path="${CLASSPATH}"
  -Dlaunch.complete=weblogic.store.internal.LockManagerImpl -cp ${WL_
  HOME}/server/lib/pcl2.jar"
          export JAVA_OPTIONS
          SERVER_CLASS="com.oracle.classloader.launch.Launcher"
  fi

  CLASSPATH=$DOMAIN_HOME/servers/$SERVER_NAME:$CLASSPATH
  JAVA_OPTIONS="-Dweblogic.ejb.container.MDBMessageWaitTime=2
  ${JAVA_OPTIONS}"
  JAVA_VM="-server"
  USER_MEM_ARGS="-Xms1024m -Xmx2048m "

  # START WEBLOGIC

  echo "starting weblogic with Java version:"

  ...
  ```

- Do not restart the Node Manager or RIB WebLogic admin and managed servers – the **Next** section requires them to be shut down.

Configure weblogic.policy for RIB Application

This section explains how to setup the security permission grants required for the rib-<app>-deployments that will be installed in order for them to be accessible. For these changes to take effect the server will have to be restarted after the changes have been made.

- Shutdown the WebLogic admin and managed servers if they are not already down.
- Take a back up of the <RIB_WEBLOGIC_HOME>/wlserver/server/lib/weblogic.policy.

  Example:

  ```
  $ cd (/u00/webadmin/products/wls_rib/wlserver/server/lib
  $ cp weblogic.policy weblogic.policybk1
  ```

- Edit the <RIB_WEBLOGIC_HOME>/wlserver/server/lib/weblogic.policy to include the following at the end of the file.

> **Note**: If copying the following text from this guide to UNIX, ensure that it is properly formatted in UNIX. Each line entry beginning with "permission" must terminate on the same line with a semicolon.

Example:

```
grant codeBase
"file:/u00/webadmin/config/domains/wls_rib/RIBDomain/servers/rib-aip-
server/-"
{
permission java.security.AllPermission;
permission
oracle.security.jps.service.credstore.CredentialAccessPermission
"credstoressp.credstore", "read,write,update,delete";
permission
oracle.security.jps.service.credstore.CredentialAccessPermission
"credstoressp.credstore.*", "read,write,update,delete";
};


grant codeBase
"file:/u00/webadmin/config/domains/wls_rib/RIBDomain/servers/rib-oms-
server/-"
{
permission java.security.AllPermission;
permission
oracle.security.jps.service.credstore.CredentialAccessPermission
"credstoressp.credstore", "read,write,update,delete";
permission
oracle.security.jps.service.credstore.CredentialAccessPermission
"credstoressp.credstore.*", "read,write,update,delete";
};
grant codeBase
"file:/u00/webadmin/config/domains/wls_rib/RIBDomain/servers/rib-rms-
server/-"
{
permission java.security.AllPermission;
permission
oracle.security.jps.service.credstore.CredentialAccessPermission
"credstoressp.credstore", "read,write,update,delete";
permission
oracle.security.jps.service.credstore.CredentialAccessPermission
"credstoressp.credstore.*", "read,write,update,delete";
};


grant codeBase
"file:/u00/webadmin/config/domains/wls_rib/RIBDomain/servers/rib-rpm-
server/-"
{
permission java.security.AllPermission;
permission
oracle.security.jps.service.credstore.CredentialAccessPermission
"credstoressp.credstore", "read,write,update,delete";
permission
oracle.security.jps.service.credstore.CredentialAccessPermission
"credstoressp.credstore.*", "read,write,update,delete";
};
```

```
grant codeBase
"file:/u00/webadmin/config/domains/wls_rib/RIBDomain/servers/rib-sim-
server/-"
{
permission java.security.AllPermission;
permission
oracle.security.jps.service.credstore.CredentialAccessPermission
"credstoressp.credstore", "read,write,update,delete";
permission
oracle.security.jps.service.credstore.CredentialAccessPermission
"credstoressp.credstore.*", "read,write,update,delete";
};
```

```
grant codeBase
"file:/u00/webadmin/config/domains/wls_rib/RIBDomain/servers/rib-tafr-
server/-"
{
permission java.security.AllPermission;
permission
oracle.security.jps.service.credstore.CredentialAccessPermission
"credstoressp.credstore", "read,write,update,delete";
permission
oracle.security.jps.service.credstore.CredentialAccessPermission
"credstoressp.credstore.*", "read,write,update,delete";
};
```

Prepare for RIB and RDMT Installations

This section covers some preliminary steps that will properly prepare for the RIB application to be installed. The intent is to organize and create a RIB_HOME directory with the proper files placed in specific directories **in** order to install the application correctly.

- Restart the Node Manager as well as the WebLogic admin and managed servers.

  Example:

  ```
  $ cd /u00/webadmin/config/domains/wls_rib/RIBDomain/bin
  $ nohup ./startNodeManager.sh &
  $ nohup ./startWebLogic.sh &
  ```

- Create a directory on ribhost1 that can be used as a staging area to create the RIB_HOME directory.

  Example:

  ```
  $ mkdir -p /u00/webadmin/RIB_INSTALL
  ```

- Create a staging directory on ribhost1 to use as a staging directory and download the RIB application distribution to that directory. Going forward this document will refer to this directory as <RIB_STAGING> and the examples in this document will show it as:

  Example:

  ```
  /u00/webadmin/media/rib/
  ```

- From the <RIB_STAGING> directory - copy the RIB Kernel package (RibKernel15.0.0ForAll15.x.xApps_eng_ga.jar) into the

/u00/webadmin/RIB_INSTALL directory that was created for the RIB_HOME and extract its contents.

Example:

```
$ cd /u00/webadmin/media/rib/RIB
$ cp RibKernel15.0.0ForAll15.x.xApps_eng_ga.jar /u00/webadmin/RIB_INSTALL
$ cd /u00/webadmin/RIB_INSTALL
$ jar -xf RibKernel15.0.0ForAll15.x.xApps_eng_ga.jar
```

- Copy the RIB packaged aqapi-12.1.0.2.jar and ojdbc7-12.1.0.2.jar to <WEBLOGIC_HOME>/wlserver/server/lib directory.

  Example:

```
$ cd /u00/webadmin/RIB_INSTALL/Rib1500ForAll15xxApps/rib-home/integration-
lib/third-party/oracle/db/12.1.0.2
$ cp aqapi-12.1.0.2.jar /u00/webadmin/products/wls_rib/wlserver/server/lib
$ cp ojdbc7-12.1.0.2.jar
/u00/webadmin/products/wls_rib/wlserver/server/lib
```

- Copy the tar file distributions for each rib-<app> application (RibPak15.0.0For<app>15.0.0_eng_ga.tar) that are going to be installed into the <RIB_HOME>/download-home/all-rib-apps directory. Do not extract/untar the files.

  NOTE: This case study does not show installing AIP, RFM, and RWMS.

  Example:

```
$ cd /u00/webadmin/media/rib/
$ cp RibPak15.0.0ForOms15.0.0_eng_ga.tar
/u00/webadmin/RIB_INSTALL/Rib1500ForAll15xxApps/rib-home/download-
home/all-rib-apps
$ cp RibPak15.0.0ForRms15.0.0_eng_ga.tar
/u00/webadmin/RIB_INSTALL/Rib1500ForAll15xxApps/rib-home/download-
home/all-rib-apps
$ cp RibPak15.0.0ForRpm15.0.0_eng_ga.tar
/u00/webadmin/RIB_INSTALL/Rib1500ForAll15xxApps/rib-home/download-
home/all-rib-apps
$ cp RibPak15.0.0ForSim15.0.0_eng_ga.tar
/u00/webadmin/RIB_INSTALL/Rib1500ForAll15xxApps/rib-home/download-
home/all-rib-apps
$ cp RibPak15.0.0ForTafr15.0.0_eng_ga.tar
/u00/webadmin/RIB_INSTALL/Rib1500ForAll15xxApps/rib-home/download-
home/all-rib-apps
```

- Copy the RIB Diagnostic and Monitoring Tools (RDMT) package (Rdmt15.0.0ForAll15.x.xApps_eng_ga.tar) and extract/untar it into the <RIB_HOME>/tools-home directory. Several files will be placed under the rdmt directory when you untar the package. This allows the installer to run the <RIB_HOME>/tools-home/rdmt/configbuilder.sh script as part of the RIB installation.

  Example:

```
$ cd /u00/webadmin/media/rib/RIB
$ cp Rdmt15.0.0ForAll15.x.xApps_eng_ga.tar
/u00/webadmin/RIB_INSTALL/Rib1500ForAll15xxApps/rib-home/tools-home
$ cd /u00/webadmin/RIB_INSTALL/Rib1500ForAll15xxApps/rib-home/tools-home
$ tar -xf Rdmt15.0.0ForAll15.x.xApps_eng_ga.tar
```

- Give execute permissions to RIB_HOME directory:

  Example:

  ```
  $ chmod -R 755 /u00/webadmin/RIB_INSTALL/
  ```

  NOTE: There will be a number of .tar files that do not get moved from the RIB-STAGING. Those are for applications that will be installed on rtghosts later.

Install RIB Application

This section contains detailed steps on running the RIB Installer.

- Set the following variables by exporting them before running the installer so that the JAVA_HOME is added to the PATH variable:
  - JAVA_HOME
  - PATH
  - DISPLAY

  Example:

  ```
  $ export JAVA_HOME=/u00/webadmin/products/jdk_java
  $ export PATH=$JAVA_HOME/bin:$PATH
  $ export DISPLAY=<pc_ip_address:0.0>
  ```

- Clear the following variables prior to running the installer to make sure that they are not already populated from some other process.
  - ANT_HOME
  - ANT_CONTRIB
  - CLASSPATH

  Example:

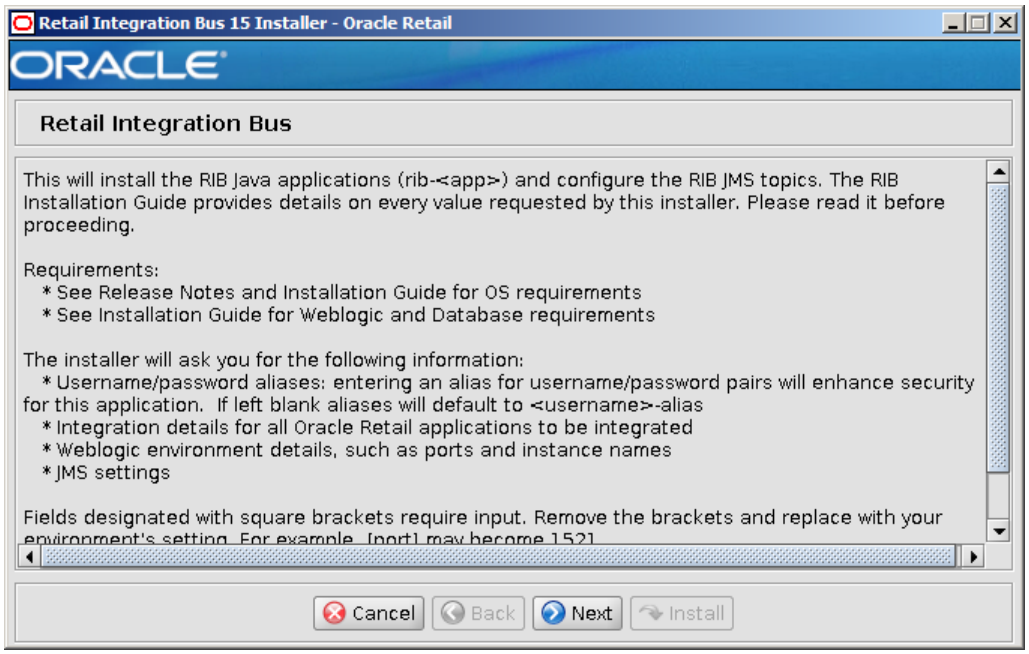
  ```
  $ unset ANT_HOME
  $ unset ANT_CONTRIB
  $ unset CLASSPATH
  ```

- Change directories to the <RIB_HOME> directory and run the rib-installer.sh script.

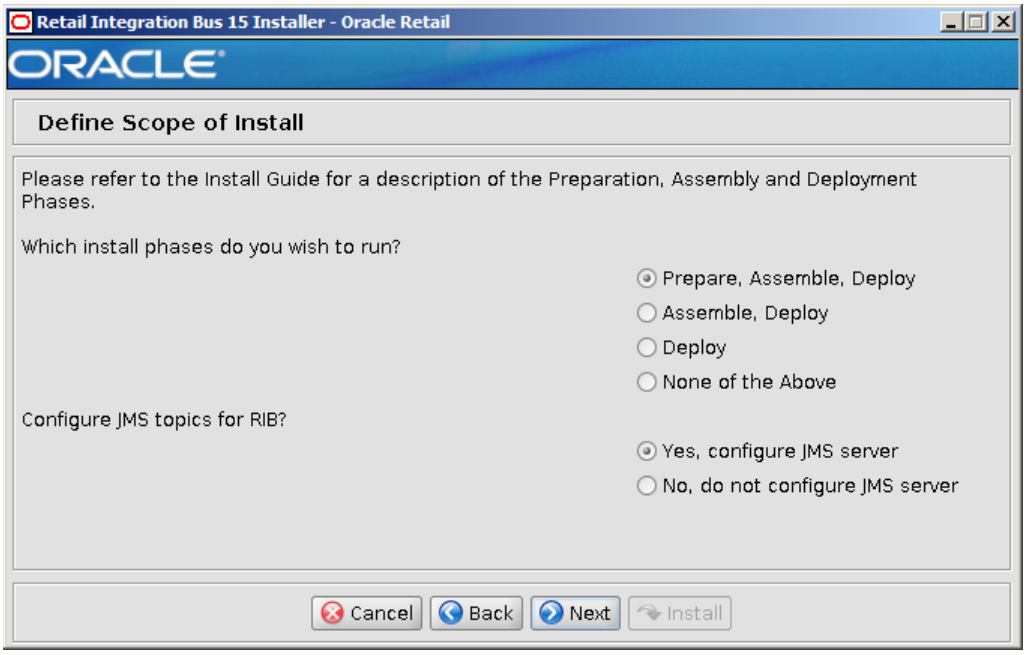
  Example:

  ```
  $ cd /u00/webadmin/RIB_INSTALL/Rib1500ForAll15xxApps/rib-home
  $./rib-installer.sh
  ```

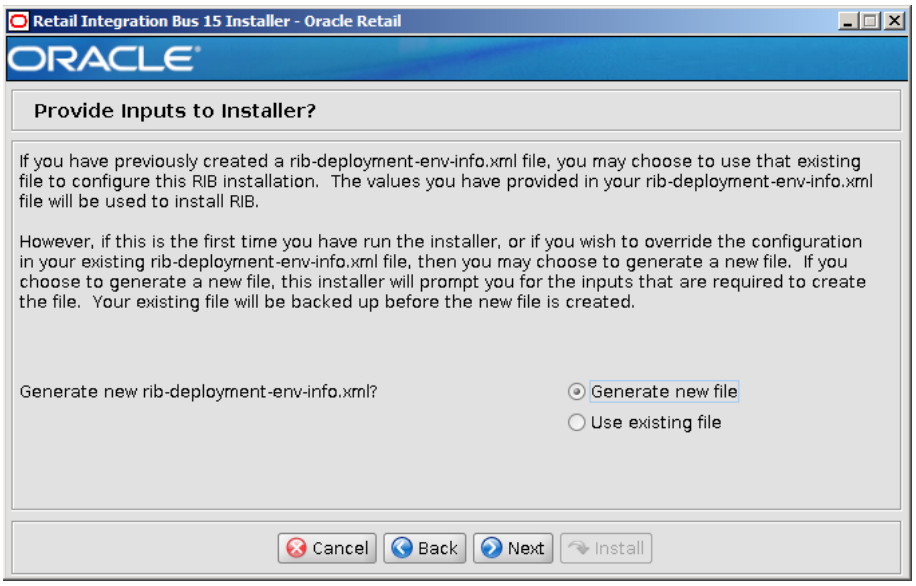
- The Retail Integration Bus installer sc appears. Click **Next**.



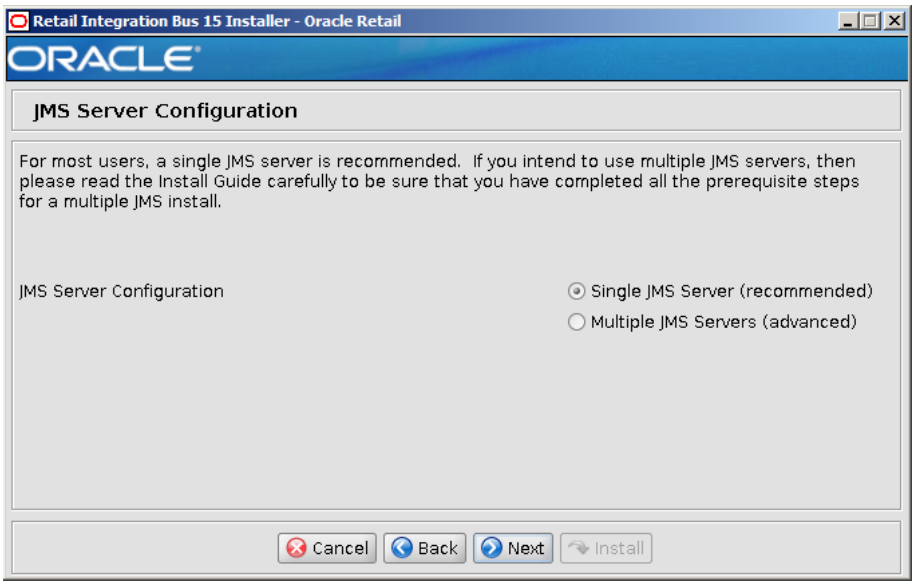
- The Define Scope of Install window opens. Select the following and click **Next**.
  - Which install phases do you wish to run?: **Prepare, Assemble, Deploy**
  - Configure JMS topics for RIB?: **Yes, configure JMS server**



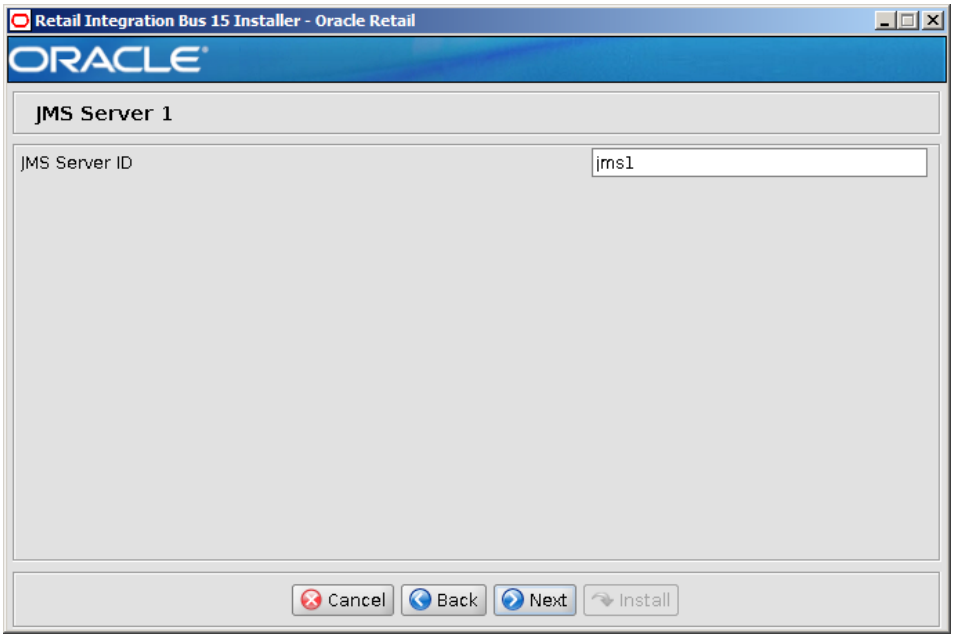
- The Provide Inputs to Installer window opens. Select **Generate new file**, and click **Next**.



- The JMS Server Configuration window opens. Select **Single JMS Server (recommended)**, and click **Next**.

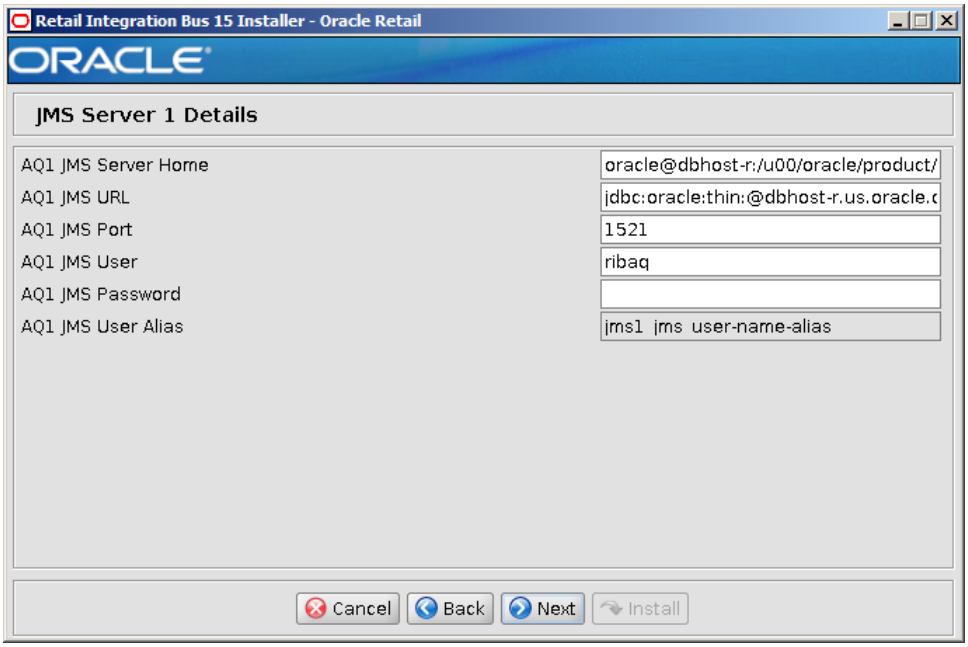


- The JMS Server 1 window opens. Provide a name for the JSM Server ID (jms1), and click **Next**.

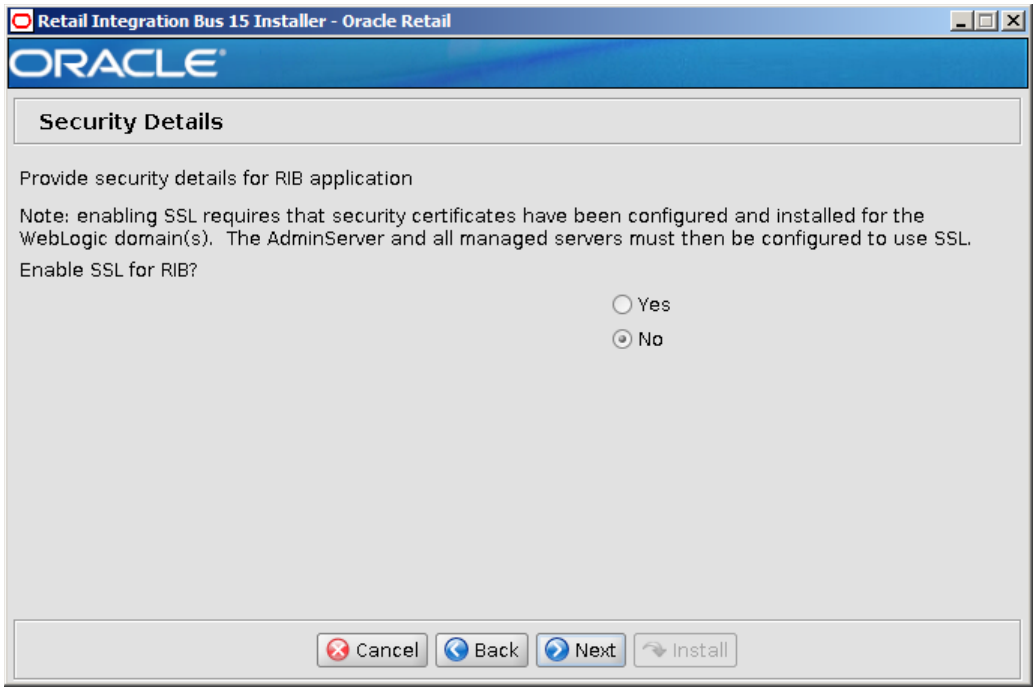


- The JMS Server 1 Details window opens. Provide the following inputs for the database connection details, and click **Next**.
  - AQ1 JMS Server Home: <dbhost_user>@dbhost-r:<DB_ORACLE_HOME>
  - AQ1 JMS URL: jdbc:oracle:thin@dbhost-r.us.oracle.com:1521/ribservicename
  - AQ1 JMS Port: 1521
  - AQ1 JMS User: ribaq
  - AQ1 JMS Password: < password>
  - AQ1 JMS User Alias: Leave as defaulted.

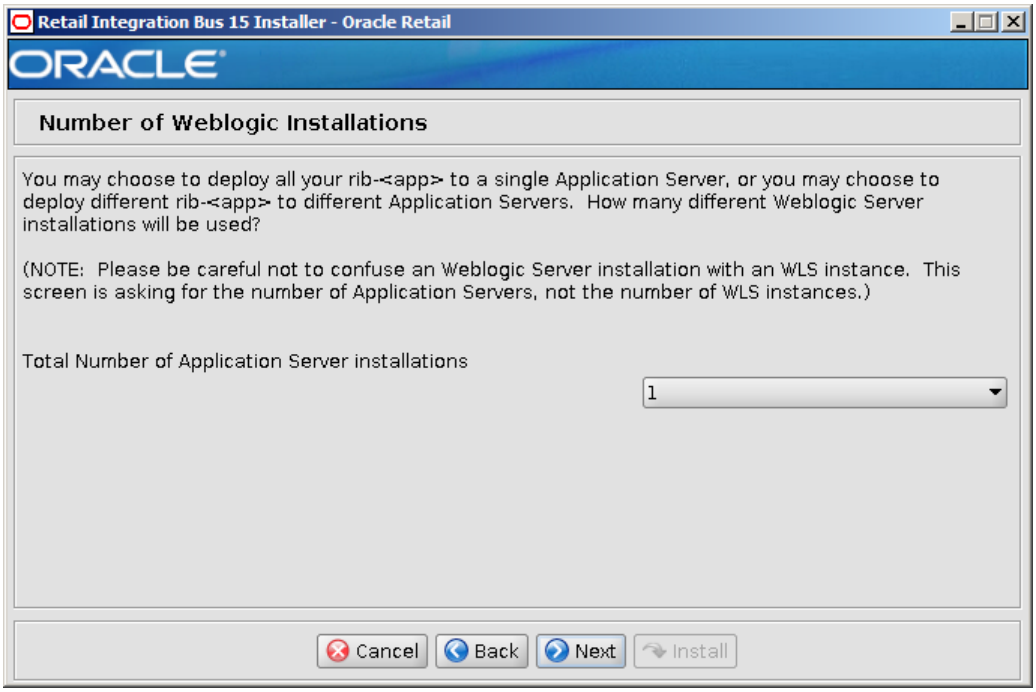
NOTE: It is assumed the ribaq user was created during database creation setup steps.



- The Security Details window opens. Select **No**, and click **Next**.



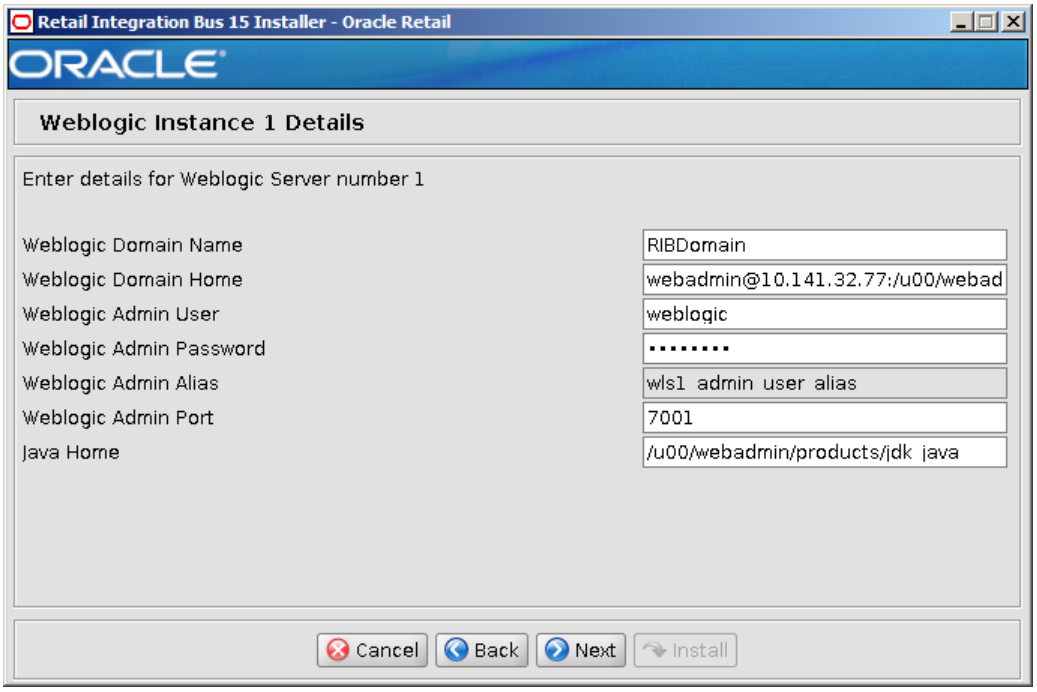
- The Number of WebLogic Installations window opens. Select 1 and click **Next**.



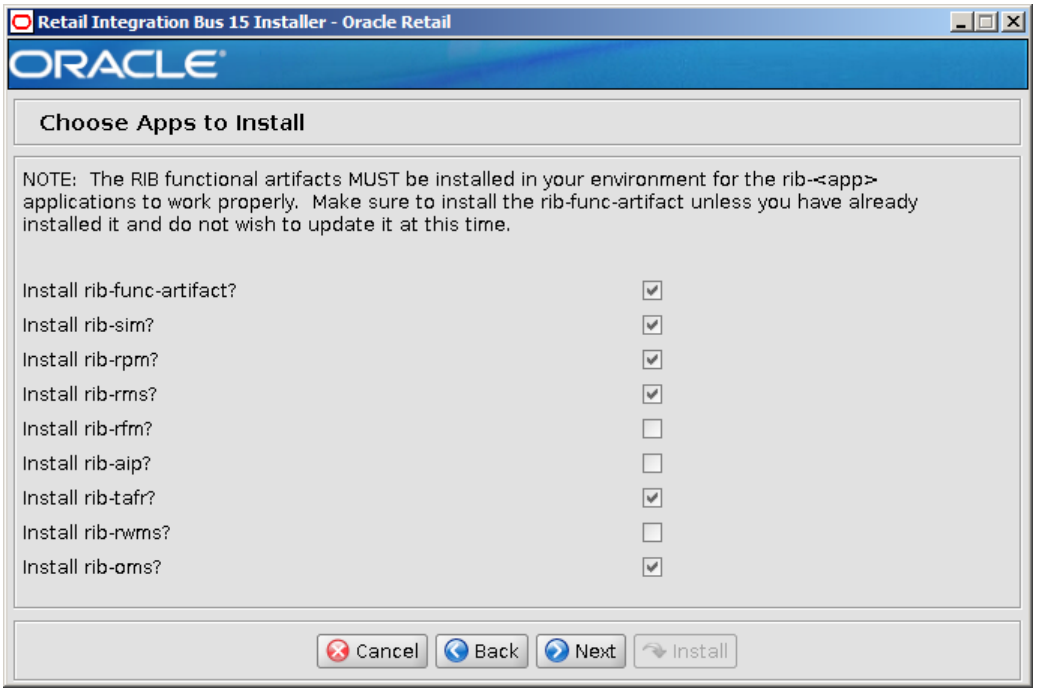
- The Weblogic Instance 1 Details window opens. Provide the details for the RIB WebLogic and click **Next**.

  Example for Weblogic Domain Home below:

  – Weblogic Domain Home: <ribhost_user>@<floating_ip>:/u00/webadmin/config/domains/wls_rib/RIBDomain

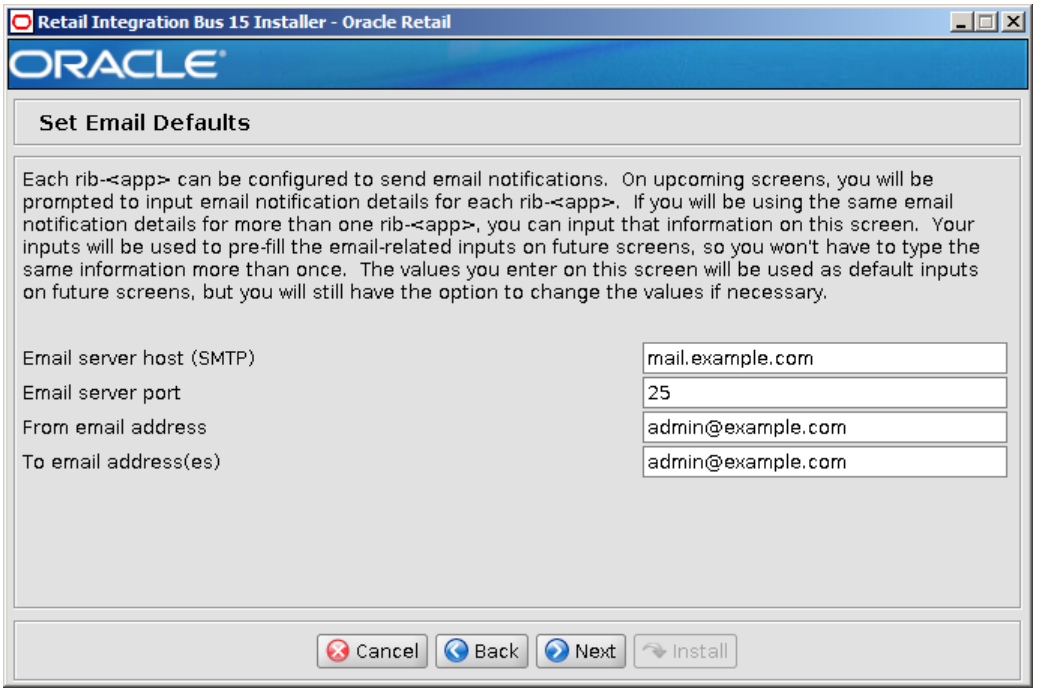


- The Choose Apps to Install window opens. Make the following selections and click **Next**.
  - Install rib-func-artifact
  - Install rib-sim
  - Install rib-rpm
  - Install rib-rms
  - Install rib-tafr
  - Install rib-oms



- The Set Email Defaults window opens. Provide inputs and click **Next**.

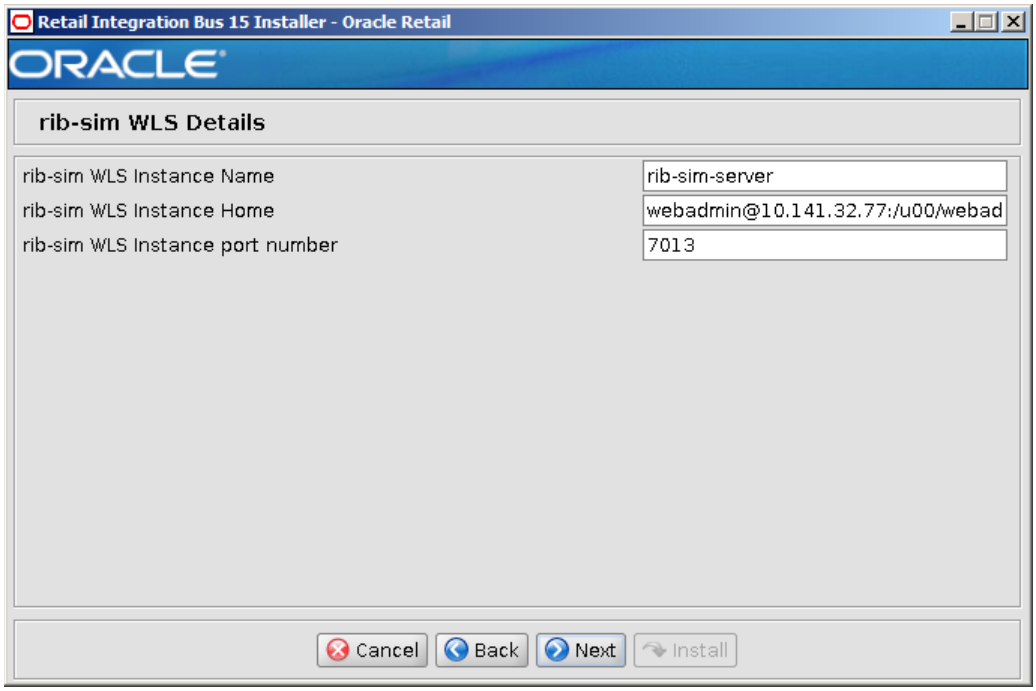
> **NOTE**: The entries made here will be added to the rest of the screens that request for information regarding emails as default entries.



- The rib-sim WLS Details window opens. Provide the required inputs and click **Next**.
  - The inputs should reflect the managed server and port setup in WebLogic for SIM in the RIBDomain.

Example for rib-sim WLS Instance Home:

– rib-sim WLS Instance Home: webadmin@<floating_ip>:/u00/webadmin/config/domains/wls_rib/RIBDomain/servers/rib-sim-server

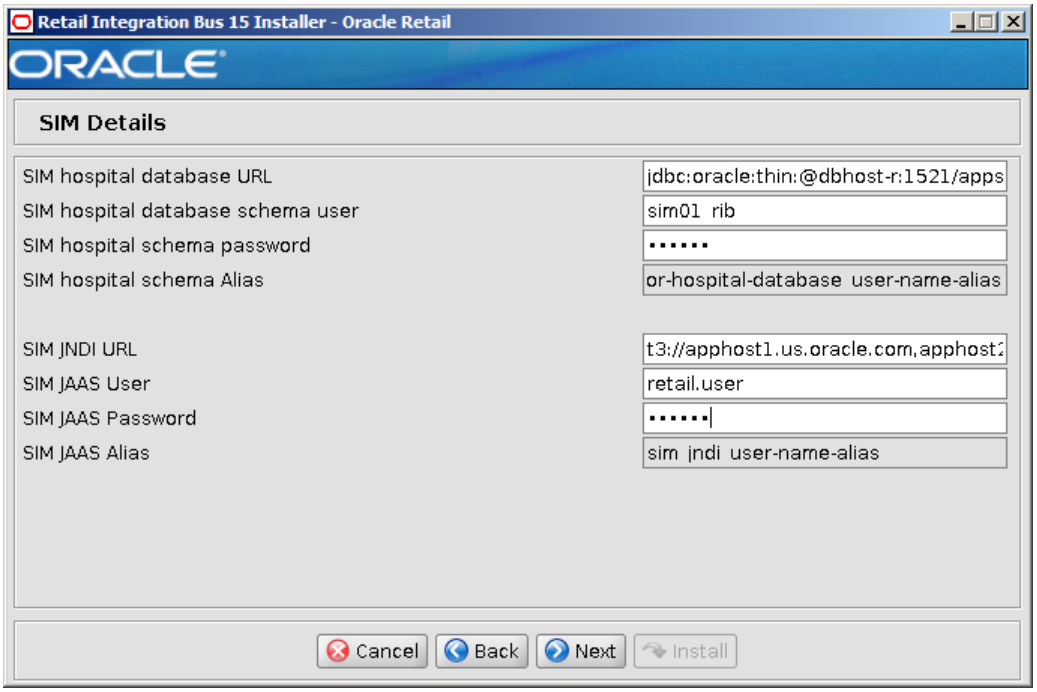


- The SIM Details window opens. Provide the database details for how the RIB user message and hospital tables were setup for SIM in the top section of the screen. Provide details of the SIM application as it is setup on apphost1 and apphost2 including port number. Click **Next**.

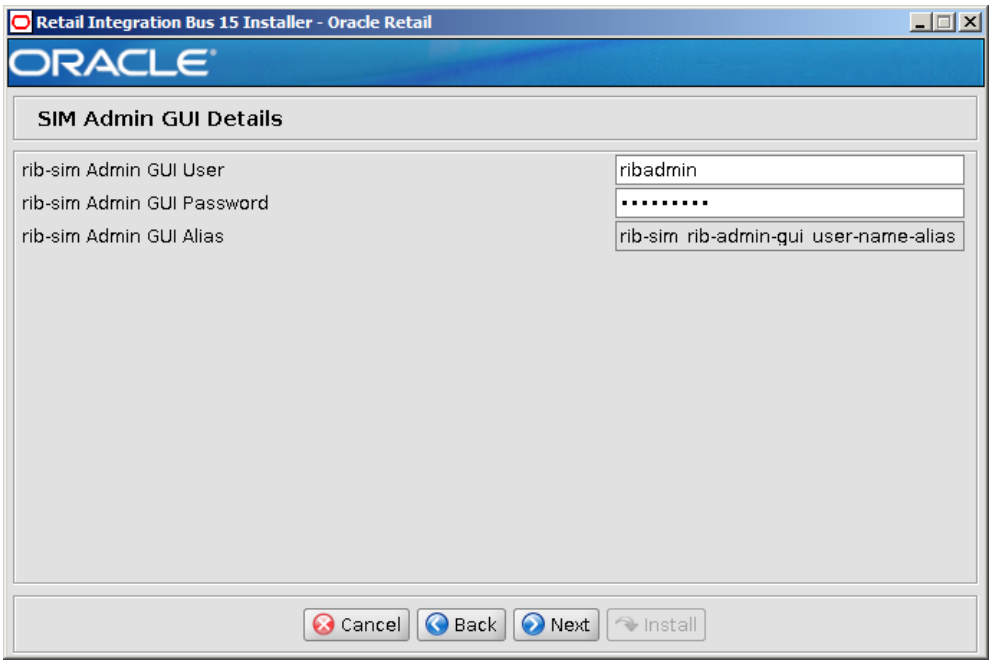
  Example for SIM hospital database URL and SIM JINDI URL below:

  – SIM hospital database URL: jdbc:oracle:thin:@dbhost-r:1521/appservicename

  – SIM JINDI URL: t3://apphost2.us.oracle.com,apphost1.us.oracle.com:7143/

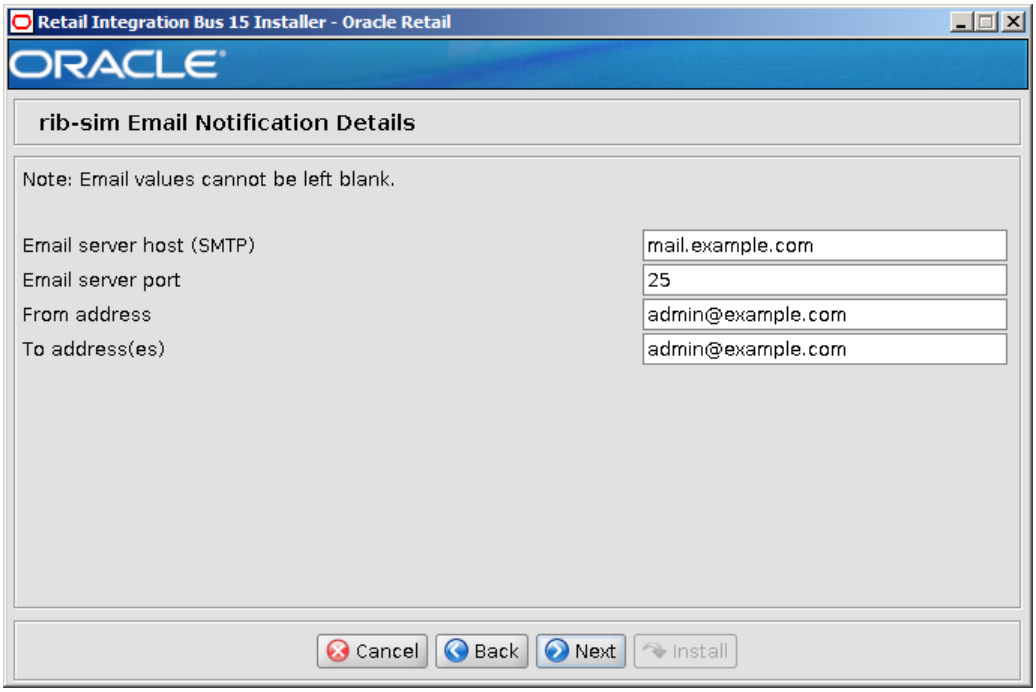


- The SIM Admin GUI Details window opens. Provide the desired credentials to be created for the RIB SIM admin screen logon. Click **Next**.



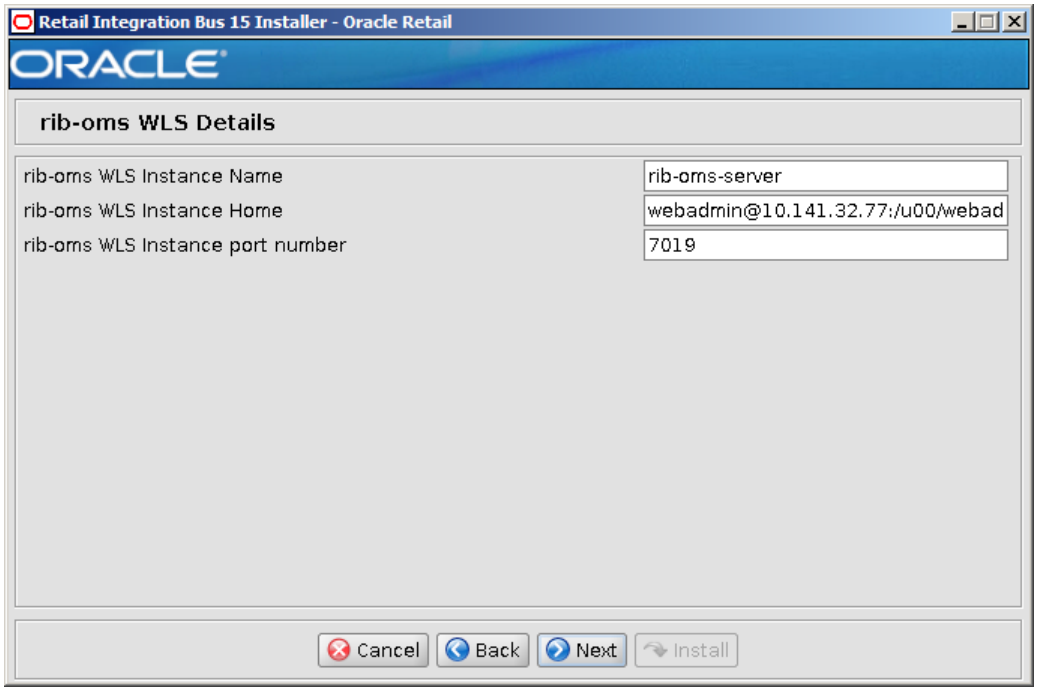
- The rib-sim Email Notification Details window opens. Provide email details and click **Next**.

NOTE: Email address does not need to be valid.



- The rib-oms WLS Details window opens. Provide the required inputs and click **Next**.
  - The inputs should reflect the managed server and port setup in WebLogic for OMS in the RIBDomain.

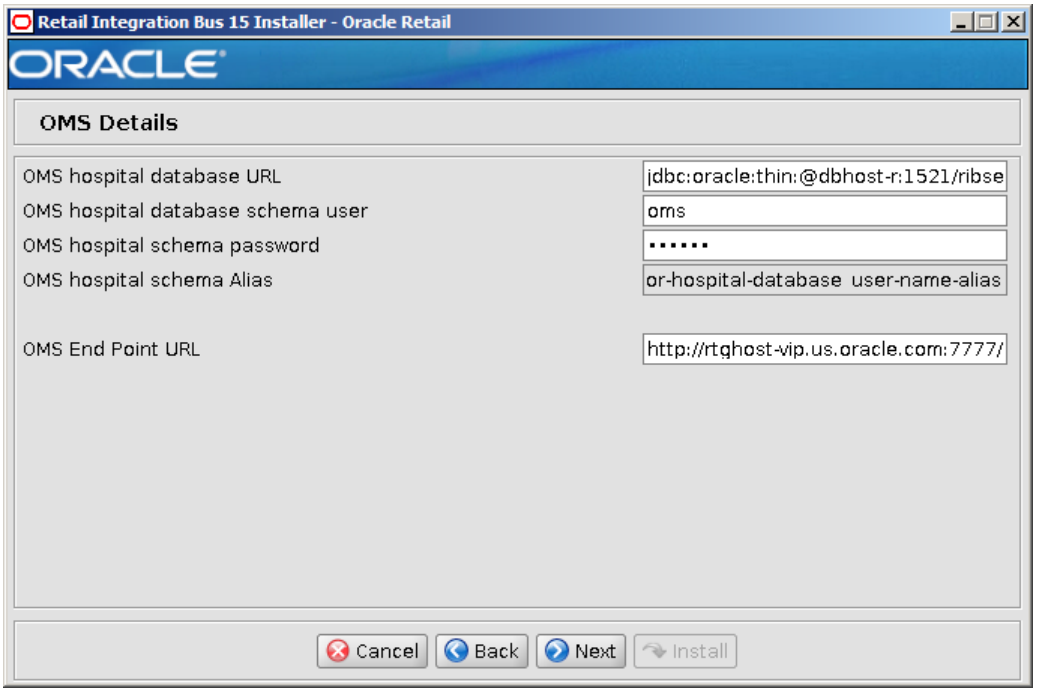
  Example for rib-oms WLS Instance Home:
  - rib-oms WLS Instance Home: webadmin@<floating_ip>:/u00/webadmin/config/domains/wls_rib/RIBDomain/servers/rib-oms-server



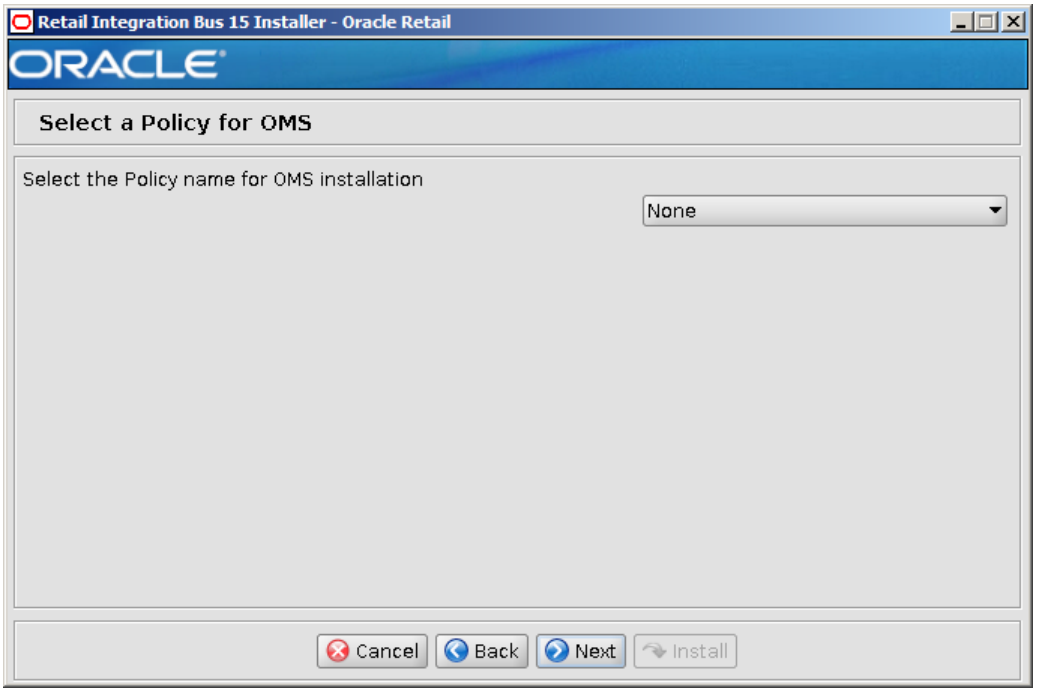
- The OMS Details window opens. Provide the database details for how the RIB user message and hospital tables were setup for OMS in the top section of the screen. Provide the End Point URL for the OMS WSDL. Click **Next**.

  Example for OMS hospital database URL and OMS End Point URL:

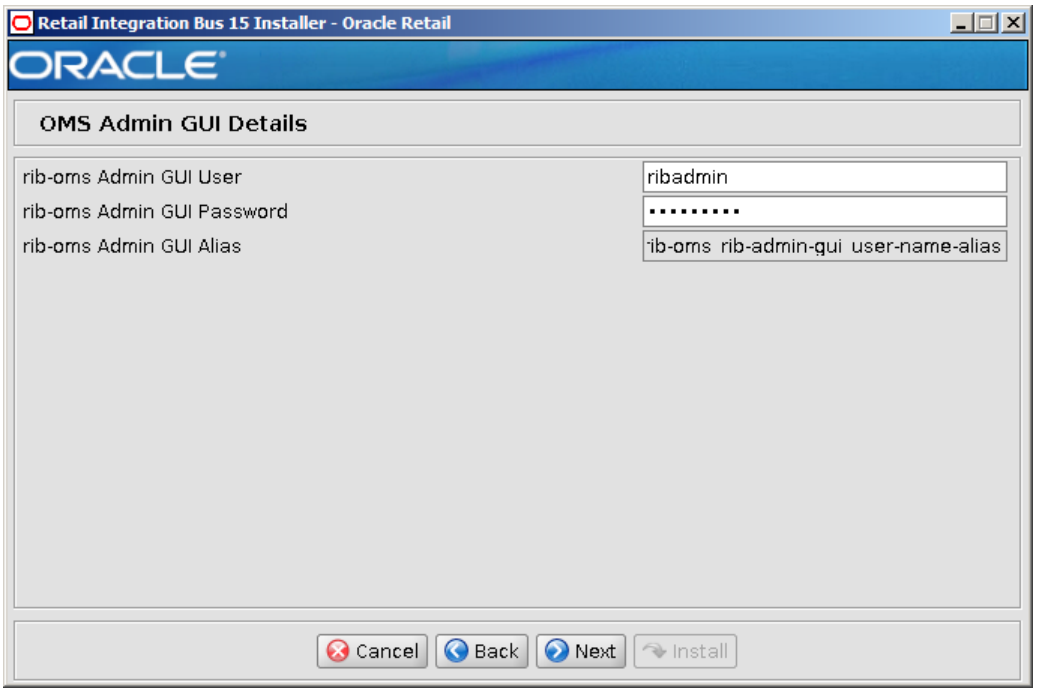
  – OMS hospital database URL: jdbc:oracle:thin:@dbhost-r:1521/ribservicename
  – OMS End Point URL:
    http://rtghost-vip.us.oracle.com:7777/RibOmsToRsbOmsRouting-ServicesIntegrationFlow/ProxyService/RibOmsToRsbOmsRoutingService?wsdl



- The Select a Policy for OMS window opens. Select **None**, and Click **Next**.

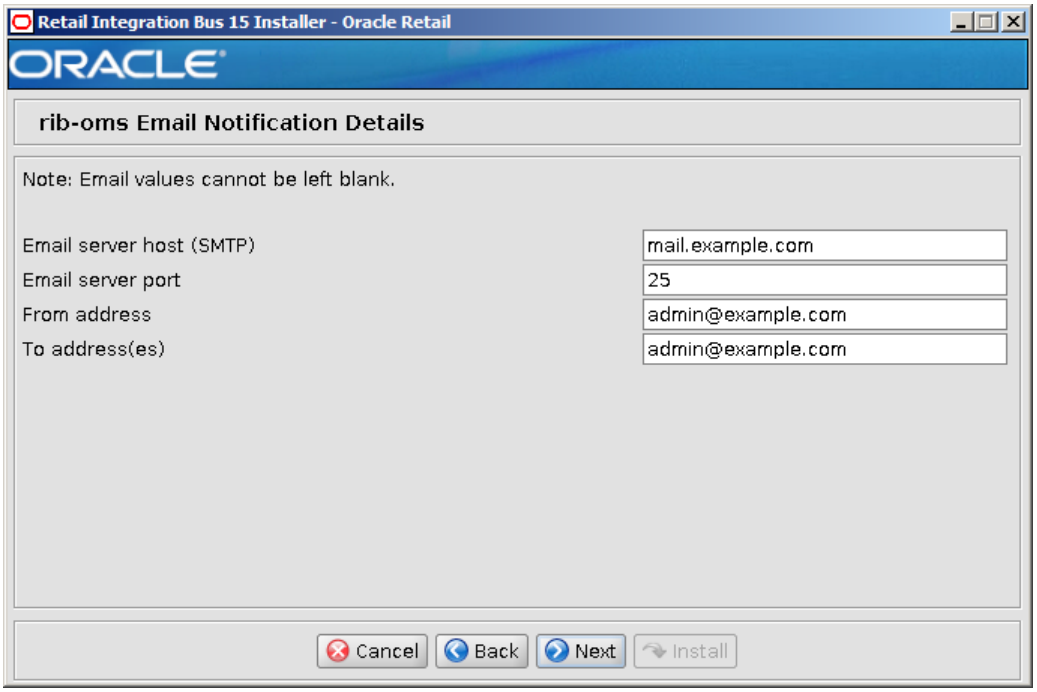


- The OMS Admin GUI Details window opens. Provide the desired credentials to be created for the RIB OMS admin screen logon. Click **Next**.



- The rib-oms Email Notification Details window opens. Provide email details and click **Next**.

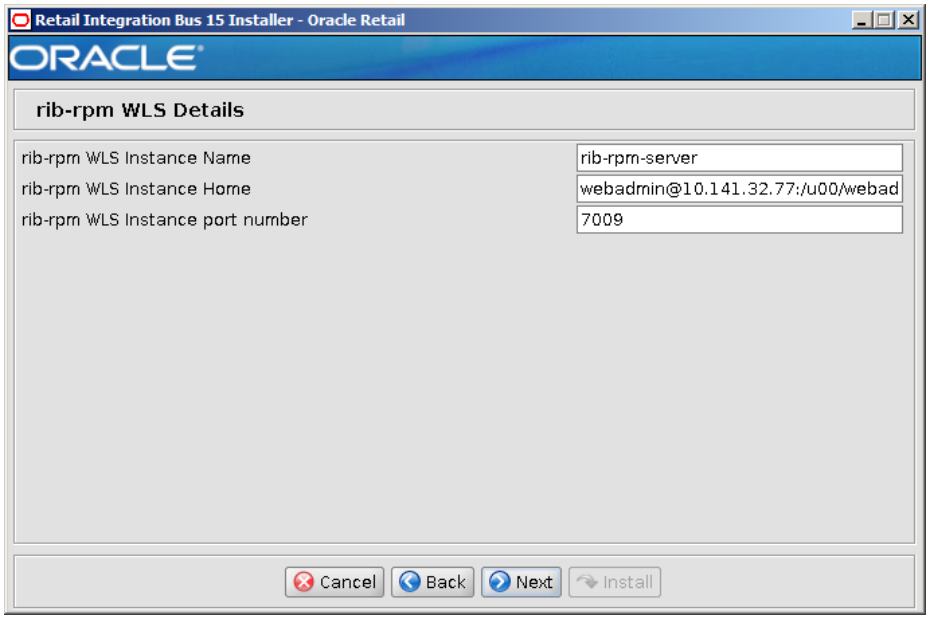
NOTE: Email address does not need to be valid.



- The rib-rpm WLS Details window opens. Provide the required inputs and click **Next**.
  - The inputs should reflect the managed server and port setup in WebLogic for RPM in the RIBDomain.

  Example for rib-rpm WLS Instance Home:

  rib-rpm WLS Instance Home:
  webadmin@<floating_ip>:/u00/webadmin/config/domains/wls_rib/RIBDomain/servers/rib-rpm-server

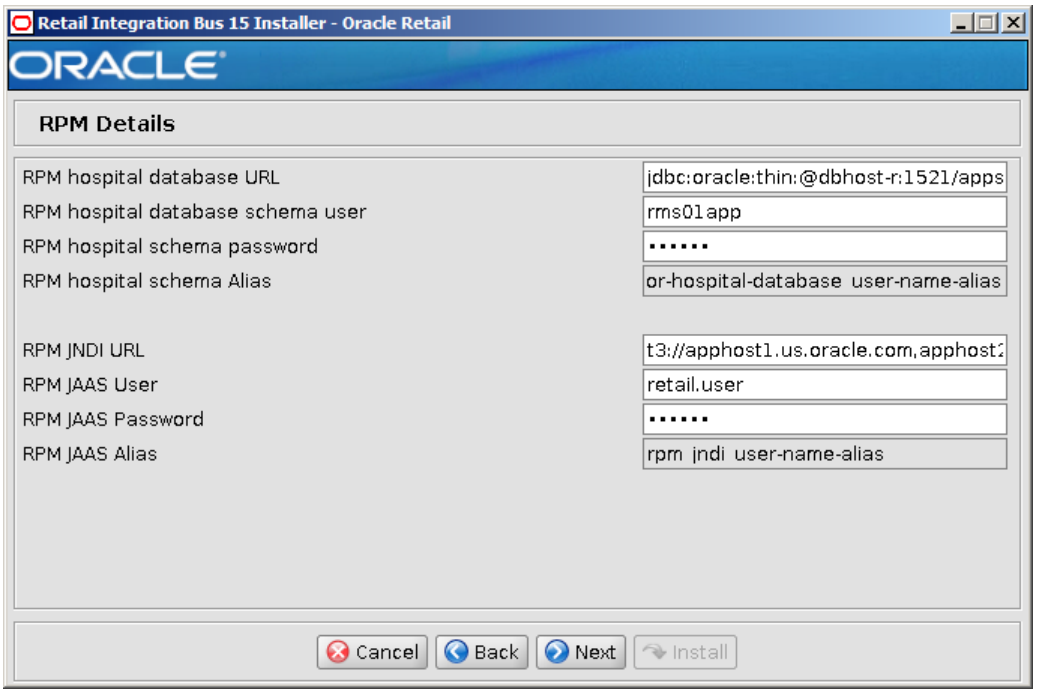


- The RPM Details window opens. Provide the database connection details for how the RIB user message and hospital tables were setup for the RMS application user in the top section of the screen. The reason RMS database details are used here is that the RMS and RPM applications share the same database schemas. Provide details of the RPM application as it is setup on apphost1 and apphost2 including port number. Click **Next**.

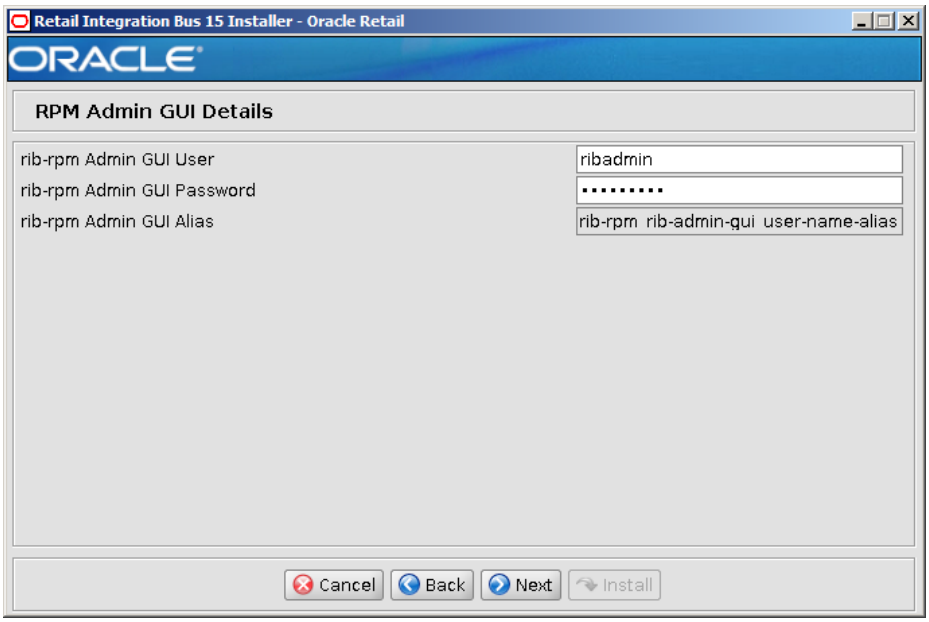
Example for RPM hospital database URL and RPM JINDI URL below:

  – RPM hospital database URL: jdbc:oracle:thin:@dbhost-r:1521/appservicename
  – RPM JINDI URL: t3://apphost1.us.oracle.com,apphost2.us.oracle.com:7133/

NOTE: The reason RMS database details are used here is that the RMS and RPM applications share the same database schemas.

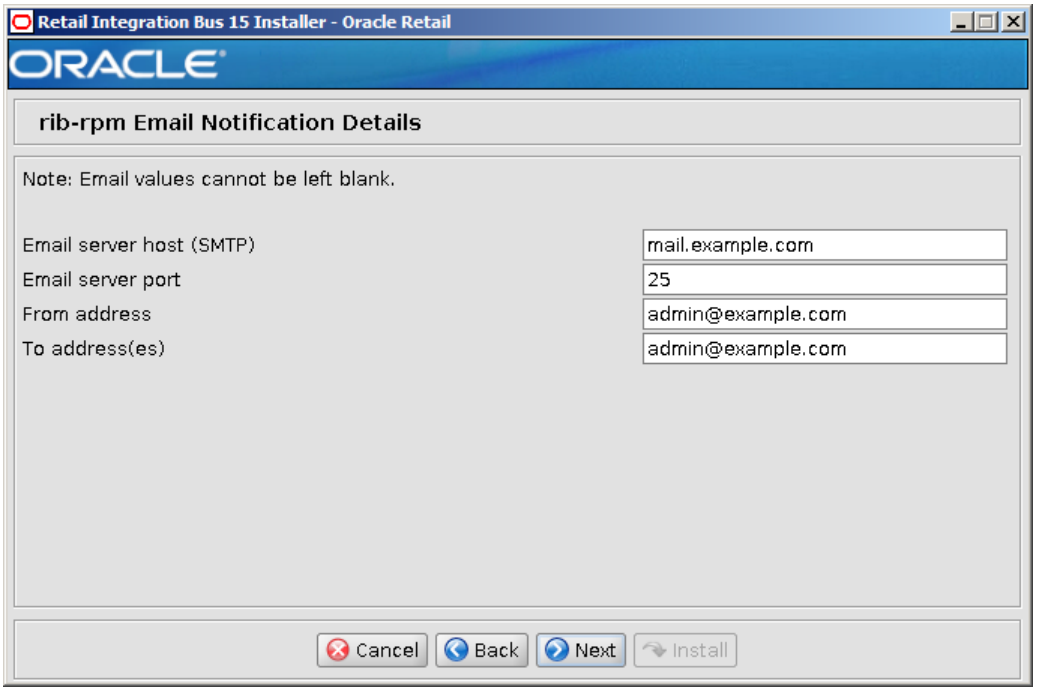


- The RPM Admin GUI Details window opens. Provide the desired credentials to be created for the RIB RPM admin screen logon. Click **Next**.



- The rib-rpm Email Notification Details window opens. Provide email details and click **Next**.

NOTE: Email address does not need to be valid.

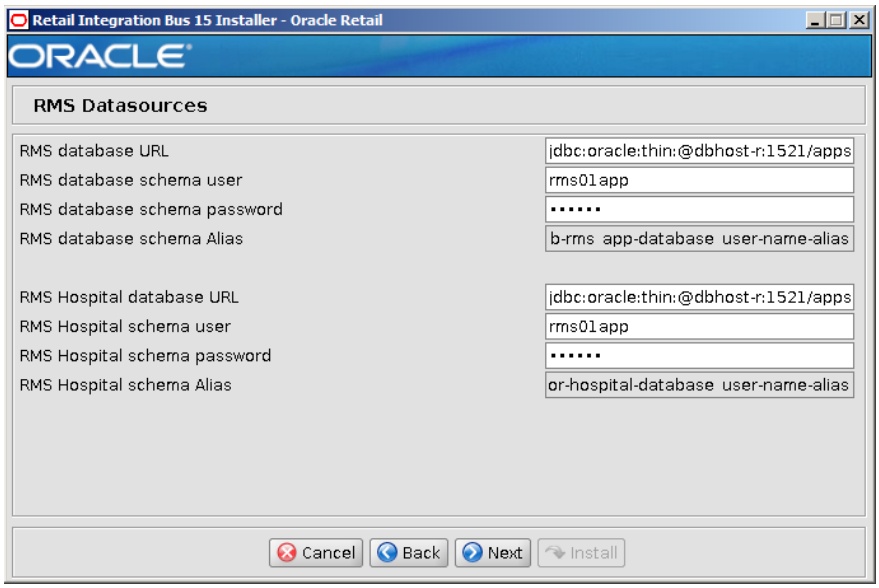


- The RMS Datasources window opens. Provide database connection details for the RMS application user in the top section of the screen. Provide the database connection details for how the RIB user message and hospital tables for RMS were setup in the bottom section of the screen. Click **Next**.

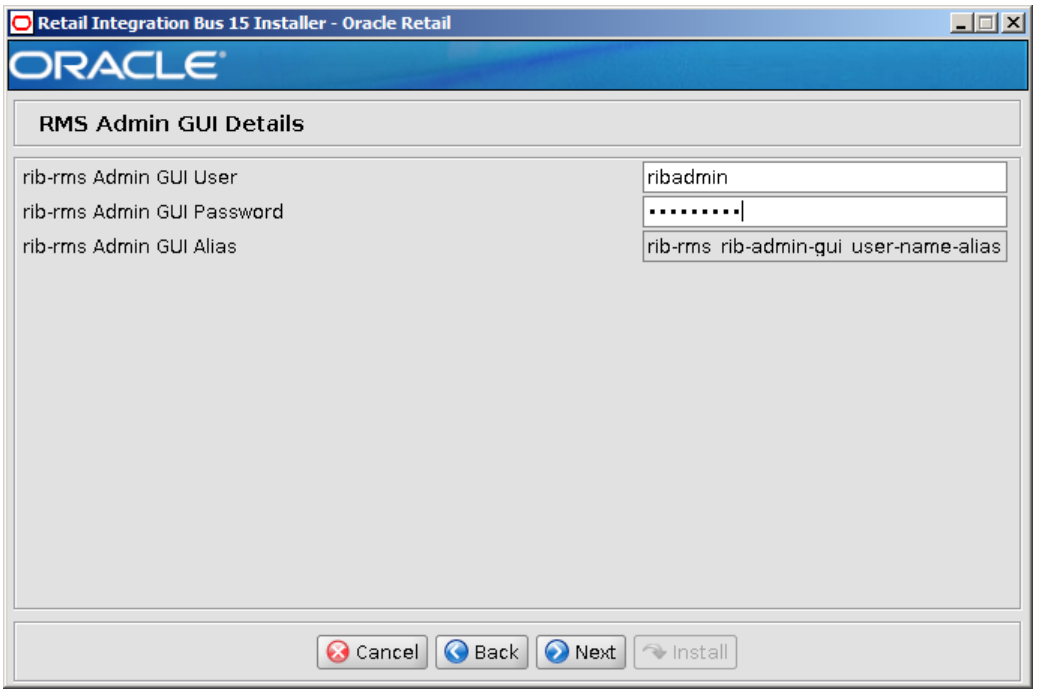
Example for RMS database URL and RMS hospital database URL below:

  – RMS database URL: jdbc:oracle:thin:@dbhost-r:1521/appservicename

– RMS hospital database URL: jdbc:oracle:thin:@dbhost-r:1521/appservicename

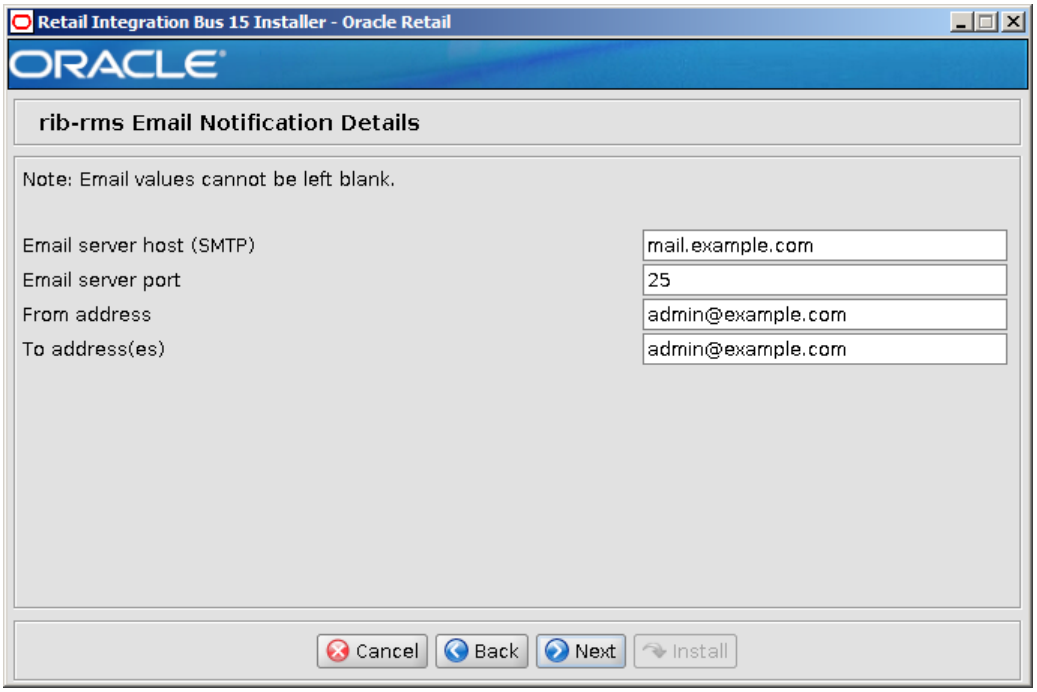


- The RMS Admin GUI Details window opens. Provide the desired credentials to be created for the RIB RMS admin screen logon. Click **Next**.



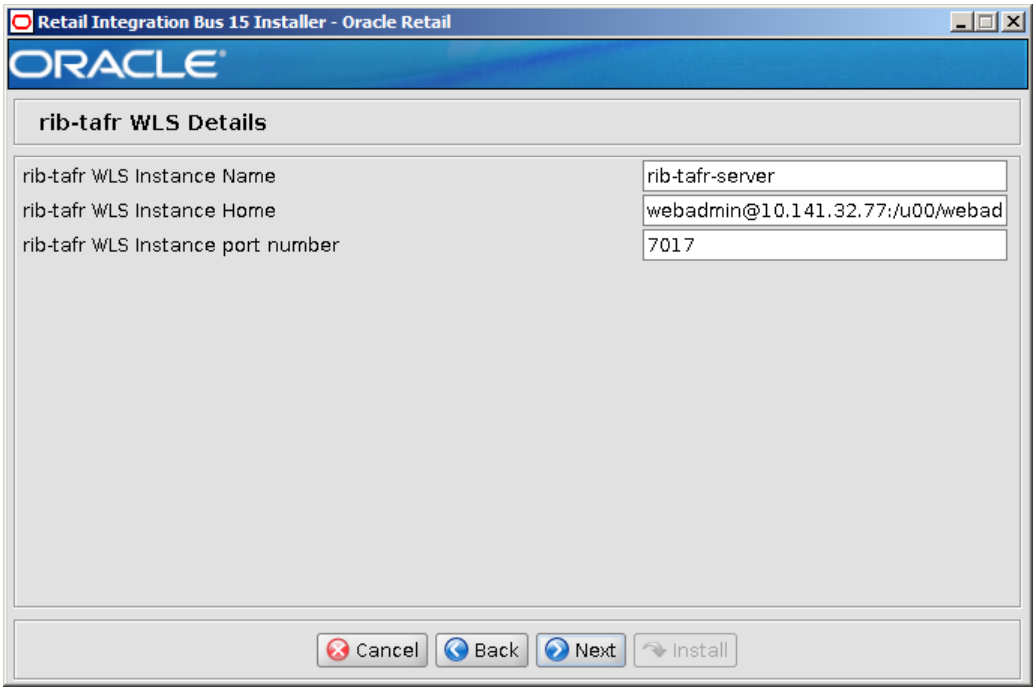
- The rib-rms Email Notification Details window opens. Provide email details and click **Next**.

NOTE: Email address does not need to be valid.



- The rib-tafr WLS Details window opens. Provide the required inputs and click **Next**.
  - The inputs should reflect the managed server and port setup in WebLogic for rib-tafr in the RIBDomain.

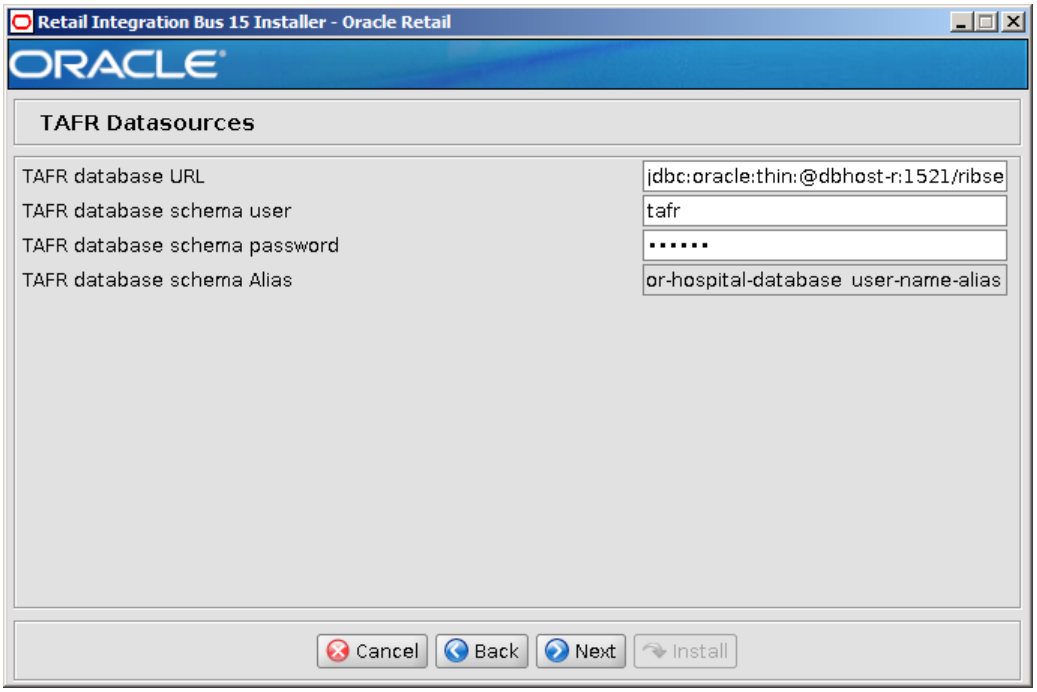
  Example for rib-tafr WLS Instance Home:
  - rib-tafr WLS Instance Home: webadmin@<floating_ip>:/u00/webadmin/config/domains/wls_rib/RIBDomain/servers/rib-tafr-server



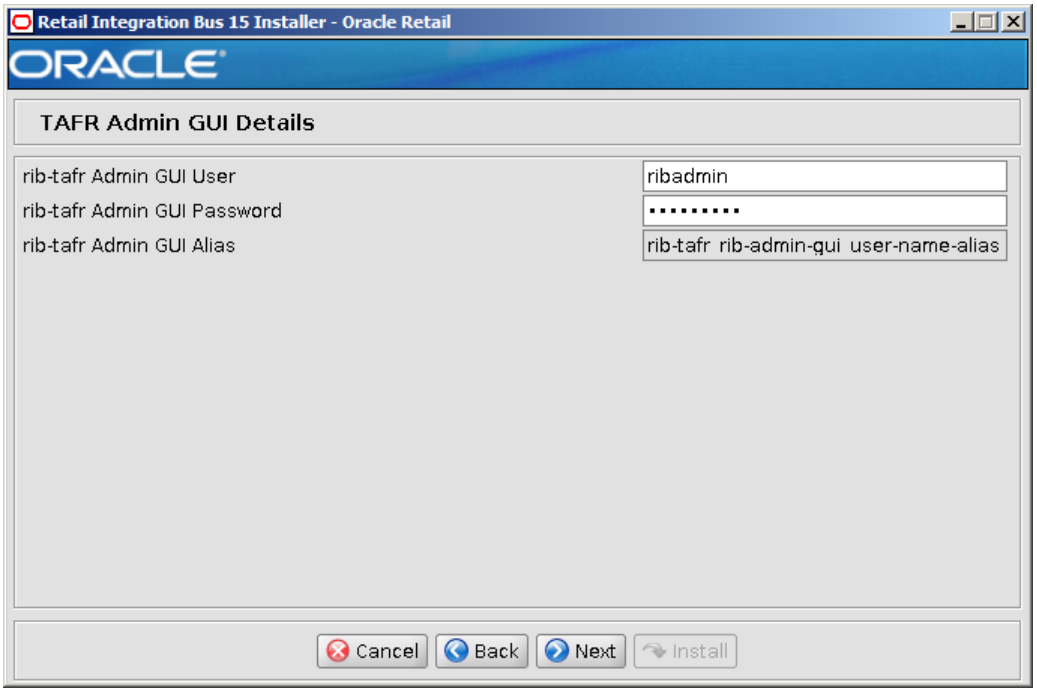
- The TAFR Datasources window opens. Provide the database connection details for how the RIB user message and hospital tables were setup for the RIB TAFR's. Click **Next**.

  Example for TAFR database URL below:

  – TAFR database URL: jdbc:oracle:thin:@dbhost-r:1521/ribservicename

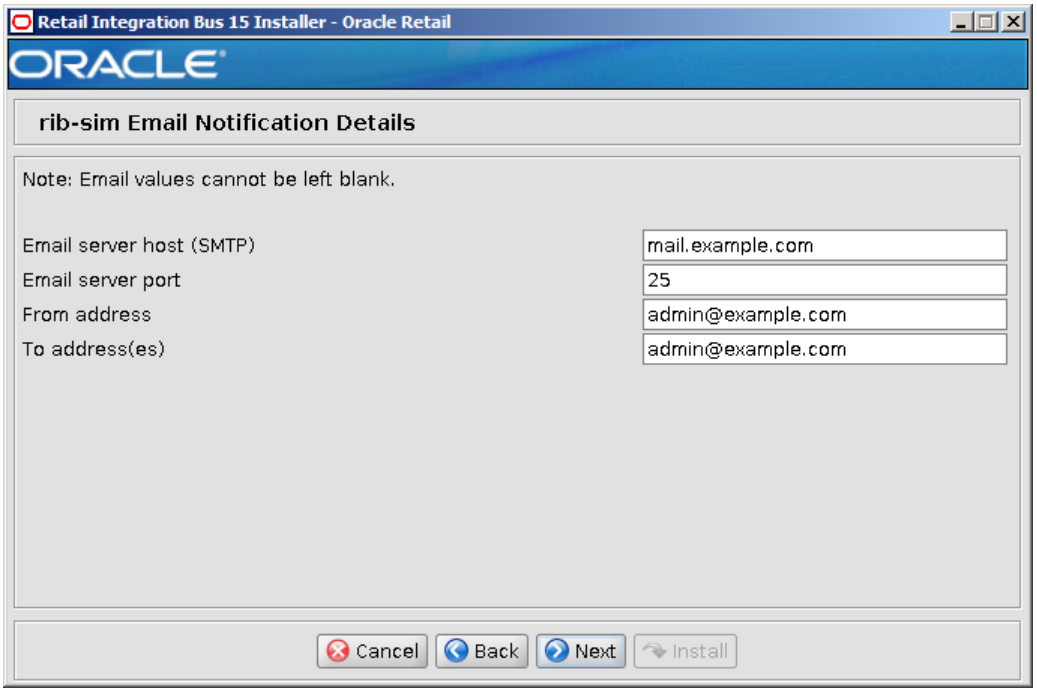


- The TAFR Admin GUI Details window opens. Provide the desired credentials to be created for the RIB TAFR admin screen logon. Click **Next**.



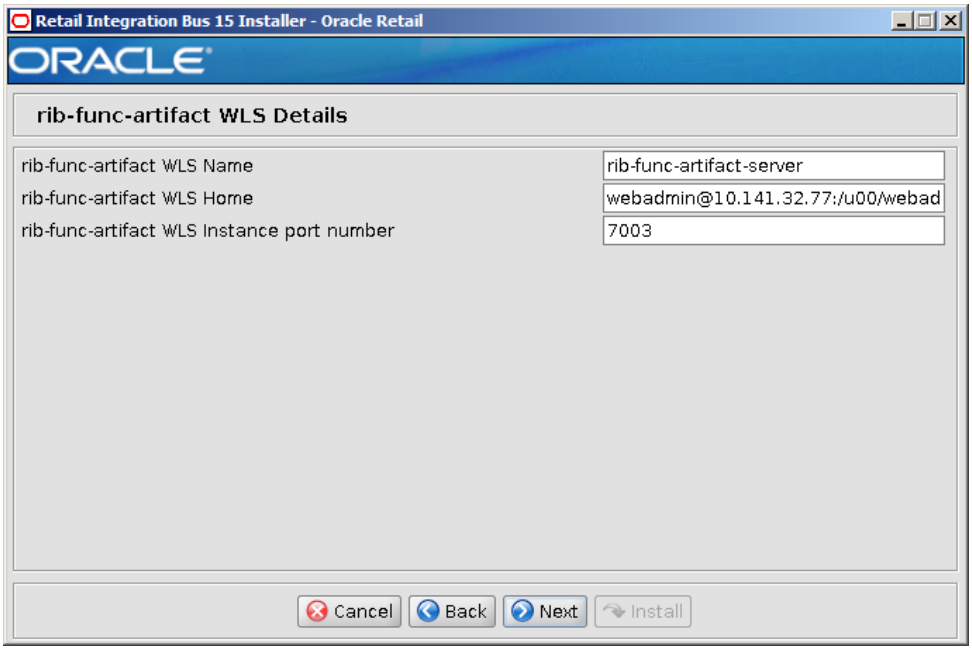
- The rib-tafr Email Notification Details window opens. Provide email details and click **Next**.

NOTE: Email address does not need to be valid.

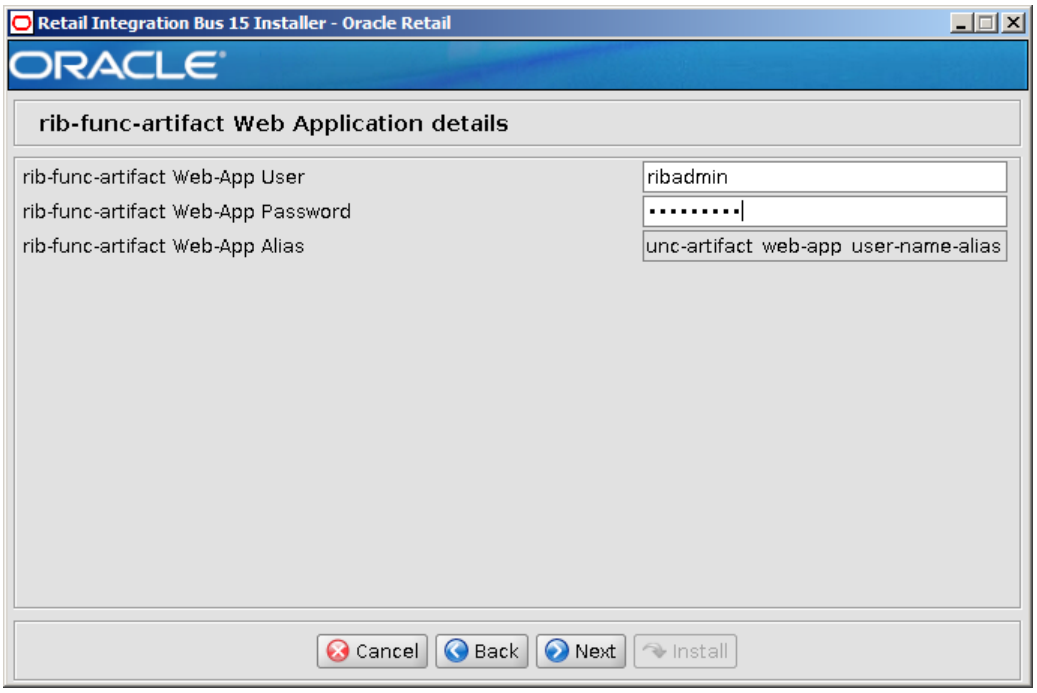


- The rib-func-artifact WLS Details window opens. Provide the required inputs and click **Next**.
  - The inputs should reflect the managed server and port setup in WebLogic for rib-func-artifact's in the RIBDomain.

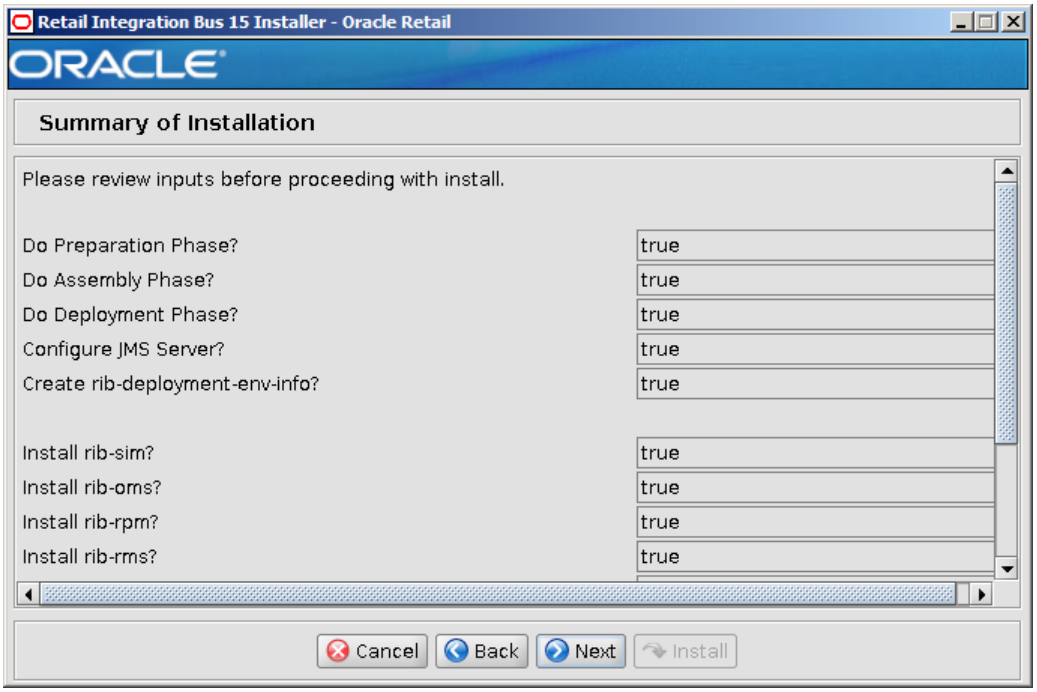
  Example for rib-func-artifact WLS Instance Home:
  - rib-func-artifact WLS Instance Home: webadmin@<floating_ip>:/u00/webadmin/config/domains/wls_rib/RIBDomain/servers/rib-func-artifact-server



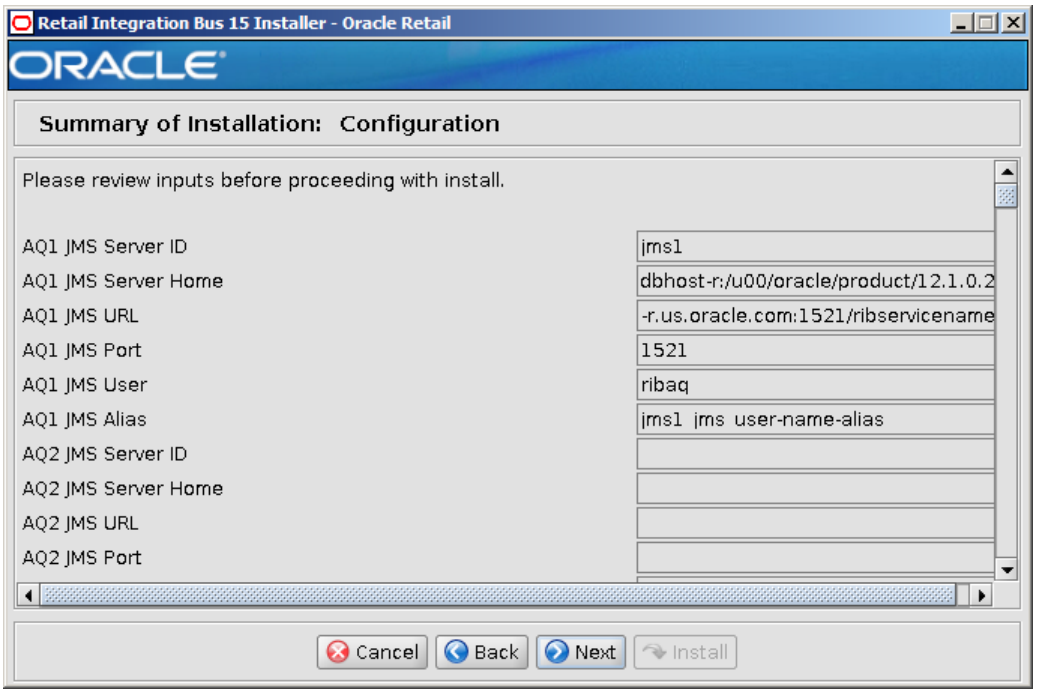
- The rib-func-artifact Web Application details window opens. Provide the desired credentials to be created for the func-artifact admin screen logon. Click **Next**.



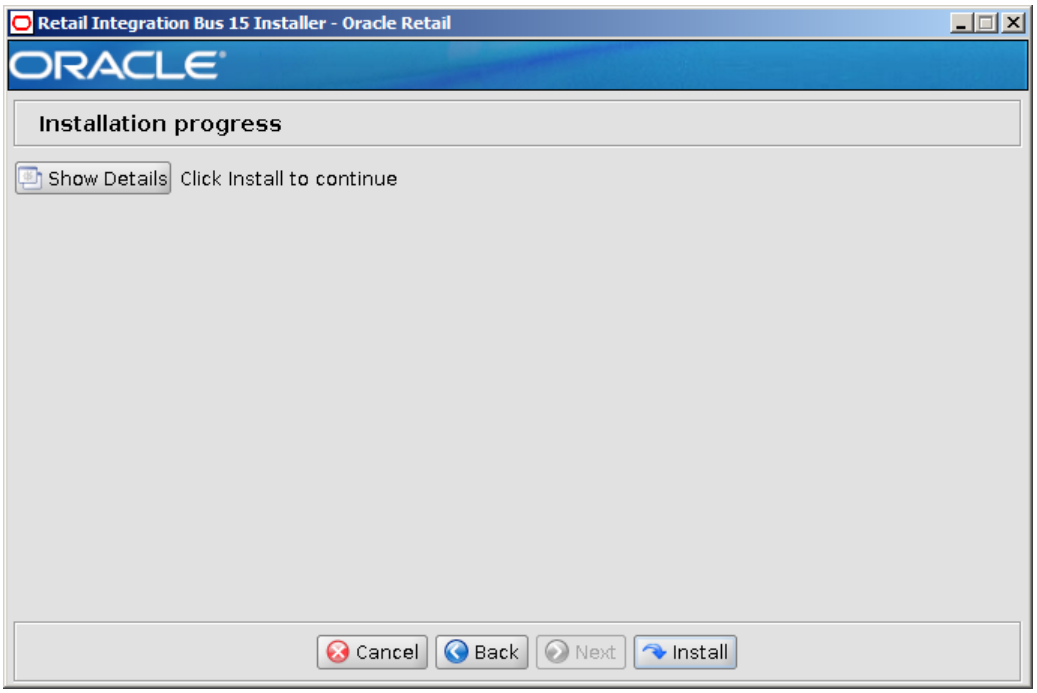
- The Summary of Installation window opens. Review the entries and click **Next** if everything looks correct otherwise use the Back button to go backwards and fix issues.



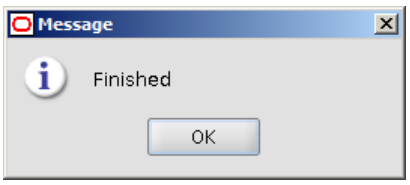
- The Summary of Installation: Configuration window opens. Review the entries and click **Next** if everything looks correct otherwise use the Back button to go backwards and fix issues.



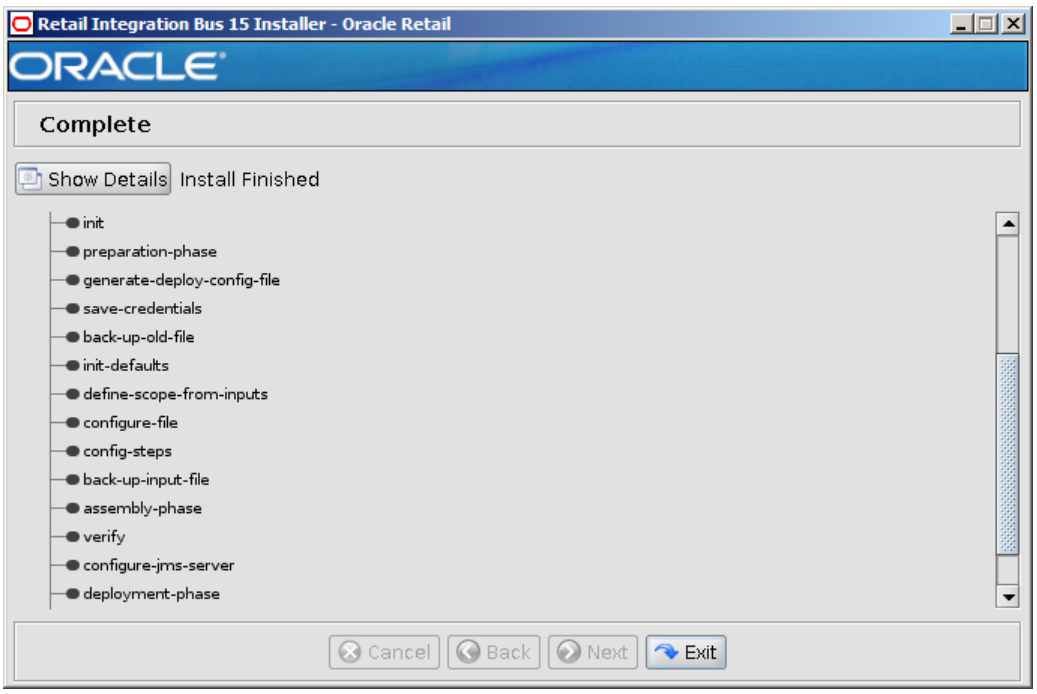
- The Installation progress window opens. Click Install.



- The following prompt is displayed when it finishes installing. Click **OK**.



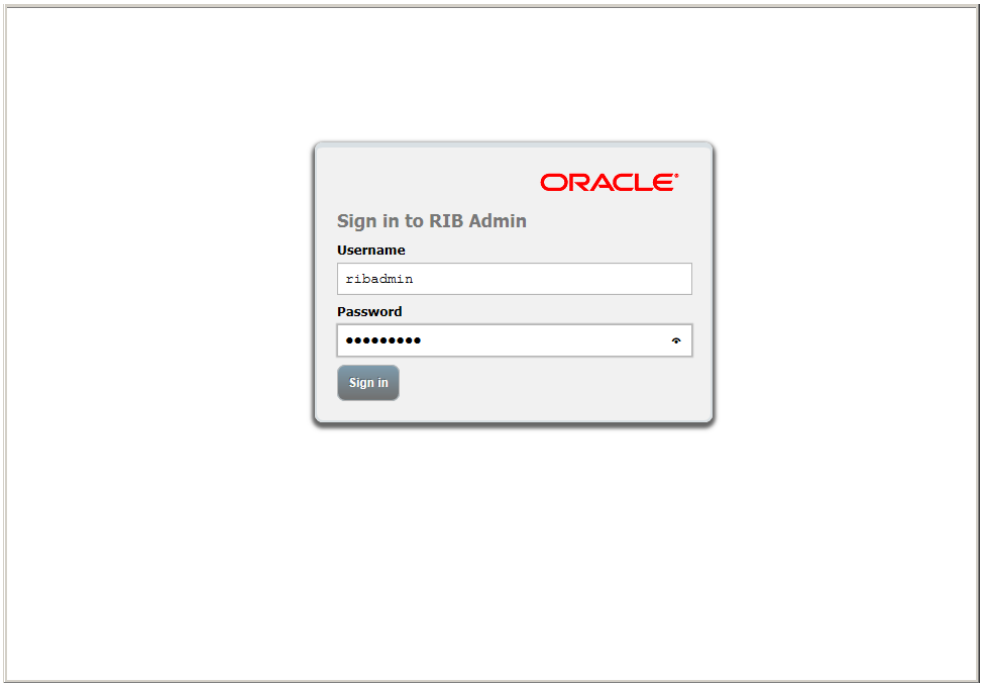
- The Complete window opens. Click **Exit**.



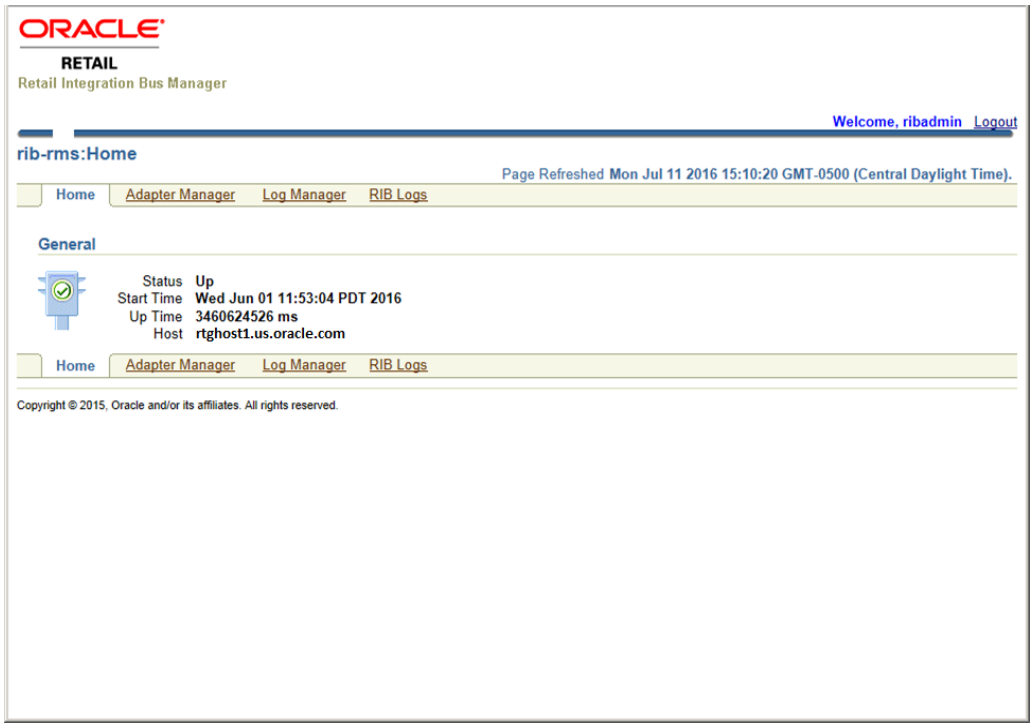
Verification – Verify the RIB Installation

Perform the following steps to validate that the RIB application was installed correctly. The following steps can be executed for each of the rib-<app>-admin-gui URLS.

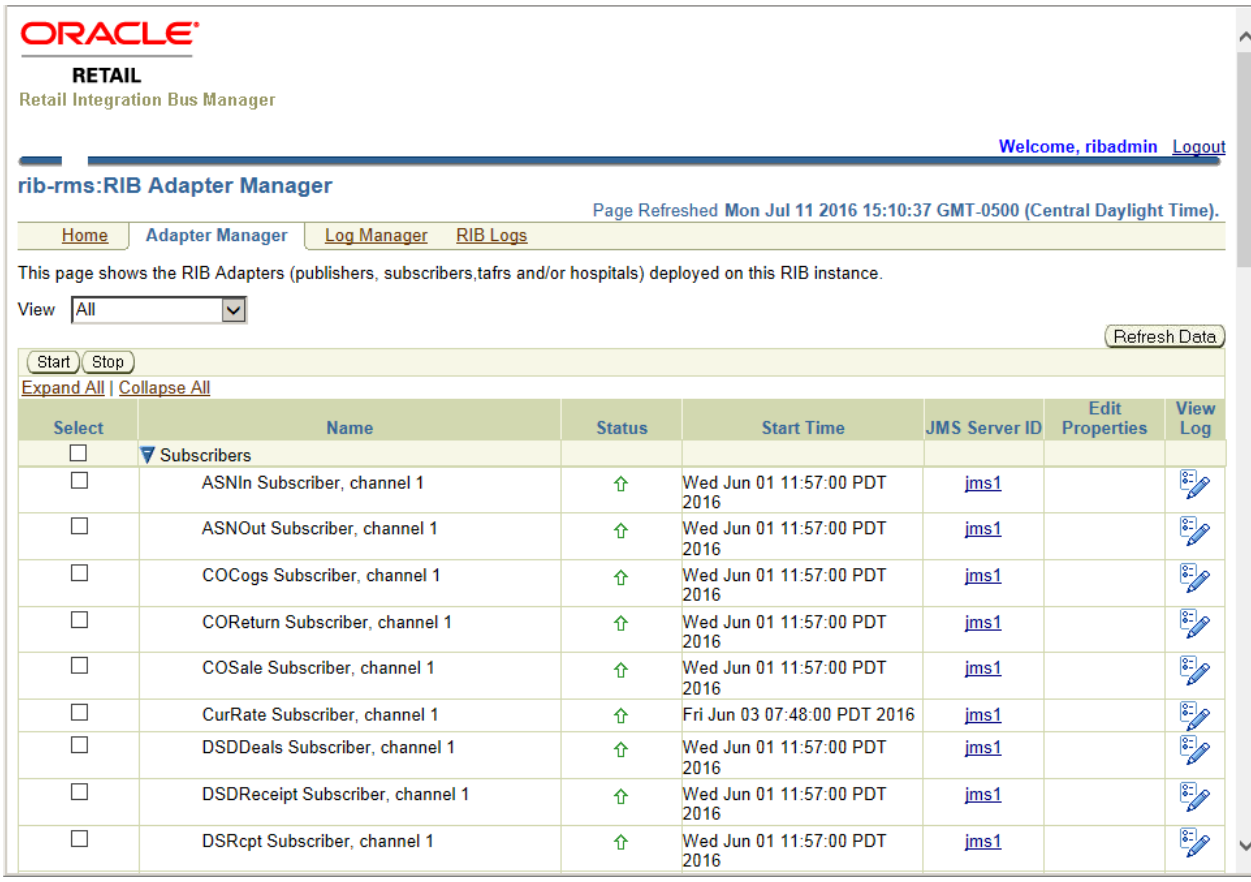
- In a browser window launch the following URL the rib-<app>-admin_gui: http://<floating_ip>:7005/rib-rms-admin-gui/index.jsp, and login using the admin gui credentials that were created during installation.



- The rib-<app>: Home window opens. Click on the Adapter Manager link.



- The Adapter Manager window opens.



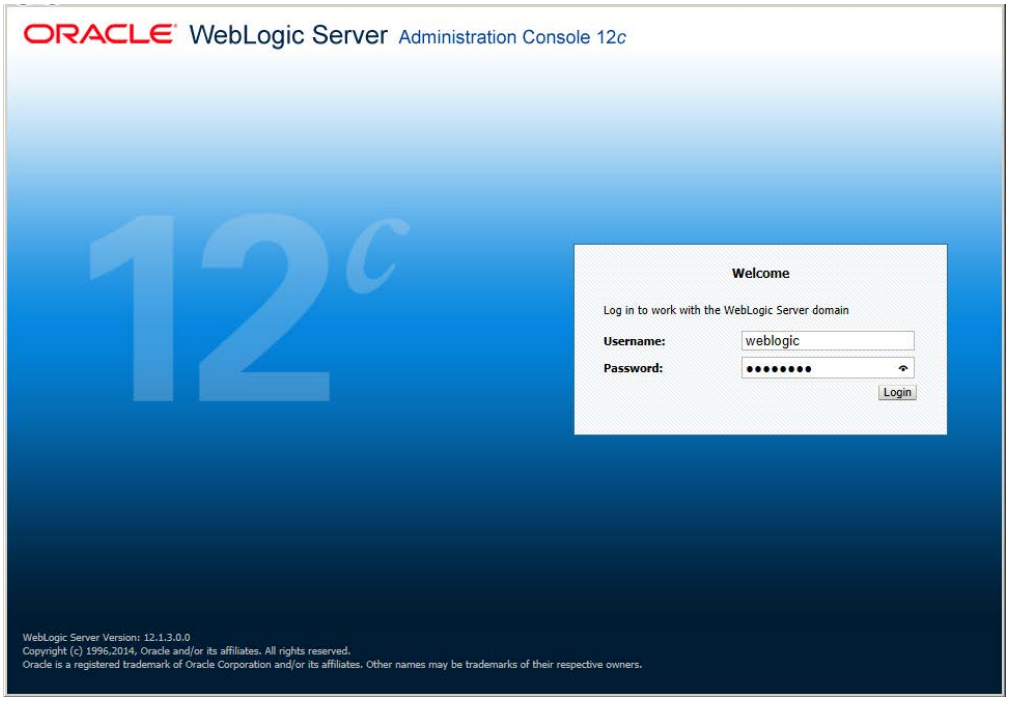
- This is a good indicator that the application has been installed correctly.

> **NOTE**: It is possible that this screen is displayed but messages are not capable of flowing from one application to another. Testing the actual integration should be done after Web Tier updates have been made. Instructions for Web Tier configuration will occur at the end of this document.

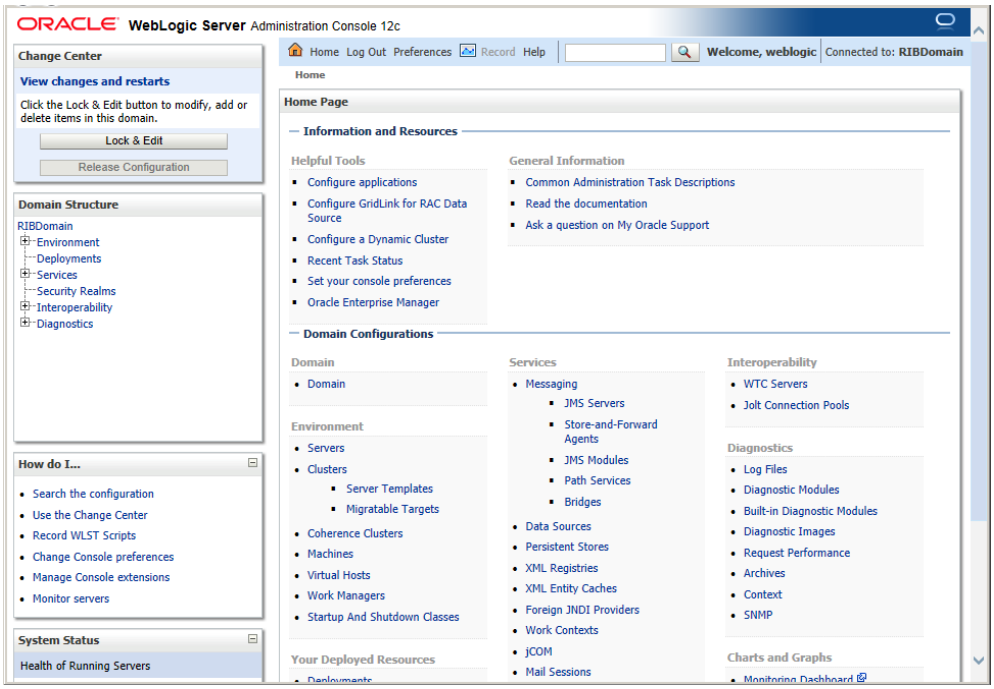
Enable RIB Functional Artifacts Directory

The functionality of accessing the RIB Functional Artifacts via the browser are not readily available after the RIB application is installed. The deployment needs to have its directory enabled in order for it to be displayed. This section provides the instructions to enable the RIB Functional Artifacts deployment.

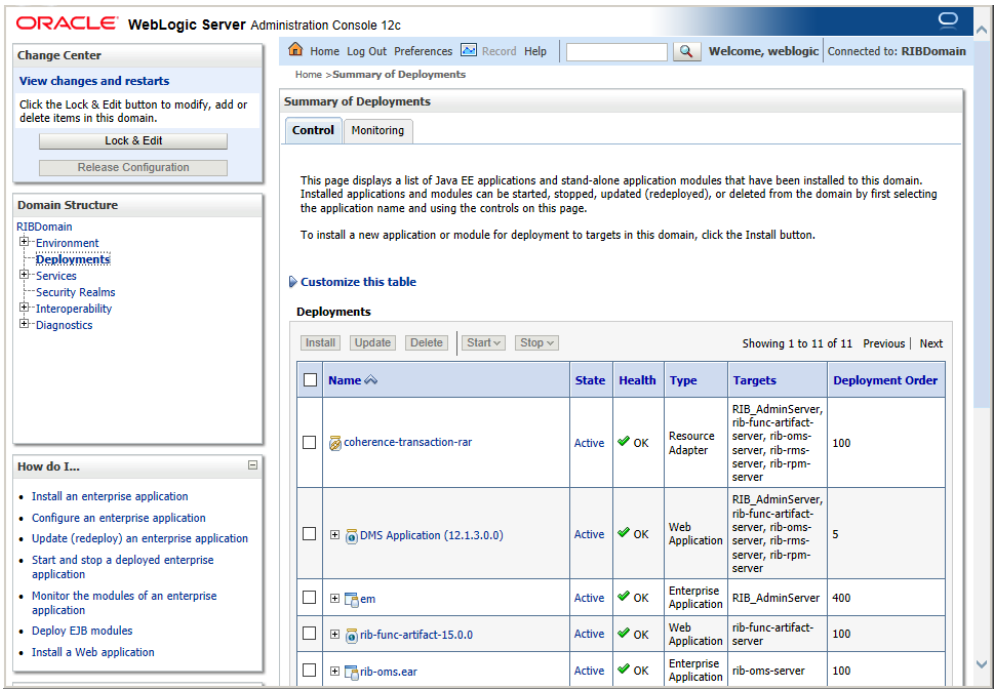
- Log onto the WebLogic Administration Console.



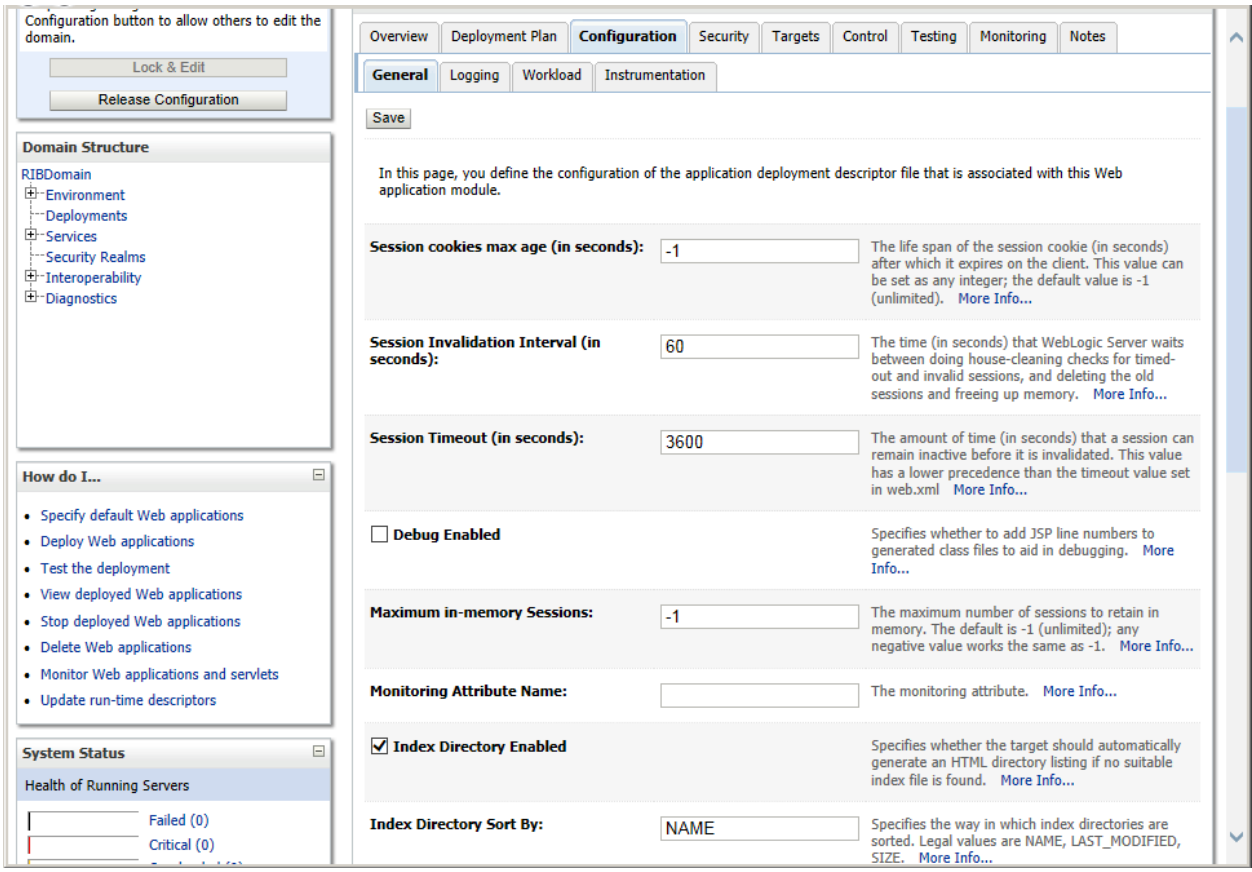
- Click the Deployments link in the left hand navigation pane.



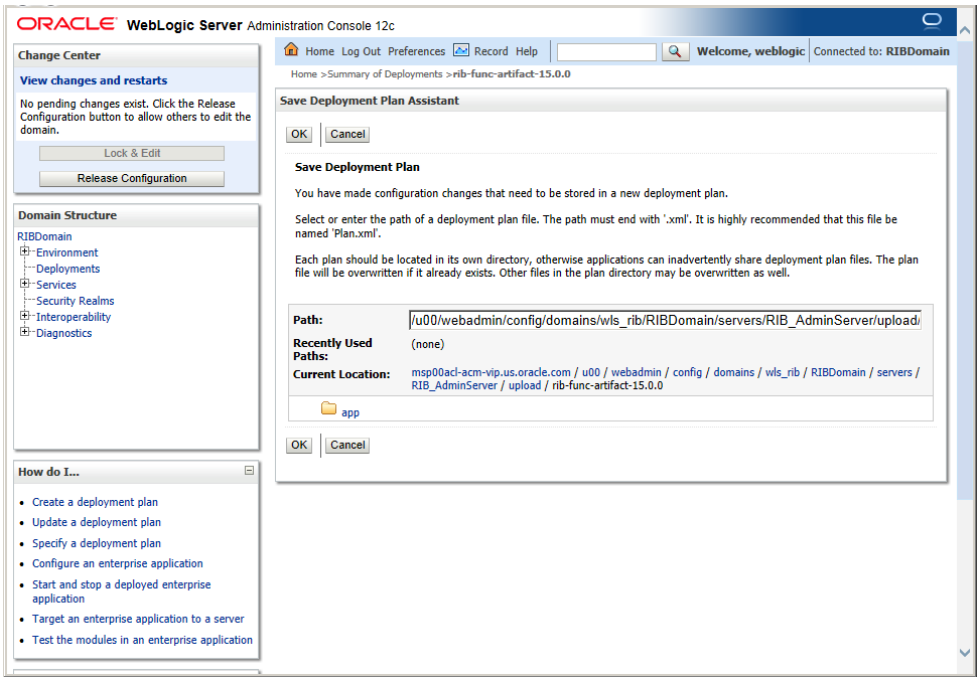
- The Summary of Deployments window opens. Click the **rib-func-artifact-15.0.0** link.



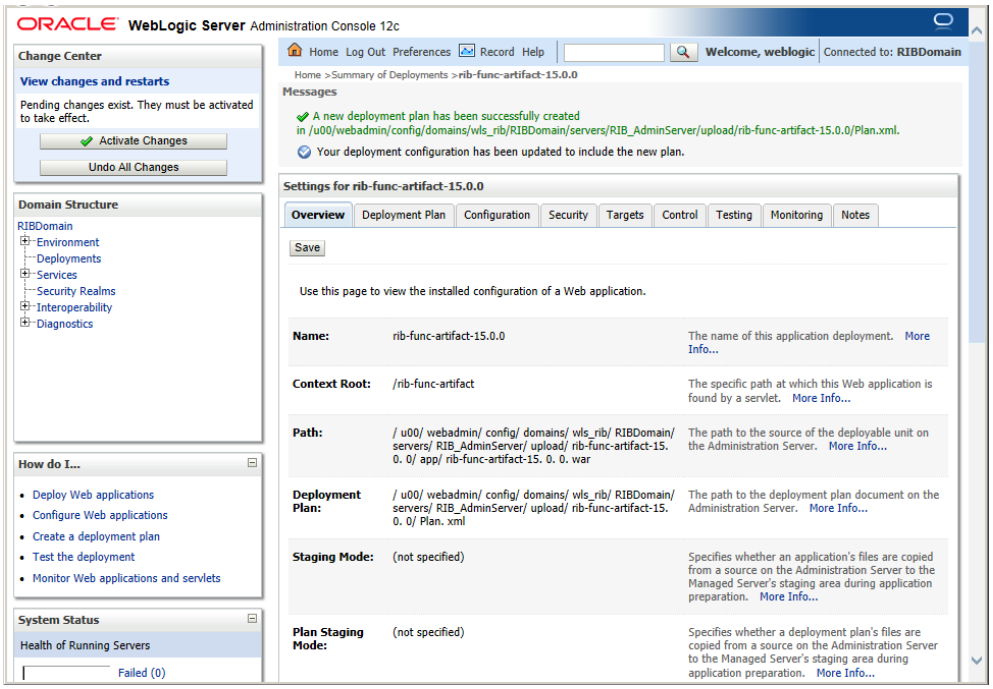
- The Settings for rib-func-artifact-15.0.0 window opens.
  - Navigate to the Configuration > General tab.
  - Click Lock and Edit at the top left hand side of the screen.
  - Select the **Index Directory Enable** option.
  - Click **Save**.



- The Save Deployment Plan Assistant window opens. Click **OK**.



- The Settings for rib-func-artifact-15.0.0 window opens. Click **Activate Changes**.

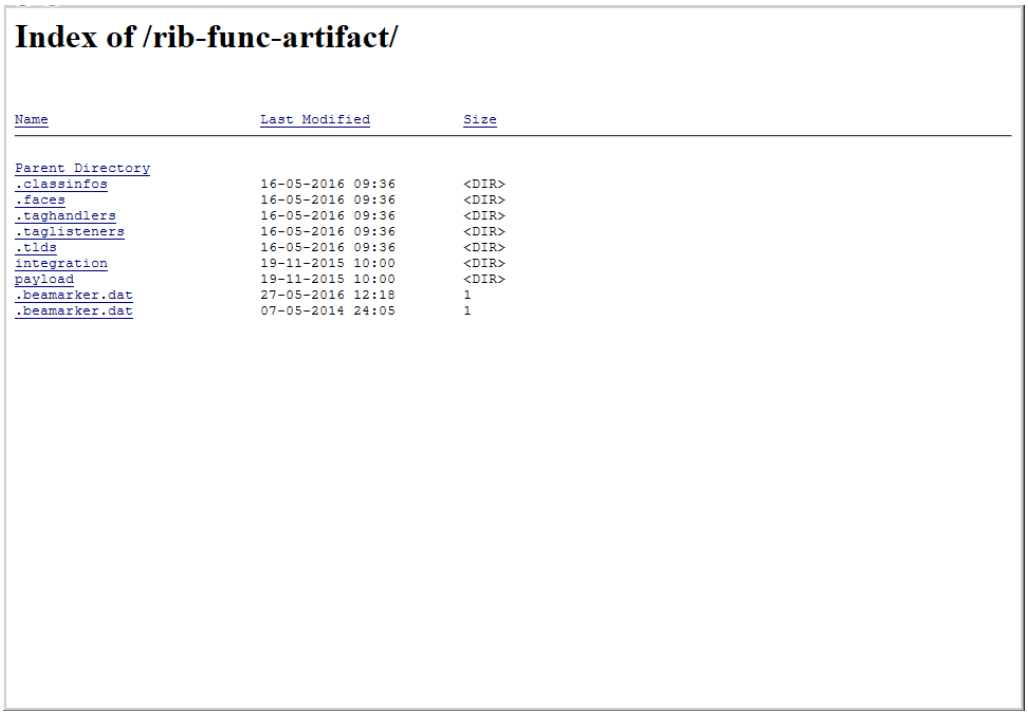


- Enabling the RIB Functional Artifacts directory is complete.

Verification – Verify the RIB Functional Artifacts

Perform the following steps to validate that the RIB Functional Artifacts directory is accessible.

- In a browser window launch the following URL: http://<floating_ip>:7003/rib-func-artifact.



- Validation complete.

Install RDMT

This section describes the steps required to install the RDMT application.

The rdmt installation preparation was already done as a part of the RIB installation preparation. Perform the following steps to install RDMT on ribhost1 only. It will only need to be installed here since the installation is intended to be on a shared mount between ribhost1 and ribhost2.

- Set the following variables by exporting them before running the installer so that the JAVA_HOME is added to the PATH variable:
  - JAVA_HOME
  - PATH

  Example:

  ```
  $ export JAVA_HOME=/u00/webadmin/products/jdk_java
  $ export PATH=$JAVA_HOME/bin:$PATH
  ```

- Change directories to the RDMT directory and execute the configbuilder.sh script supplied with the toolkit (configbuilder.sh). As a part of this you will need to make sure that you are using bash shell, because rdmt was originally written using bash and it sometimes has issues when running in other shells.

  Example:

  ```
  $ cd /u00/webadmin/RIB_INSTALL/Rib1500ForAll15xxApps/rib-home/tools-
  home/rdmt
  $ bash
  $ configbuilder.sh
  ```

  NOTE: Once executed, it checks if the RDMT has been extracted under rib-home/tools-home directory. If so, it fetches all the necessary configuration information from rib-deployment-env-info.xml present under rib-home/deployment-home/conf directory and it automatically completes the RDMT installation.

- RDMT installation is complete.

Verification – Verify RDMT Installation

Perform the following steps to validate the RDMT installation. This test should also verify parts of the RIB installation by running a test message publish and consume.

- Navigate to the rdmt installation directory and launch the RDMT menu.

  Example:

  ```
  $ cd /u00/webadmin/RIB_INSTALL/Rib1500ForAll15xxApps/rib-home/tools-
  home/rdmt
  $./rdmtmenu
  ```

- The Main Menu is displayed as shown below. Select menu item 3 and press enter.

  ```
  RIB Diagnostic & Monitoring Tools
  Main Menu
  ```

```
 This Host: ribhost1.us.oracle.com
 RDMTLOGS : /u00/webadmin/RIB_INSTALL/Rib1500ForAll15xxApps/rib-
home/tools-home/rdmt/RDMTLOGS [eof]
 RMS  DB  : rib-rms_error-hospital-database_user-name-
alias@jdbc:oracle:thin:@dbhost-r:1521/appservicename
 SIM  DB  : rib-sim_error-hospital-database_user-name-
alias@jdbc:oracle:thin:@dbhost-r:1521/appservicename
 TAFR DB  : rib-tafr_error-hospital-database_user-name-
alias@jdbc:oracle:thin:@dbhost-r:1521/ribservicename
 RPM DB   : rib-rpm_error-hospital-database_user-name-
alias@jdbc:oracle:thin:@dbhost-r:1521/appservicename
 OMS DB   : rib-oms_error-hospital-database_user-name-
alias@jdbc:oracle:thin:@dbhost-r:1521/ribservicename
 JMS Type : ojms
 JMS URL  : jdbc:oracle:thin:@dbhost-r.us.oracle.com:1521/ribservicename
 JMX CFG  : jmx1.conf
 WLS     : 10.141.32.77:7005
 Instance : rib-rms-server
 RIB APP  : rib-rms

1  - WLS/JMX Utilities Menu        7 - RIB Health Menu         13 - RIB
Admin menu
2  - JMS Utilities Menu            8 - Hospital Scan Menu      14 -
Archive RDMTLOGS
3  - PUB/SUB/TAFR Msg Menu         9 - JMS Topic Scan          15 - View
history log
4  - Switch WLS/JMX Config         10 - JMS Switch             16 - RIB
Health scan
5  - Scan RIB Logs                 11 - JMS Config             17 - Config
Multi Channels
6  - Scan RIB Logs - delta           12 - View Scan Log Exceptions

99 - Logout
Selection: 3
```

- The PUB/SUB/TAFR Utilities Submenu is displayed as shown below. Select menu item 5, press enter and confirm y if prompted.

```
RIB Diagnostic & Monitoring Tools
PUB/SUB/TAFR Utilities Submenu

 This Host: ribhost1.us.oracle.com
 RDMTLOGS : /u00/webadmin/RIB_INSTALL/Rib1500ForAll15xxApps/rib-
home/tools-home/rdmt/RDMTLOGS [eof]
home/rdmt/RDMTLOGS [eof]
 RMS  DB  : rib-rms_error-hospital-database_user-name-
alias@jdbc:oracle:thin:@dbhost-r:1521/appservicename
 SIM  DB  : rib-sim_error-hospital-database_user-name-
alias@jdbc:oracle:thin:@dbhost-r:1521/appservicename
 TAFR DB  : rib-tafr_error-hospital-database_user-name-
alias@jdbc:oracle:thin:@dbhost-r:1521/ribservicename
 RPM DB   : rib-rpm_error-hospital-database_user-name-
alias@jdbc:oracle:thin:@dbhost-r:1521/appservicename
 OMS DB   : rib-oms_error-hospital-database_user-name-
alias@jdbc:oracle:thin:@dbhost-r:1521/ribservicename
 JMS Type : ojms
 JMS URL  : jdbc:oracle:thin:@dbhost-r.us.oracle.com:1521/ribservicename
 JMX CFG  : jmx1.conf
 WLS     : 10.141.32.77:7005
 Instance : rib-rms-server
 RIB APP  : rib-rms
```

```
1 - Publish Msg Utility       4 - SUB Receipt test msg         7 - PUB
Multiple - dir
2 - Subscribe Msg Utility     5 - PUB Receipt test msg         8 - EJB
Publish Utility
3 - JMS Topic Scan            6 - PUB Multiple - file          9 - EJB
PUB Multiple - dir
10 - EJB PUB Multiple - file  11 - TAFR Msg Utility

99 - Main Menu

Selection: 5
```

- The PUB/SUB/TAFR Utilities Submenu is still displayed as shown below. Select menu item 4, press enter and confirm y if prompted.

```
 RIB Diagnostic & Monitoring Tools
 PUB/SUB/TAFR Utilities Submenu

 This Host: ribhost1.us.oracle.com
 RDMTLOGS : /u00/webadmin/RIB_INSTALL/Rib1500ForAll15xxApps/rib-
home/tools-home/rdmt/RDMTLOGS [eof]
home/rdmt/RDMTLOGS [eof]
 RMS  DB  : rib-rms_error-hospital-database_user-name-
alias@jdbc:oracle:thin:@dbhost-r:1521/appservicename
 SIM  DB  : rib-sim_error-hospital-database_user-name-
alias@jdbc:oracle:thin:@dbhost-r:1521/appservicename
 TAFR DB  : rib-tafr_error-hospital-database_user-name-
alias@jdbc:oracle:thin:@dbhost-r:1521/ribservicename
 RPM DB   : rib-rpm_error-hospital-database_user-name-
alias@jdbc:oracle:thin:@dbhost-r:1521/appservicename
 OMS DB   : rib-oms_error-hospital-database_user-name-
alias@jdbc:oracle:thin:@dbhost-r:1521/ribservicename
 JMS Type : ojms
 JMS URL  : jdbc:oracle:thin:@dbhost-r.us.oracle.com:1521/ribservicename
 JMX CFG  : jmx1.conf
 WLS     : 10.141.32.77:7005
 Instance : rib-rms-server
 RIB APP  : rib-rms

1 - Publish Msg Utility       4 - SUB Receipt test msg         7 - PUB
Multiple - dir
2 - Subscribe Msg Utility     5 - PUB Receipt test msg         8 - EJB
Publish Utility
3 - JMS Topic Scan            6 - PUB Multiple - file          9 - EJB
PUB Multiple - dir
10 - EJB PUB Multiple - file  11 - TAFR Msg Utility

99 - Main Menu

Selection: 4

-j ojms -x jdbc:oracle:thin:@dbhost-r.us.oracle.com:1521/ribservicename -w
/u00/webadmin/RIB_INSTALL/Rib1500ForAll15xxApps/rib-home/deployment-
home/conf/security -c DEFAULT_KEY_PARTITION_NAME -k jms1_jms_user-name-
alias -t etReceiving -n
/u00/webadmin/RIB_INSTALL/Rib1500ForAll15xxApps/rib-home/tools-
home/rdmt/RDMTLOGS/tmp/msg.tmp -tv 1 -to 5
Enter to Continue? y/n/q [y]: y
```

- No errors should be shown and the logs can be reviewed to make sure no errors are in there. The location of the log files are specified as part of the menu in the output above or see example below:

  Example:

```
RDMTLOGS : /u00/webadmin/RIB_INSTALL/Rib1500ForAll15xxApps/rib-home/tools-
home/rdmt/RDMTLOGS
```

- RDMT verification complete.

4

Installing RTG Applications

The chapter contains instructions needed to install and configure some of the RTG applications. Included in the instructions are steps to setup the infrastructure, install the applications and verify the installations.

Install WebLogic Server 12c with ADF on rtghost1 and rtghost2

This section describes the steps required for installing WebLogic 12.1.3 on rtghost1 and rtghost2.

Perform the following steps to install WebLogic 12.1.3 on rtghost1 and rtghost2. Unlike the RIB installation the rtghost's are configured as active/active where all the nodes of the clustered environment will have the applications installed.

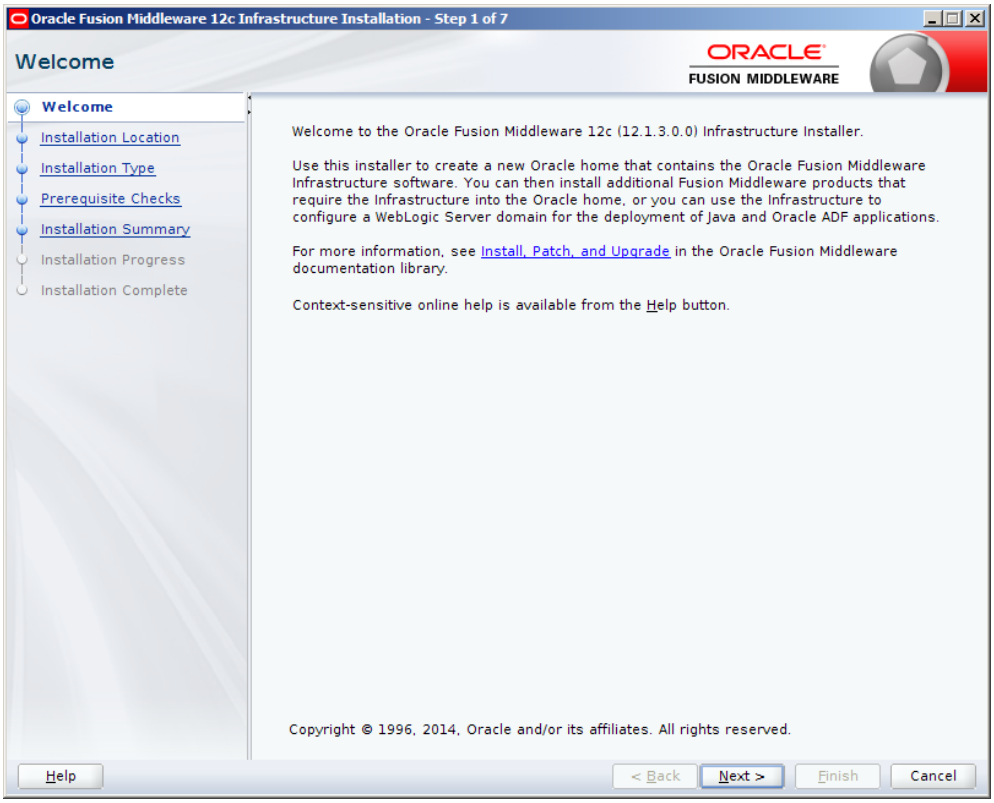
- Download software installation for WebLogic 12.1.3 that is bundled with ADF and place it in an installation staging directory on rtghost1 and rtghost2.

  Example:
  <INSTALL_DIR>/WLS-ADF12c_12.1.3.
- Set the following variables by exporting them before running the installer so that the JAVA_HOME is added to the PATH variable:
  - JAVA_HOME
  - PATH
  - DISPLAY

  Example:
  ```
  $ export JAVA_HOME=/u00/webadmin/products/jdk_java
  $ export PATH=$JAVA_HOME/bin:$PATH
  $ export DISPLAY=<pc_ip_address:0.0>
  ```
- Run the following commands to run the installer:

  Example:
  ```
  $ cd <INSTALL_DIR>/ WLS-ADF12c_12.1.3
  $ java -jar ./fmw_12.1.3.0.0_infrastructure.jar
  ```
- The Installation Inventory Setup window displays. Update the Inventory Directory and Operation System Group fields according to the instructions on the screen and click **Next**. See screen shot for examples.

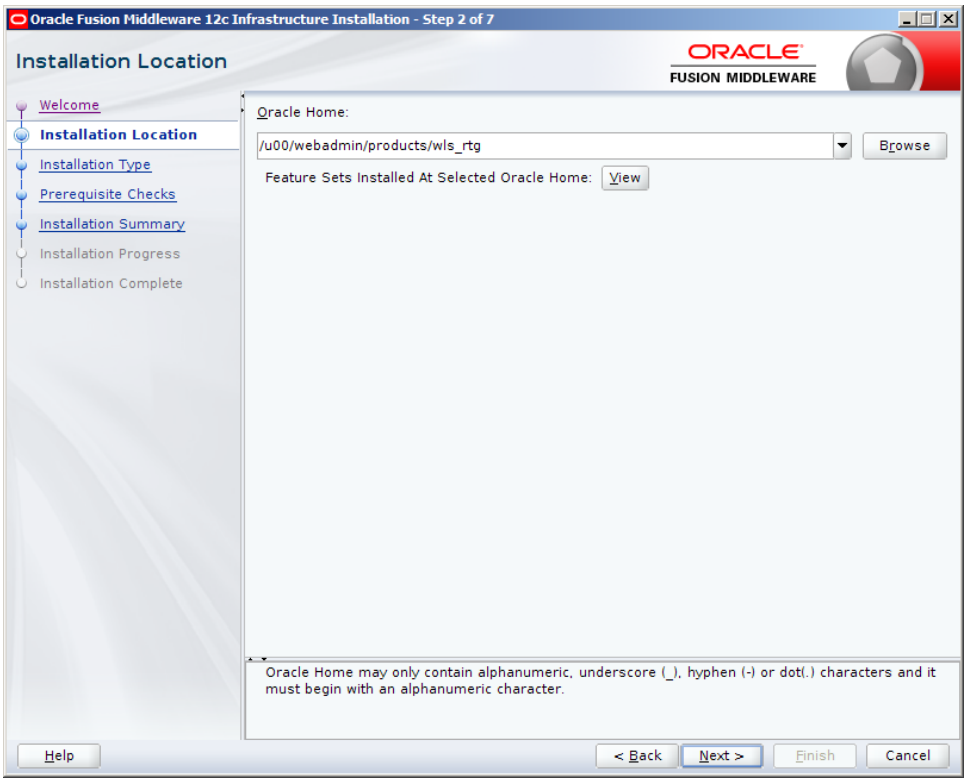
- The Welcome window of the WebLogic 12c installer displays. Click **Next**.



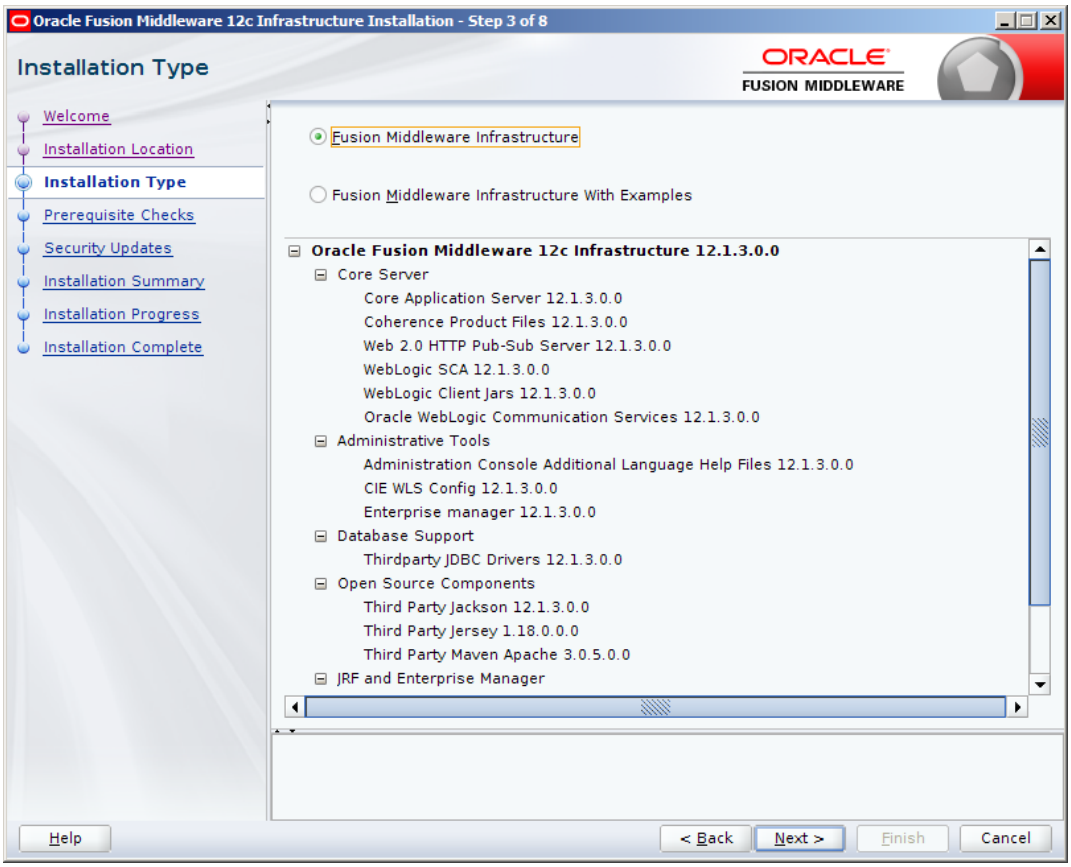
- The Installation Location window displays. Enter the path to where the ORACLE_HOME for RTG should be and click **Next**.

  Example:

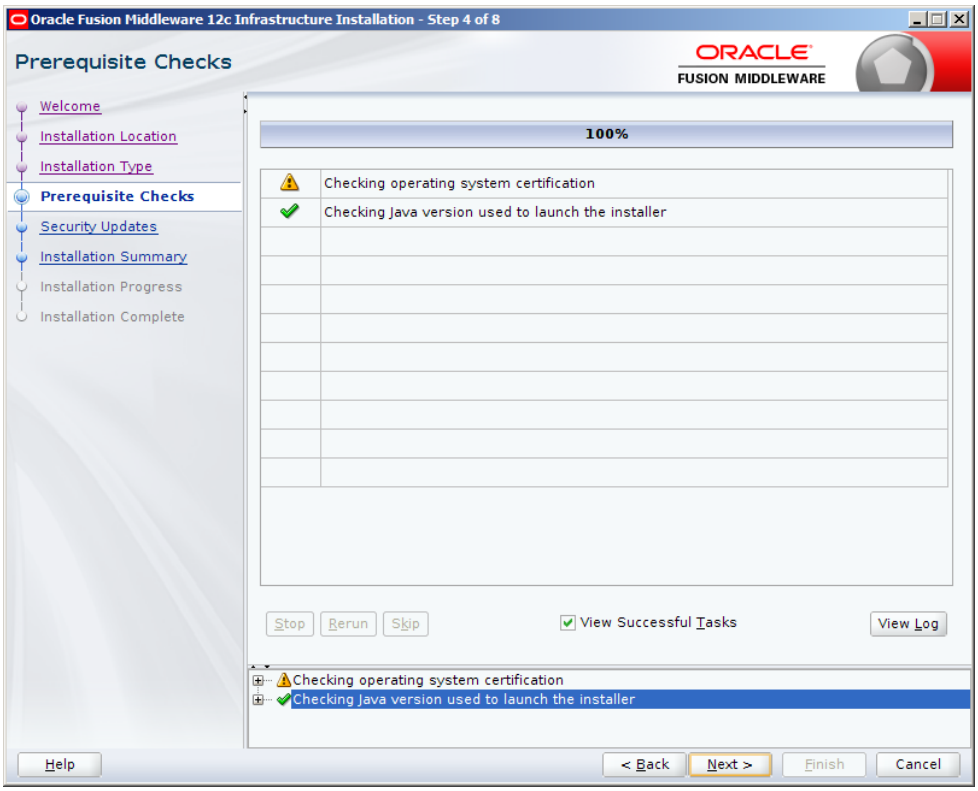
  /u00/webadmin/products/wls_rtg



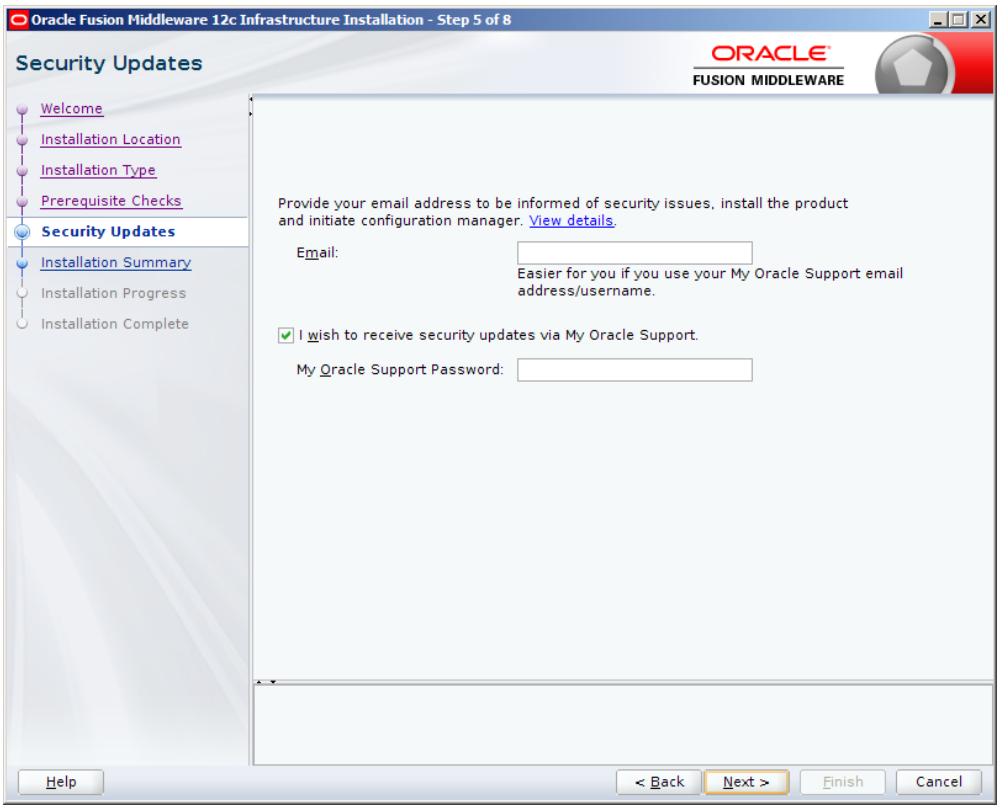
- The Installation Type window opens. Select **Fusion Middleware Infrastructure,** and click **Next**.



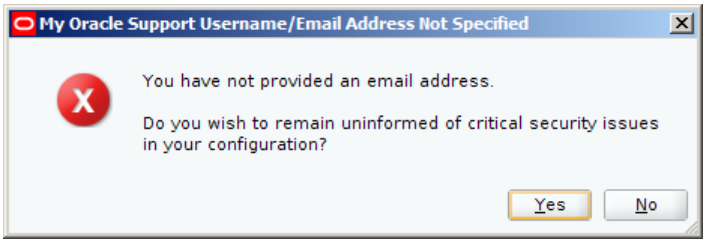
- The Prerequisite Checks window opens. When the checks are completed – click **Next**.



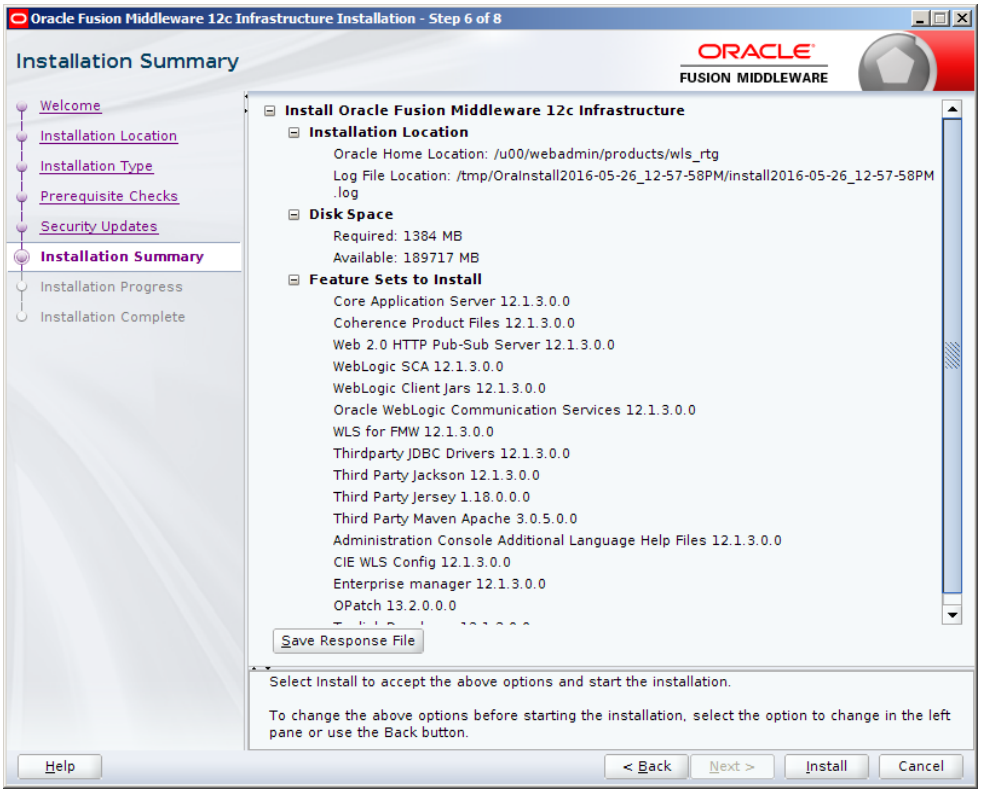
- The Security Updates window opens. Leave the screen blank and click **Next**.



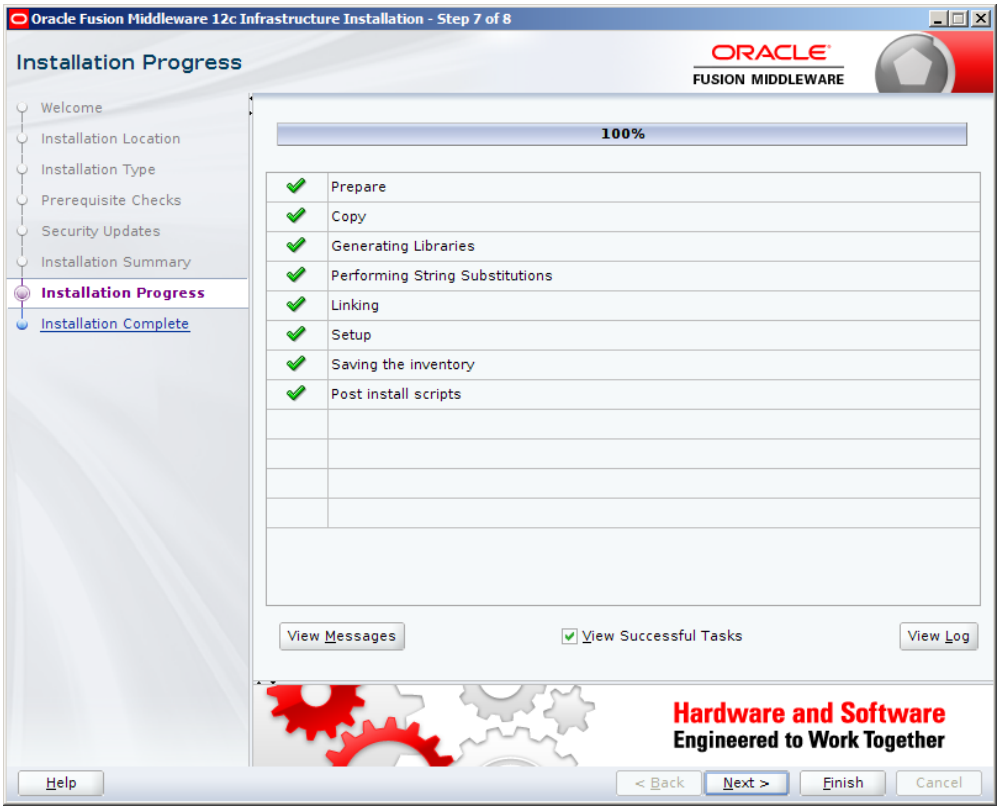
- A confirmation prompt is displayed. Click Yes to continue.



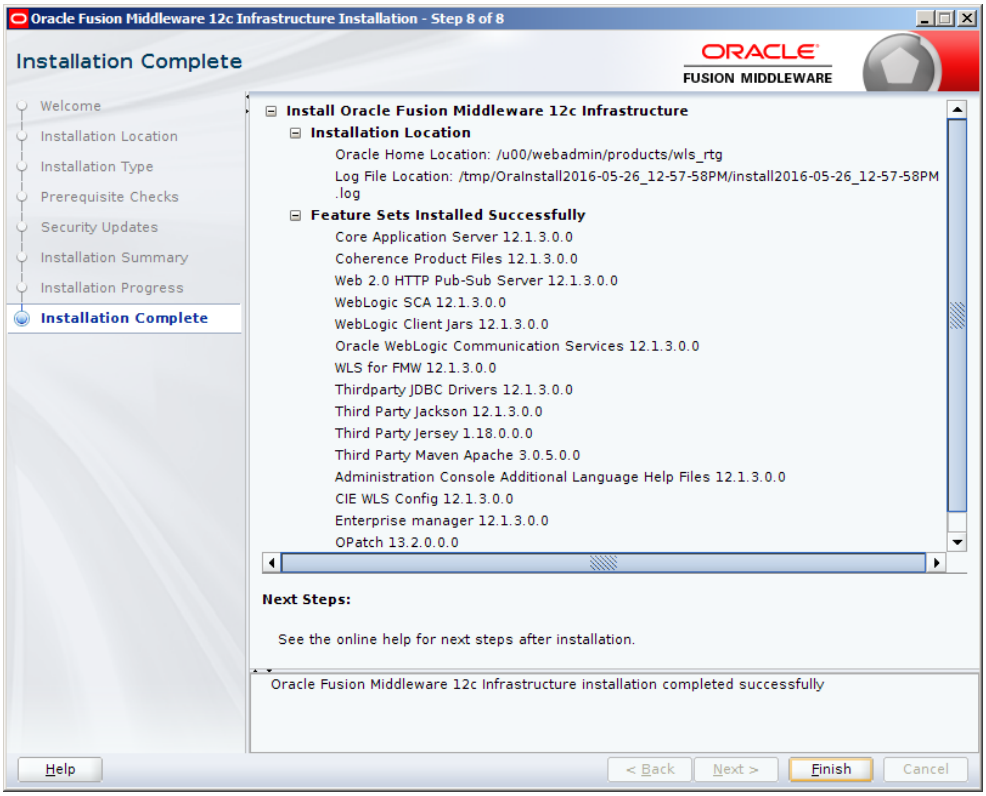
- The Installation Summary window opens. Confirm that Installation Location is correct and click **Next**.



- The Installation Progress window opens. Click **Next** once it has reached 100%.



- The Installation Complete window opens. Click **Finish**.



- Installation of WebLogic 12c is complete.

Creating Database Repository for RTGDomain

This chapter describes the steps to create the proper schemas required for the managed servers of the RTG installation. In order to run the Repository Creation Utility (RCU) - It is required to use the database credentials for the SYS user. Perform the following steps to create a database repository using Repository Creation Utility:

- Set the DISPLAY variable before running the installer:

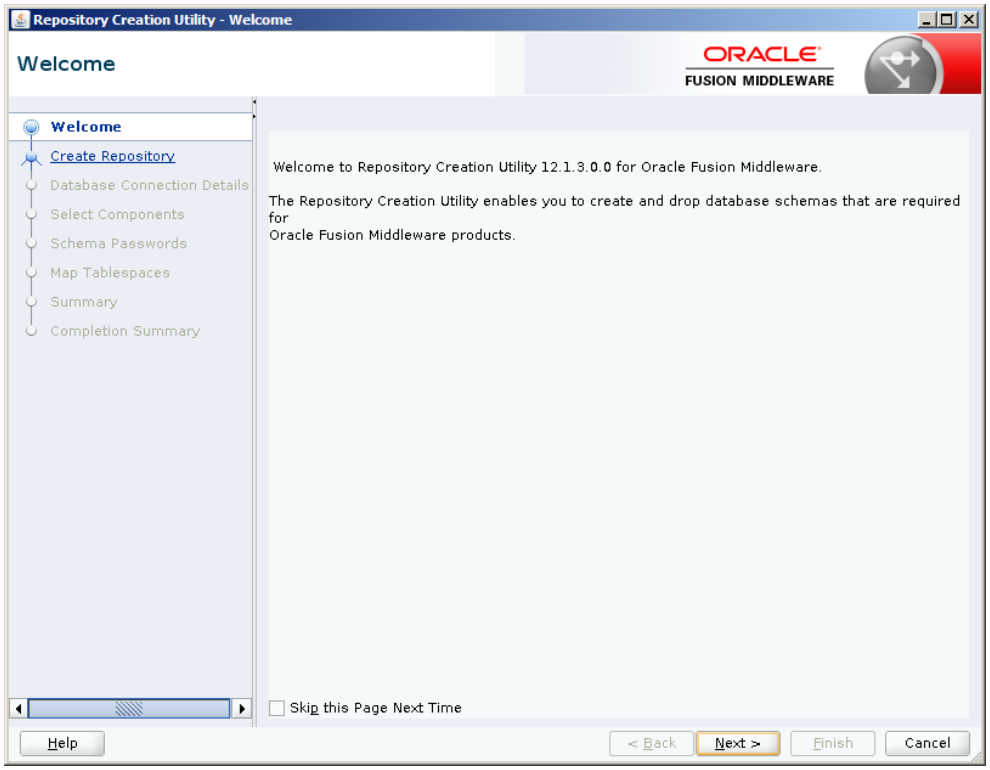
  Example:

  ```
  $ export DISPLAY=<pc_ip_address:0.0>
  ```

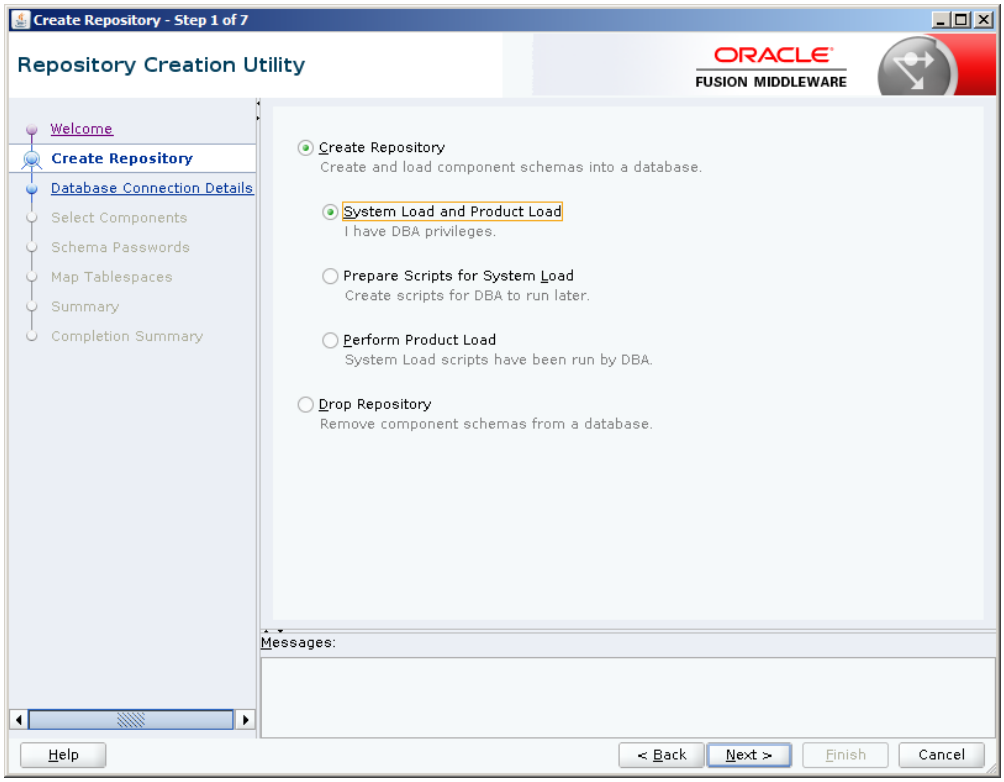
- RCU is packaged with 12c WebLogic installations so it is easy to make sure you are running the correct version of RCU. Navigate to the location of the rcu executable and run the installer by executing the following commands:

  ```
  $ cd /u00/webadmin/products/wls_rtg/oracle_common/bin
  $ ./rcu
  ```

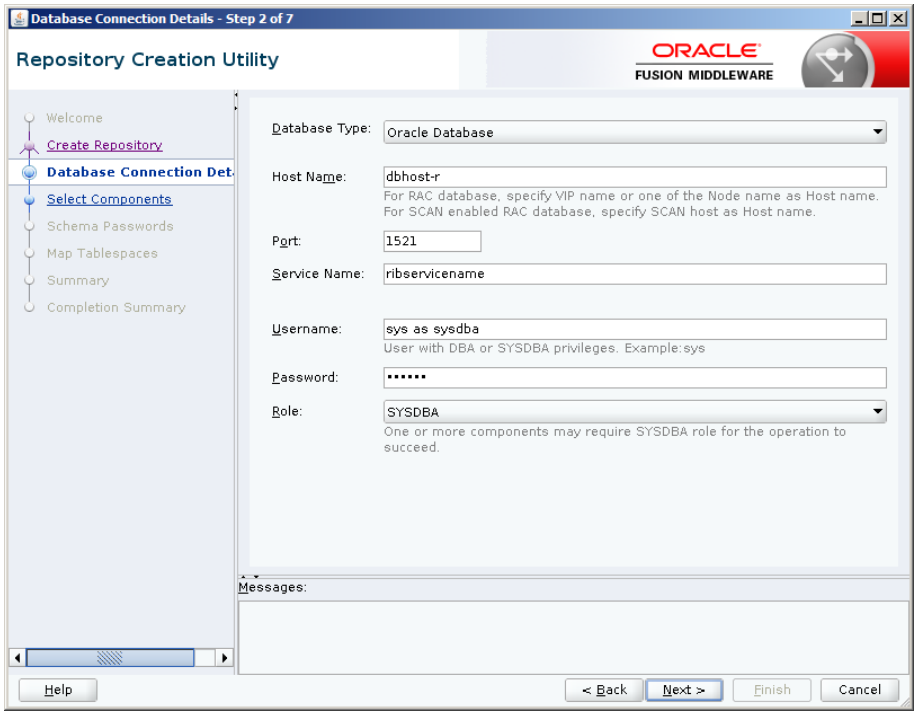
- The Welcome Window displays. Click **Next**.



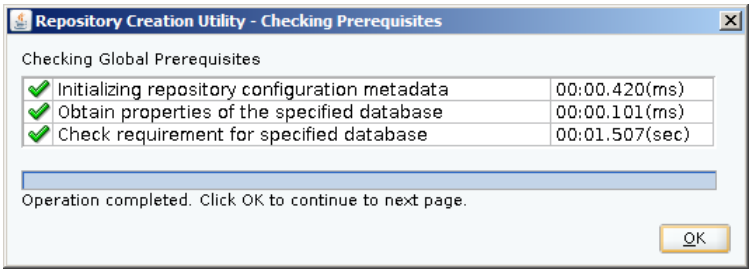
- The Create Repository window displays. Select the **Create** option, and click **Next**.



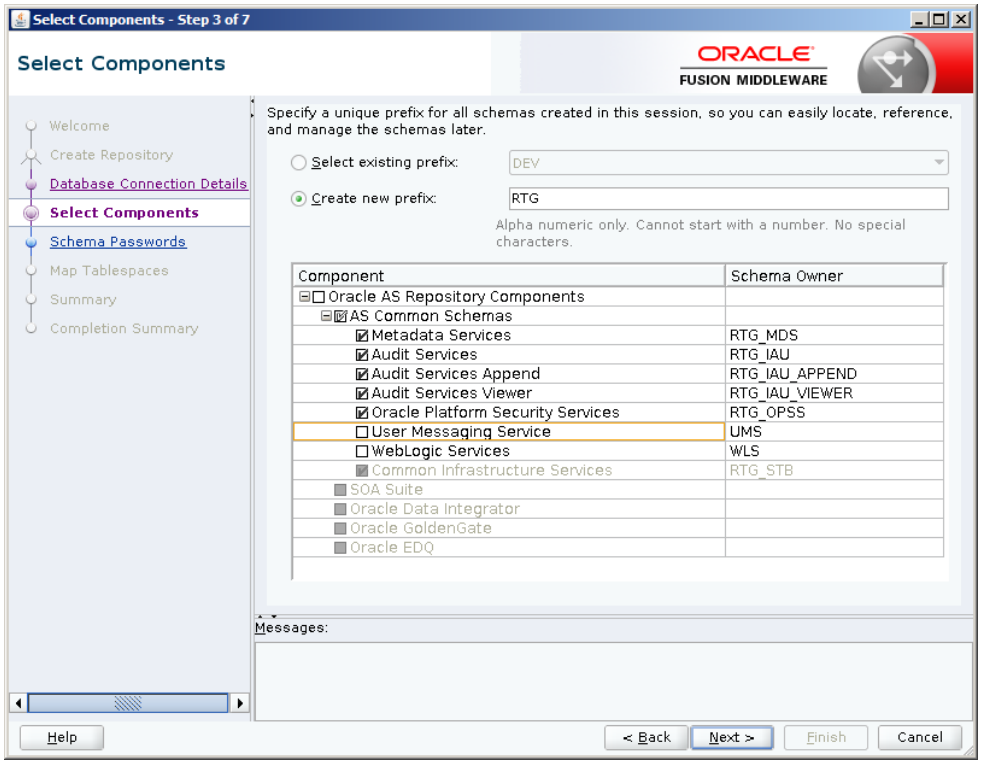
- The Database Connection Details window opens. Provide the database details and click **Next**:
  - Host Name: dbhost-r
  - Port: 1521
  - Service Name: ribservicename
  - Username (must have DBA or SYSDBA privileges)
  - Password



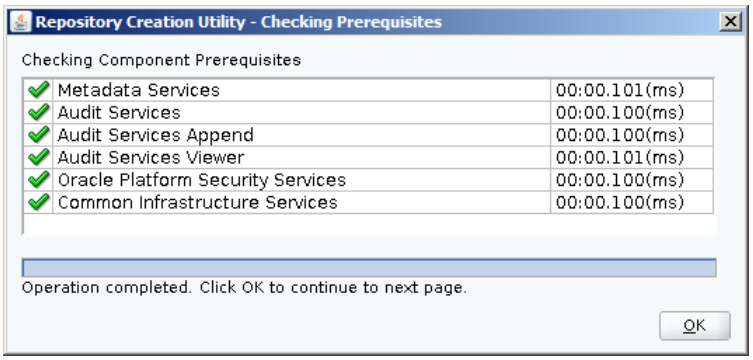
- RCU checks database prerequisites. Click **OK**.



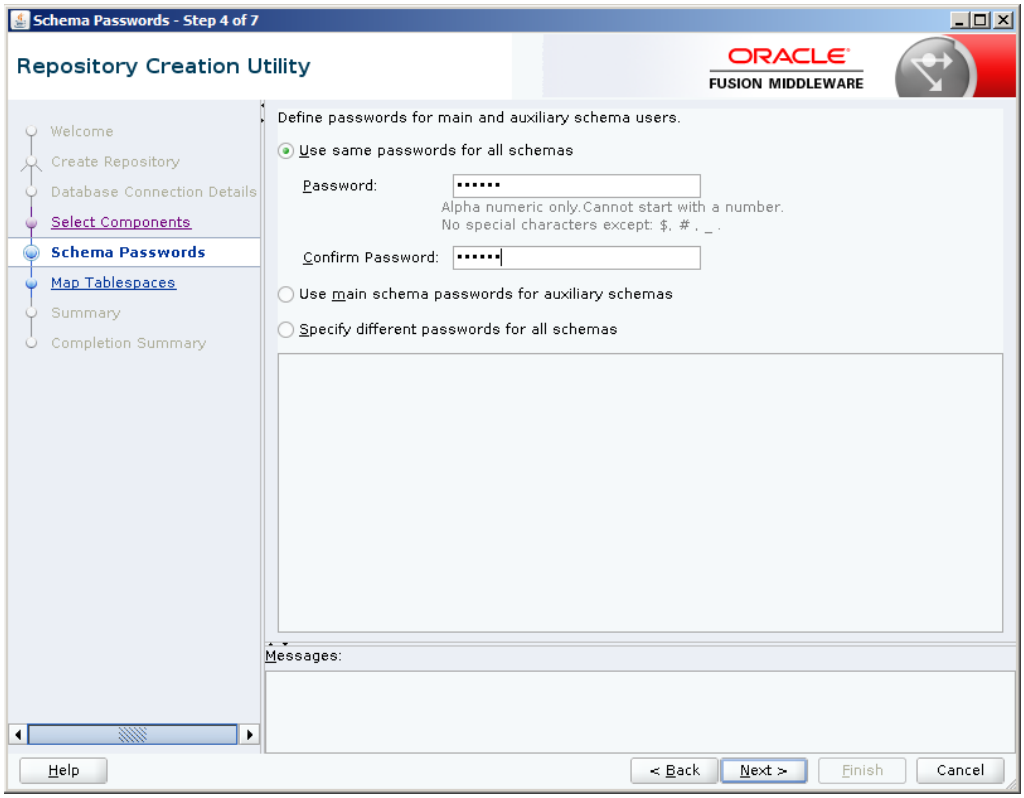
- The Select Components window opens. Select just the following items below and click **Next**.
  - Select **Create a new Prefix: RTG**
  - Select the following components:
    a. Metadata Services
    b. Audit Services
    c. Audit Services Append
    d. Audit Services Viewer
    e. Oracle Platform Security Services
    f. Common Infrastructure Services



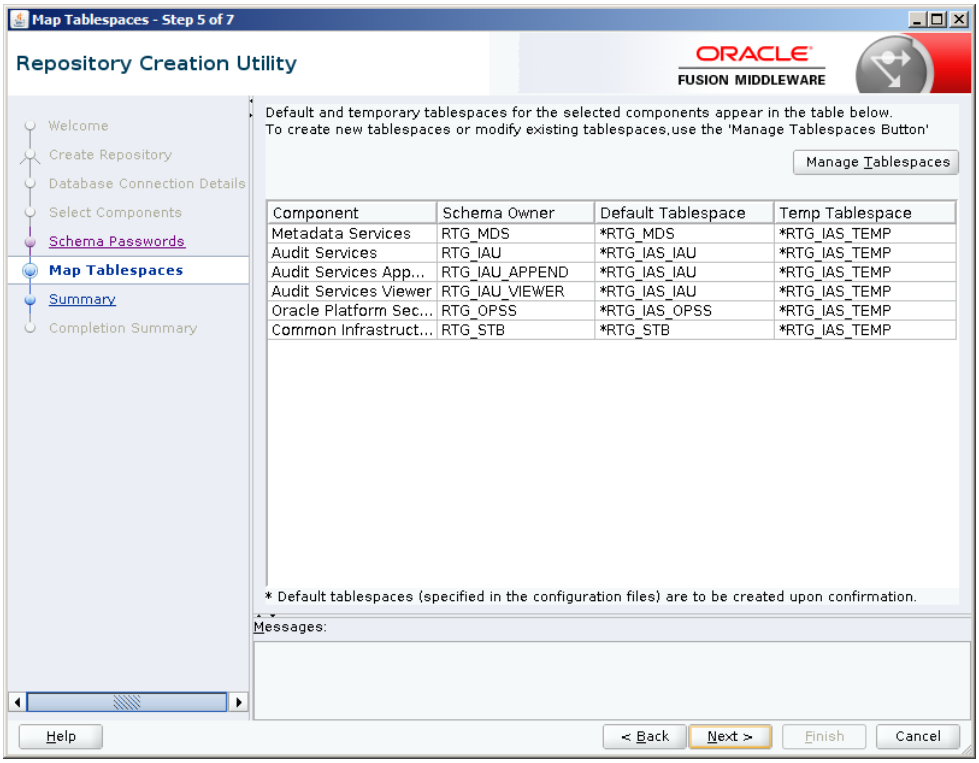
- RCU will check database prerequisites to make sure it can create the selected components. Click **OK**.



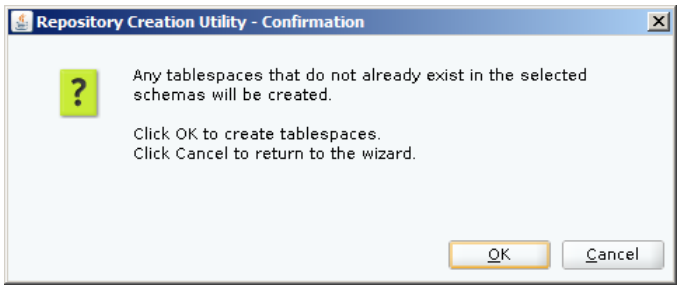
- The Schema Passwords window opens.
  - Select the Use same passwords for all schemas option.
  - Enter in the password.
  - Click **Next**.



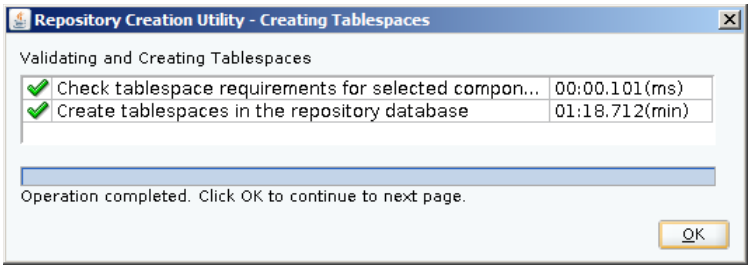
- The Map Tablespaces window opens. Click **Next**.



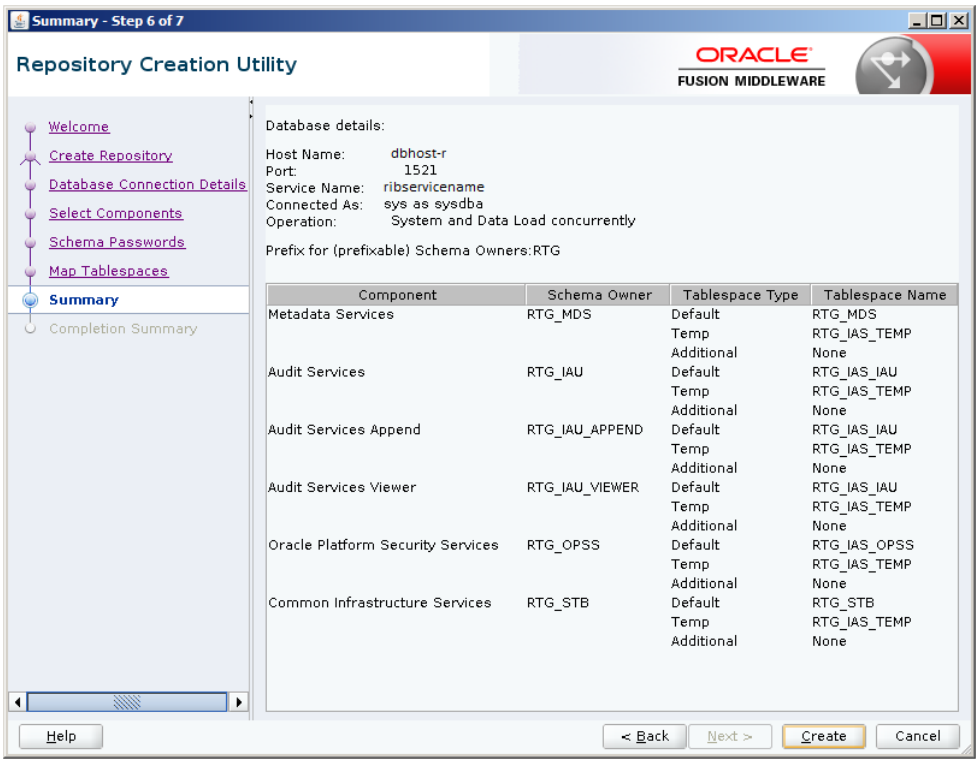
- The Repository Creation Utility - Confirmation window displays. Click **OK**.



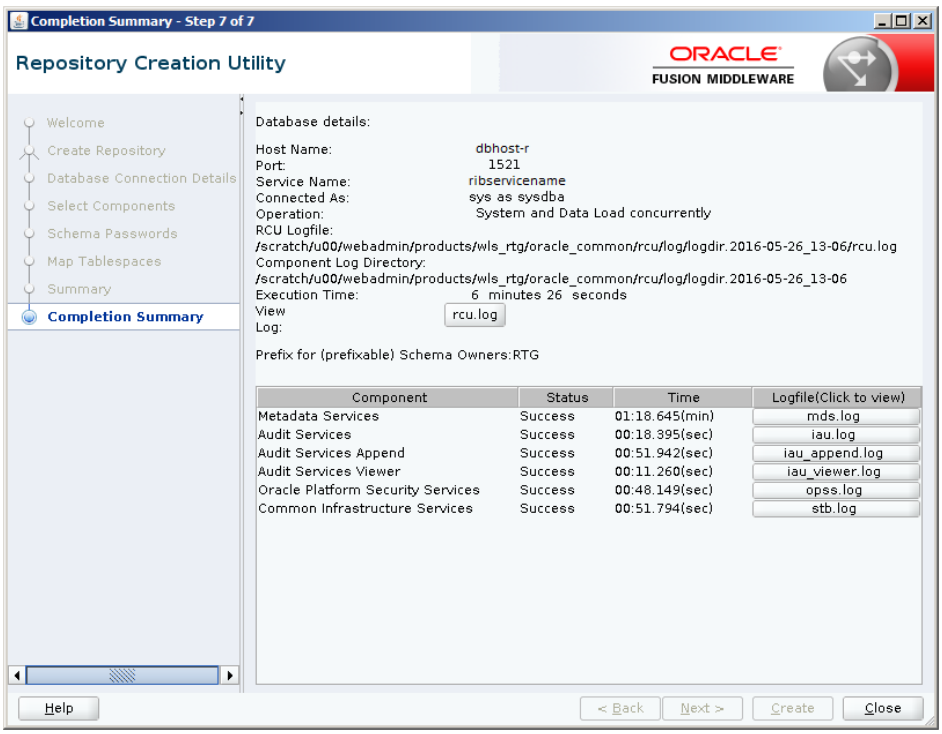
- The Repository Creation Utility – Creating Tablespaces window opens. Click **OK**.



- The Summary window opens. Click **Create**.



- The Completion Summary window opens. Click **Close**.



- The RCU schema creations are complete.

Configure RTGDomain on rtghost1

This section describes the steps to configure the admin and managed servers for the RTG applications. The steps should only be performed on rtghost1. The configuration will then be propagated to rtghost2 once configuration has been completed.

Perform the following steps to configure RIB admin and managed servers on rtghost1 only.

- Set the following variables by exporting them before running the installer so that the JAVA_HOME is added to the PATH variable:
  - JAVA_HOME
  - PATH
  - DISPLAY

  Example:

  ```
  $ export JAVA_HOME=/u00/webadmin/products/jdk_java
  $ export PATH=$JAVA_HOME/bin:$PATH
  $ export DISPLAY=<pc_ip_address:0.0>
  ```

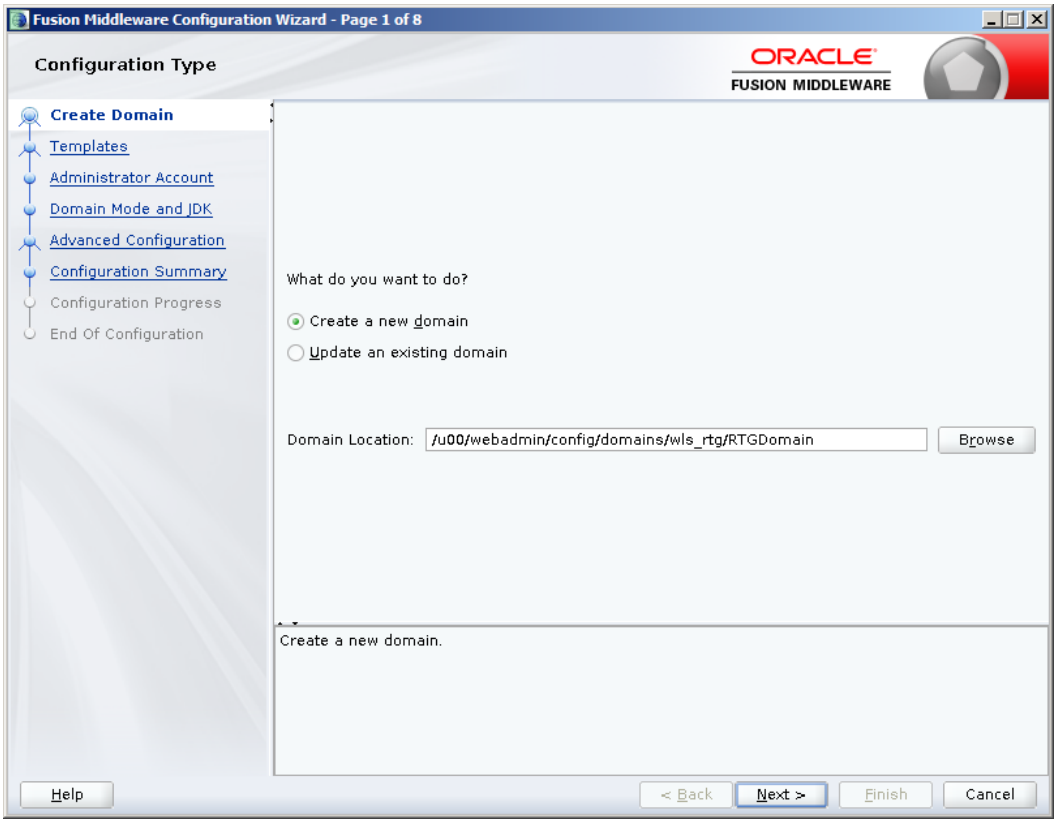
- Navigate to the Navigate to <RTG_WEBLOGIC_HOME>/oracle_common/common/bin and run config.sh:

  Example:

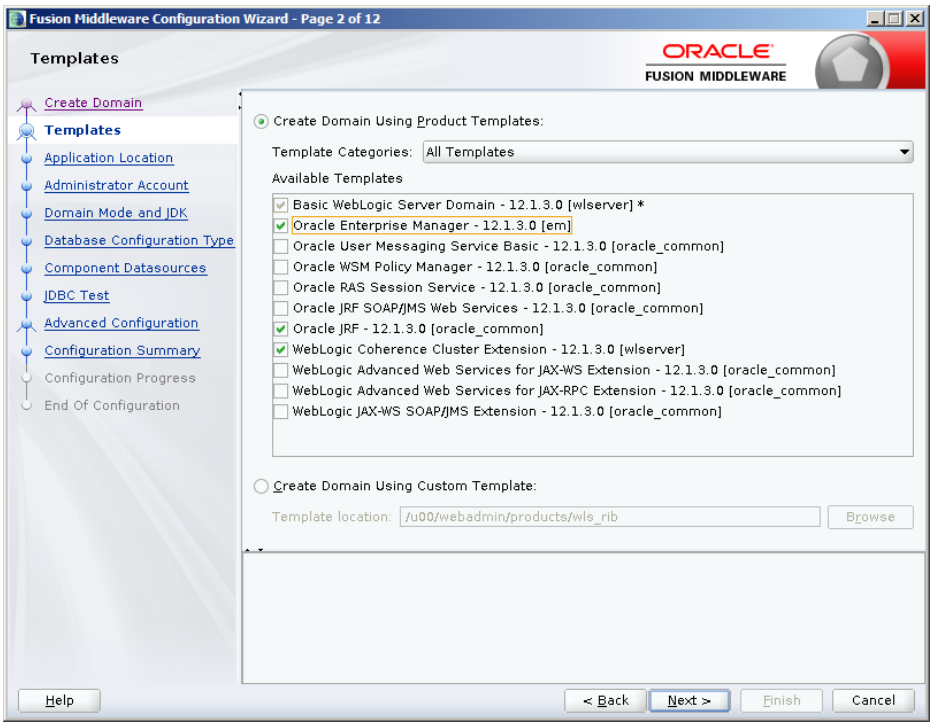
  ```
  $ cd /u00/webadmin/products/wls_rtg/wlserver/common/bin
   $ ./config.sh
  ```

- The Configuration Type window opens. Update the screen as follows and click **Next**.

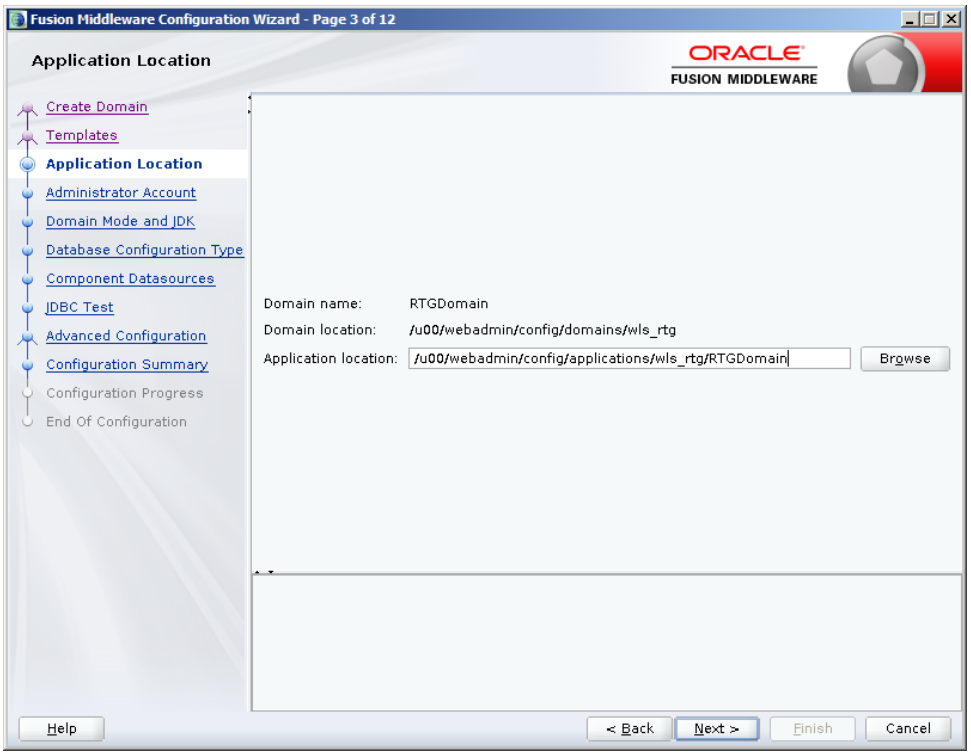
- Select **Create a new domain**.
- Domain Location: /u00/webadmin/config/domains/wls_rtg/RTGDomain



- The Templates window opens. Select the following components and click **Next**.
  - Oracle Enterprise Manager – 12.1.3.0 [EM]
  - Oracle JRF – 12.1.3.0 [oracle_common]
  - WebLogic Coherence Cluster Extension – 12.1.3.0 [wlsserver]



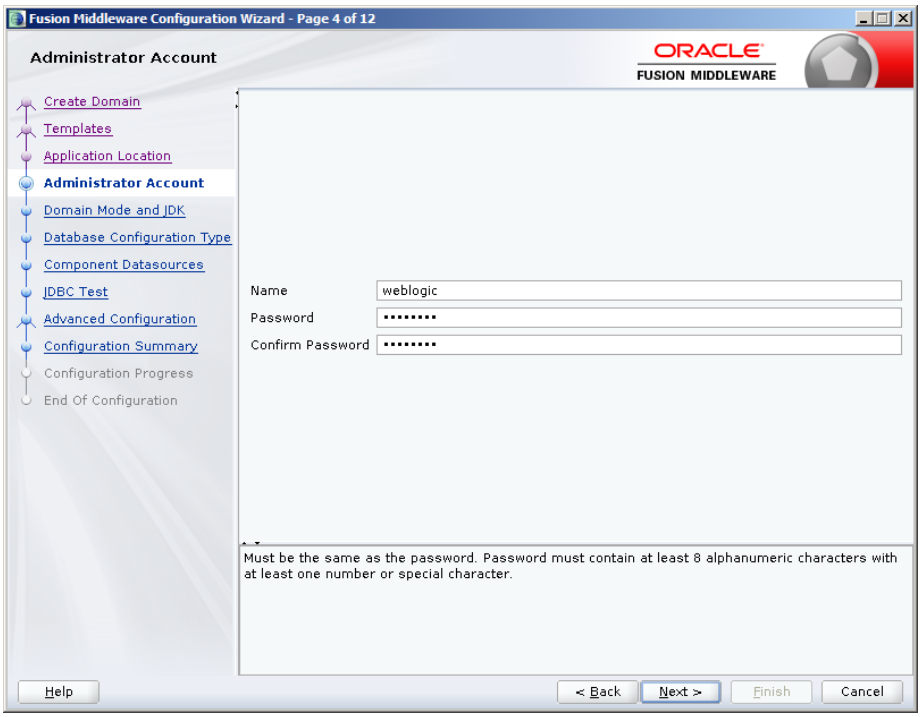
- The Application Location window opens. Update the Application Location as follows and click **Next**.
  - Application Location: /u00/webadmin/config/applications/wls_rtg/RTGDomain



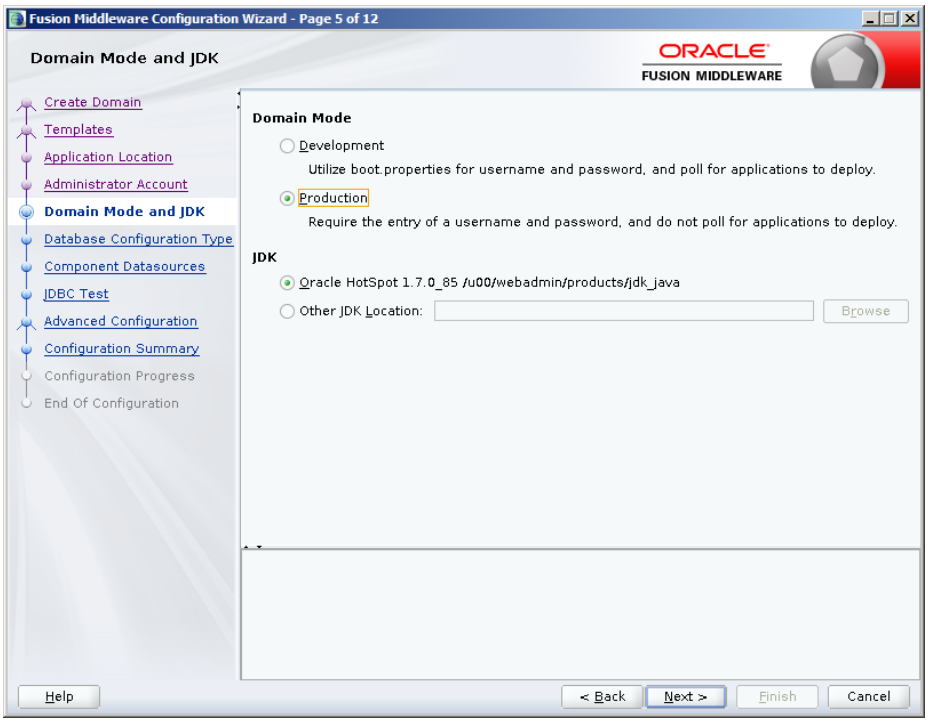
- The Administrator Account window opens. Provide the WebLogic administrator users credentials and click **Next**.

Example:

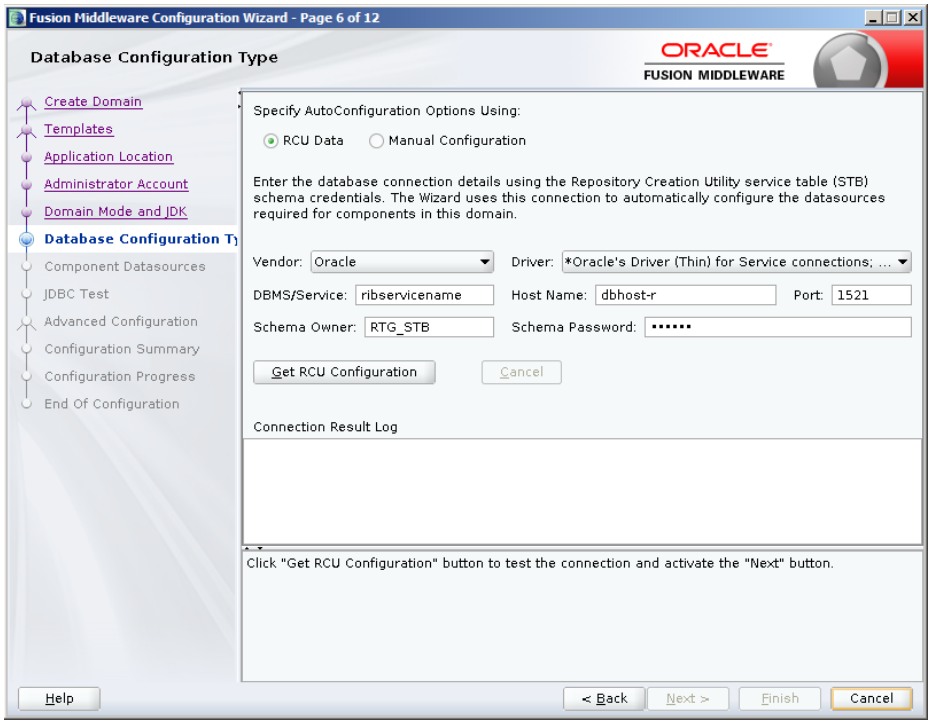
- Name: weblogic
- Password: <password>
- Confirm Password: <password>



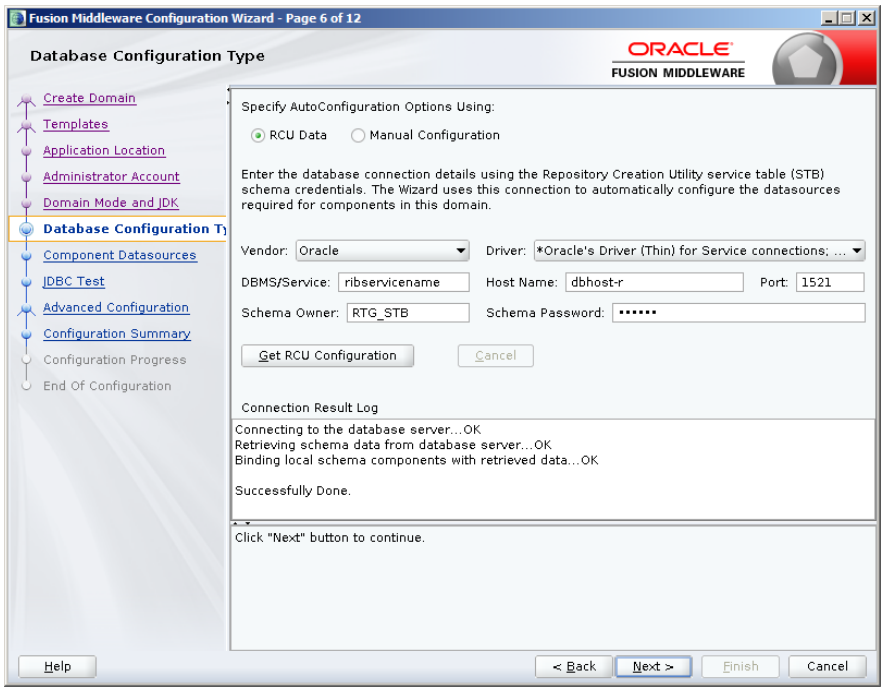
- The Domain Mode and JDK window opens. Select the **Production** Mode and the appropriate JDK options. Click **Next**.



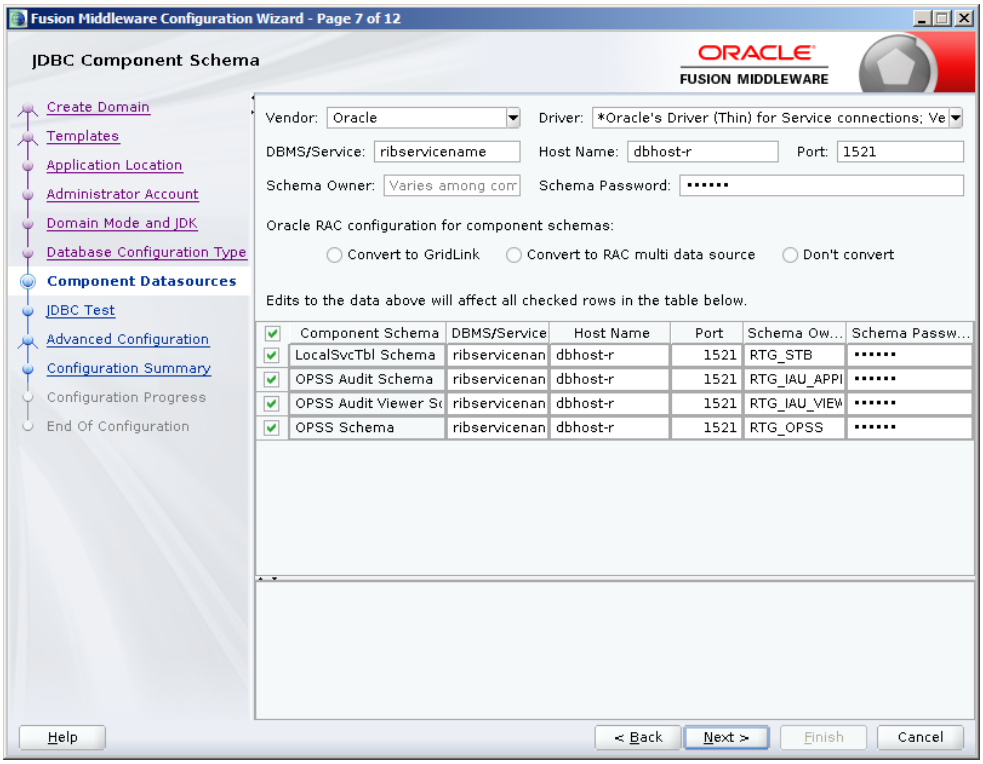
- The Database Configuration Type window opens. Update it as follows and click the Get RCU Configuration button.
  - Driver: *Oracle's Driver (Thin) for Service connections; Versions 9.0.1 and later
  - DBMS/Service: ribservicename
  - Host Name: dbhost-r
  - Port: 1521
  - Schema Owner: RTG_STB
  - Schema Password: <password>



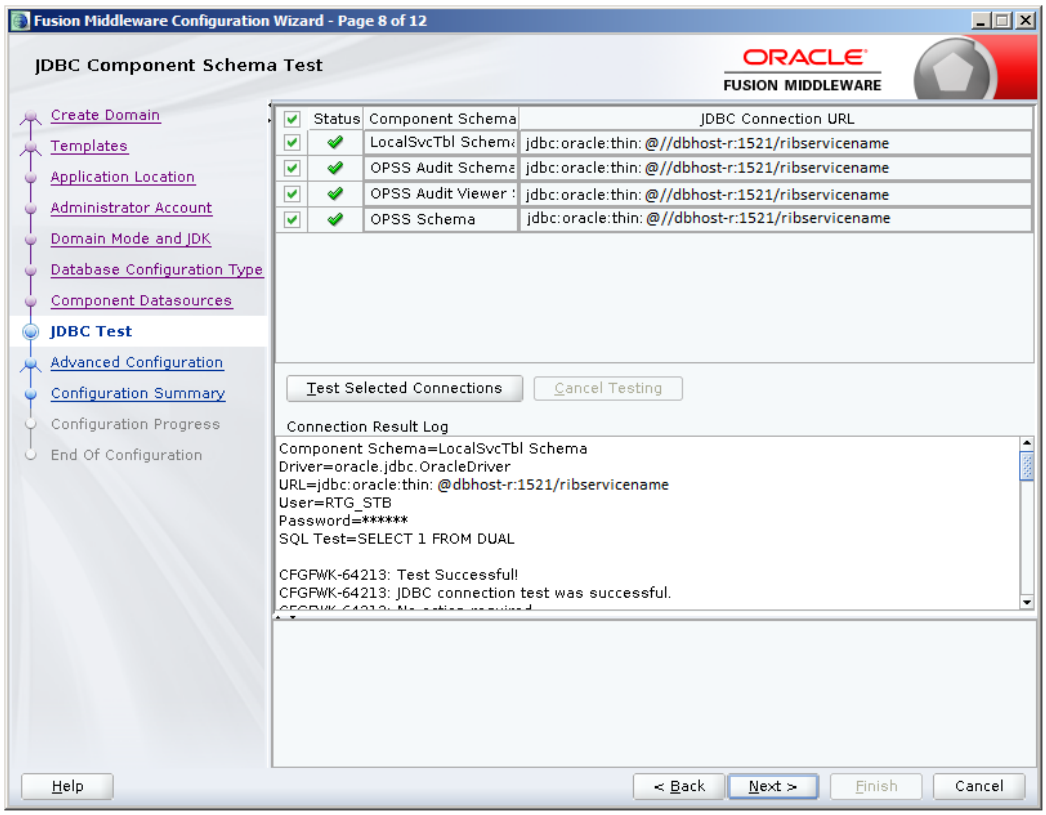
- After the screen validates the entries on the screen the **Next** button is enabled. Click **Next**.



- The JDBC Component Schema is displayed. Review the entries and click **Next**.

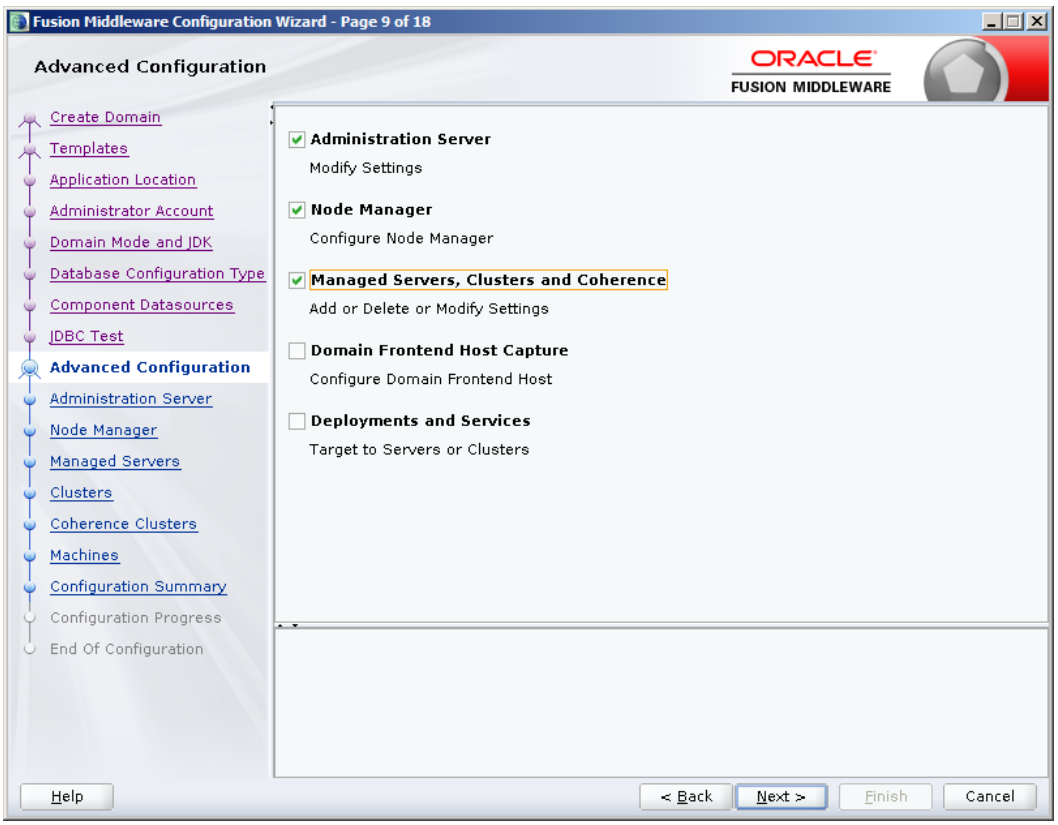


- The JDBC Component Schema Test window opens. Once all the connection tests are successful, click **Next**.

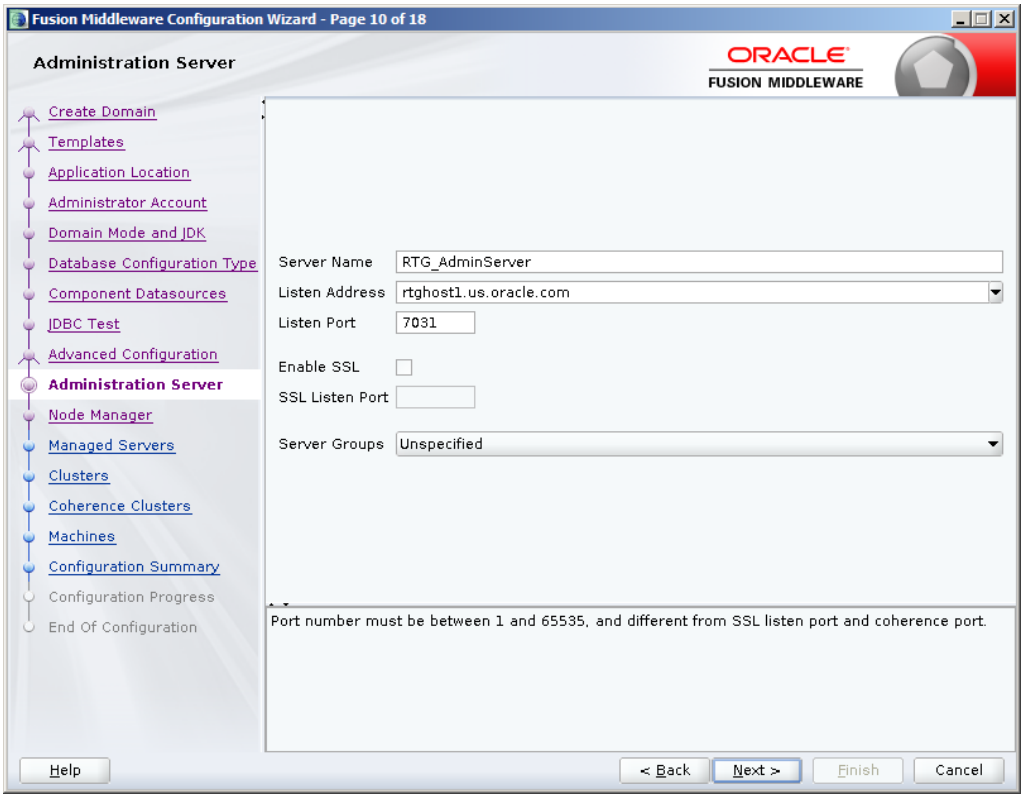


The Advanced configuration window opens. Select the following and click **Next**.

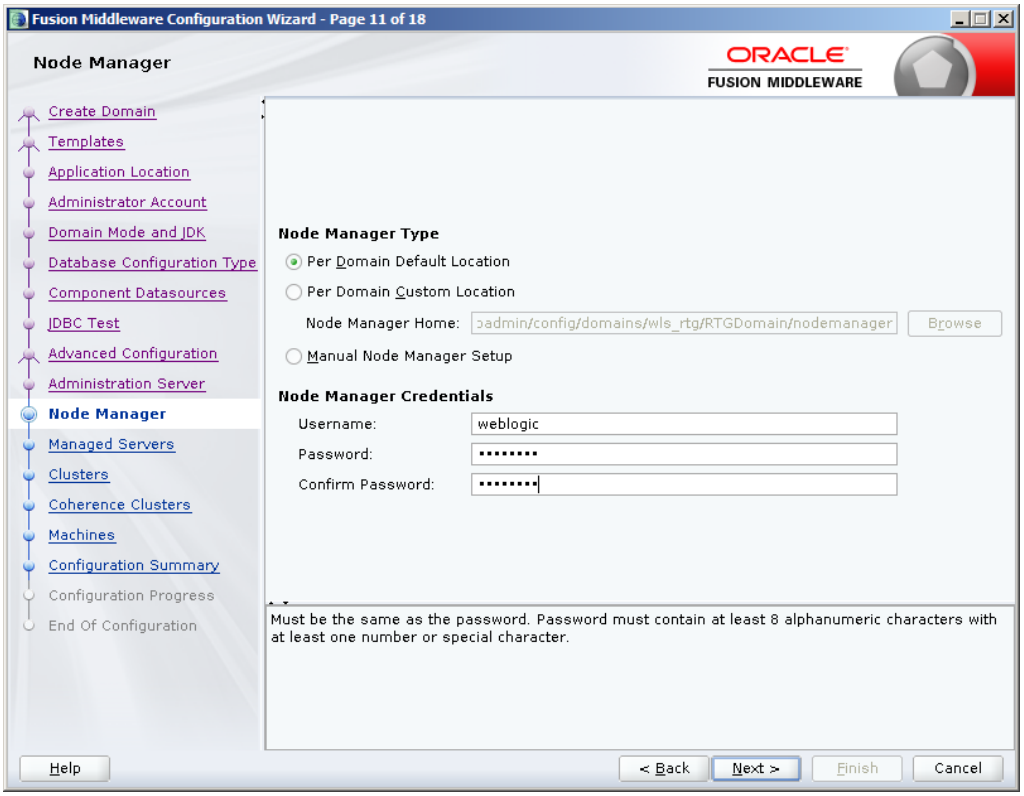
- Administration Server
- Node Manager
- Managed Servers, Clusters and Coherence



- The Administration Server window opens. Provide the Server Name, Listen Address, and Listen Port. The Listen address should be the rtghost1. See screen shot for examples. Click **Next**.

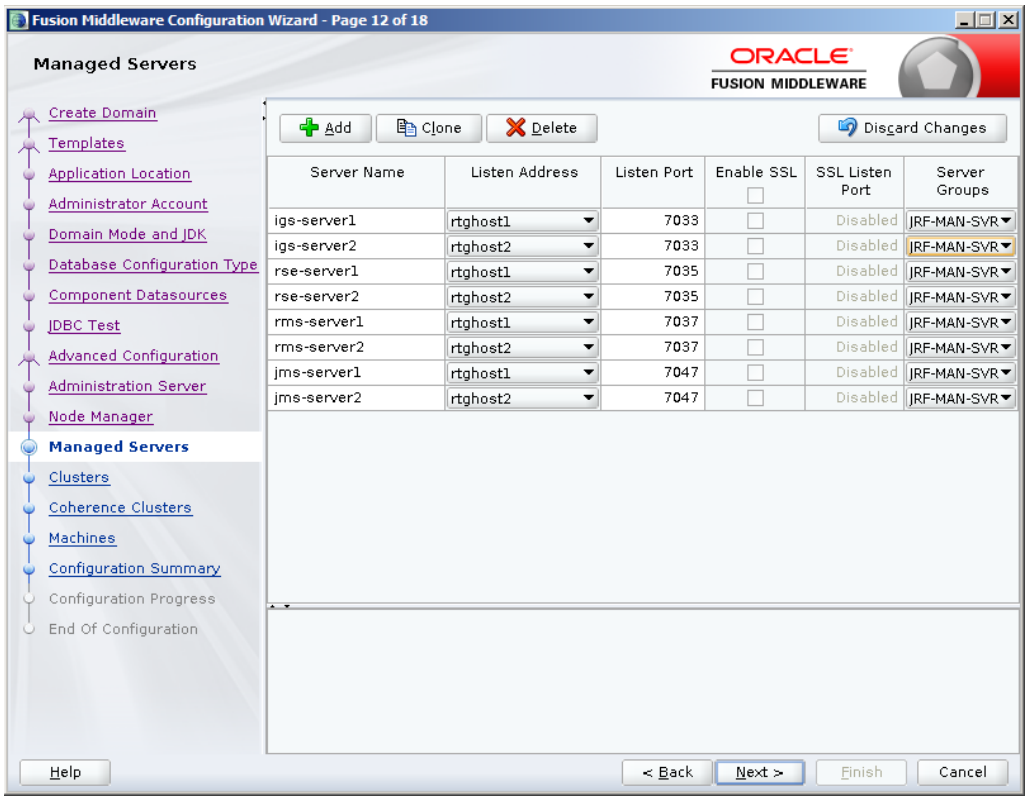


- The Node Manager window opens. Select **Per Domain Default Location** as a Node Manager Type and enter the Node Manager Credentials. Click **Next**.

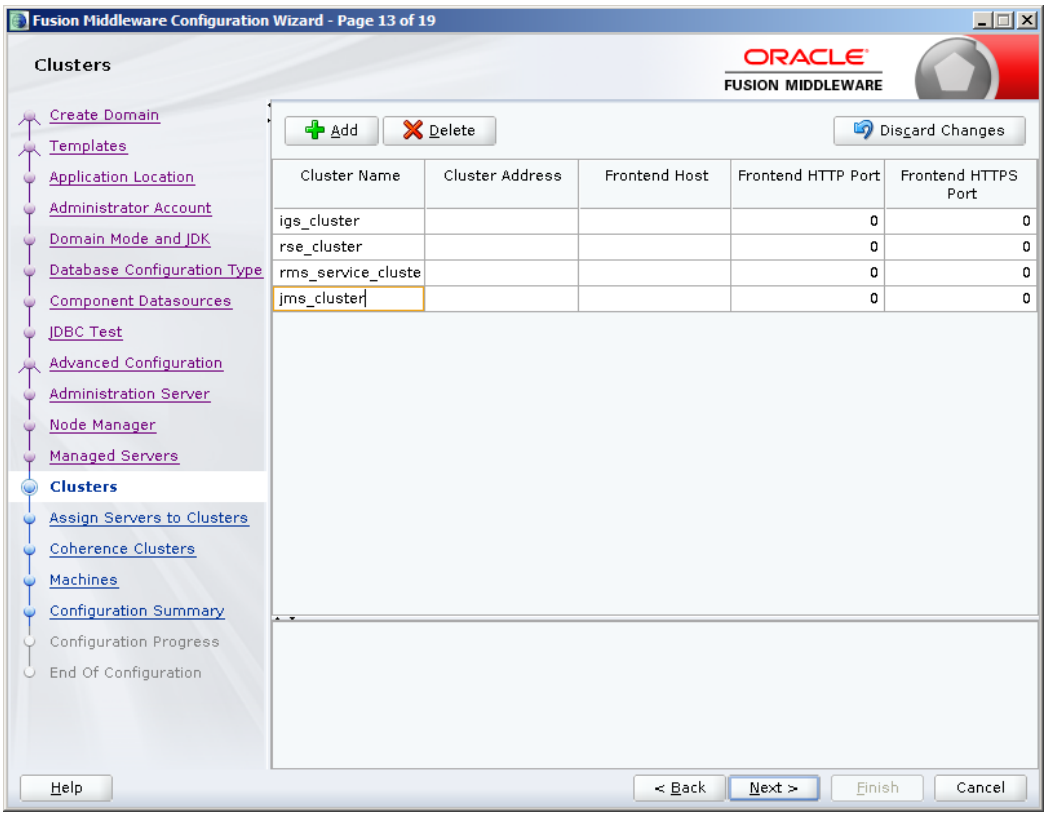


- The Managed Servers window opens. Using the Add button create an entry for each of the managed server and port specifications shown in the screen shot below. Keeping in mind that the managed servers will be created on both rtghost1 and rtghost2 - Make sure to specify the appropriate rtghost for each of the managed servers Listen Address and Select JRF-MAN-SRV for the Server Groups. See screen shot below for example. Click **Next**.

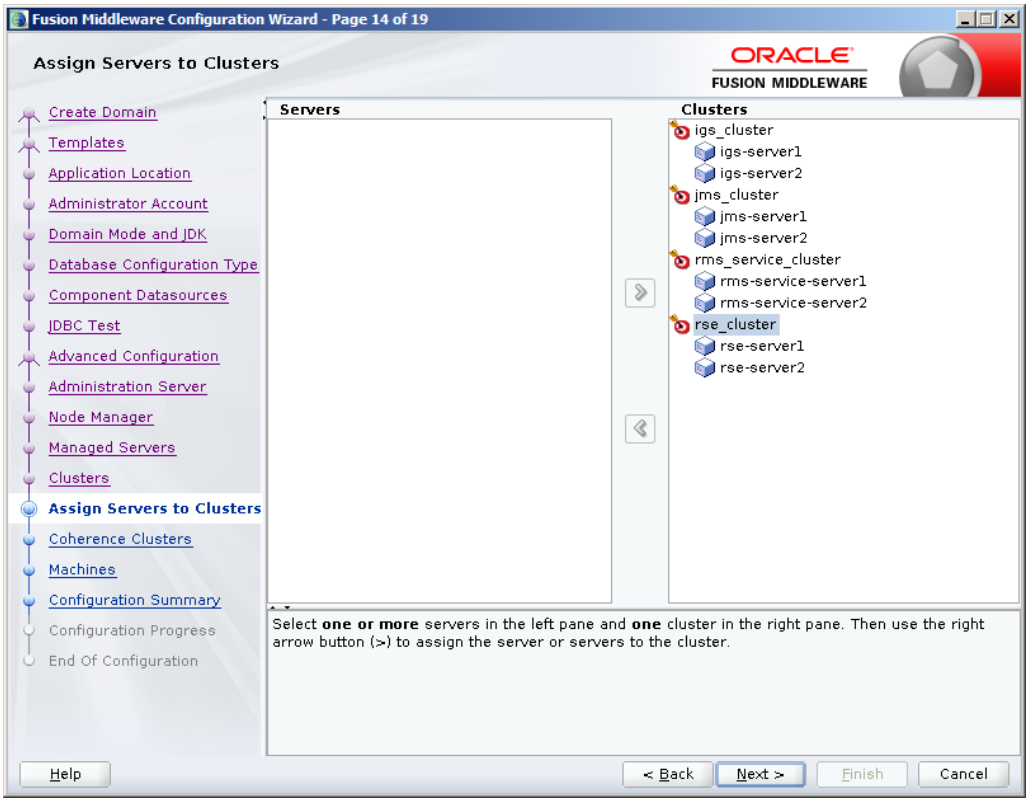
> **NOTE**: Server ports can be anything but they do have dependencies in other applications so be careful to consider that when creating them. The ports used in this document are in alignment with RSB, RIB and RGBU Application ports used in examples from other 15.0.1 HA documents.



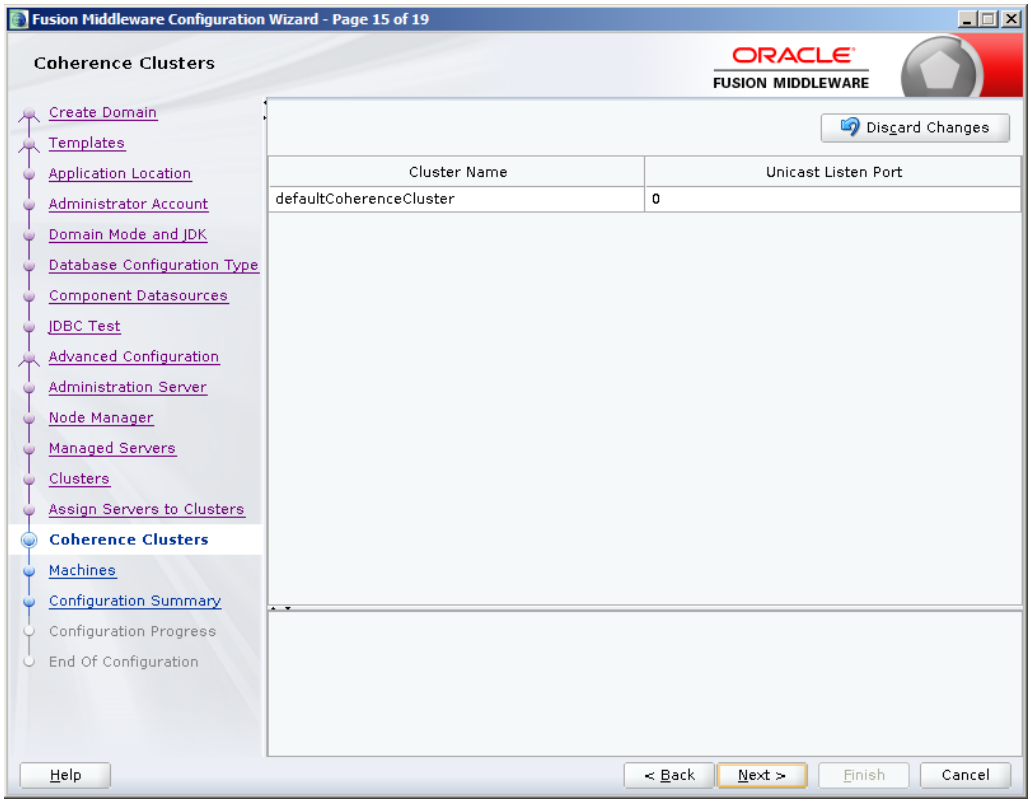
- The Clusters window opens. Using the Add button – create the following clusters and click **Next**.
  - igs_cluster
  - rse_cluster
  - rms_service_cluster
  - jms_cluster



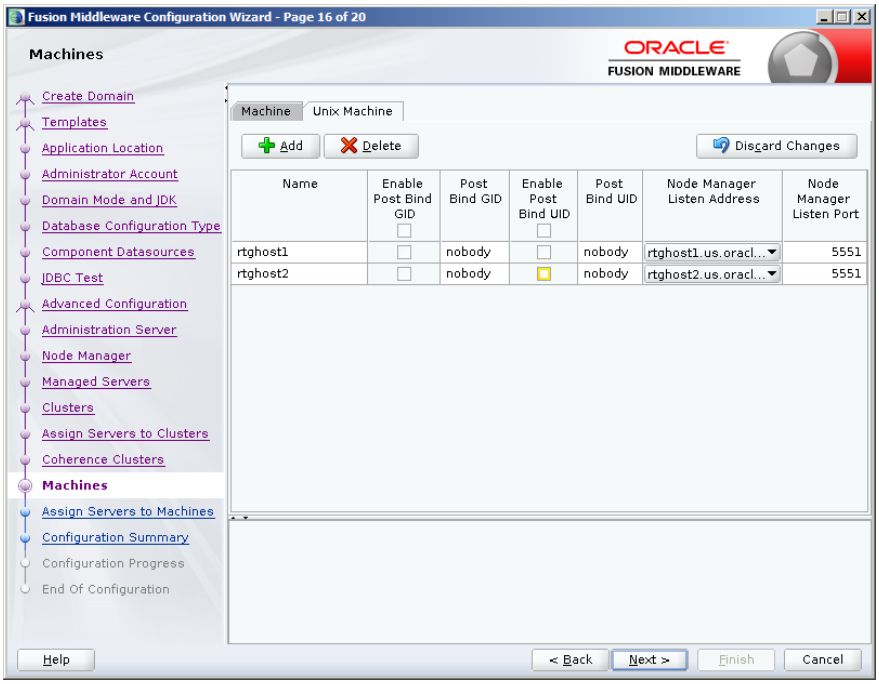
- The Assign Servers to Clusters window opens. Following the examples in the screen shot below assign the managed servers with the appropriate clusters and click **Next**.



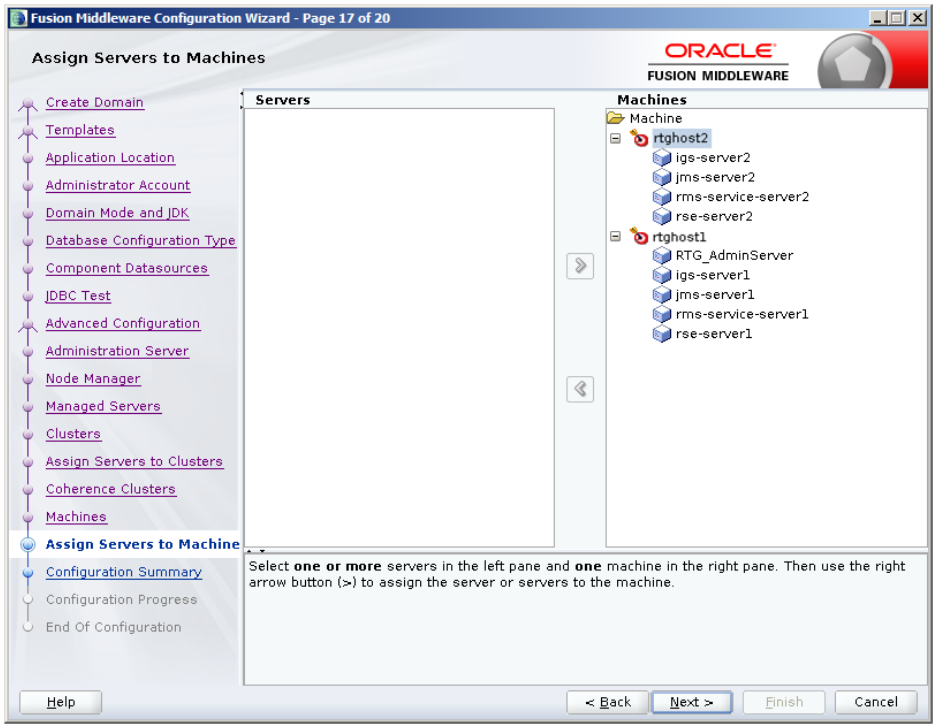
- The Coherence Clusters window opens. Click **Next**.



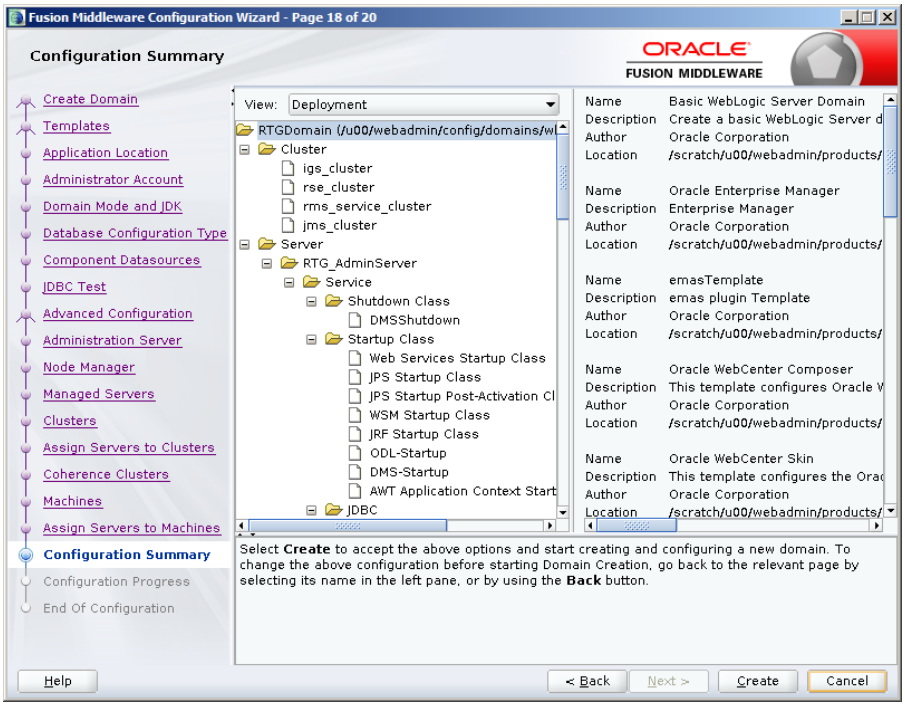
- The Machines window opens. Make sure to click on the Unix Machine tab so the screen looks like the one shown below. Then using the screen shot as an example – provide the details to create a Node Manager for each rtghost. Click **Next**.



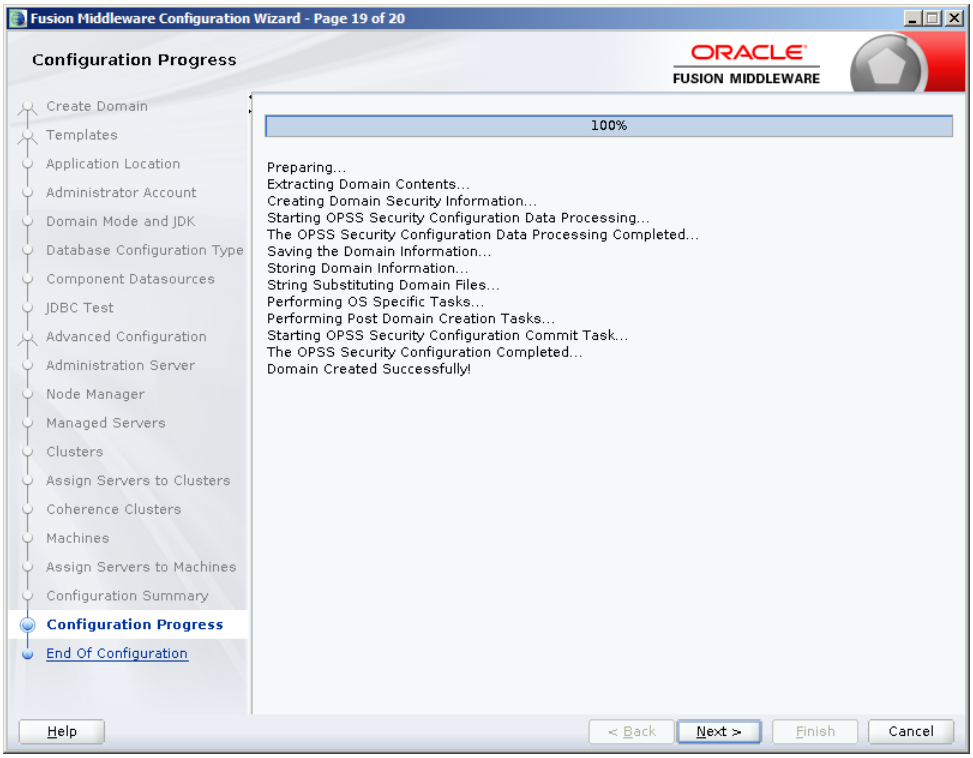
- The Assign Servers to Machines window opens. Organize and assign managed servers to appropriate rtghost and click **Next**.



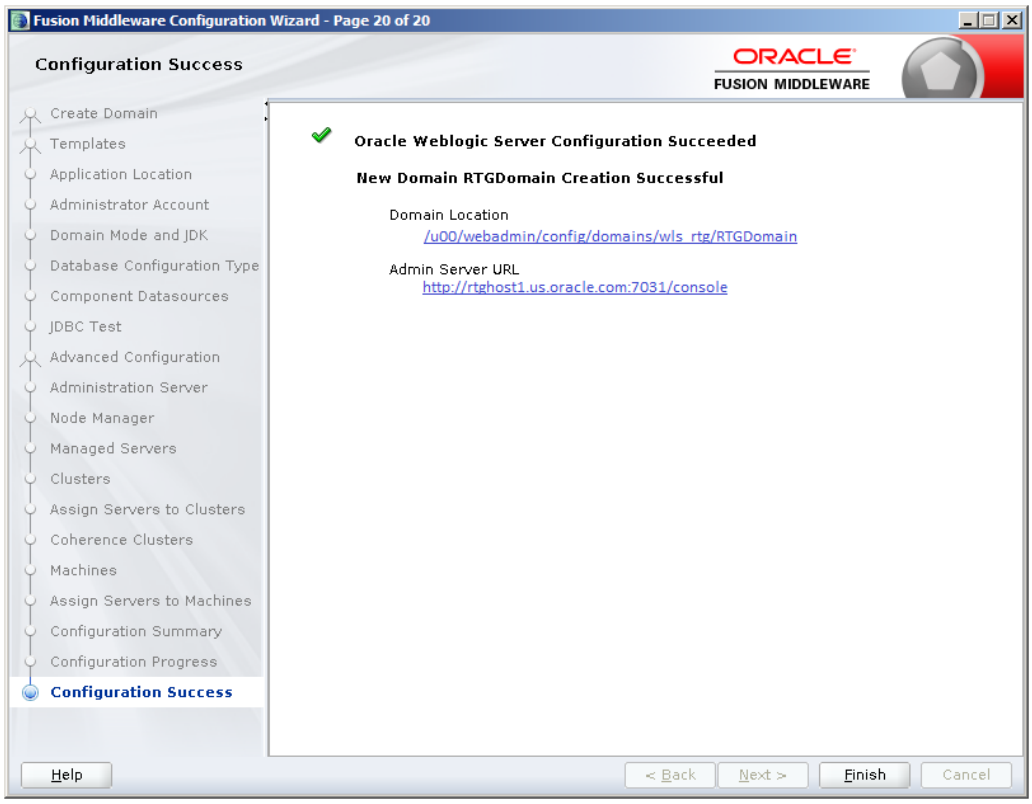
- The Configuration Summary window opens. Click **Create**.



- The Configuration Progress window opens. When progress reaches 100% - click **Next**.



- The Configuration Success window opens. Take note of the Admin Server URL and click **Finish**.



- RTGDomain configuration setup is complete.

Start WLS and Managed Servers - Create boot.properties for RTGDomain

Start the Node Manager, WebLogic admin server and create WebLogic boot.properties file with username and password so that the startup script does not prompt or require user input to start the admin server.

- Navigate to <RTG_DOMAIN_HOME>/bin and run the startNodeManager.sh script to start the Node Manager.

  Example:

  ```
  $ cd /u00/webadmin/config/domains/wls_rtg/RTGDomain/bin
  $ nohup ./startNodeManager.sh &
  ```

Create WebLogic boot.properties as follows:

- On rtghost1 - navigate to <RTG_DOMAIN_HOME>/bin and run the startWebLogic.sh script. Enter in the admin server name user and password when prompted

  Example:

  ```
  $ cd /u00/webadmin/config/domains/wls_rtg/RTGDomain/bin
  $ ./startWebLogic.sh
  username=weblogic
  password=<password>
  ```

- Shut down the admin server.
- Navigate to <RTG_DOMAIN_HOME>/servers/RTG_AdminServer/security. If the security directory does not yet exist – create it.
- Create a new file called boot.properties and add the WLS admin username and password credentials.

Example:
```
$ cd
/u00/webadmin/config/domains/wls_rtg/RTGDomain/servers/RTG_AdminServer/sec
urity
$ vi boot.properties

username=weblogic
password=<password>
```

- Start the WebLogic admin server.

  Example:
  ```
  $ cd /u00/webadmin/config/domains/wls_rtg/RTGDomain/bin
  $ nohup ./startWebLogic.sh &
  ```

- Log onto the WebLogic admin server and start the managed servers that have a listen address of rtghost1 using the following URL: http://rtghost1:7031/console.

Propagate RTGDomain from rthost1 to rtghost2

This section includes the instructions to be performed that will propagate the configuration of the RTGDomain configuration from rtghost1 to rtghost2.

- Shut down the admin and managed servers for the RTGDomain on rtghost1. This will make sure there are no changes that can take place and cause the hosts to be out of sync while copying the RTGDomain from rtghost1 to rtghost2.

- Add the path of the WebLogic pack.sh script to the PATH variable on rtghost1.

  Example:
  ```
  $ export
  PATH=/u00/webadmin/products/wls_rtg/oracle_common/common/bin:$PATH
  ```

- Run the pack.sh script on rtghost1 to pack up the RTGDomain and place it in a jar file called RTGDomain.jar. Following the examples provided for these steps place it in the same directory as the RTGDomain.

  Example:
  ```
  $ pack.sh -domain=/u00/webadmin/config/domains/wls_rtg/RTGDomain -
  template=/u00/webadmin/config/domains/wls_rtg/RTGDomain.jar -
  template_name=RTGDomain -managed=true
  ```

  Output:
  ```
  << read domain from "/u00/webadmin/config/domains/wls_rtg/RTGDomain"
  >>  succeed: read domain from
  "/u00/webadmin/config/domains/wls_rtg/RTGDomain"
  << set config option Managed to "true"
  >>  succeed: set config option Managed to "true"
  << write template to "/u00/webadmin/config/domains/wls_rtg/RTGDomain.jar"
  ......................................................................
  ........................
  >>  succeed: write template to
  "/u00/webadmin/config/domains/wls_rtg/RTGDomain.jar"
  << close template
  >>  succeed: close template
  ```

- Log onto rtghost2 – Then create a directory like the one the RTGDomain.jar exists in on rtghost1 and then copy the RTGDomain.jar there.

  Example:
  ```
  $ mkdir -p /u00/webadmin/config/domains/wls_rtg
  $ cd /u00/webadmin/config/domains/wls_rtg
  $ scp -r
  webadmin@rtghost1:/u00/webadmin/config/domains/wls_rtg/RTGDomain.jar .
  ```

- Still on rtghost2 - Add the path of the WebLogic pack.sh script to the PATH variable.

  Example:
  ```
  $ export
  PATH=/u00/webadmin/products/wls_rtg/oracle_common/common/bin:$PATH
  ```

- Un-pack the RTGDomain on rtghost2 using unpack.sh script.

  Example:
  ```
  $ unpack.sh -domain=/u00/webadmin/config/domains/wls_rtg/RTGDomain -
  template=/u00/webadmin/config/domains/wls_rtg/RTGDomain.jar -
  overwrite_domain=true -
  app_dir=/u00/webadmin/config/applications/wls_rtg/RTGDomain
  ```

  Output:
  ```
  << read template from "/u00/webadmin/config/domains/wls_rtg/RTGDomain.jar"
  >>  succeed: read template from
  "/u00/webadmin/config/domains/wls_rtg/RTGDomain.jar"
  << set config option OverwriteDomain to "true"
  >>  succeed: set config option OverwriteDomain to "true"
  << set config option AppDir to
  "/u00/webadmin/config/applications/wls_rtg/RTGDomain"
  >>  succeed: set config option AppDir to
  "/u00/webadmin/config/applications/wls_rtg/RTGDomain"
  << set config option DomainName to "RTGDomain"
  >>  succeed: set config option DomainName to "RTGDomain"
  << write Domain to "/u00/webadmin/config/domains/wls_rtg/RTGDomain"
  ..........................................................................
  ................
  >>  succeed: write Domain to
  "/u00/webadmin/config/domains/wls_rtg/RTGDomain"
  << close template
  >>  succeed: close template
  ```

- Start the Node manager on rtghost2 by navigating to <RTG_DOMAIN_HOME>/bin and running the startNodeManager.sh script.

  Example:
  ```
  $ cd /u00/webadmin/config/domains/wls_rtg/RTGDomain/bin
  $ nohup ./startNodeManager.sh &
  ```

- On rtghost1 restart the WebLogic admin server.

  Example:
  ```
  $ cd /u00/webadmin/config/domains/wls_rtg/RTGDomain/bin
  $ nohup ./startWebLogic.sh &
  ```

- Log onto the WebLogic Administration Console and start all of the managed servers for RTG on both rtghost1 and rtghost2.

- Propagating the RTGDomain from rtghost1 to rtghost2 is complete.

Prepare RTG Application Installations for IGS and JMS

This section covers some preliminary steps that will prepare for the IGS and JMS RTG applications to be installed. The intent is to recreate the RIB_HOME directory from ribhost1 on rtghost1 and then organize files in the specific directories that are required to install the applications correctly.

Perform the following steps in order to prepare the RTG applications for installation. While performing the steps pay careful attention to what host the steps are intended for. The steps will be carried out on ribhost1 and rtghost1. The host is in bold text to highlight the intended host for each step.

- Starting out on ribhost1 - Move the IntegrationGatewayService15.0.0ForAll15.0.0Apps_eng_ga.tar and JmsConsole15.0.0ForAll15.x.xApps_eng_ga.zip files from the <RIB_STAGING> to the <RIB_HOME>/tools-home directory. Do not extract the files.

  Example:
  ```
  $ cd /u00/webadmin/media/rib/IGS
  $ cp IntegrationGatewayService15.0.0ForAll15.0.0Apps_eng_ga.tar
  /u00/webadmin/RIB_INSTALL/Rib1500ForAll15xxApps/rib-home/tools-home
  $ cd /u00/webadmin/media/rib/RIB
  $ cp JmsConsole15.0.0ForAll15.x.xApps_eng_ga.zip
  /u00/webadmin/RIB_INSTALL/Rib1500ForAll15xxApps/rib-home/tools-home
  ```

- On ribhost1 - Copy the RIB_INSTALL directory to rtghost1.

  Example:
  ```
  $ cd /u00/webadmin
  $ scp -r RIB_INSTALL webadmin@rtghost1:/u00/webadmin/
  ```

- On rtghost1 – Navigate to <RIB_HOME>/tools-home/ and extract the following 2 files IntegrationGatewayService15.0.0ForAll15.0.0Apps_eng_ga.tar and JmsConsole15.0.0ForAll15.x.xApps_eng_ga.zip

  Example:
  ```
  $ cd /u00/webadmin/RIB_INSTALL/Rib1500ForAll15xxApps/rib-home/tools-home
  $ tar xf IntegrationGatewayService15.0.0ForAll15.0.0Apps_eng_ga.tar
  $ unzip JmsConsole15.0.0ForAll15.x.xApps_eng_ga.zip
  ```

- On rtghost1 - Copy the RIB packaged aqapi-12.1.0.2.jar and ojdbc7-12.1.0.2.jar to <RTG_WEBLOGIC_HOME>/wlserver/server/lib directory.

  Example:
  ```
  $ cd /u00/webadmin/RIB_INSTALL/Rib1500ForAll15xxApps/rib-home/integration-
  lib/third-party/oracle/db/12.1.0.2
  $ cp aqapi-12.1.0.2.jar /u00/webadmin/products/wls_rtg/wlserver/server/lib
  $ cp ojdbc7-12.1.0.2.jar
  /u00/webadmin/products/wls_rtg/wlserver/server/lib
  ```

Install IGS Application

Perform the following steps on rtghost1 to install the IGS application.

- Navigate to <RIB_HOME>/tools-home/integration-bus-gateway-services/conf and edit the IgsConfig.properties as follows:
  - Change the value of WlsUrl to point to the WebLogic server where IGS is going to be deployed. The port in the WlsUrl should be the administration port.

- Change the value of WlsTarget to the instance name where IGS is going to be deployed (for example, igs-service_wls_instance or rtg_cluster).

  Example: Highlighted text

```
...

# Connection string for Weblogic
WlsUrl=t3://rtghost1.us.oracle.com:7031
# The target server name or cluster where Igs should be deployed
WlsTarget=igs_cluster

...
```

- Navigate to <RIB_HOME>/tools-home/integration-bus-gateway-services/bin and run the igs-install.sh script to install IGS.

  Example:

```
$ cd /u00/webadmin/RIB_INSTALL/Rib1500ForAll15xxApps/rib-home/tools-home/integration-bus-gateway-services/bin
$ ./igs-install.sh
```

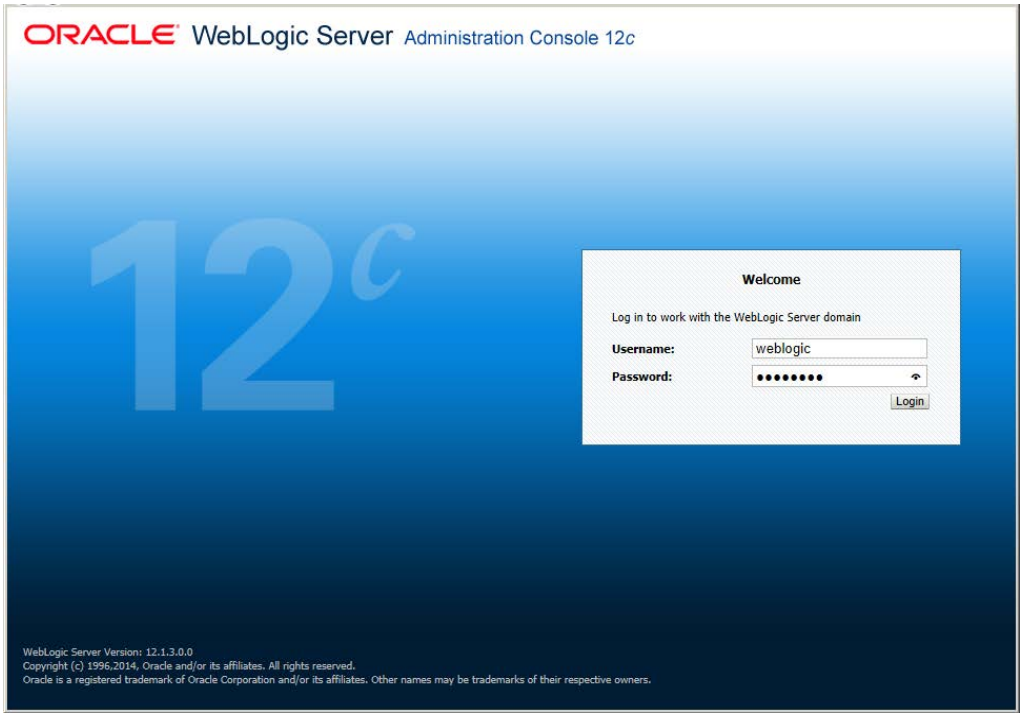
- The installer will display prompts for the WebLogic administrators user name and password. Provide the credentials and press enter.

  Example:

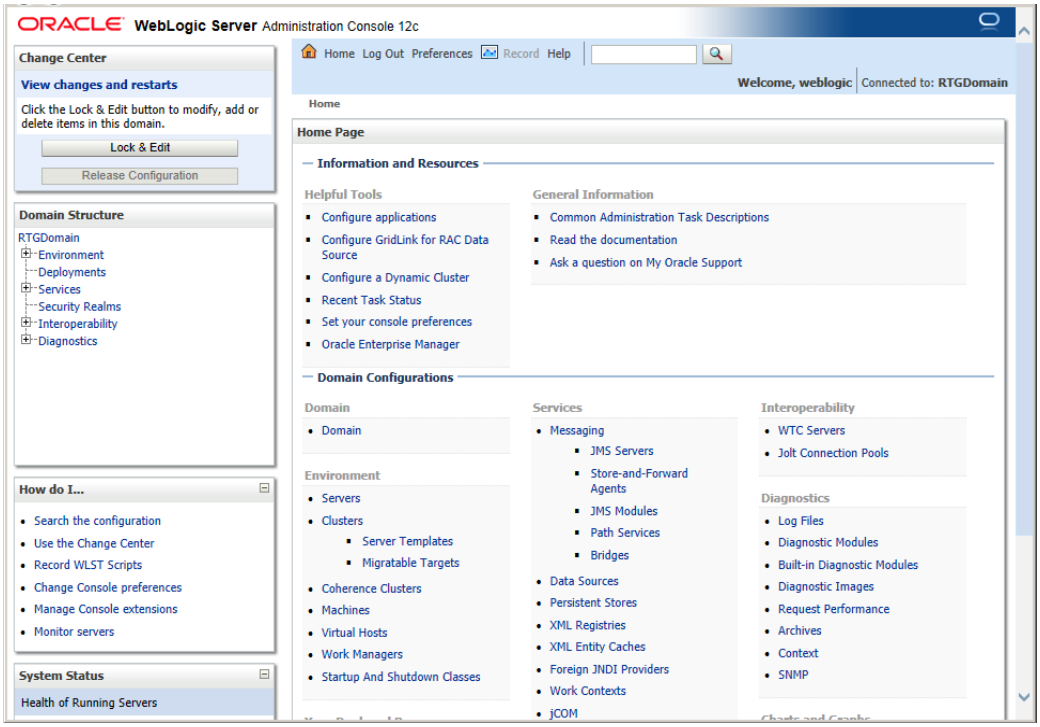
```
Enter a username for the weblogic server where IGS would be deployed
weblogic
Loading log4j.xml from
file:/u00/webadmin/RIB_INSTALL/Rib1500ForAll15xxApps/rib-home/log4j.xml
Credential Store type is FILE_STORE
Credential Store location is file://../conf/security

Enter password: <password>
```

After the installer finishes running – There are some server start parameters for IGS that need to be set before the application installation is complete.

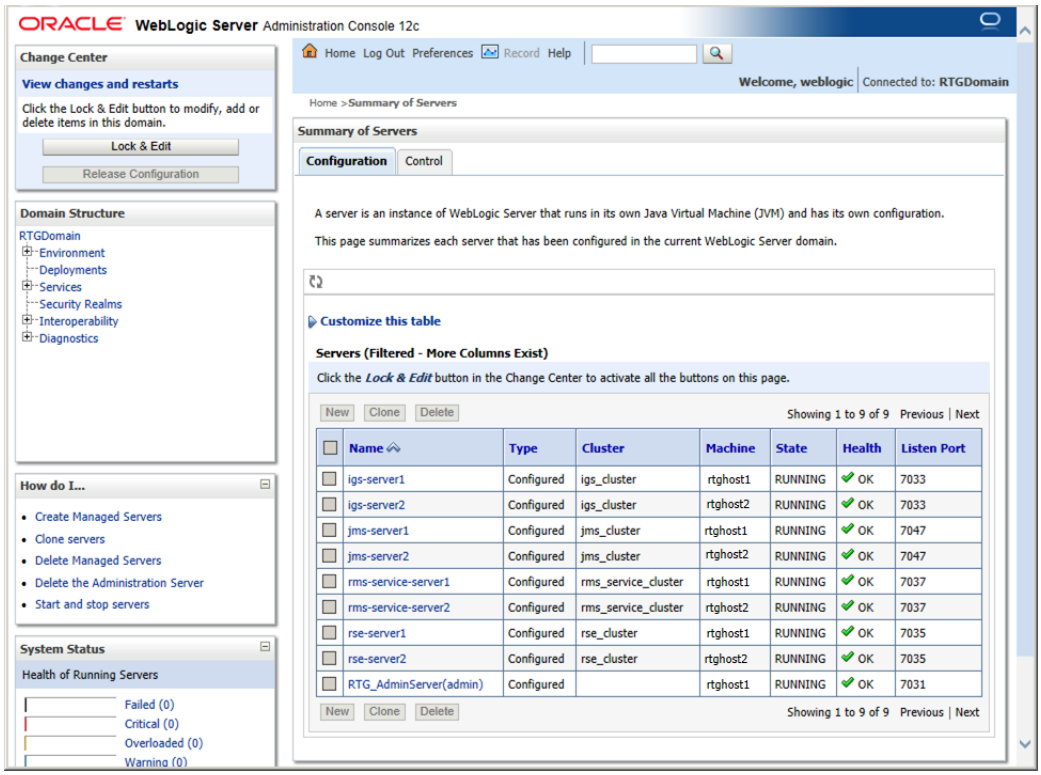
- Log onto the WebLogic Administration Console.



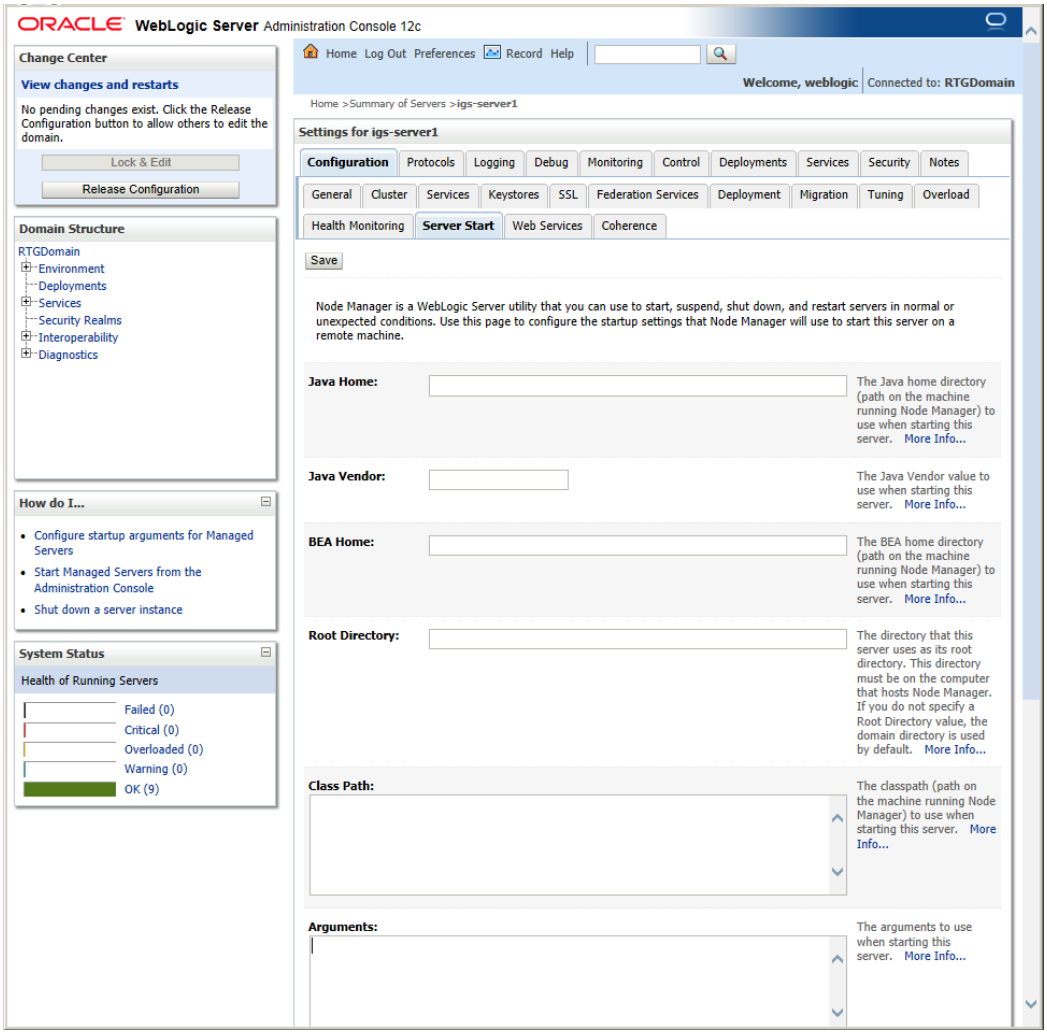
- The Home Page is displayed. Click the **Servers** link.



- The Summary of Servers window opens. Click on the **igs-server1** link.

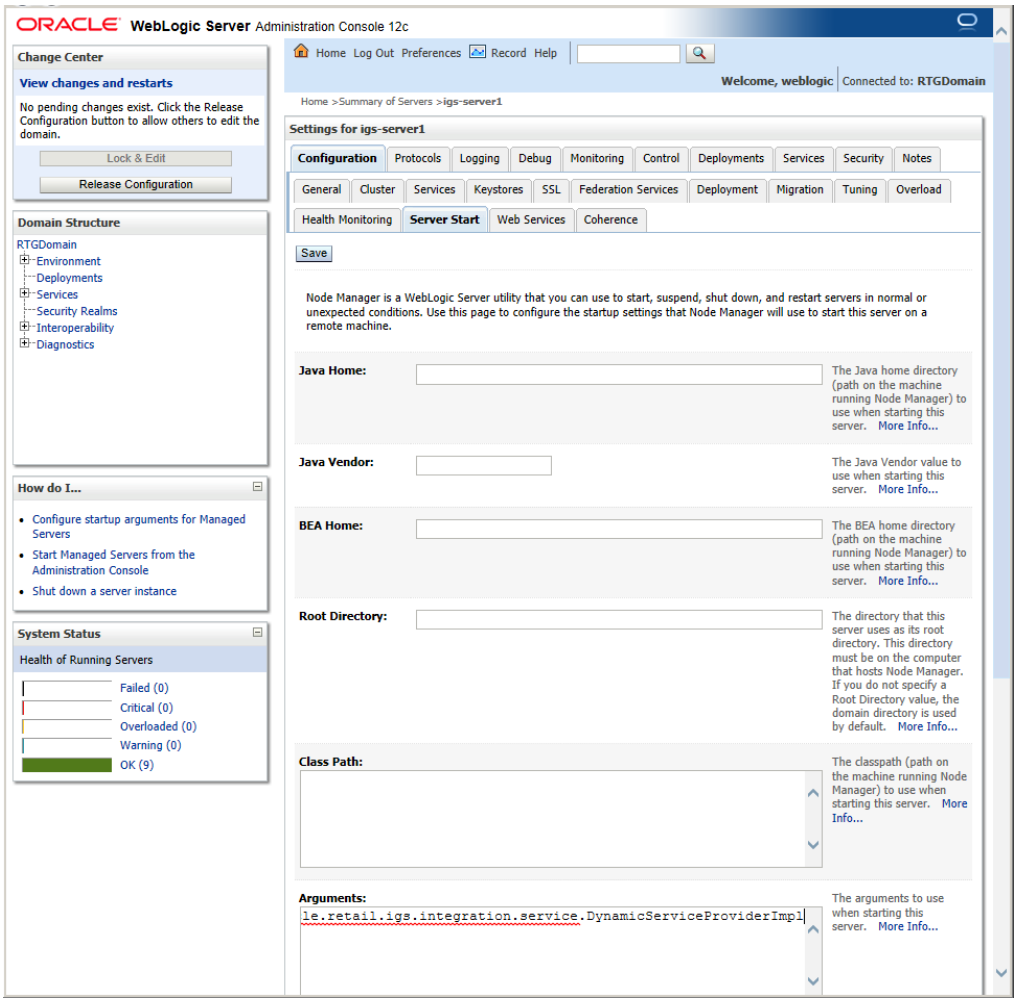


- The Setting for igs-server1 window opens. Under the Configuration, navigate to the **Server** tab and then click **Lock & Edit**.



- Input the following Arguments, click **Save** and then click **Activate Changes**. Arguments should all be in one line including the '–' at the front.
  - Arguments:

Doracle.retail.soa.enabler.service.provider.engine.ServiceProviderImplLookupFactory.interceptor=com.oracle.retail.igs.integration.service.DynamicServiceProviderImpl



- Repeat the steps above and add the arguments to igs-server2.
- Stop and restart the IGS managed servers.
- Installation for IGS is complete.

Verification - Verify IGS Installation

Perform the following steps to validate the IGS installation. This test is just to validate that the web services are accessible from the rtghosts.

- Launch the following two URLs to make sure that at least one of the web services is accessible via rtghost1 and rtghost2.
  - http://rtghost1.us.oracle.com:7033/ASNInPublishingBean/ASNInPublishingService?WSDL
  - http://rtghost2.us.oracle.com:7033/ASNInPublishingBean/ASNInPublishingService?WSDL

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- Published by JAX-WS RI (http://jax-ws.java.net). RI's version is JAX-WS RI 2.2.10-b140319.1121 svn-
revision#7b34660d6bc70e587e8ec81a17e4d76af68816a6. -->
<!-- Generated by JAX-WS RI (http://jax-ws.java.net). RI's version is JAX-WS RI 2.2.10-b140319.1121 svn-
revision#7b34660d6bc70e587e8ec81a17e4d76af68816a6. -->
+ <definitions xmlns="http://schemas.xmlsoap.org/wsdl/" name="ASNInPublishingService"
targetNamespace="http://www.oracle.com/retail/igs/integration/services/ASNInPublishingService/v1"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:tns="http://www.oracle.com/retail/igs/integration/services/ASNInPublishingService/v1"
xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/" xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"
xmlns:wsam="http://www.w3.org/2007/05/addressing/metadata"
xmlns:wsp1_2="http://schemas.xmlsoap.org/ws/2004/09/policy" xmlns:wsp="http://www.w3.org/ns/ws-policy"
xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd">
```


- Test complete.

Install RSE Application

This section covers the steps required to install the RSE application. The installation steps for installing the RSE application include deploying the retail-soa-enabler-gui-15.0.0.war file and then setting up a user id that can access the application.

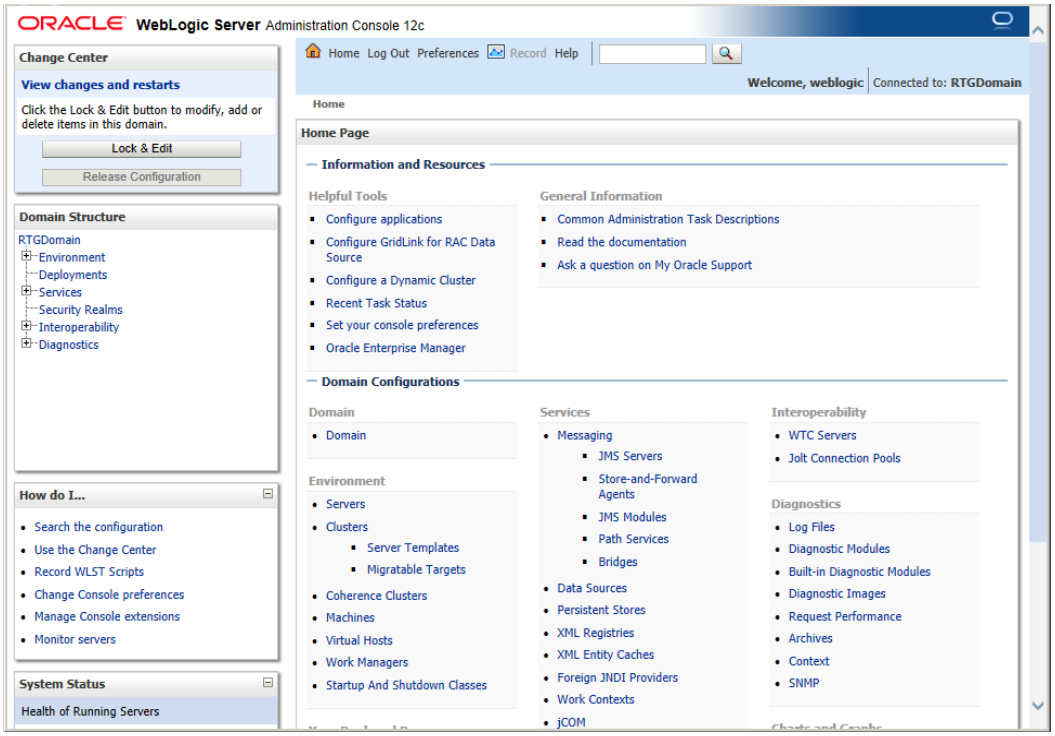
Perform the following steps on rtghost1 to install the Retail SOA Enabler application.

- On rtghost1 - Create a directory to use as a staging location and download the RSB application distribution to that directory. Going forward this document will refer to this directory as <RSB_STAGING> and the examples in this document will show it as:

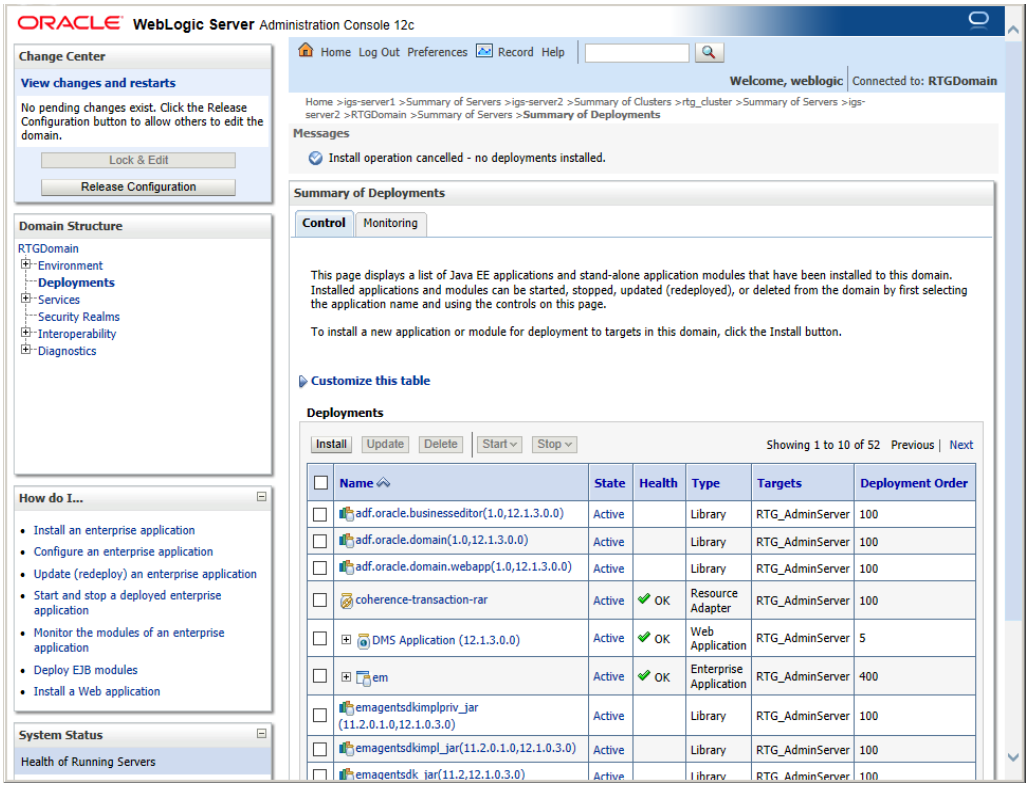
  Example:
  `/u00/webadmin/media/rsb/`

- From the <RSB_STAGING>/15.0/CDROM/rsb150c/CDROM directory - Copy the RetailSOAEnabler15.0.0ForAll15.0.0Apps_eng_ga.tar file and extract the contents to a location on your local computer that is accessible by a browser for deployment. (You could place it on your Desktop for example.)
- Log onto the WebLogic admin console.

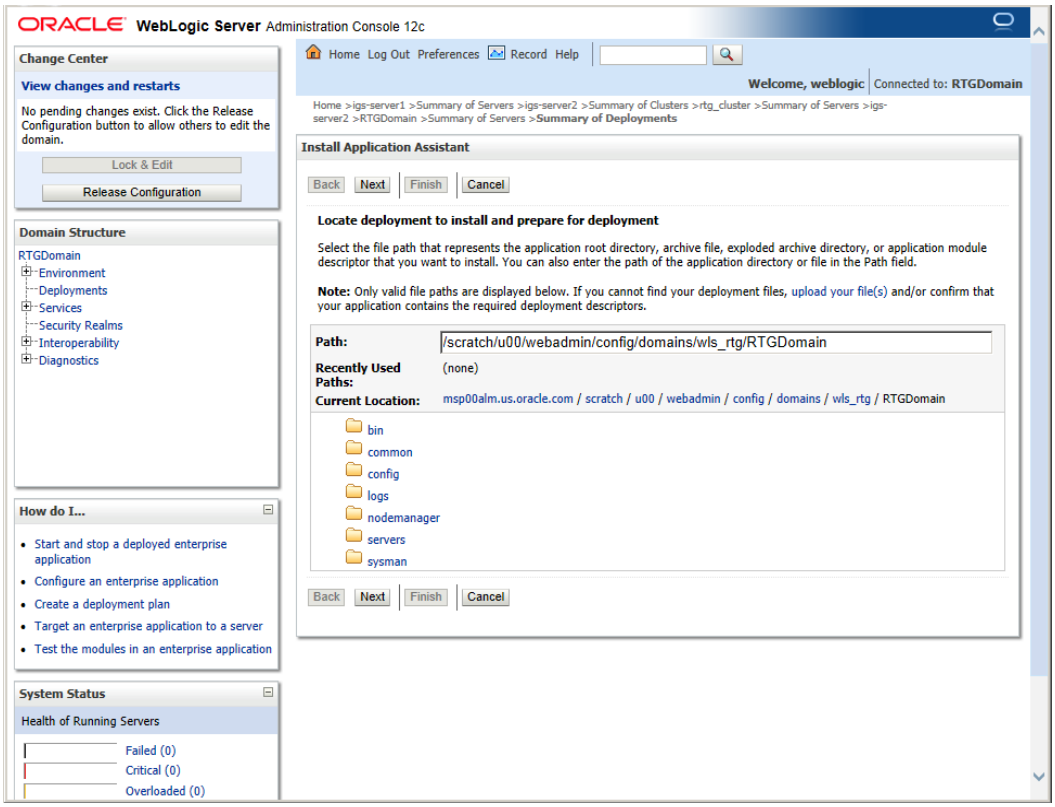
- The Home Page window opens. Click **Deployments** in the navigation pane on the left side of the screen.



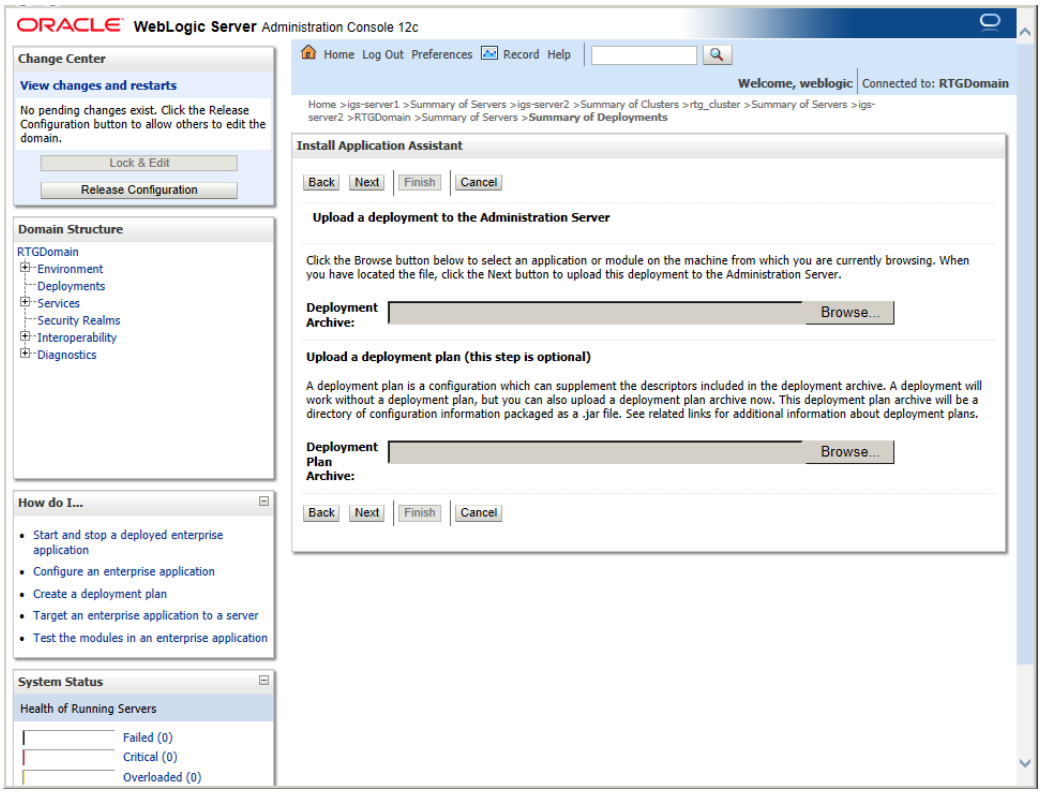
- The Summary of Deployments window opens. Click the **Lock & Edit** button on the top left of the screen and then click the **Install** button.



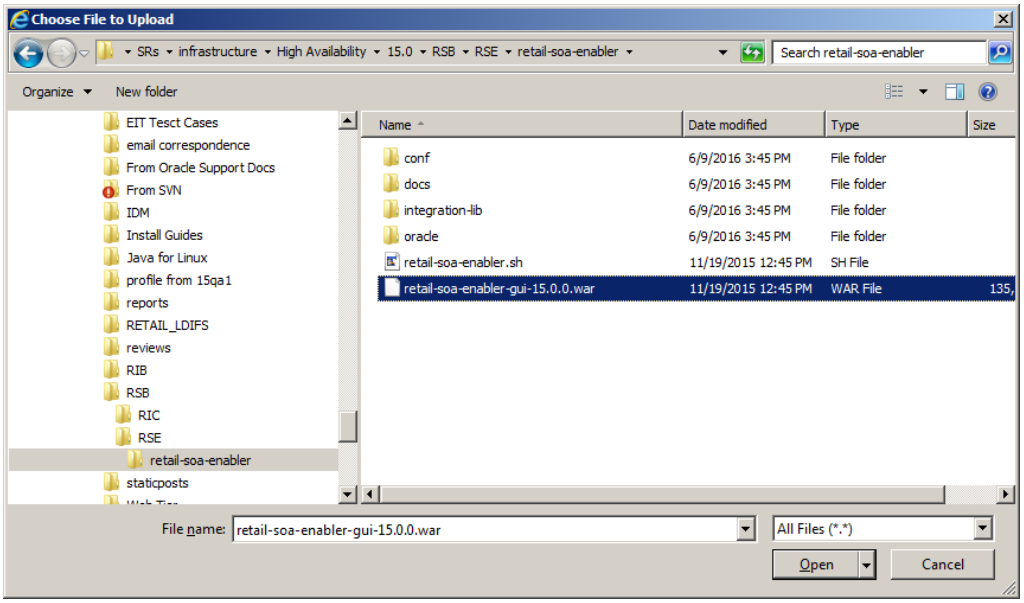
- The Install Application Assistant window opens. Click the upload your file(s) link that is located in the middle of the screen.



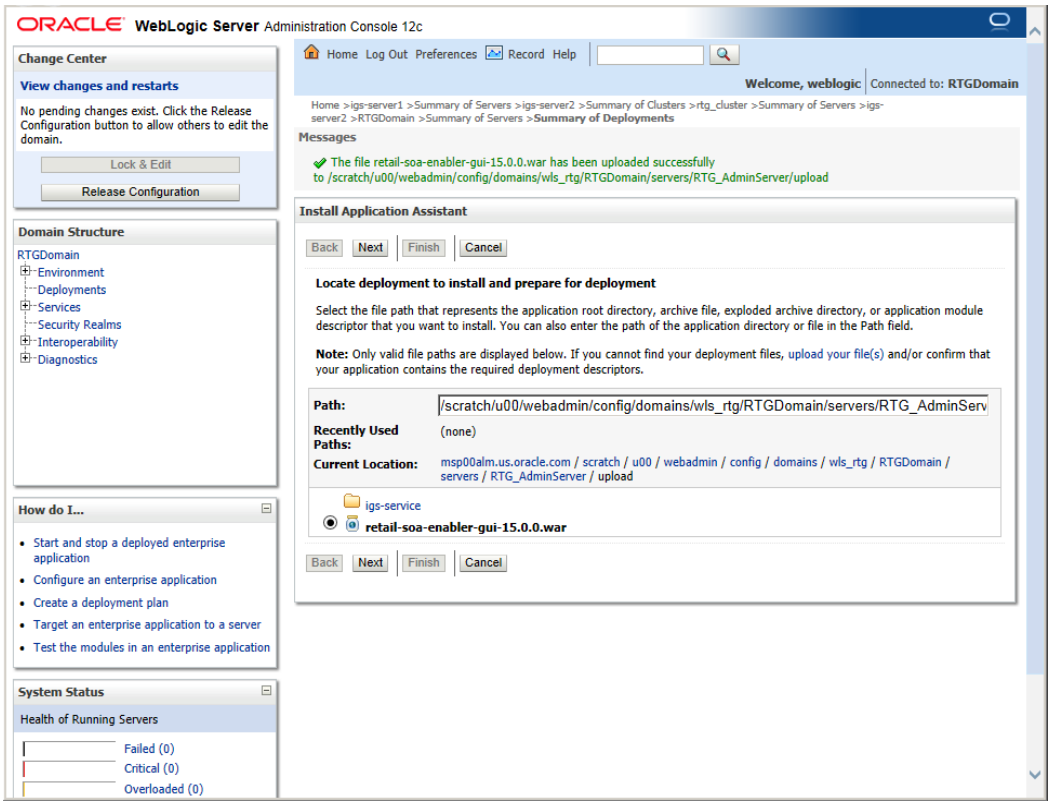
- Click the **Browse…** button located to the right of the Deployment Archive text box.



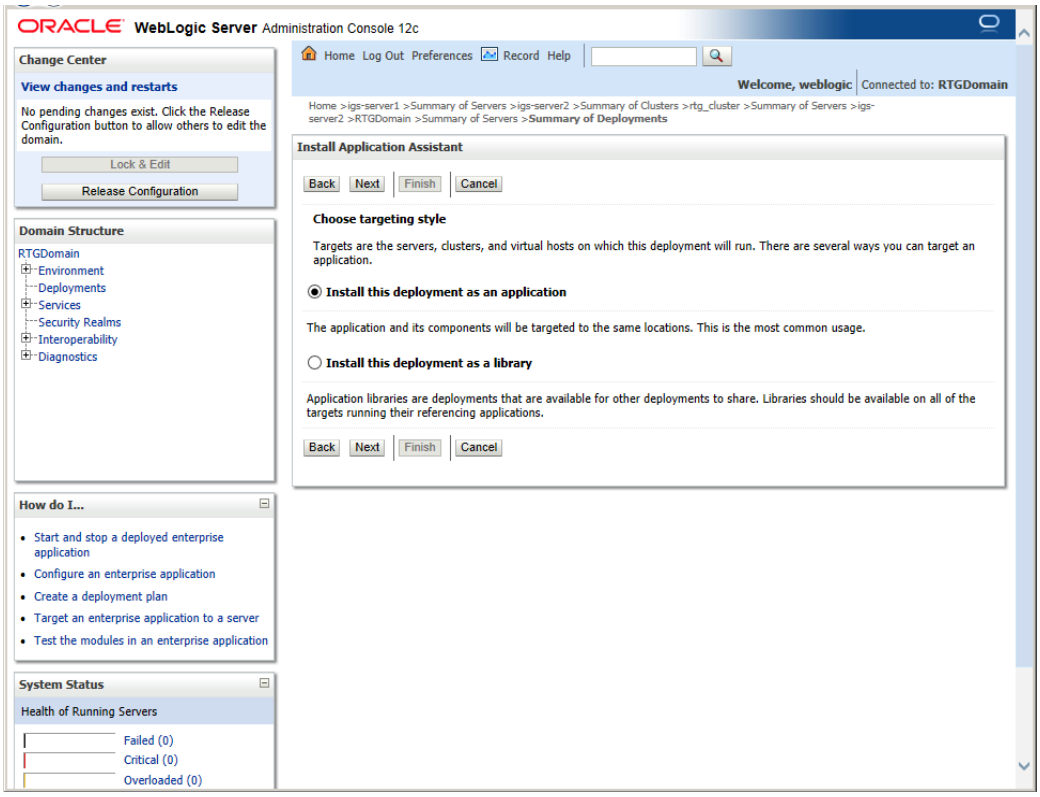
- An explorer window opens. Navigate to the location of the retail-soa-enabler-gui-15.0.0.war file and select it. The retail-soa-enabler-gui-15.0.0.war file will be inside the retail-soa-enabler directory that was extracted from the RetailSOAEnabler15.0.0ForAll15.0.0Apps_eng_ga.tar file in the initial steps for this section.



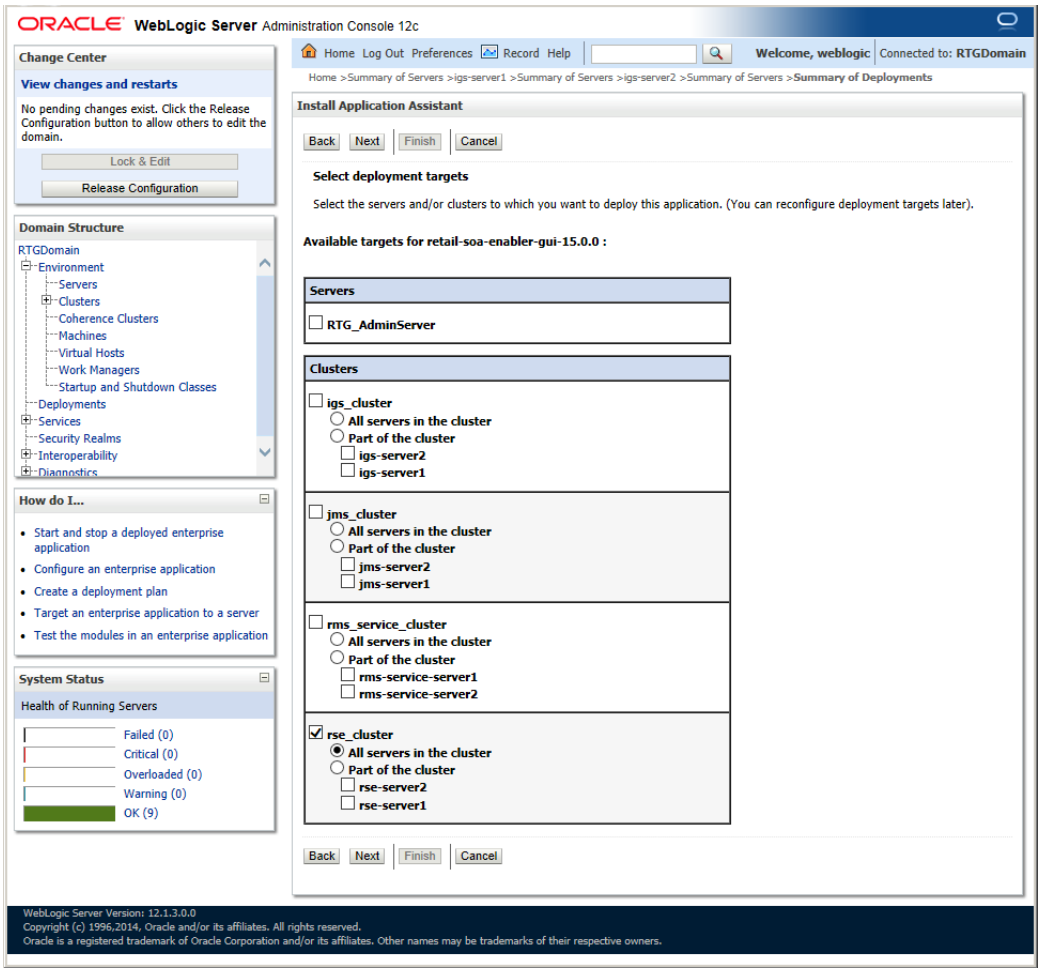
- In the Install Application Assistant window, click **Next**.



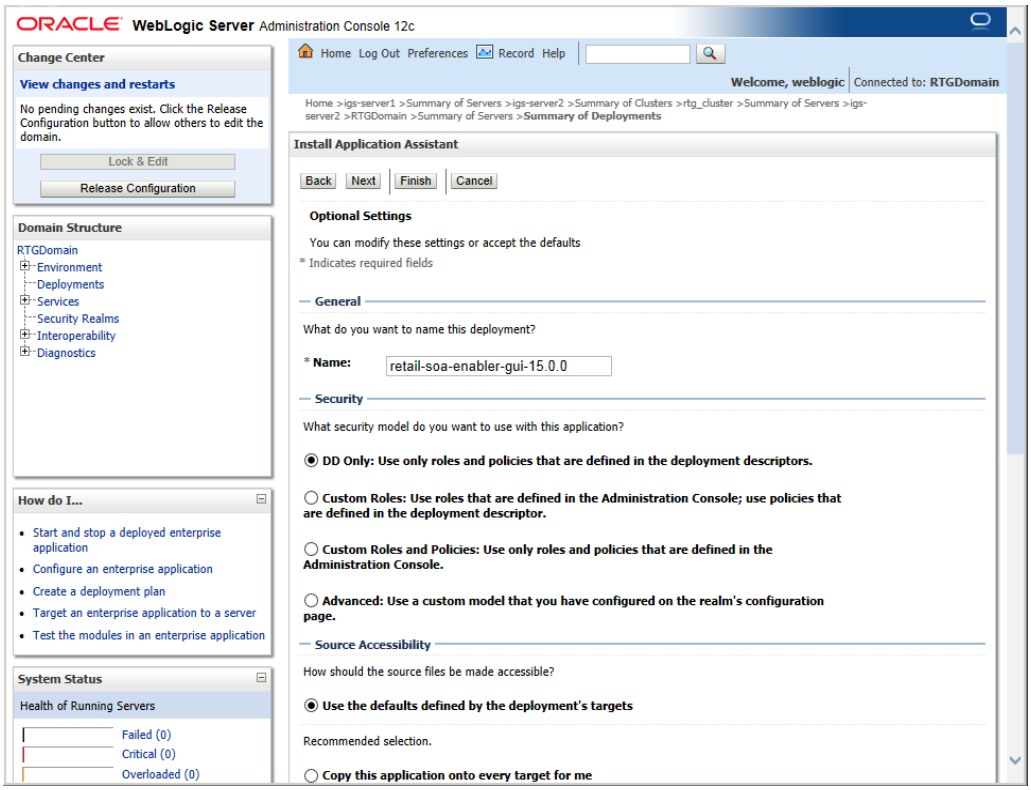
- The Choose Targeting Style window opens. Make sure the **Install this deployment as an application** option is selected and click **Next**.



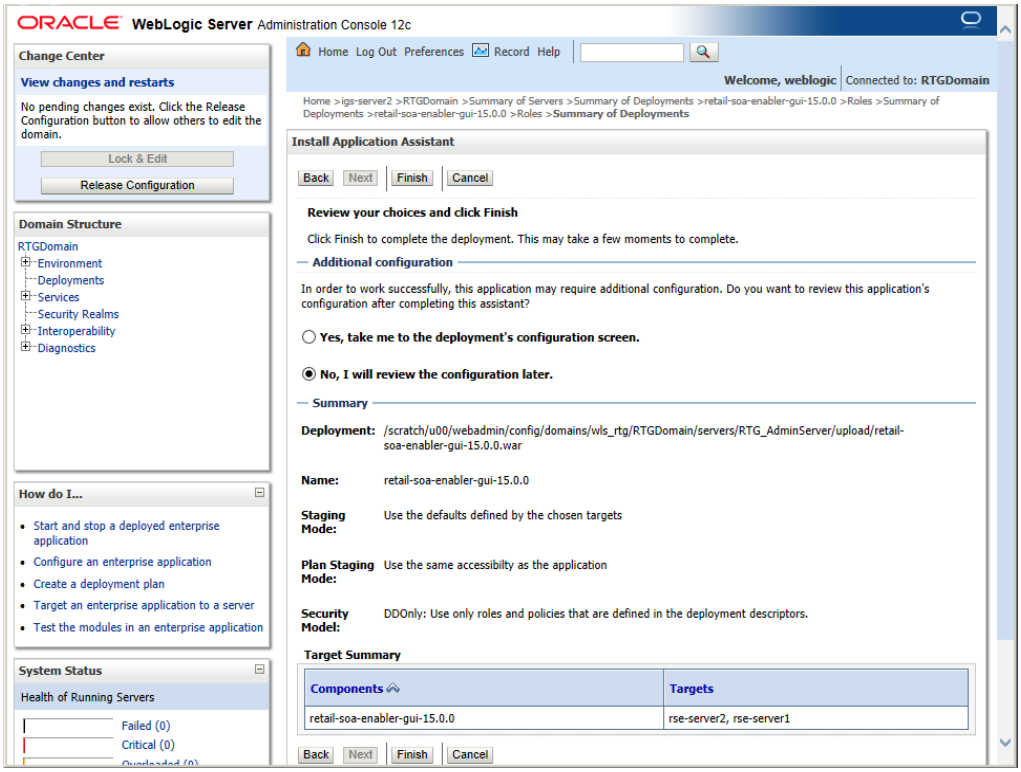
- The Select deployment targets window opens. Select **rse_cluster,** and click **Next**.



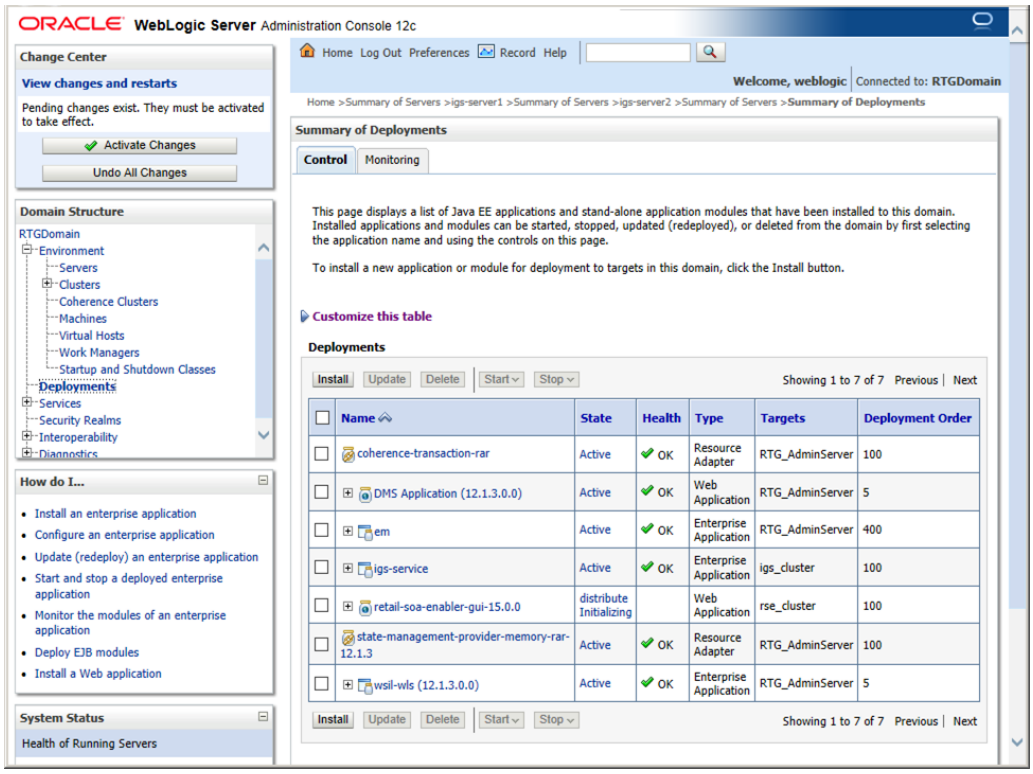
- The Optional Settings window opens. Click **Next**.



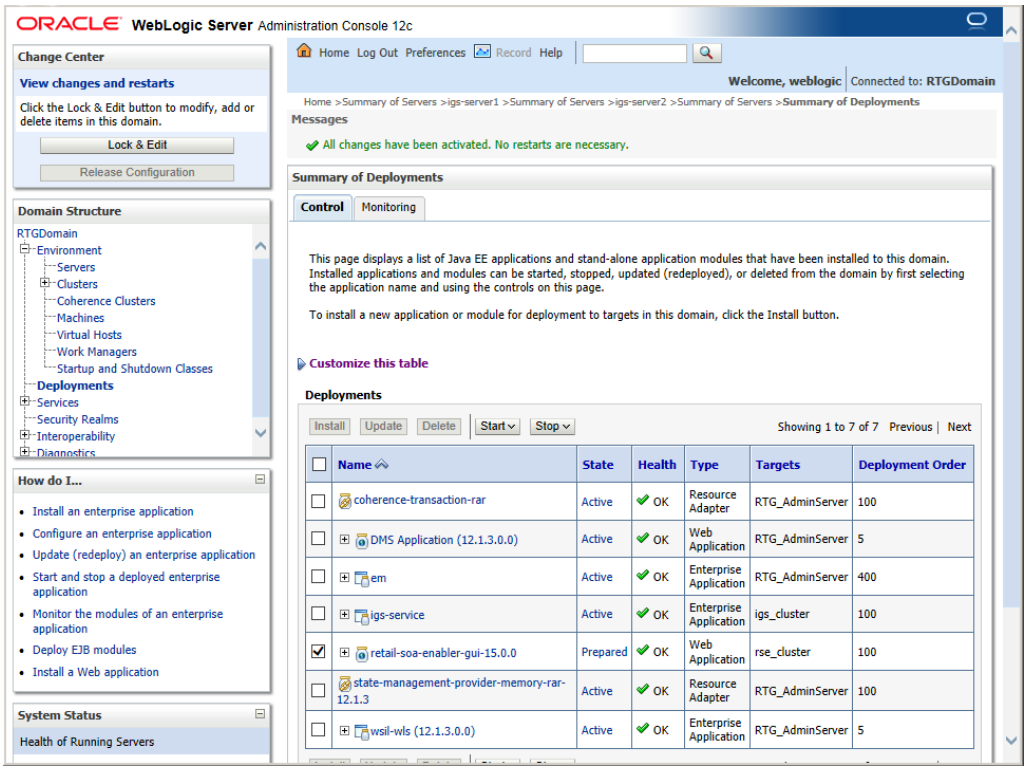
- The Review your choices window opens. Click **Finish** and then click the Deployments link located in the navigation pane on the left side of the screen.



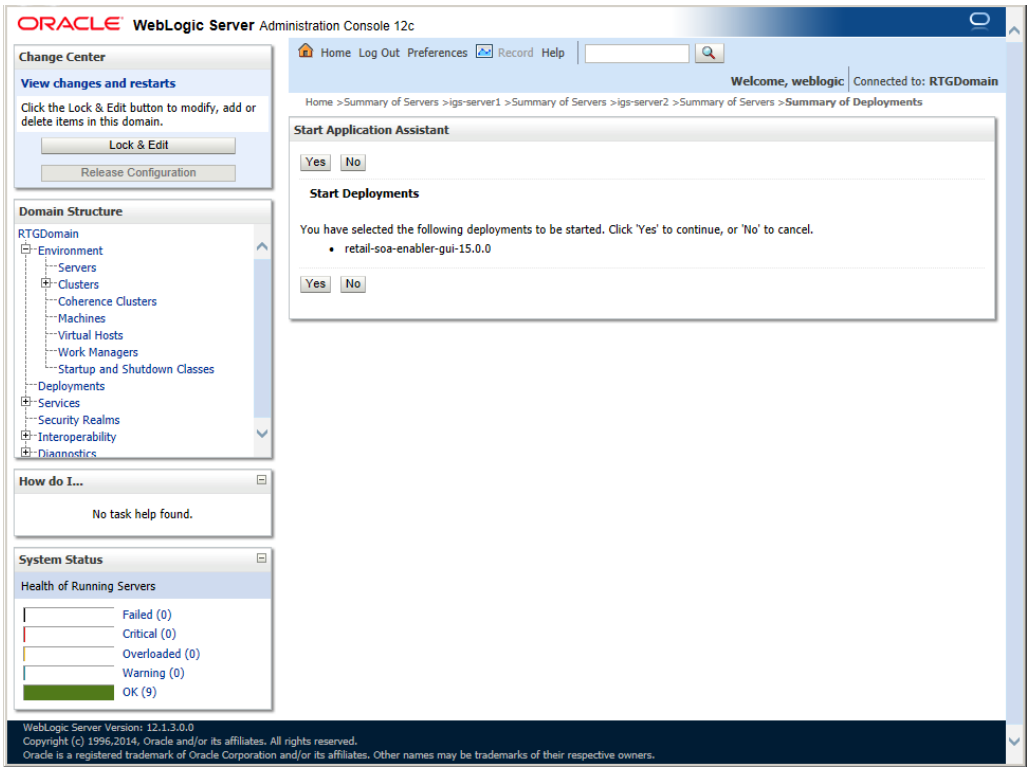
- The Summary of Deployments window opens. Click **Activate Changes**.



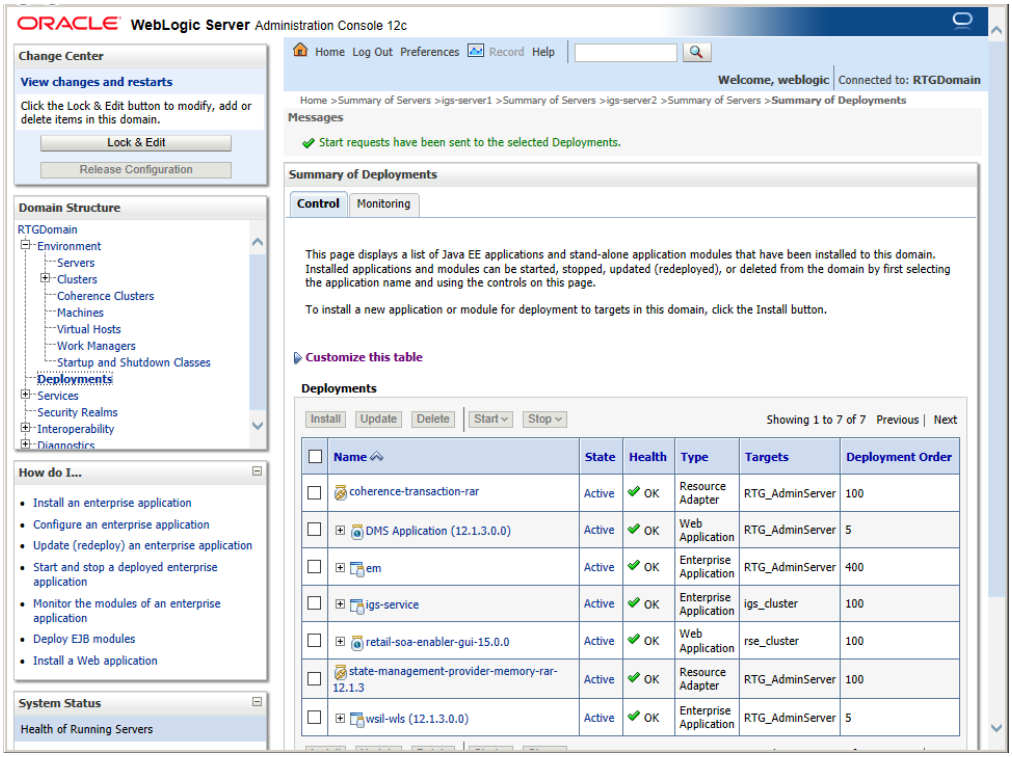
- The Start button is enabled for the deployments. Select the new retail-soa-enabler-gui-15.0.0 deployment and then use the Start button to start the deployment.



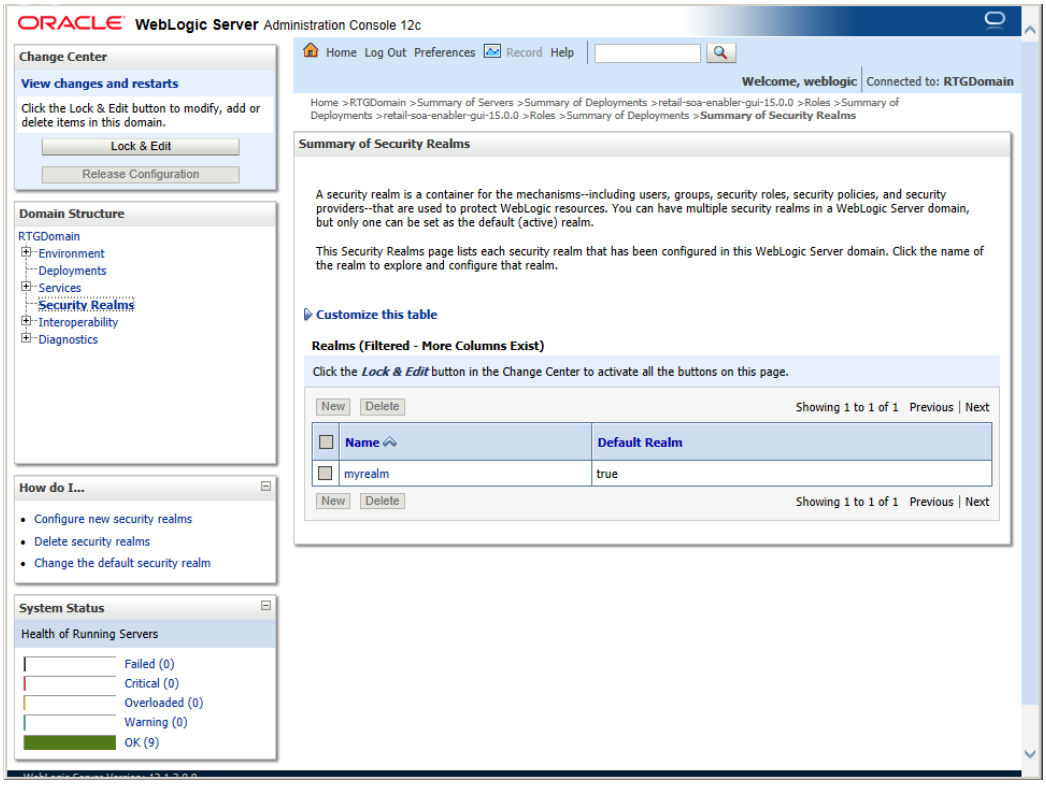
- The Start Application Assistant window opens. Click **Yes**.



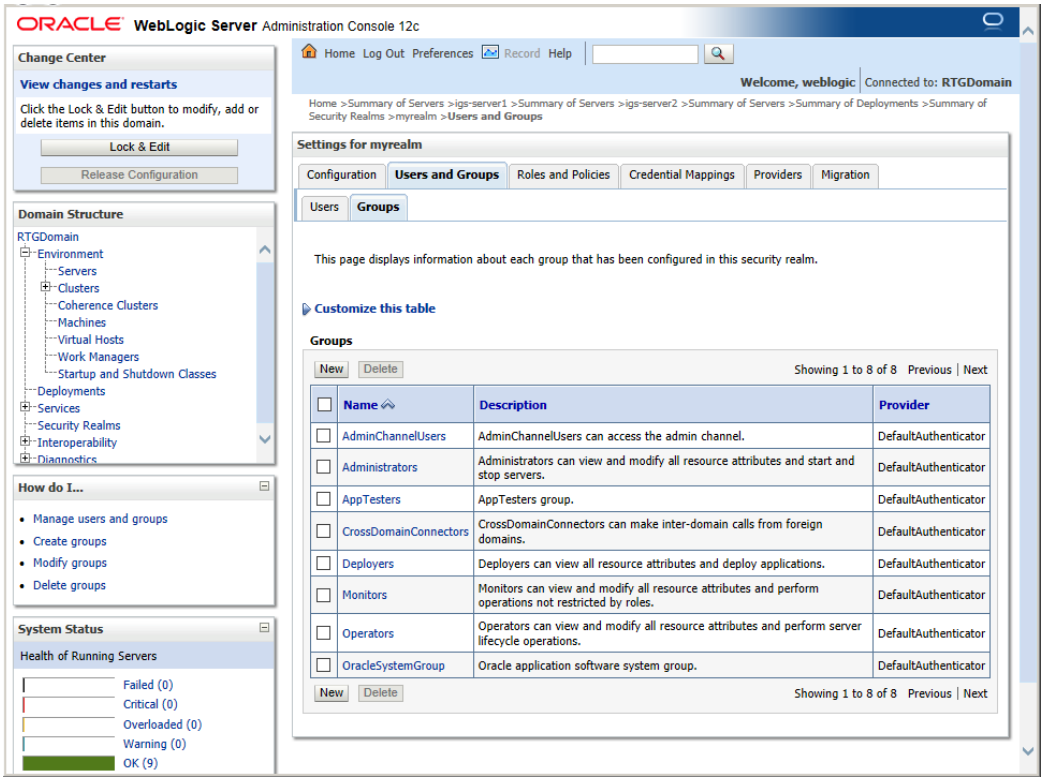
- The summary of Deployments window opens. Click on the **Security Realms** link located in the navigation pane on the left side of the screen.



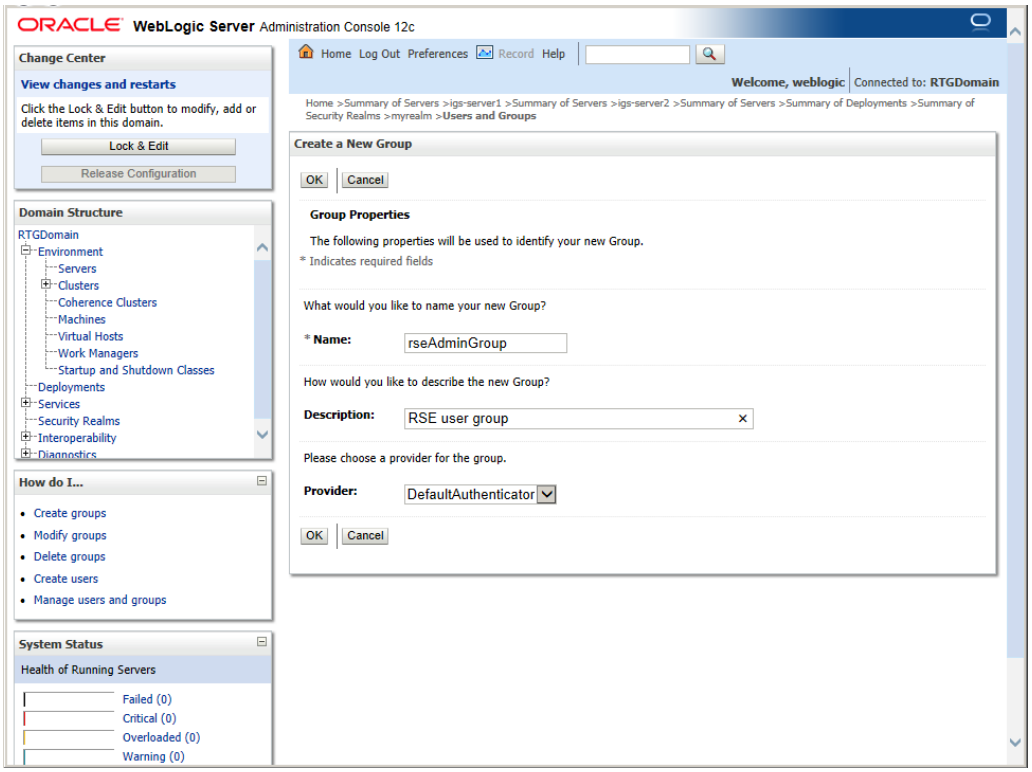
- The Summary of Security Realms window opens. Click on the **myrealm** link.



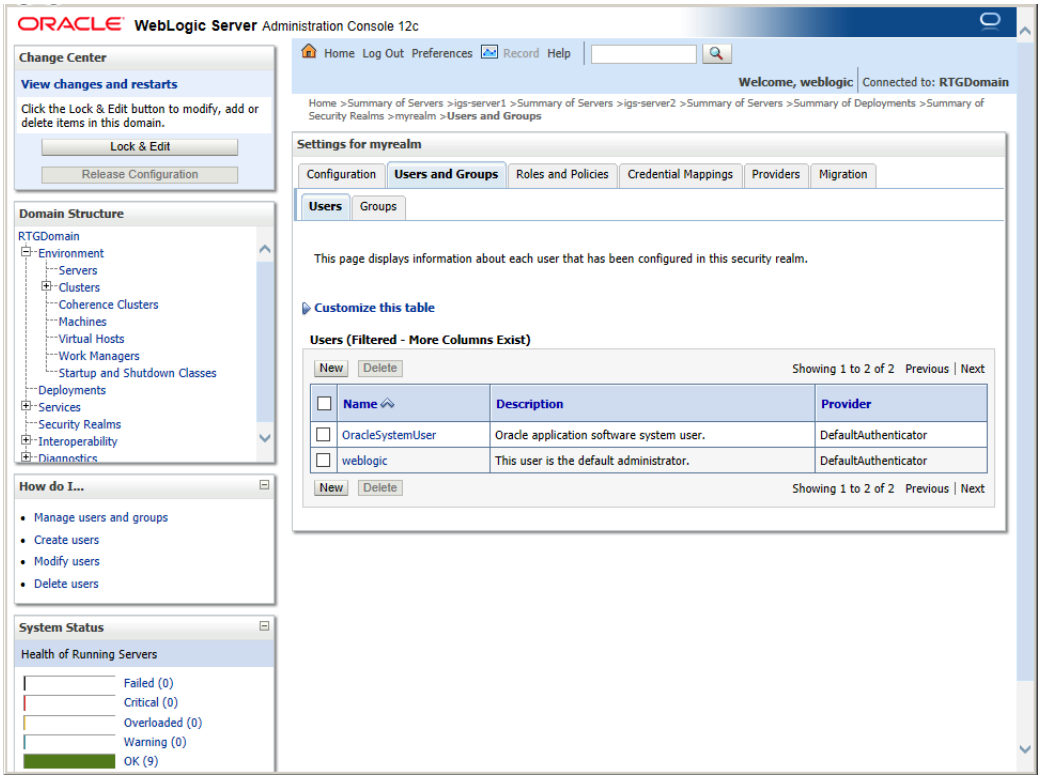
- The Settings for myrealm window opens. Under the **Users and Groups > Groups** tab, click **New**.



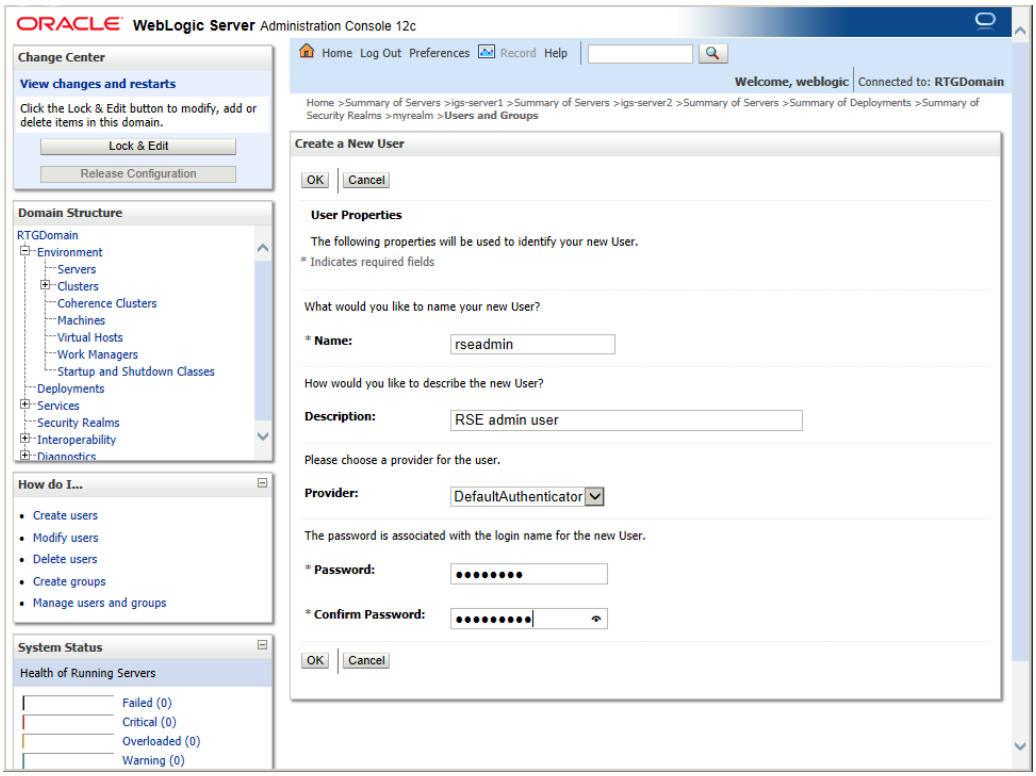
- The Create a New Group window opens. Enter in the Name: rseAdminGroup, then add a description and select **DefaultAuthentication** as the Provider. Click **OK**.



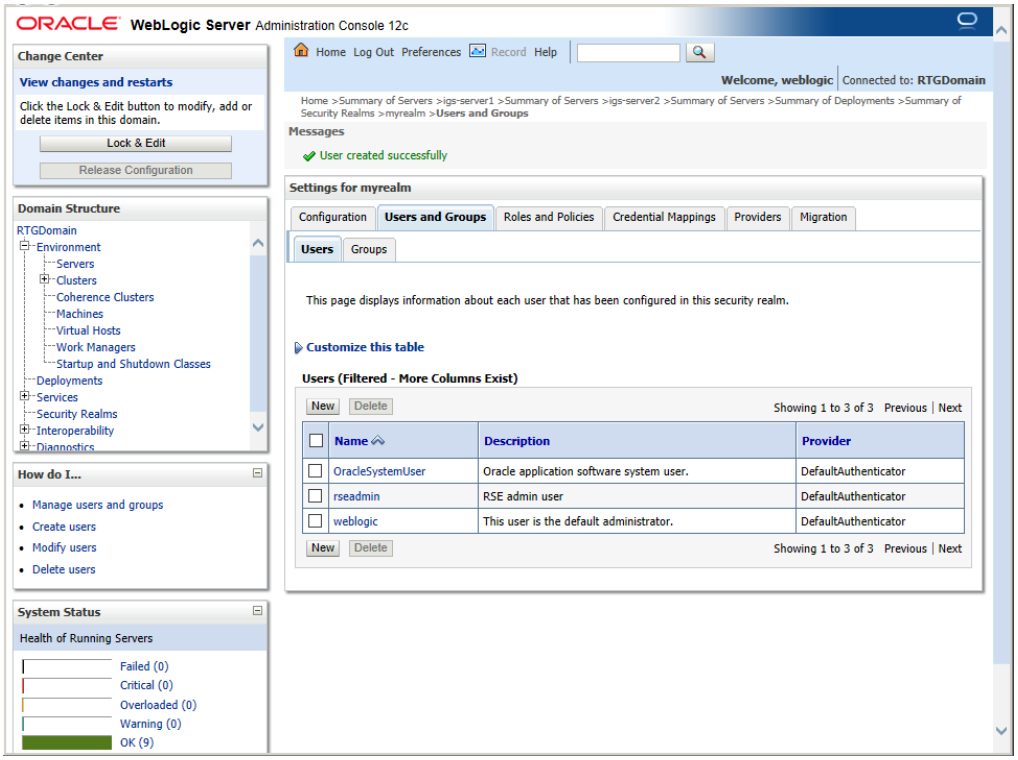
- The Settings for myrealm window opens. Under the Users tab, click **New**.



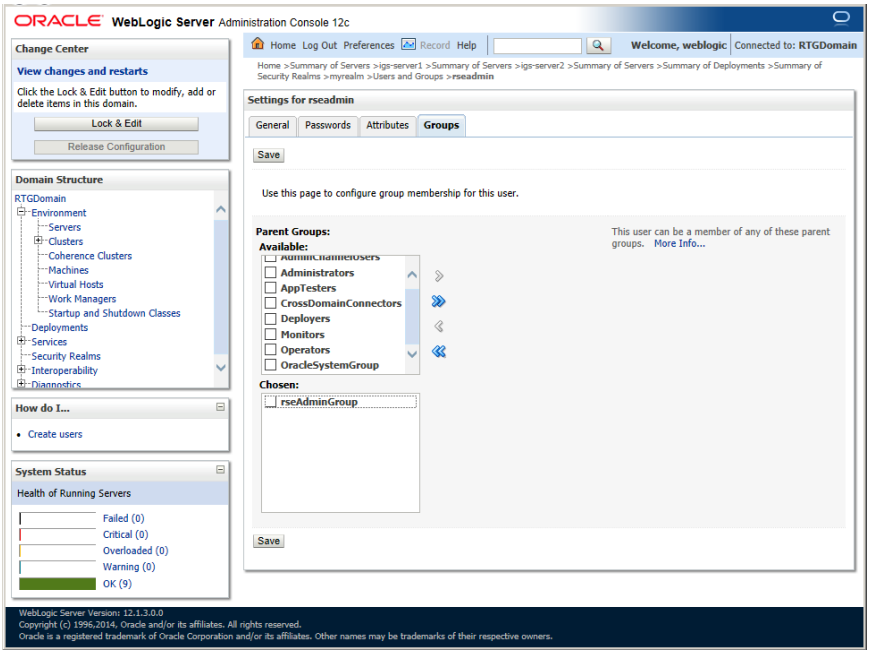
- The Create a New User window opens. Enter in the details for a user called rseadmin with the Provider set as DefaultAuthenticator and click **OK** .



- The Settings for myrealm window opens. Click the **rseadmin** user link.



- Select the **rseAdminGroup** and move it to the Chosen section of the screen, and click **Save**.



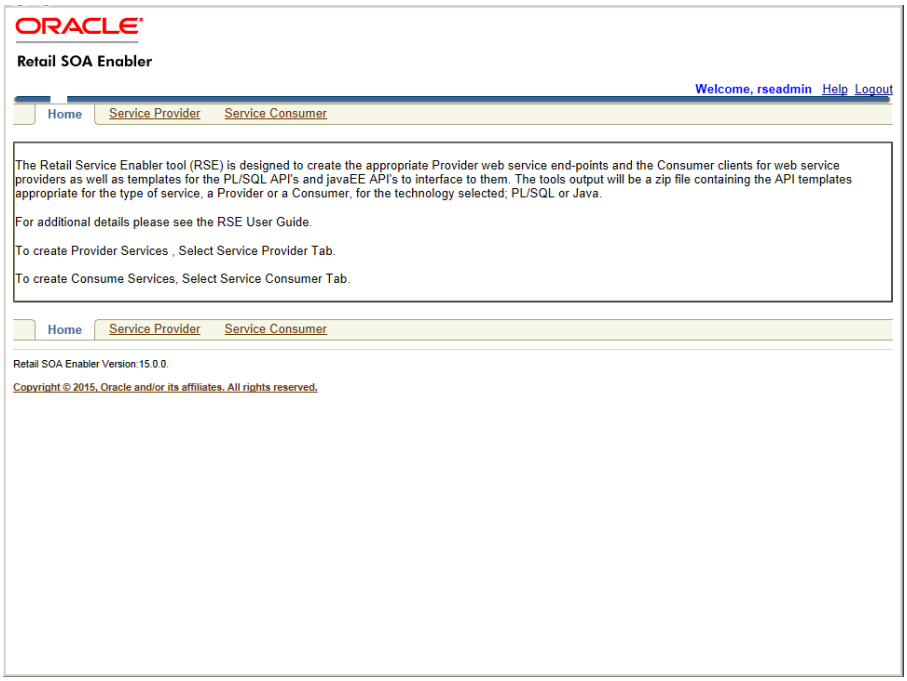
- Installation of RSE is complete.

Verification - Verify RSE Installation

Perform the following steps to validate the RSE installation by logging onto the application.

- Launch the following URL's to make sure the application is accessible from both rtghost1 and rtghost2.
  - http://rtghost1.us.oracle.com:7035/retail-soa-enabler-gui-15.0.0/index.jsp
  - http://rtghost2.us.oracle.com:7035/retail-soa-enabler-gui-15.0.0/index.jsp

- Validate that the applications home page is displayed. Click on the other tabs to see if they are accessible as well.



- Test complete.

Install JMS Application

Perform the following steps on rtghost1 to install the RSE application.

- Navigate to the <RIB_HOME>/tools-home/tools-home/jms-console/conf directory and edit the jms-console-deployment-env-info.json file so that the database connection and application server details are correct. By default the file contains 3 database connection specifications. For this case study only 1 is being used to remove the extras.
- Example: Highlighted text

```
{
    "JmsConsoleDeploymentEnvInfo":{
        "DataSourceDef":{
            "AqDataSource":[
            {
                "dataSourceName":"AqDataSource1",
                "dataSourceClass":"oracle.jdbc.pool.OracleDataSource",
                "dataSourceJndiName":"jdbc/AqDataSource1",
                "jdbcUrl":"jdbc:oracle:thin:@//dbhost-
r:1521/ribservicename",
                "jdbcUserAlias":"aqDataSourceUserAlias1",
                "jdbcUser":"GET_FROM_WALLET",
                "jdbcPassword":"GET_FROM_WALLET"
            }
            ]

        },
        "MiddlewareServerDef":{
            "JmsConsoleAppServer":{

                "weblogicDomainName":"RTGDomain",
```

```
"weblogicDomainHome":"/u00/webadmin/config/domains/wls_rtg/RTGDomain",

"weblogicDomainAdminServerUrl":"t3://rtghost1.us.oracle.com:7031",
                "weblogicDomainAdminServerProtocol":"t3",
                "weblogicDomainAdminServerHost":"rtghost1.us.oracle.com",
                "weblogicDomainAdminServerPort":"7031",

"weblogicDomainAdminServerUserAlias":"jmsConsoleAppServerAdminServerUserAl
ias",
                "weblogicDomainTargetManagedServerName":"jms-server1",


"jmsConsoleAdminUiUrl":"http://rtghost1.us.oracle.com:7047/jms-console",
                "jmsConsoleAdminUiUserGroup":"JmsConsoleAdminGroup",
                "jmsConsoleAdminUiUserAlias":"jmsConsoleAdminUiUserAlias",
                "jmsConsoleAdminUiUser":"GET_FROM_WALLET",
                "jmsConsoleAdminUiPassword":"GET_FROM_WALLET",


            }
        },
        "JmsConsoleApplication":{
            "jmsConsoleAppUses":[
                "AqDataSource",
                "JmsConsoleAppServer"
            ]
        }
    }
}
```

- Set the following variables by exporting them before running the installer so that the JAVA_HOME is added to the PATH variable:
  - JAVA_HOME
  - PATH
  - WL_HOME

  Example:

  ```
  $ export JAVA_HOME=/u00/webadmin/products/jdk_java
  $ export PATH=$JAVA_HOME/bin:$PATH
  $ export WL_HOME=/u00/webadmin/products/wls_rtg/wlserver
  ```

- Navigate to <RIB_HOME>/tools-home/ jms-console/bin and run the jms-console-deployer.sh script to install JMS.

  Example:

  ```
  $ cd /u00/webadmin/RIB_INSTALL/Rib1500ForAll15xxApps/rib-home/tools-
  home/jms-console/bin
  $ sh jms-console-deployer.sh -setup-credentials -deploy-jms-console-app
  ```

- The installer will display prompts for the WebLogic administrator's user name and password, JMS application user and also the database credentials for the AqDataSource. Provide the credentials for the RTG WebLogic administrator and ribaq database users. Take note of the values that are provided for the jmsadmin user because it gets created based on what is provided for the prompts.

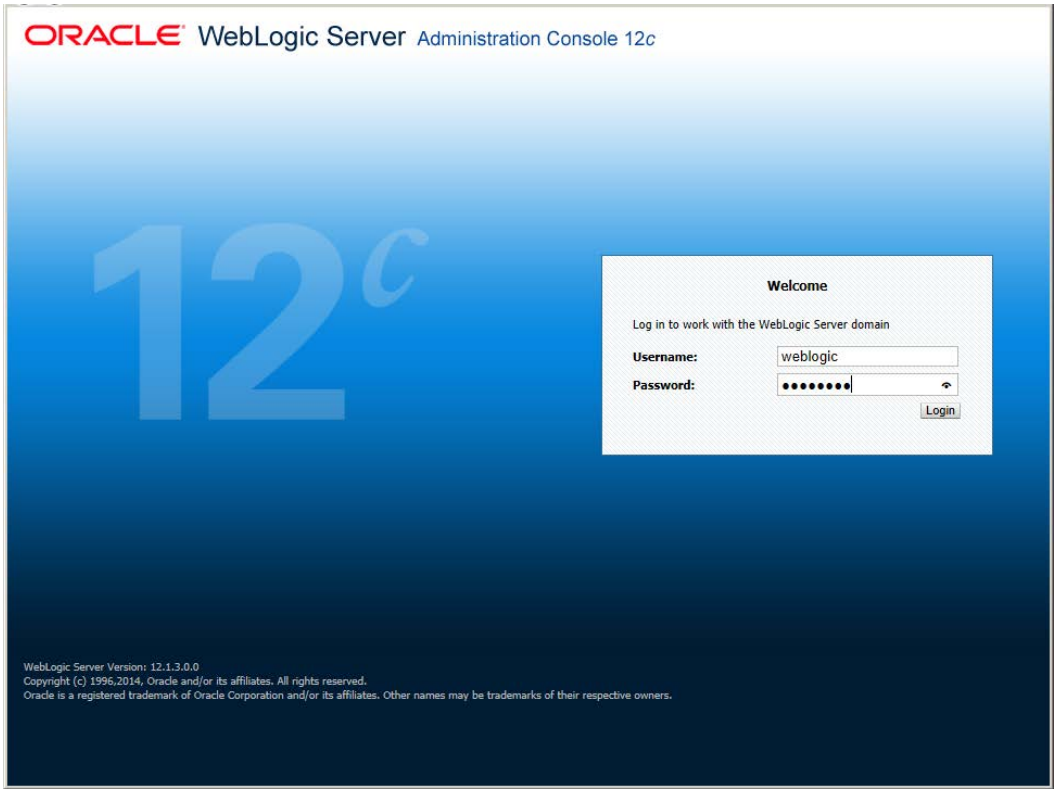
  Example:

  ```
  Credential required for
  weblogicDomainAdminServerHost(rtghost1.us.oracle.com)
  weblogicDomainAdminServerPort(7031):
  ```

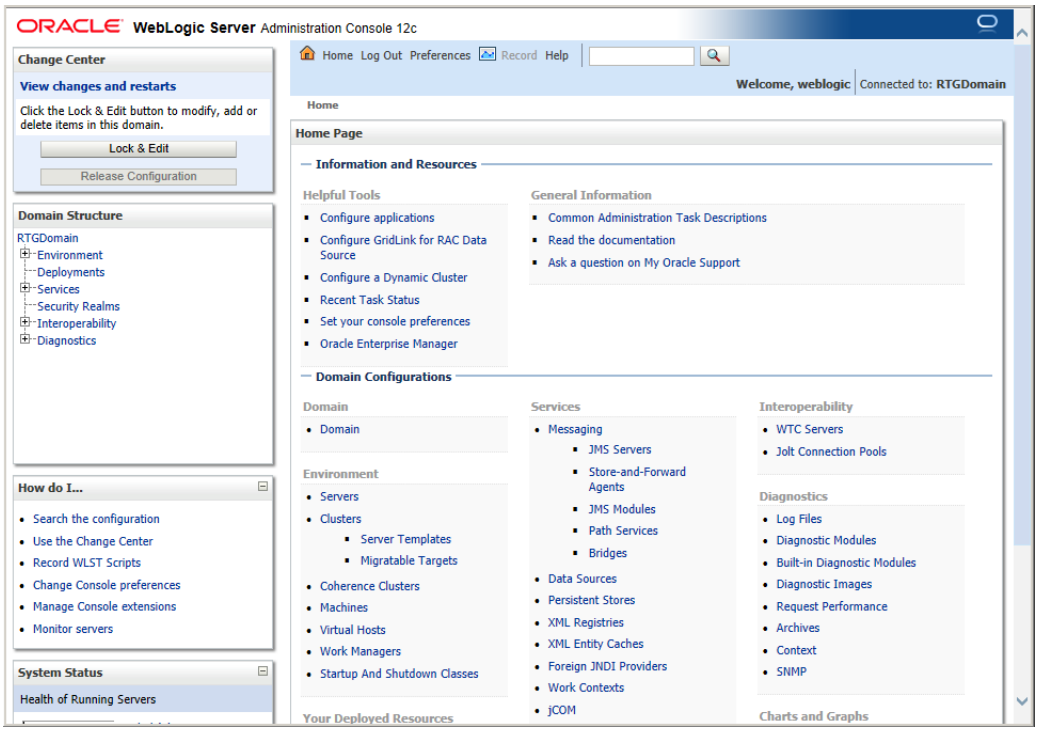
```
Enter username for alias
(jmsConsoleAppServerAdminServerUserAlias):weblogic
Enter Password: <password>

Credential required for
jmsConsoleAdminUiUrl(http://rtghost1.us.oracle.com:7047/jms-console):
Enter username for alias (jmsConsoleAdminUiUserAlias):jmsadmin
Enter Password: <password>

Credential required for dataSource(jdbc/AqDataSource1)
jdbcUrl(jdbc:oracle:thin:@//dbhost-r:1521/ribservicename):
Enter username for alias (aqDataSourceUserAlias1):retail
Enter Password: <password>
```

After the installer is finished running the deployment will have to have its target adjusted to the jms-cluster so that it gets distributed across both rtghost's. In addition to that the managed servers in the jms-cluster will need to have -Djava.awt.headless=true added to its startup tab. The following instructions are for performing these tasks.

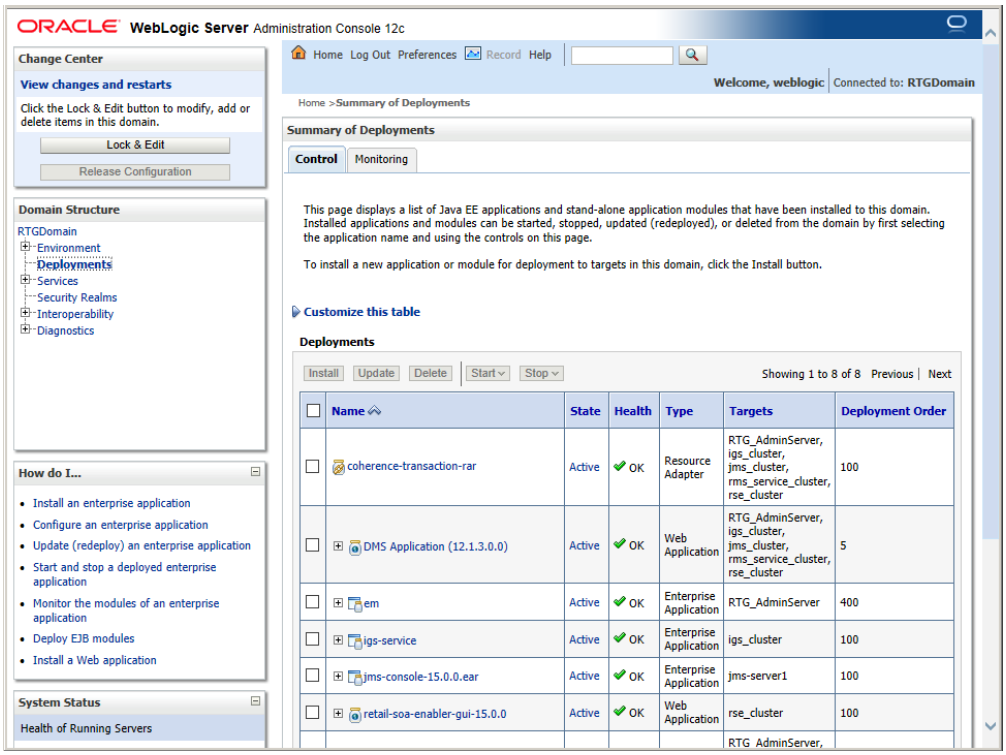
- Logon to the RTG WebLogic Administration Console.



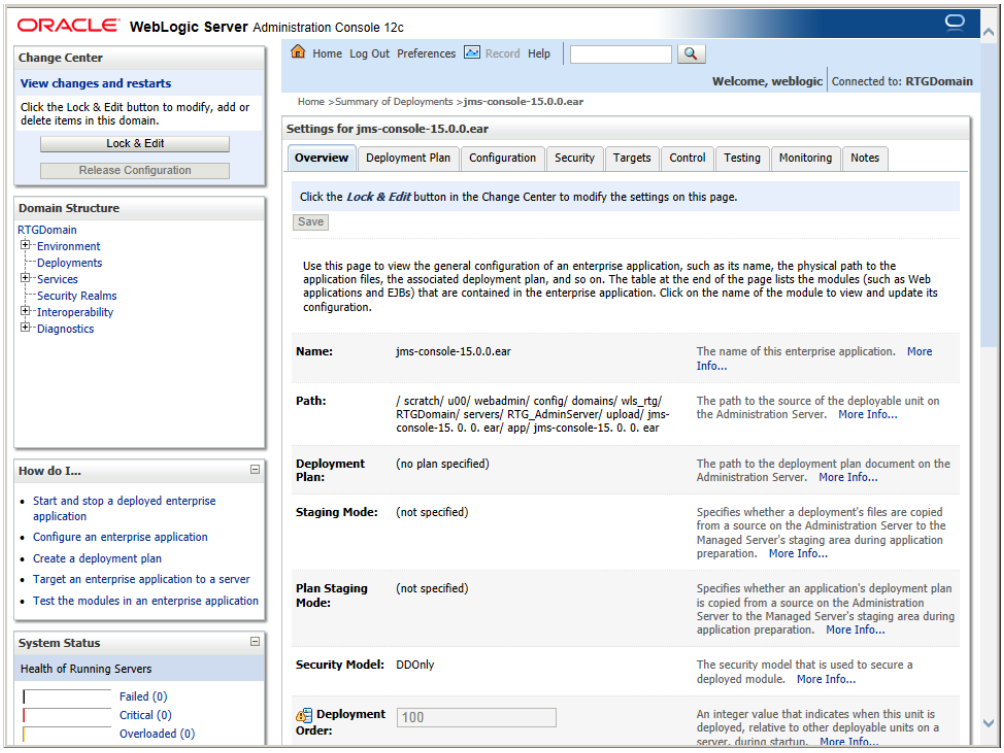
- The Home Page window opens. Click on the **Deployments** link located in the navigation pane on the left side of the screen.



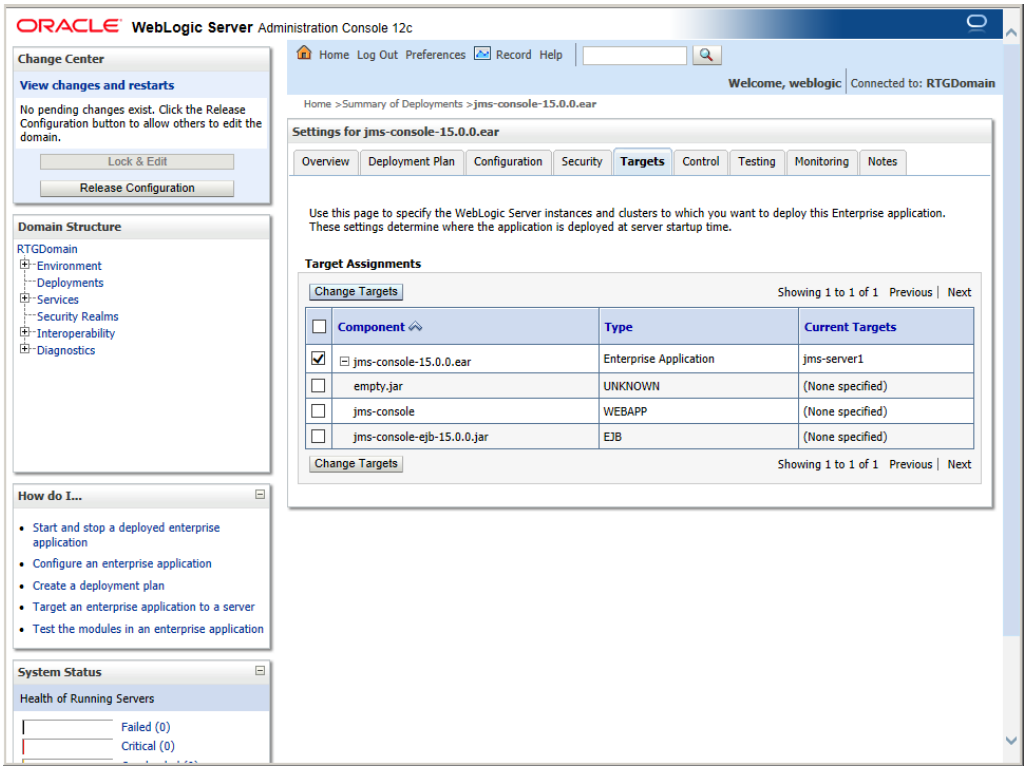
- The Summary of Deployments window opens. Click on the **jms-console-15.0.0.ear** link in the list of deployments.



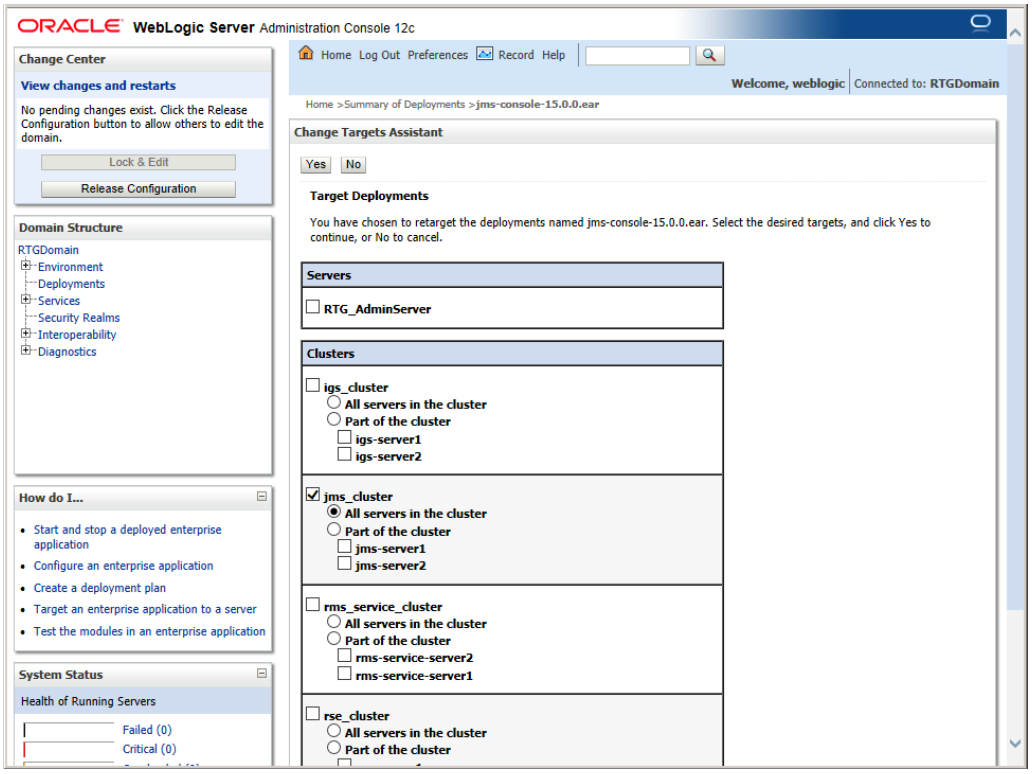
- The Settings for jms-console-15.0.0.ear window opens. Click on the **Targets** tab, and then click the **Lock & Edit** button located in the left corner of the screen.



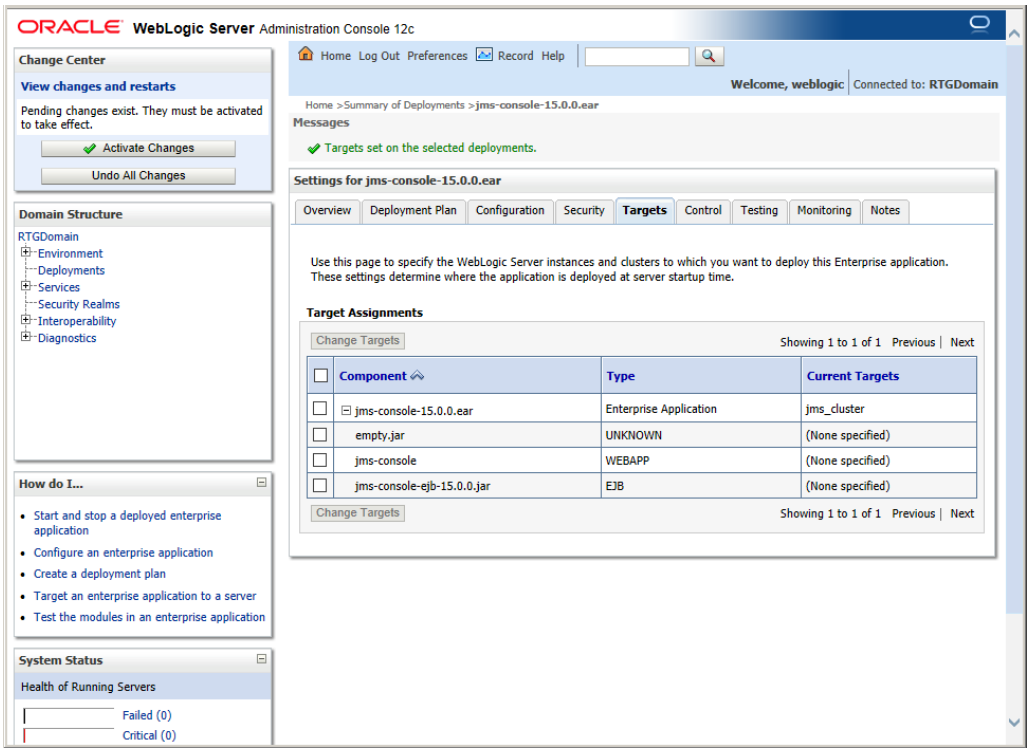
- The Target Assignments window opens. Select **jms-console-15.0.0.ear**, and click the **Change Targets** button.



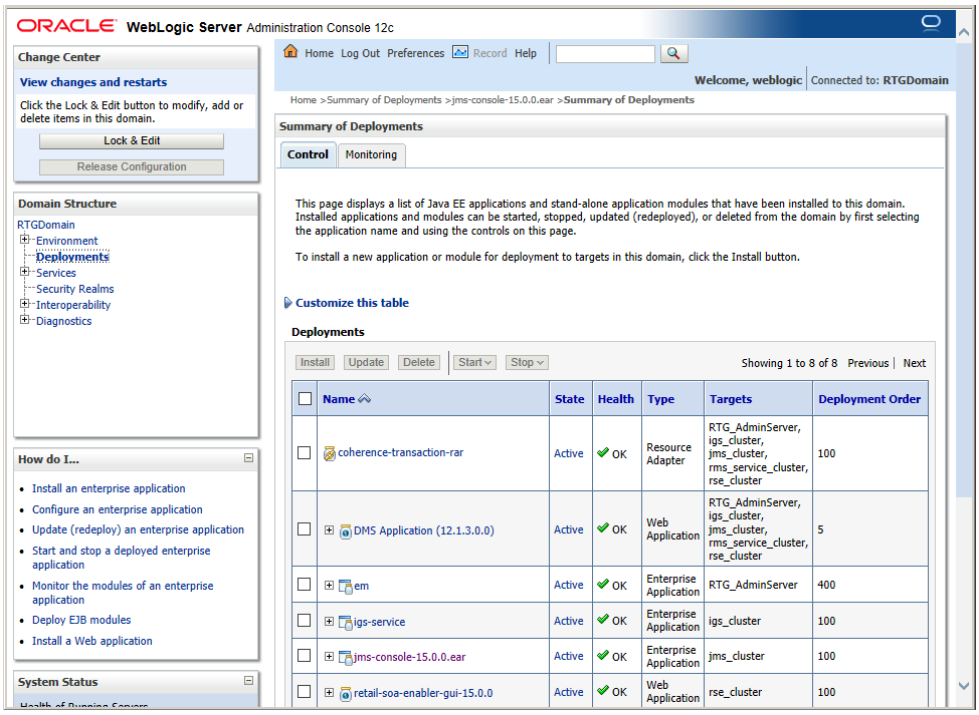
- The Change Target Assistant window opens. Select the **jms-cluster** option, and click **Yes**.



- The Target Assignments window opens. Click **Activate Changes** and then click the Deployments link located in the navigation pane on the left side of the screen.



- The Summary of Deployments window opens. Validate that the state is Active for the jms-console-15.0.0.ear deployment. If not start it.

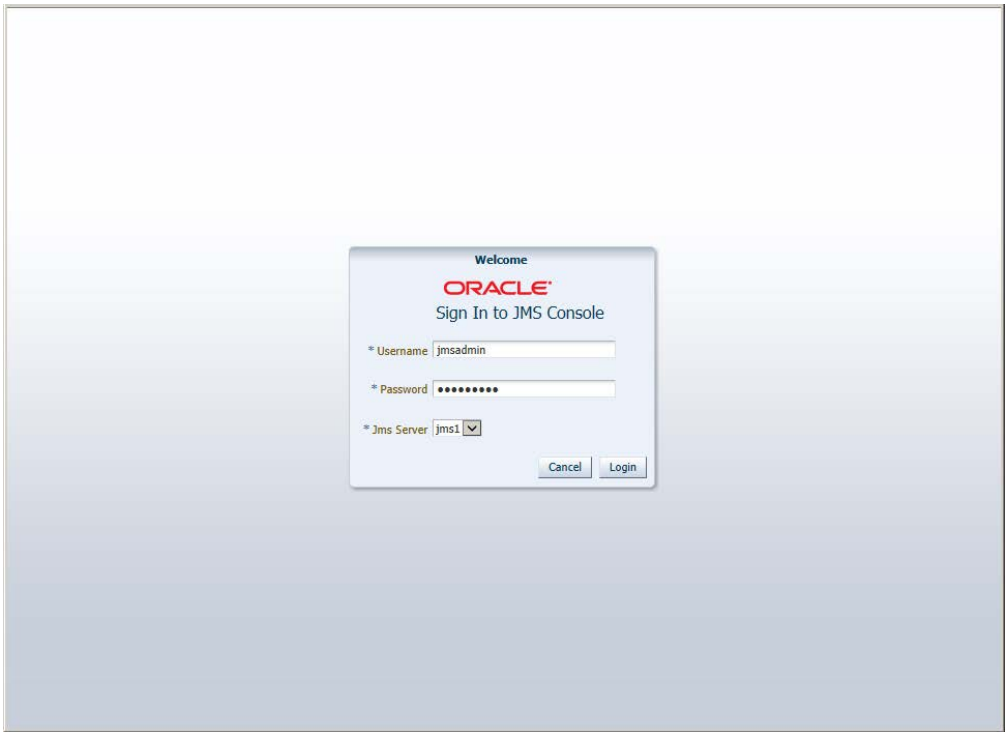


- Installation of RSE is complete.

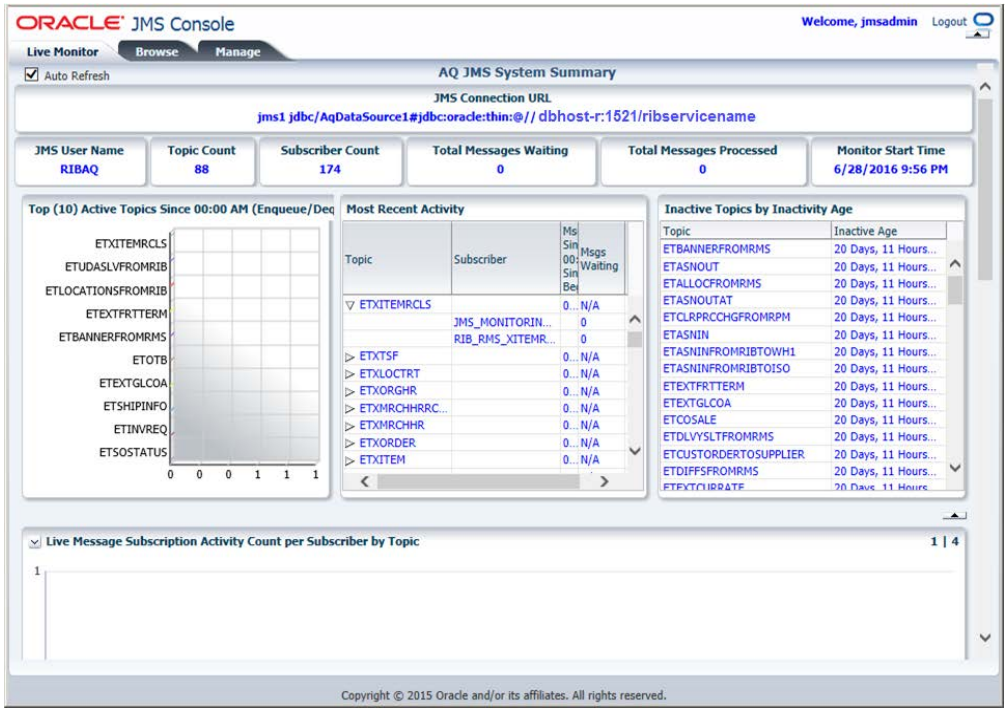
Verification - Verify JMS Installation

Perform the following steps to validate the JMS installation by logging onto the application.

- Launch the following URL's to make sure the application is accessible from both rtghost1 and rtghost2 and logon using the credentials use to create the jmsadmin user.
  - http://rtghost1.us.oracle.com:7047/jms-console/
  - http://rtghost2.us.oracle.com:7047/jms-console/



- The JMS Console window opens without errors.



- Test complete.

Deploy RMS Web Services on rms-service-server

This section covers the steps required to deploy RMS Web Services on the rms-service-server. The steps for deploying the web services include updates to the RMS database schema, creating a data source in WebLogic, deploying the jax-rs-2.0.war and deploying the rms-services.ear file to the rms-service-cluster in WebLogic.

Perform the following steps to deploy RMS Web Services on the rms-service-server.

- Logon to dbhost1 and copy the DrillBackForwardUrlServiceConsumer_grant.sql and GlAccountValidationServiceConsumer_grant.sql scripts to dbhost1 from rms application distribution installation directory located on apphost1.

  Example:

  ```
  $ mkdir -p /u00/webadmin/rms_sql_scripts/
  $ cd /u00/webadmin/rms_sql_scripts/

  $ scp
  webadmin@apphost1.us.oracle.com:/u00/webadmin/media/RMS/CDROM/rms/installe
  r/mom15/Cross_Pillar/webservice_objects/consumer/sql/DrillBackForwardUrlSe
  rviceConsumer_grant.sql .
  $ scp webadmin@apphost1.us.oracle.com:
  /u00/webadmin/media/RMS/CDROM/rms/installer/mom15/Cross_Pillar/webservice_
  objects/consumer/sql/GlAccountValidationServiceConsumer_grant.sql .
  ```

- Update the newly copied files DrillBackForwardUrlServiceConsumer_grant.sql and GlAccountValidationServiceConsumer_grant.sql on dbhost1 by changing all occurrence of <USER> to the RMS schema owner.

  Example:

  ```
  ...
  dbms_java.grant_permission('RMS01', 'SYS:java.lang.RuntimePermission',
  'getClassLoader', '' );
  ...
  ```

- On dbhost1 – setup environment variables that will allow the sqlplus to be able to log on as the sys user by running the oraenv script and providing database details for the prompts.

  Example: (Inputs highlighted)

  ```
  $ . oraenv

  ORACLE_SID = [webadmin] ? appservicename
  Cannot locate ORACLE_HOME.
  ORACLE_HOME = [] ? /u00/oracle/product/12.1.0.2/
  ORACLE_BASE environment variable is not being set since this
  information is not available for the current user ID webadmin.
  You can set ORACLE_BASE manually if it is required.
  Resetting ORACLE_BASE to its previous value or ORACLE_HOME
  The Oracle base has been set to /u00/oracle/product/12.1.0.2/
  ```

- Log onto the sqlplus as the sys user and set the session to the app server's database container.

  ```
  $ sqlplus sys/sysdolrc04@dolrc041 as SYSDBA
  SQL> alter session set container = dolrp04app;
  Session altered.
  ```

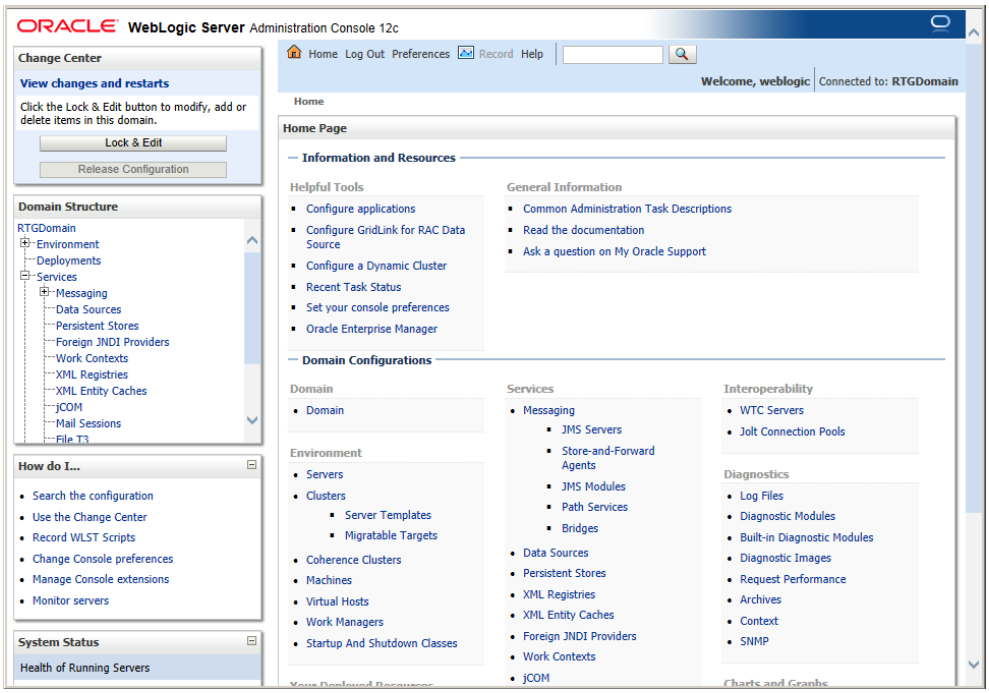
- Run the DrillBackForwardUrlServiceConsumer_grant.sql and GlAccountValidationServiceConsumer_grant.sql scripts

  ```
  SQL> @DrillBackForwardUrlServiceConsumer_grant.sql
  PL/SQL procedure successfully completed.
  SQL> @GlAccountValidationServiceConsumer_grant.sql
  ```

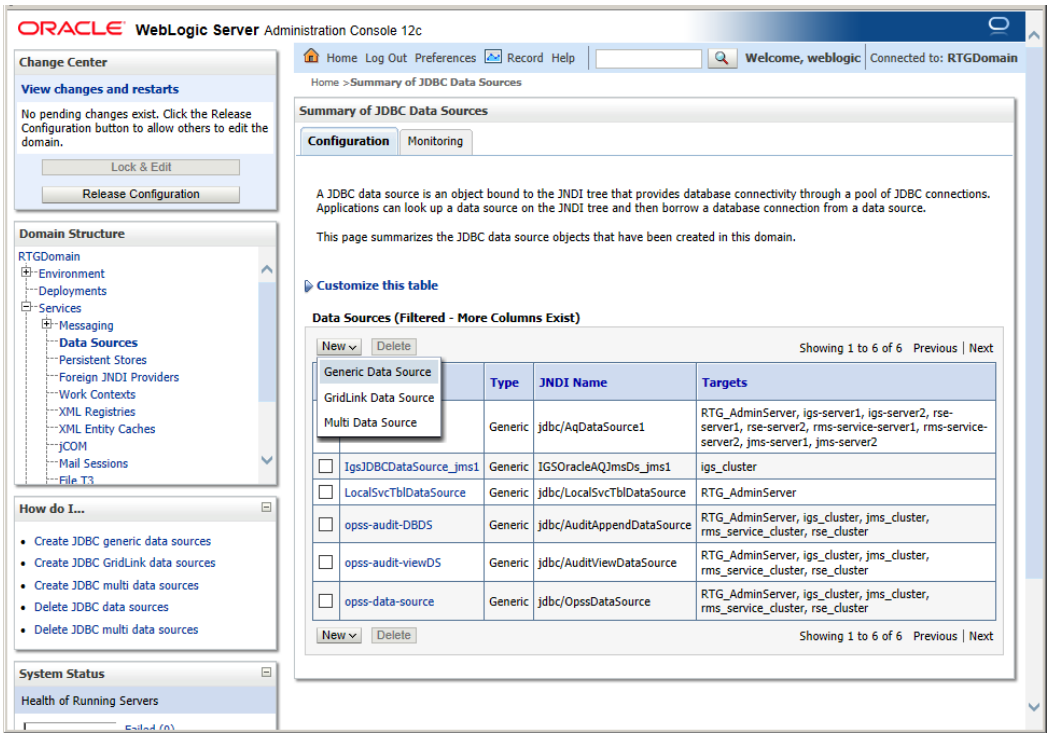
```
PL/SQL procedure successfully completed.
```

- On apphost1 – Navigate to the <RMS_STAGING>/ /rms/installer/mom15/Cross_Pillar/webservice_objects/provider/ear directory and upload the rms-service.ear file to local computer where it can be accessed by an explorer window and deployed to the rms-service-server.
- Login to the WebLogic Administration Console.

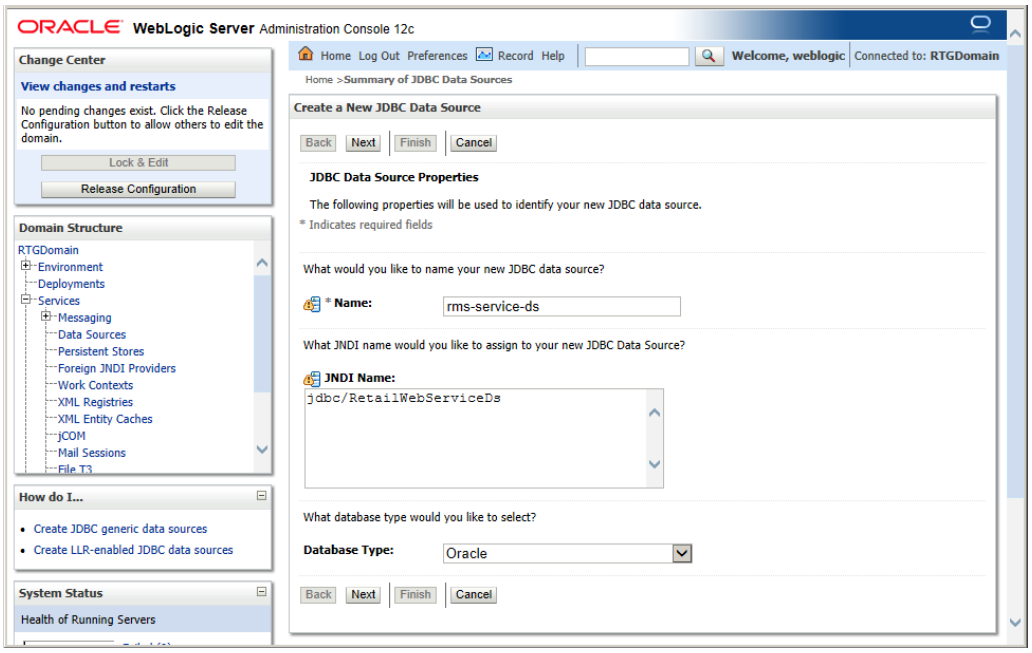
- The Home Page window opens. Click the **Data Sources** link in the navigation pane on the left side of the screen.



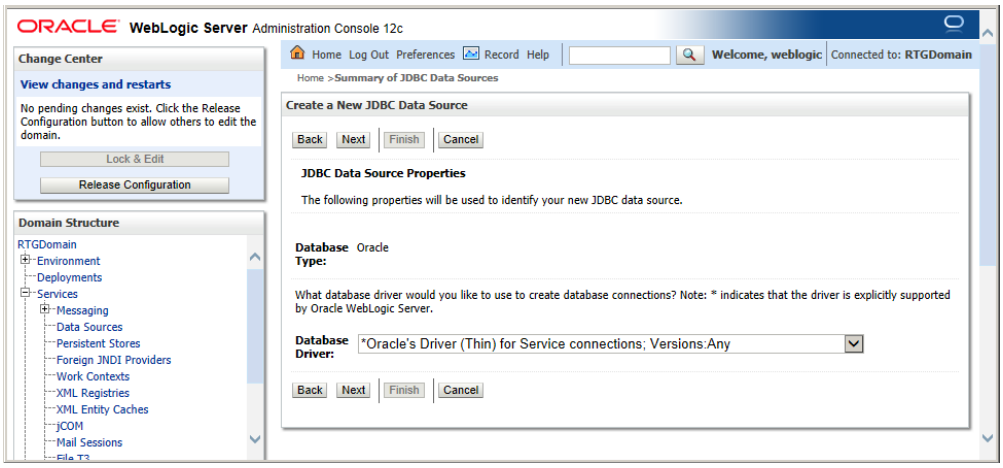
- The Summary of JDBC Data Sources window opens. Click **Lock & Edit** and then click **New > Generic data Source** buttons.



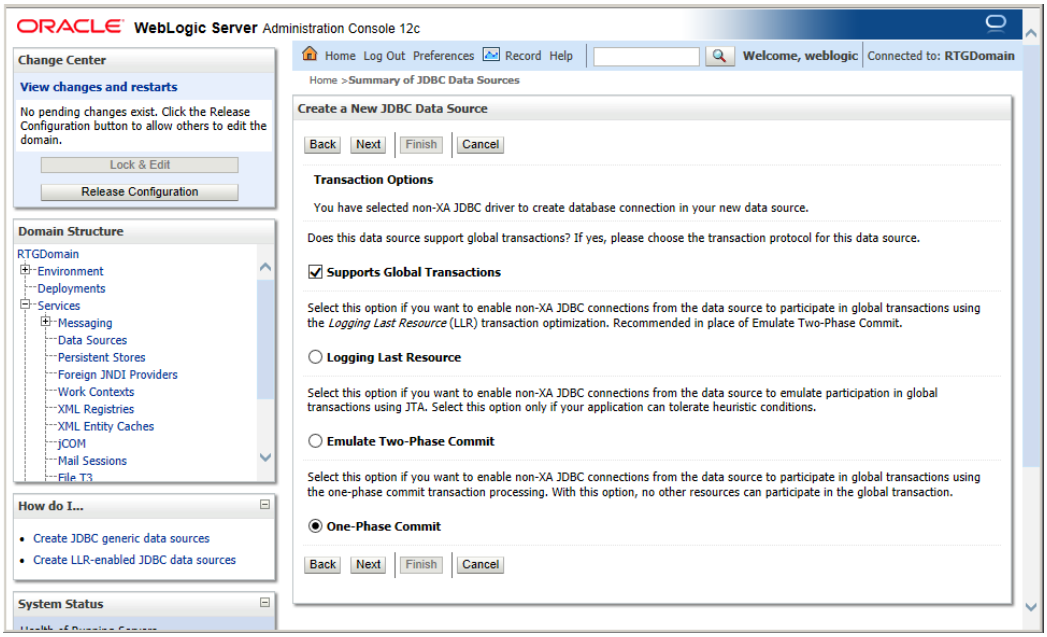
- The Create a New JDBC Data Source window opens. Provide the following details and click **Next**.
  - Name: rms-service-ds
  - JINDI Name: jdbc/RetailWebServiceDs
  - Database Type: Oracle



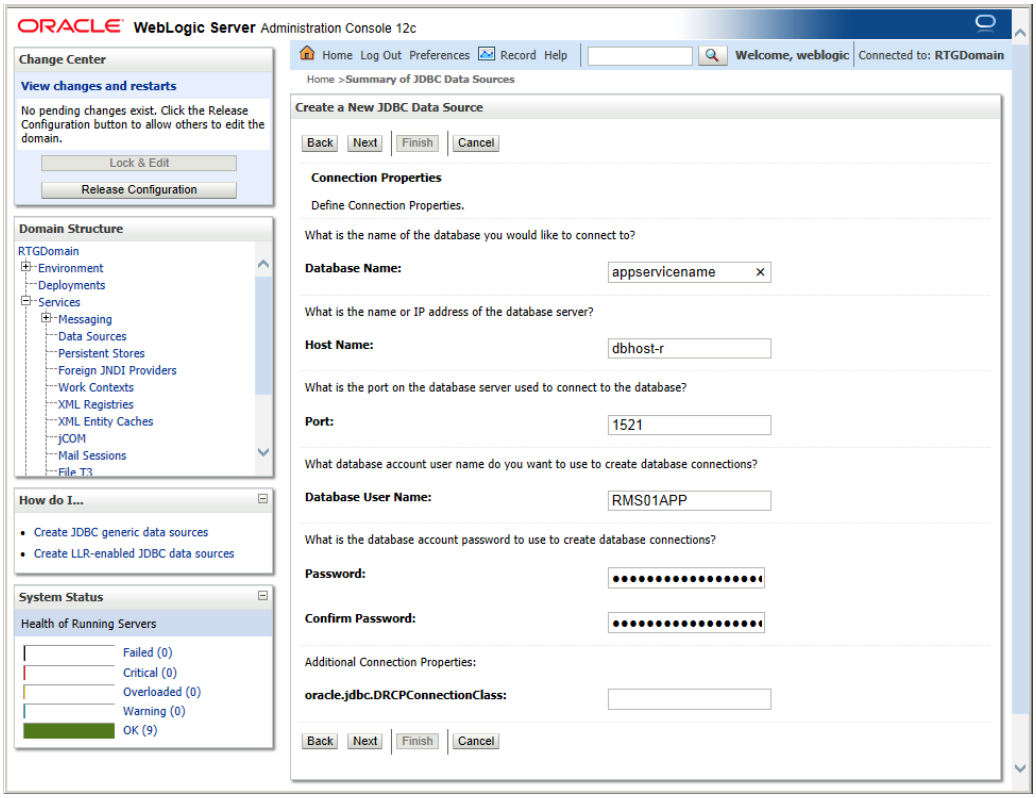
- Select the **Oracle's Driver (Thin) for Service connections, Version Any** option for Database Driver, and click **Next**.



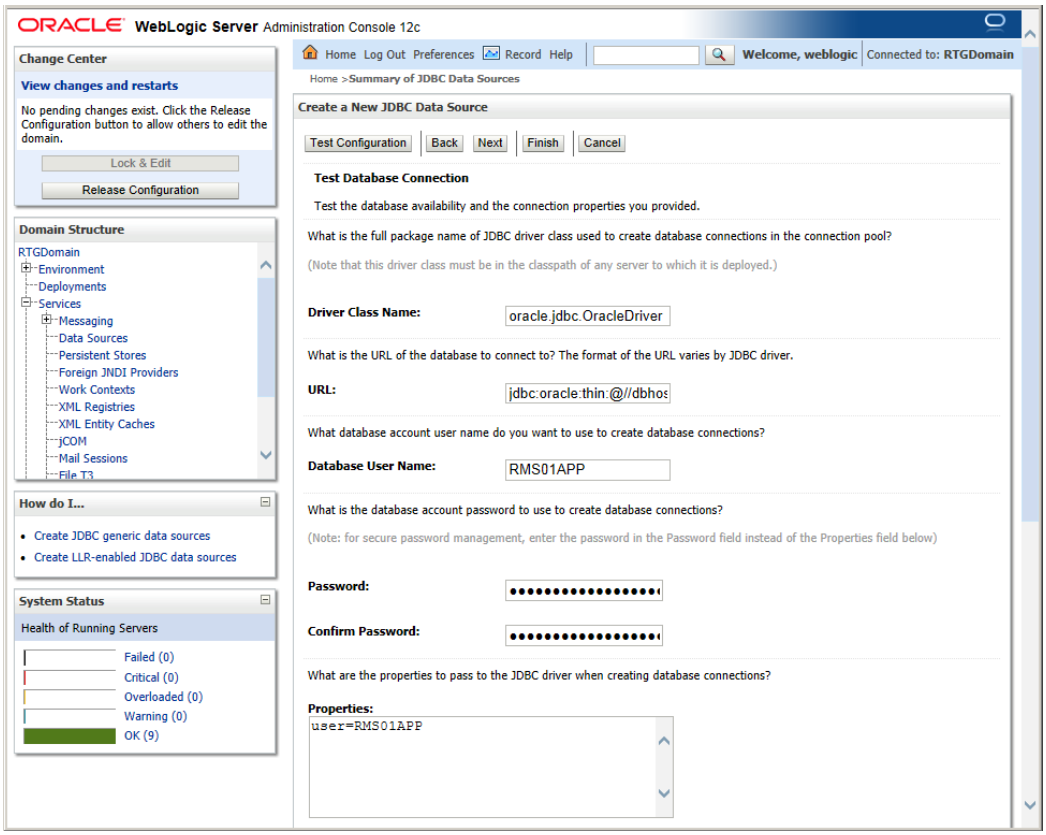
- Select the Supports Global Transactions and the One-Phase commit options. Click **Next**.



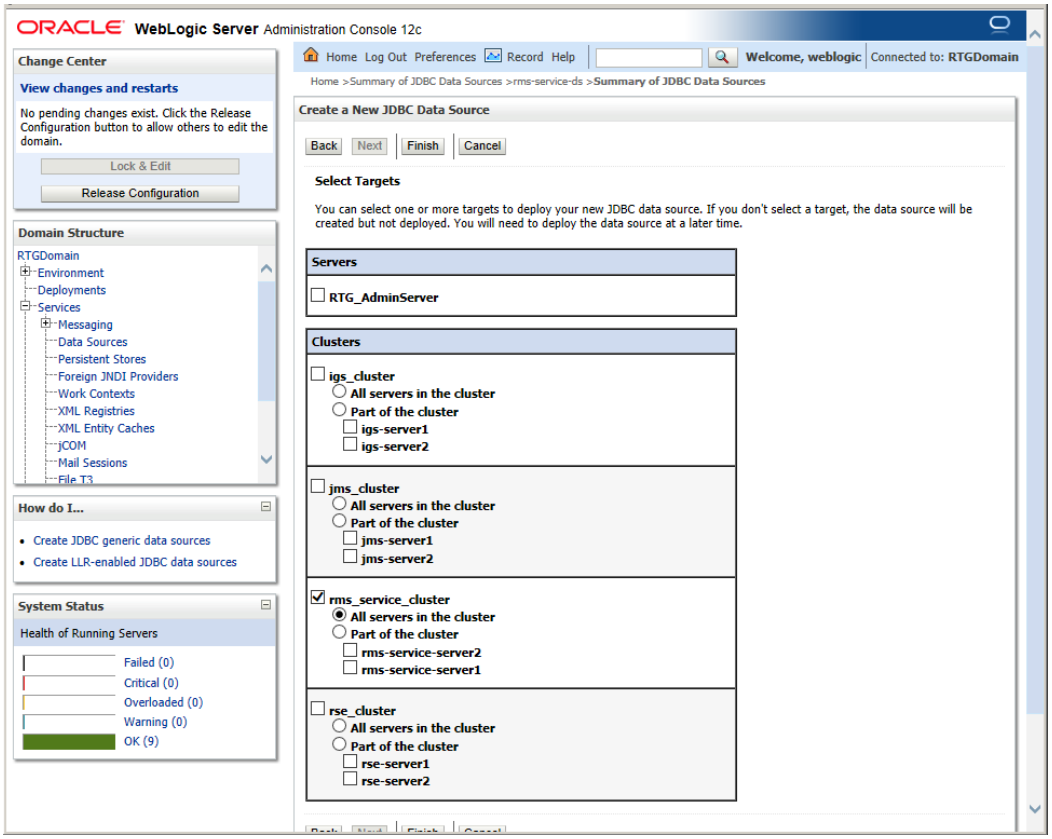
- The Connection Properties window opens. Provide inputs for the RMS application database schema and click **Next**. Use the screen shot below as an example on how to fill in the properties.



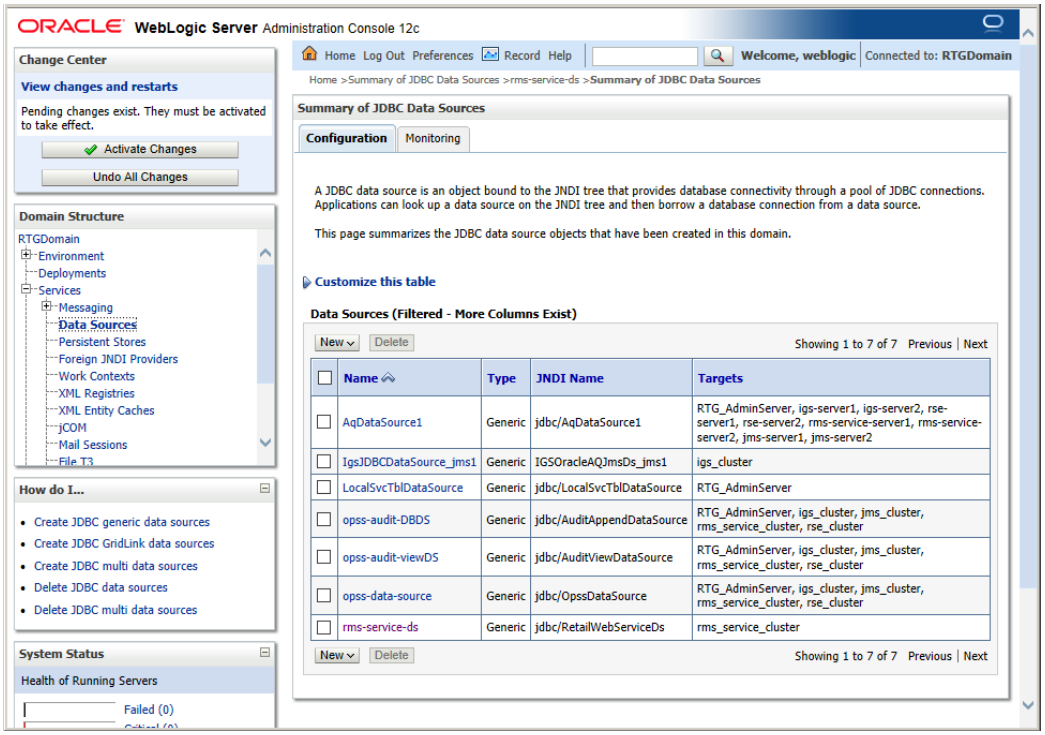
- The Test Database Connection window opens. Click the **Test Configuration** button to test and make sure it was entered in correctly. If yes click **Next**.



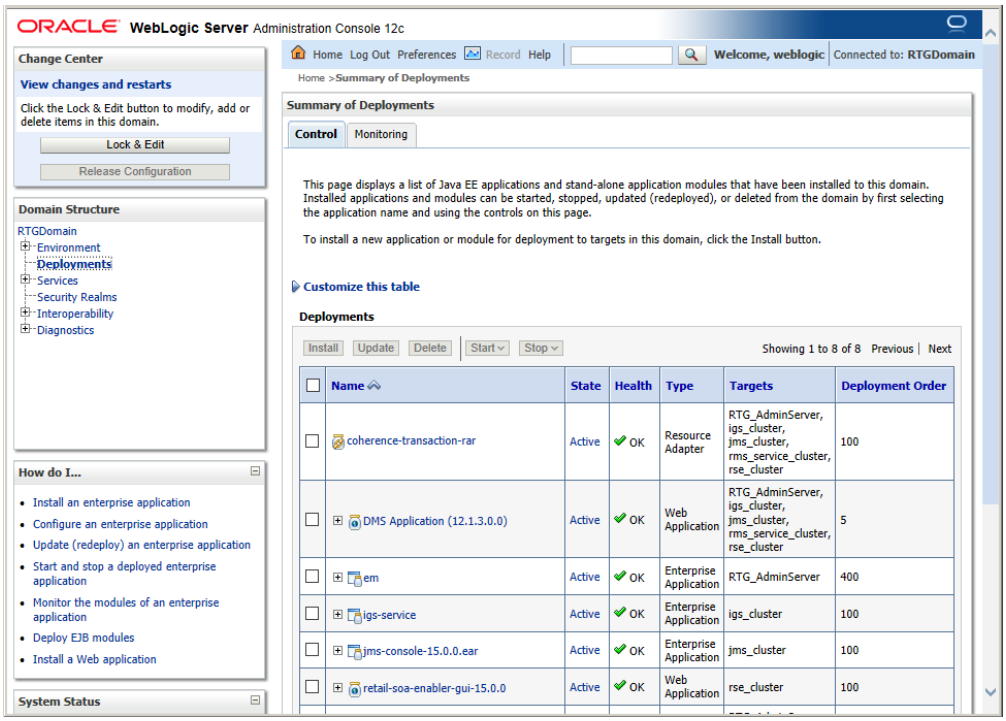
- The Select Targets window opens. Select **rms-service-cluster** and click **Finish**.



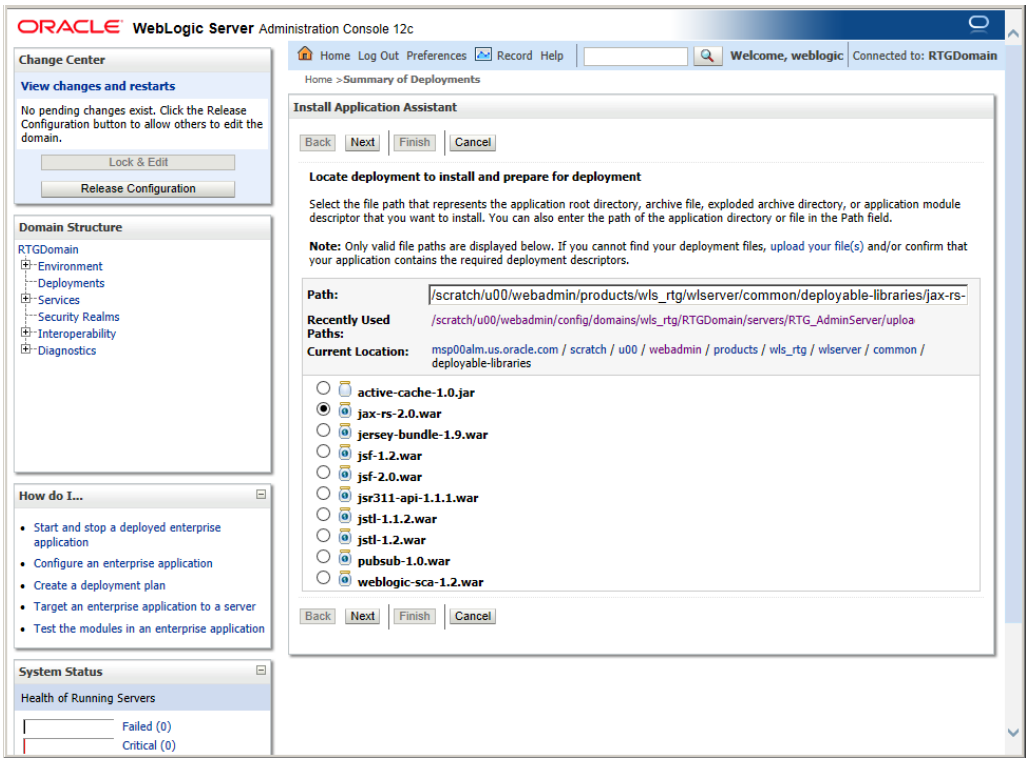
- The Summary of JDBC Data Sources window opens. Click **Activate Changes,** and then click the **Deployments** link in the navigation pane on the left side of the screen.



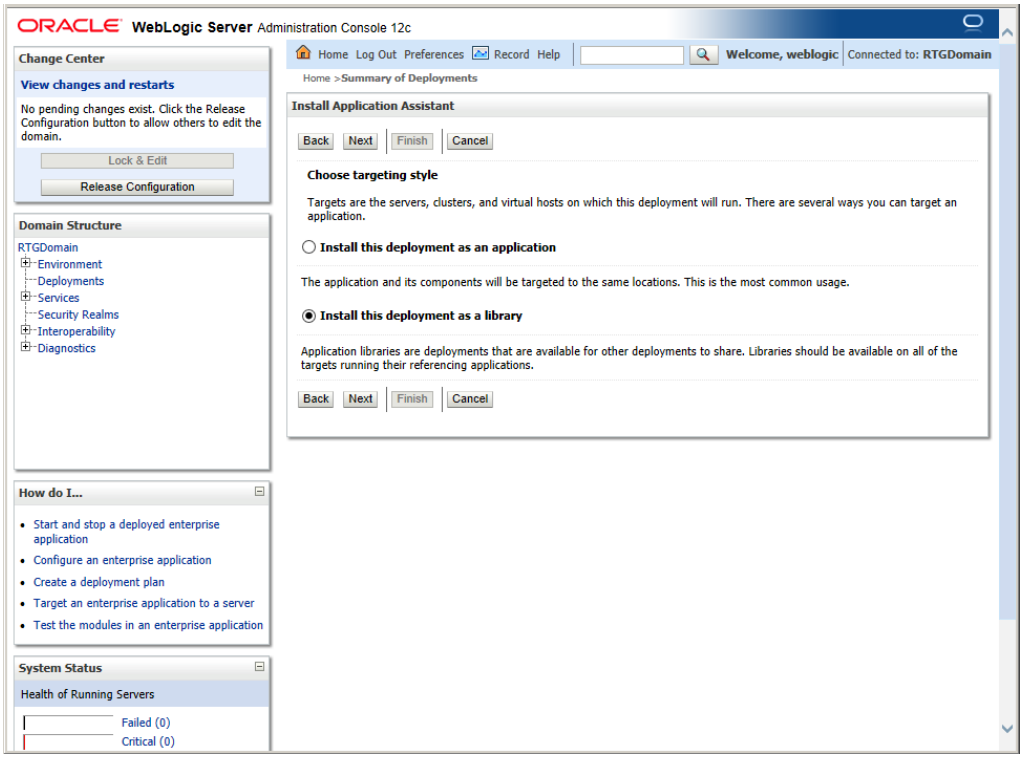
- The Summary of Deployments window opens. Click **Lock & Edit,** and then click **Install**.



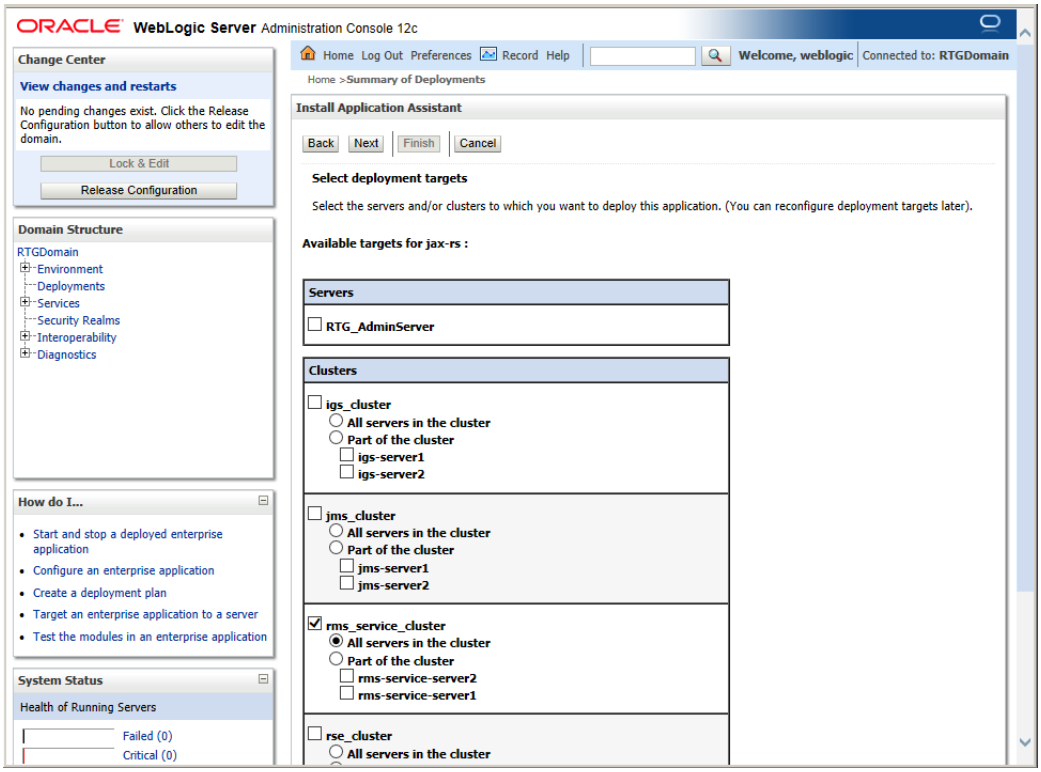
- The Locate deployment to install and prepare for deployment window opens. Set the path to <RTG_WEBLOGIC_HOME>/ wlserver/common/deployable-libraries of rtghost1, select the jax-rs-2.0.war option and click **Next**.



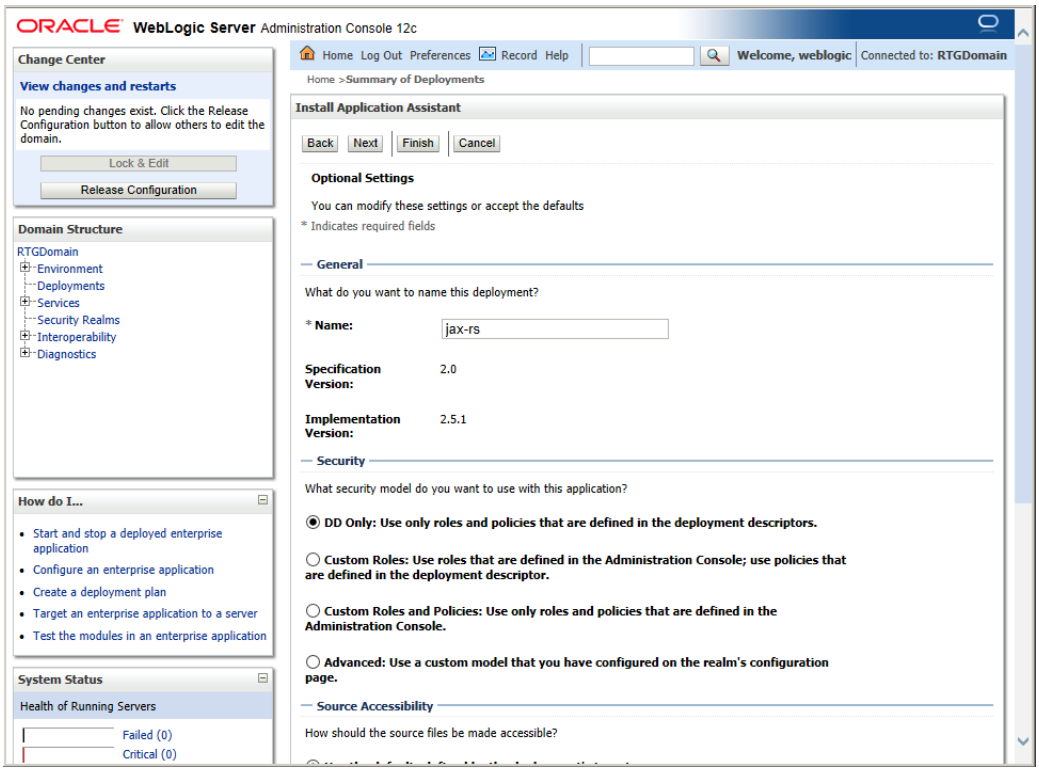
- The Choose targeting style window opens. Select **the Install this deployment as a library** option and click **Next**.



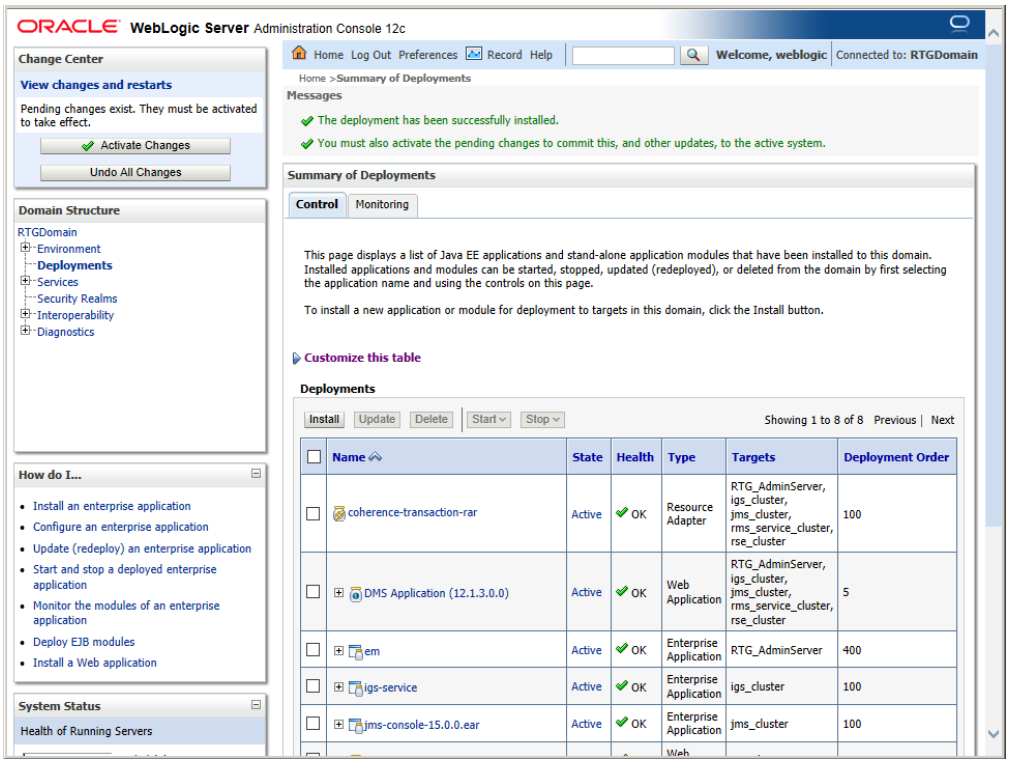
- The Select deployment targets window opens. Select the rms-service-cluster option and click **Next**.



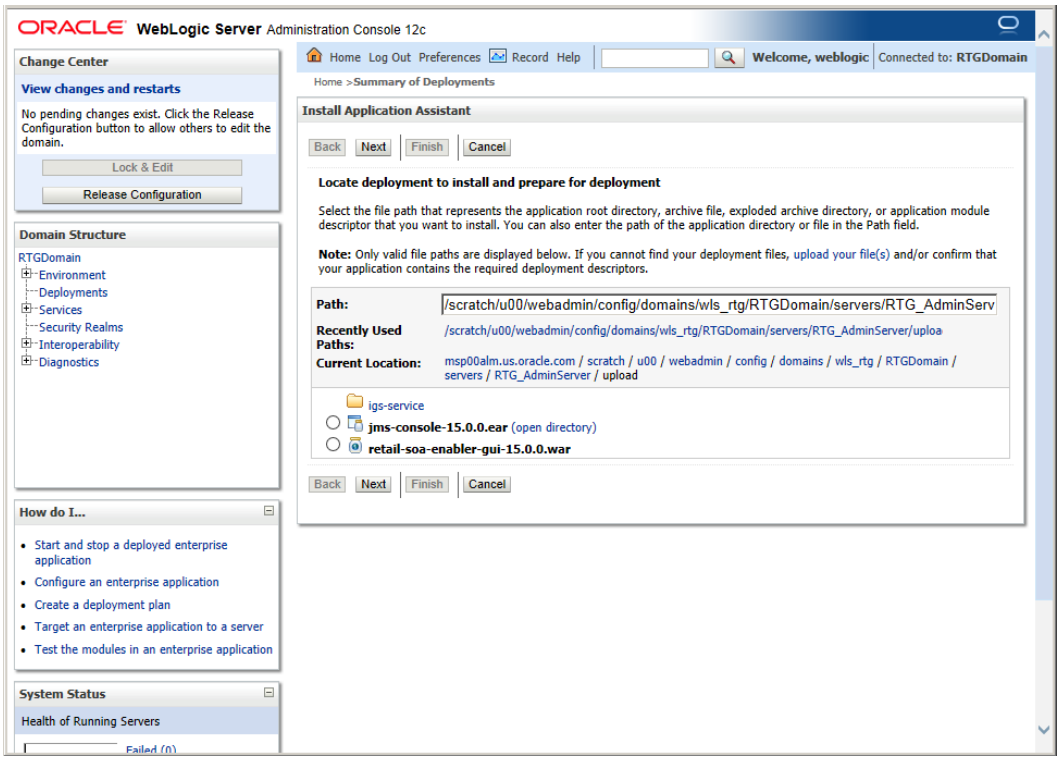
- The Optional settings window opens. Click **Finish**.



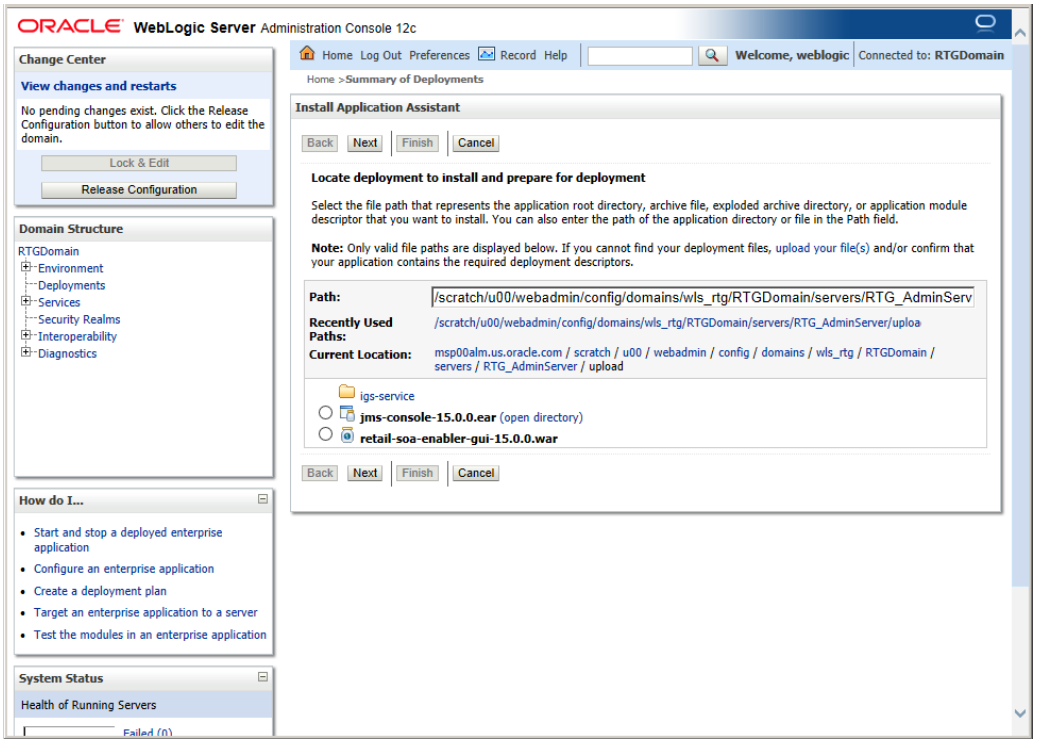
- The Summary of deployments window opens. Click **Activate Changes**.
- When the changes get activated, the Lock & Edit button gets activated. Click **Lock & Edit,** and then click the **Install** button.



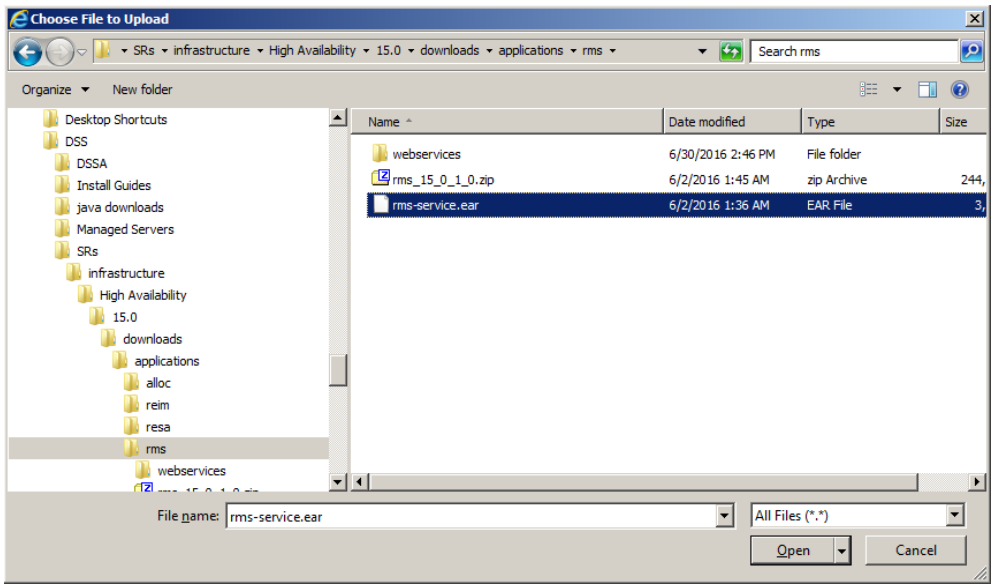
- The Locate deployment to install and prepare for deployment window opens. Click the upload your file(s) link in the center of the screen.



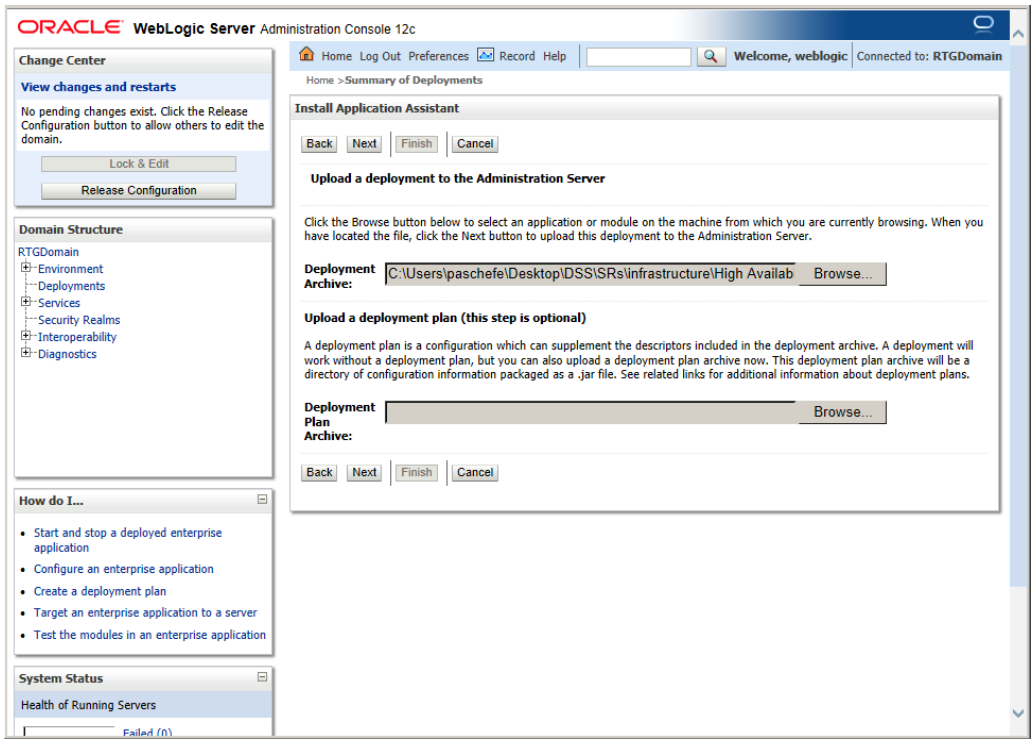
- The Upload a deployment to the Administration Server window opens. Click **Browse**.



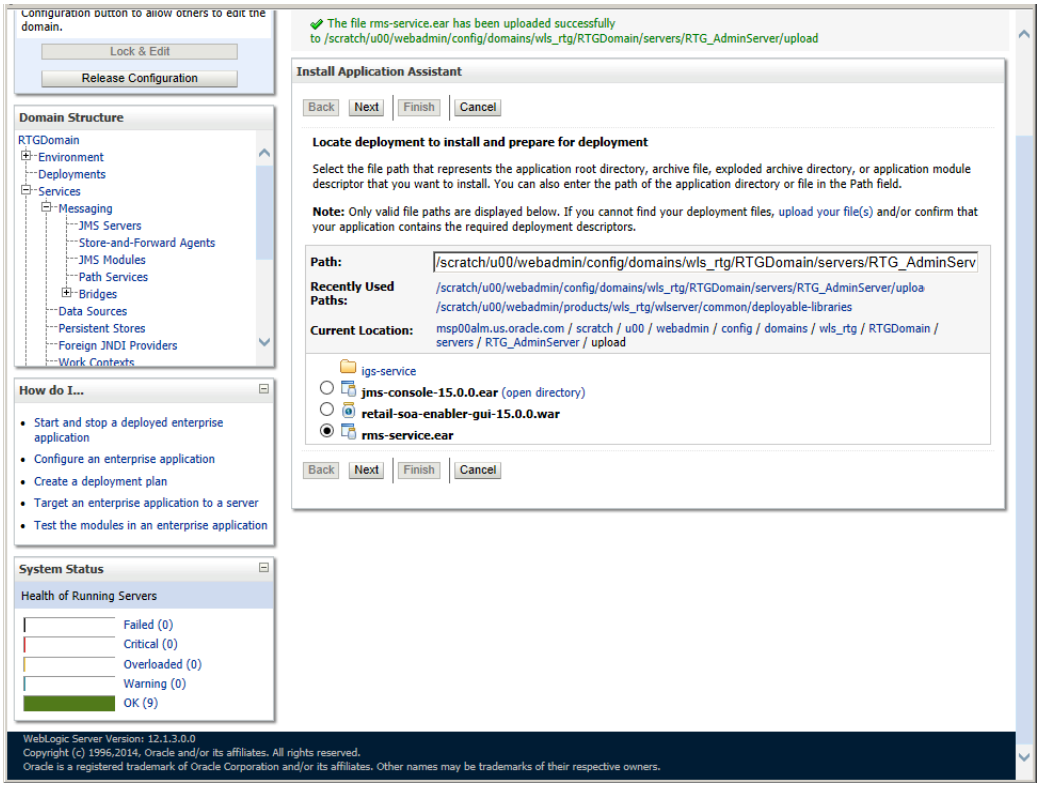
- An explorer window opens. Navigate to where the rms-service.ear was saved at on your computer during the first few steps of this section and open it.



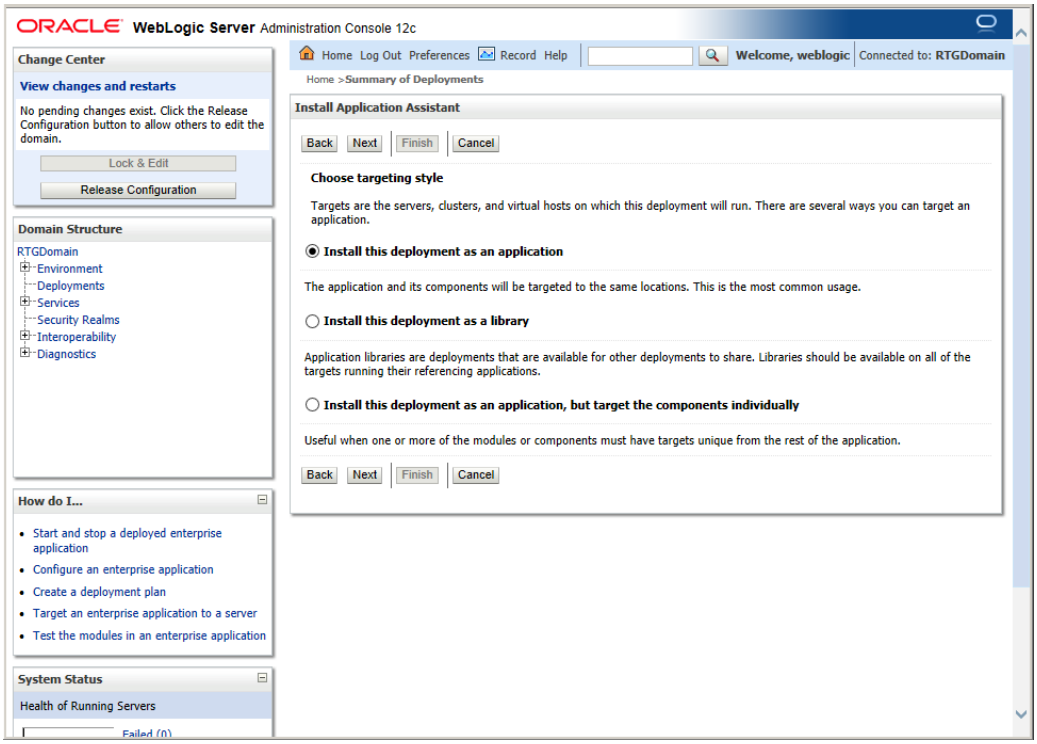
- Click **Next**.



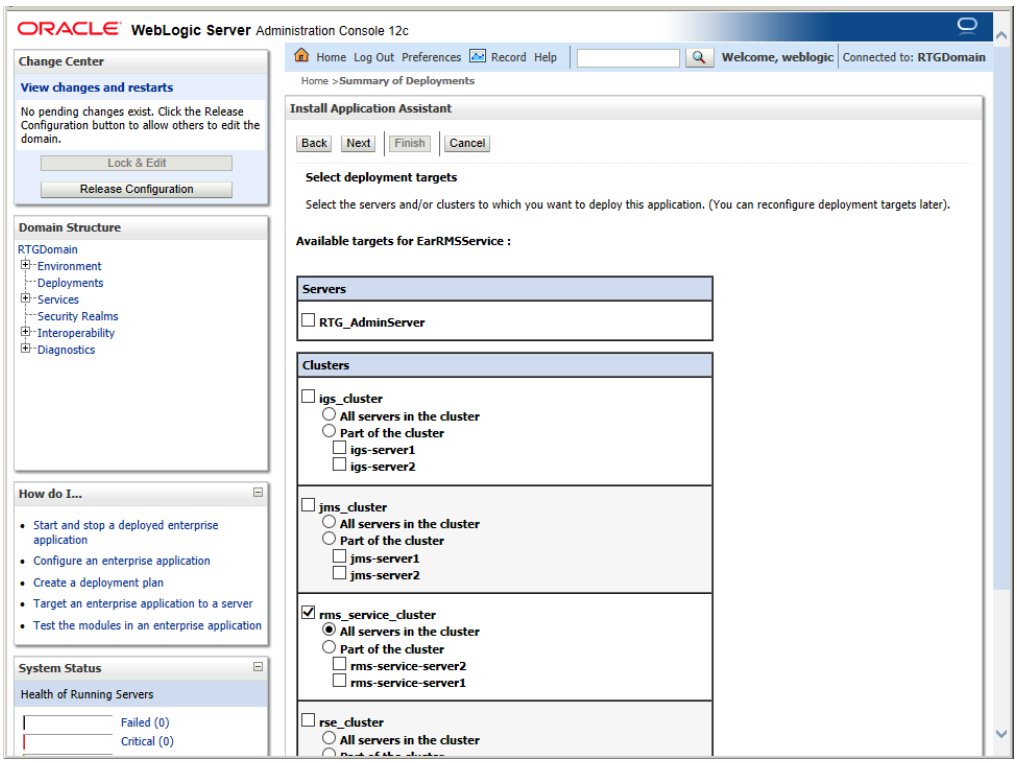
- The Locate deployment to install and prepare for deployment window opens. Click **Next**.



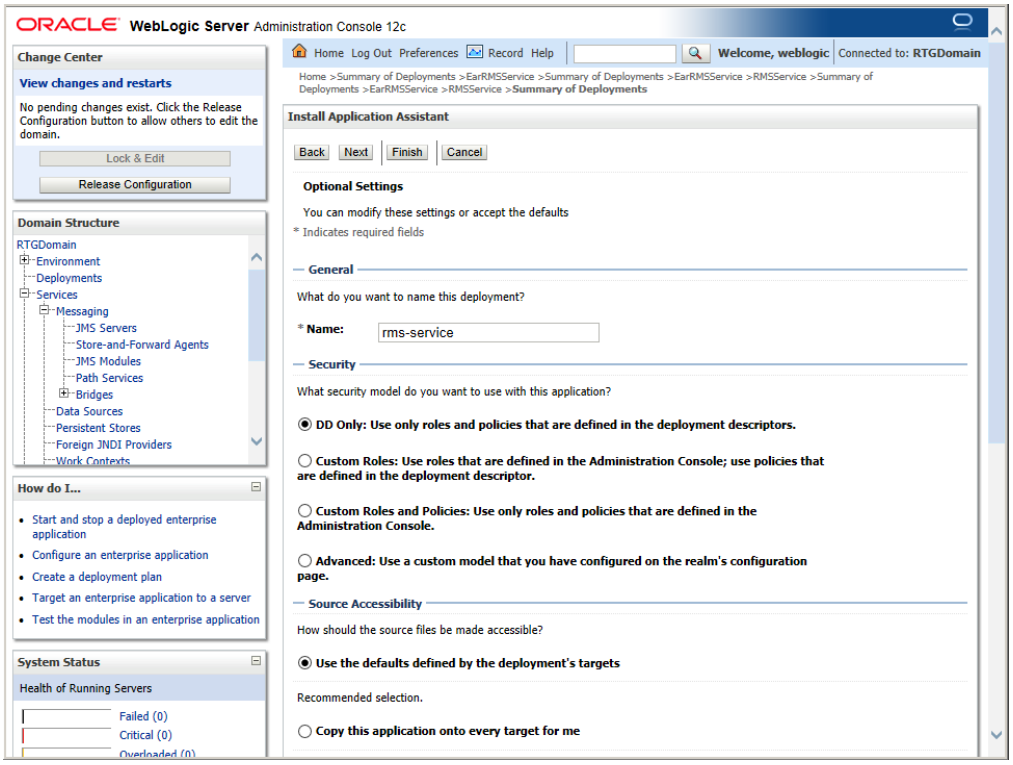
- The Choose targeting style window opens. Select the 'Install this deployment as an application' option and click **Next**.



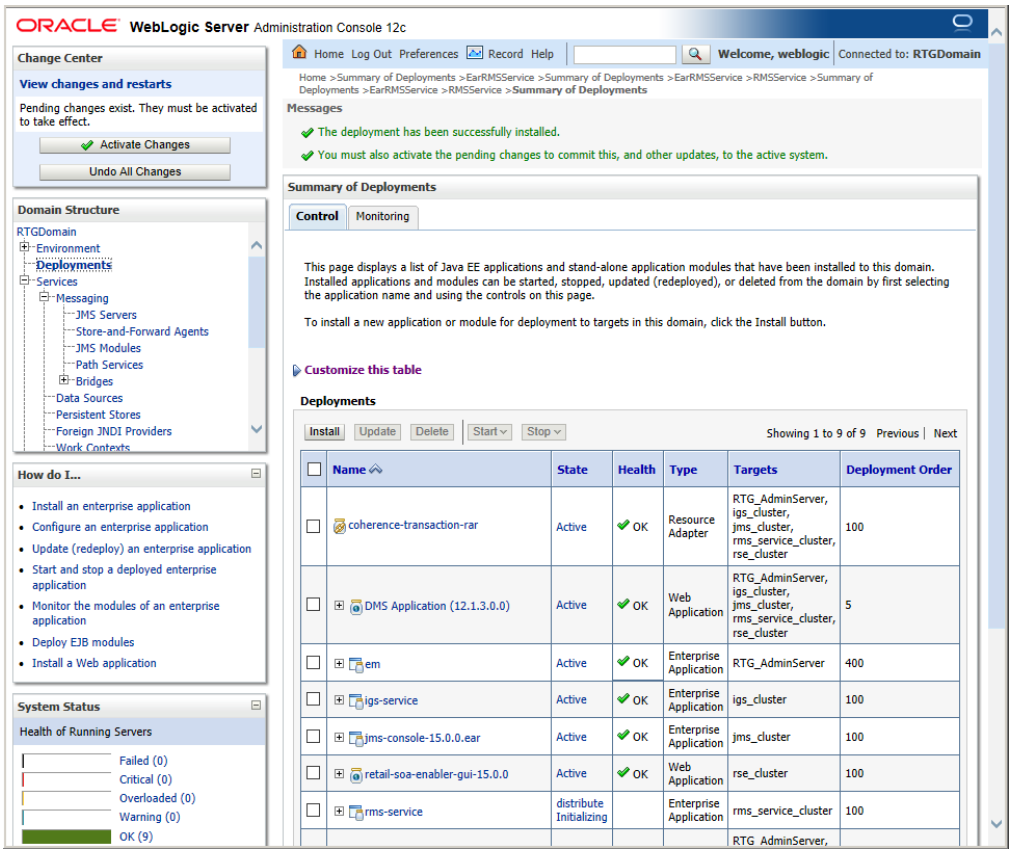
- The Select deployment targets window opens. Select the 'rms-server-cluster' and click **Next**.



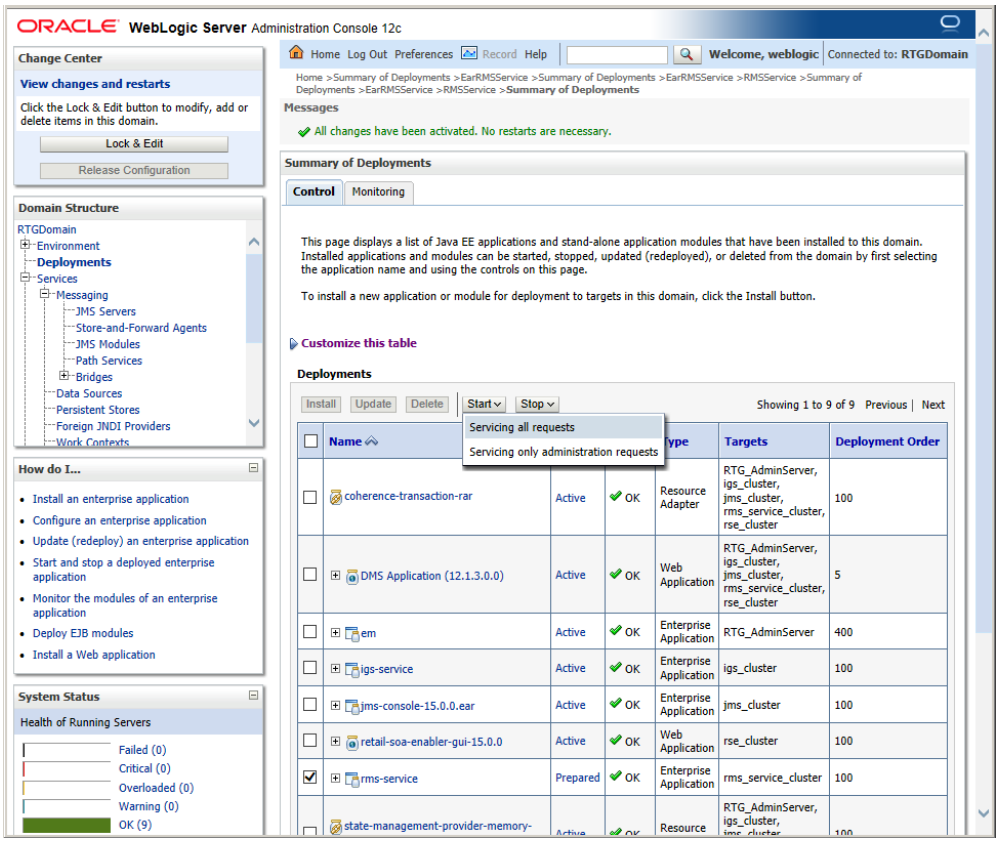
- The Optional Settings window opens. Click **Finish**.



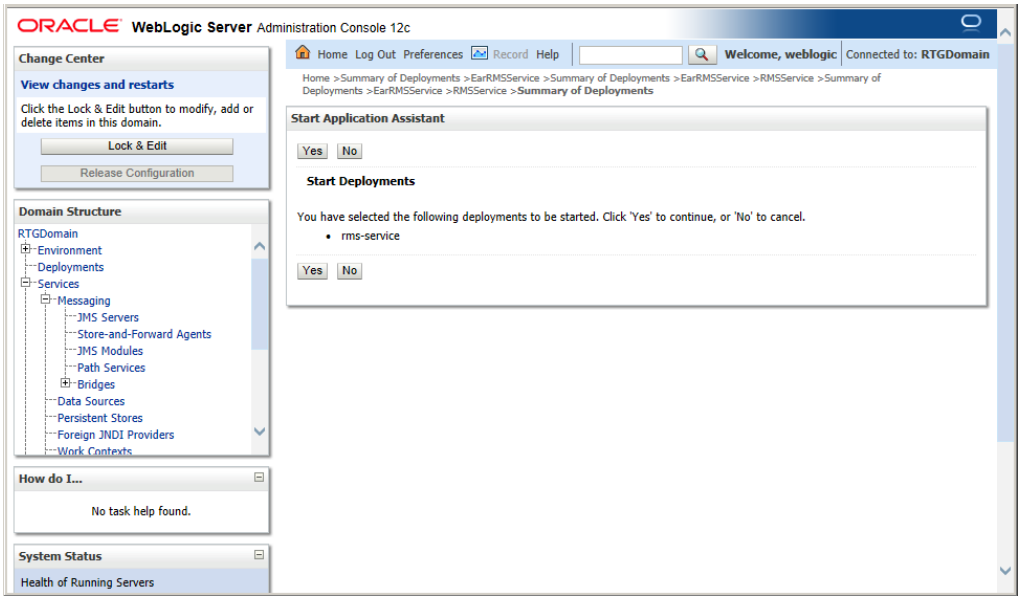
- The summary of Deployments window opens. Click **Activate Changes**.



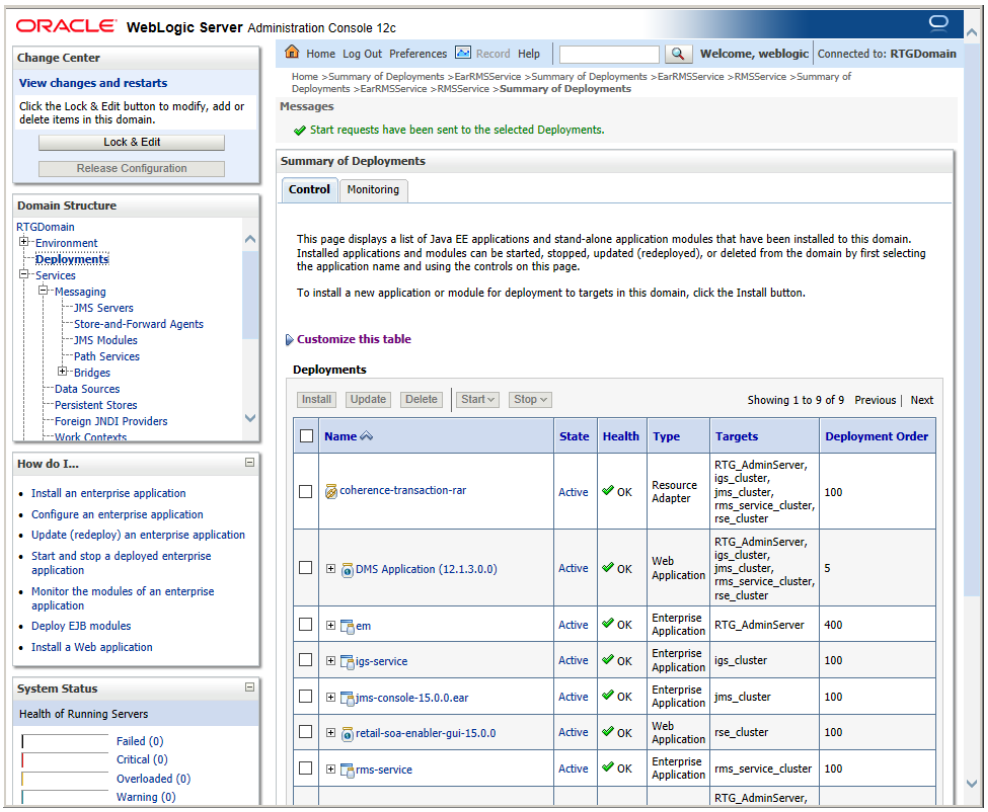
- Once the changes are activated the rms-service may be in Prepared state. If it is in Prepared state, Select it and then click **Start > Servicing** all requests button.



- The Start Application Assistant window opens. Click **Yes**.



- The Summary of Deployments window opens. Confirm that the rms-service deployment is in Active state.



- Installation for RMS web services is complete.

Verification - Verify RMS Web Services

Perform the following steps to validate that RMS web services have been deployed successfully.

- Launch the following URL's to determine if the web service WSDL is accessible from both rtghost1 and rtghost2.
  - http://rtghost1.us.oracle.com:7037/CustomerCreditCheckBean/SupplierBean/SupplierService?WSDL
  - http://rtghost2.us.oracle.com:7037/CustomerCreditCheckBean/SupplierBean/SupplierService?WSDL

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- Published by JAX-WS RI (http://jax-ws.java.net). RI's version is JAX-WS RI 2.2.10-b140319.1121 svn-
revision#7b34660d6bc70e587e8ec81a17e4d76af68816a6. -->
<!-- Generated by JAX-WS RI (http://jax-ws.java.net). RI's version is JAX-WS RI 2.2.10-b140319.1121 svn-
revision#7b34660d6bc70e587e8ec81a17e4d76af68816a6. -->
+ <definitions xmlns="http://schemas.xmlsoap.org/wsdl/" name="SupplierService"
targetNamespace="http://www.oracle.com/retail/rms/integration/services/SupplierService/v1"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:tns="http://www.oracle.com/retail/rms/integration/services/SupplierService/v1"
xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/" xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"
xmlns:wsam="http://www.w3.org/2007/05/addressing/metadata"
xmlns:wsp1_2="http://schemas.xmlsoap.org/ws/2004/09/policy" xmlns:wsp="http://www.w3.org/ns/ws-policy"
xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd">
```

RMS web service test complete.

5

Installing RSB and RIC Applications

The chapter contains instructions needed to install and configure the RSB and RIC applications. Included in the instructions are steps to setup the infrastructure, install the applications and verify the installations.

Install WebLogic Server 12c with ADF on rtghost1 and rtghost2

This section describes the steps required for installing WebLogic 12.1.3 on rtghost1 and rtghost2.

Perform the following steps to install WebLogic 12.1.3 on rtghost1 and rtghost2. Unlike the RIB installation the rtghost's are configured as active/active where all the nodes of the clustered environment will have the applications installed.

- Download software installation for WebLogic 12.1.3 that is bundled with ADF and place it in an installation staging directory on rtghost1 and rtghost2.

  Example:
  <INSTALL_DIR>/WLS-ADF12c_12.1.3.

- Set the following variables by exporting them before running the installer so that the JAVA_HOME is added to the PATH variable:
  - JAVA_HOME
  - PATH
  - DISPLAY

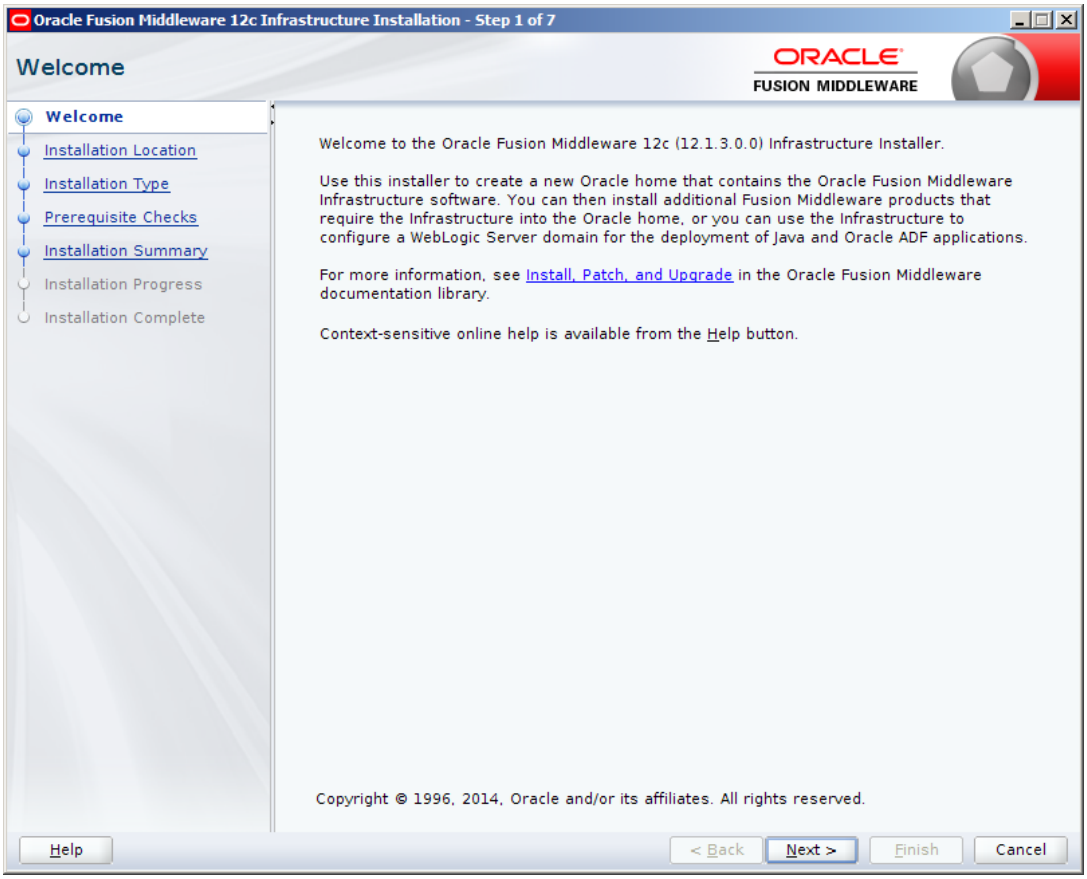
  Example:
  ```
  $ export JAVA_HOME=/u00/webadmin/products/jdk_java
  $ export PATH=$JAVA_HOME/bin:$PATH
  $ export DISPLAY=<pc_ip_address:0.0>
  ```

- Run the following commands to run the installer:

  Example:
  ```
  $ cd <INSTALL_DIR>/ WLS-ADF12c_12.1.3
  $ java -jar ./fmw_12.1.3.0.0_infrastructure.jar
  ```

- The Installation Inventory Setup window displays. Update the Inventory Directory and Operation System Group fields according to the instructions on the screen and click **Next**. See screen shot for examples.

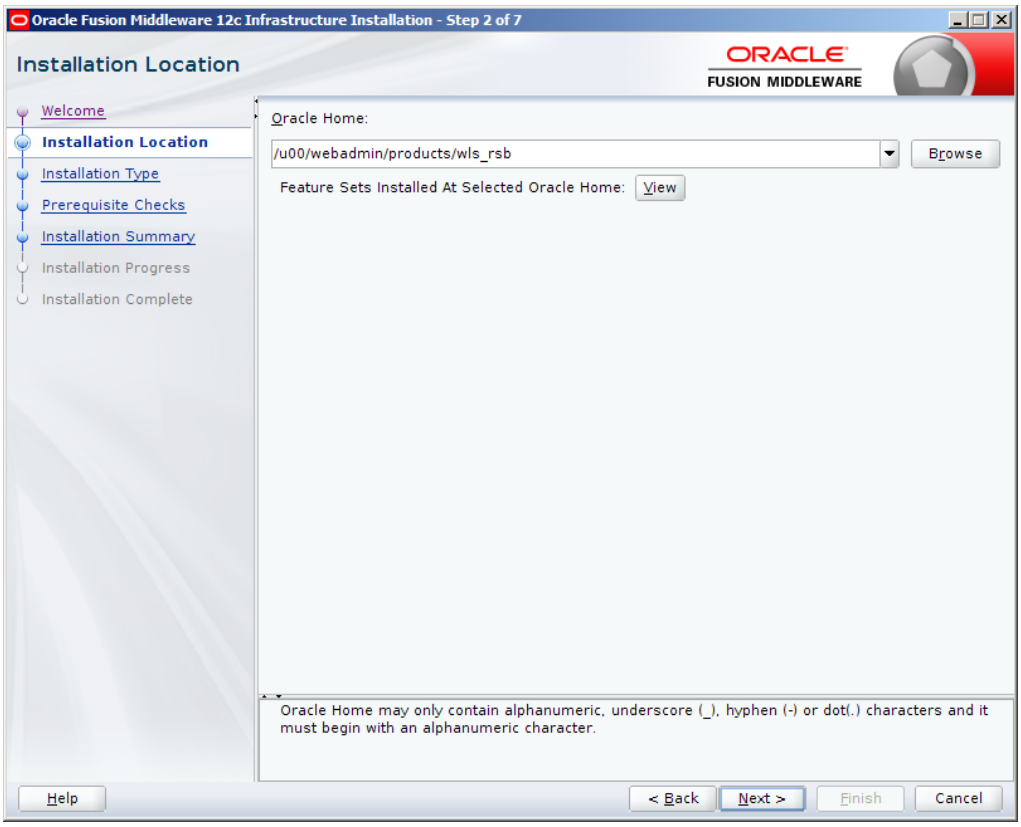
- The Welcome window of the WebLogic 12c installer displays. Click **Next**.



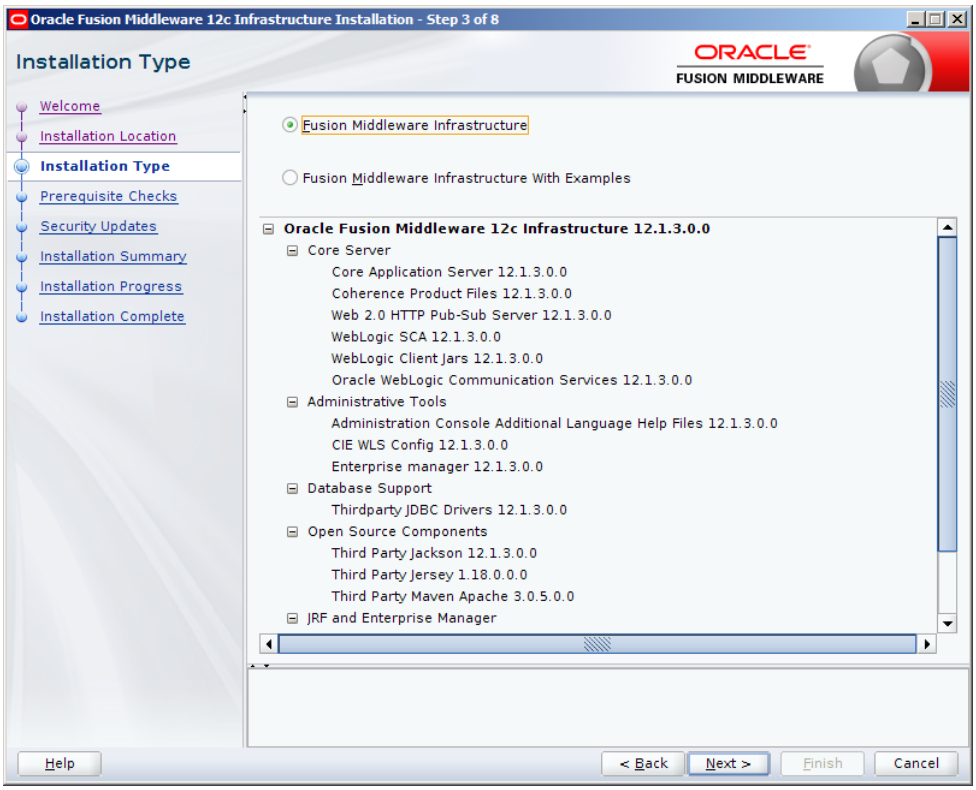
- The Installation Location window displays. Enter the path to where the ORACLE_HOME for RSB should be and click **Next**.

  Example:

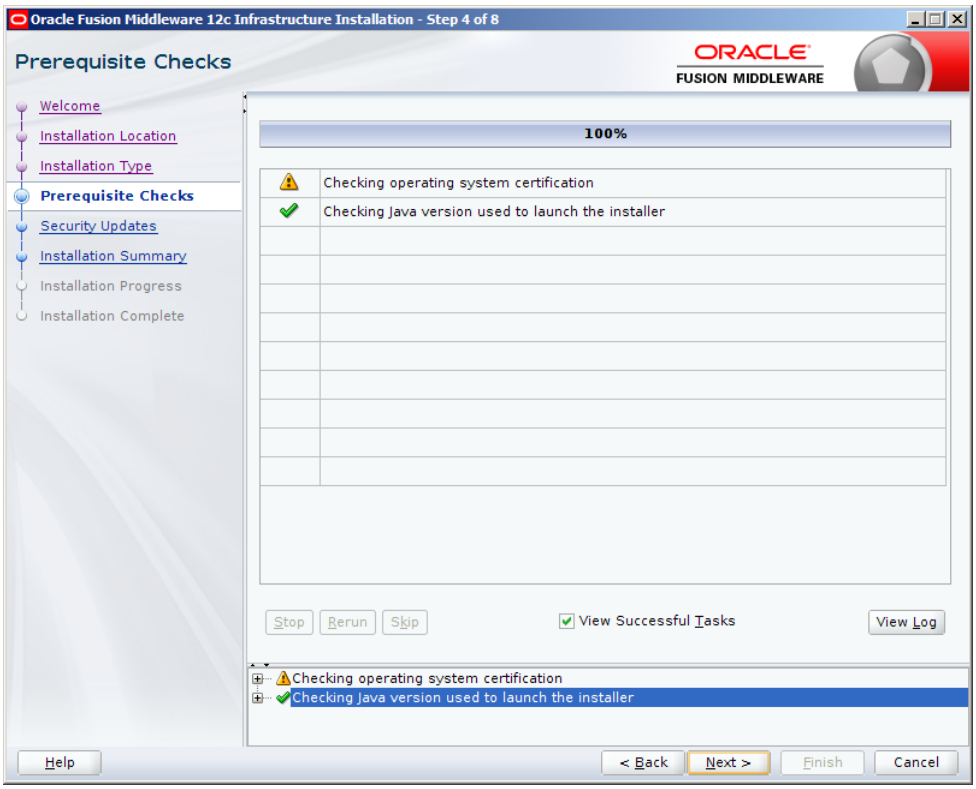
  /u00/webadmin/products/wls_rsb



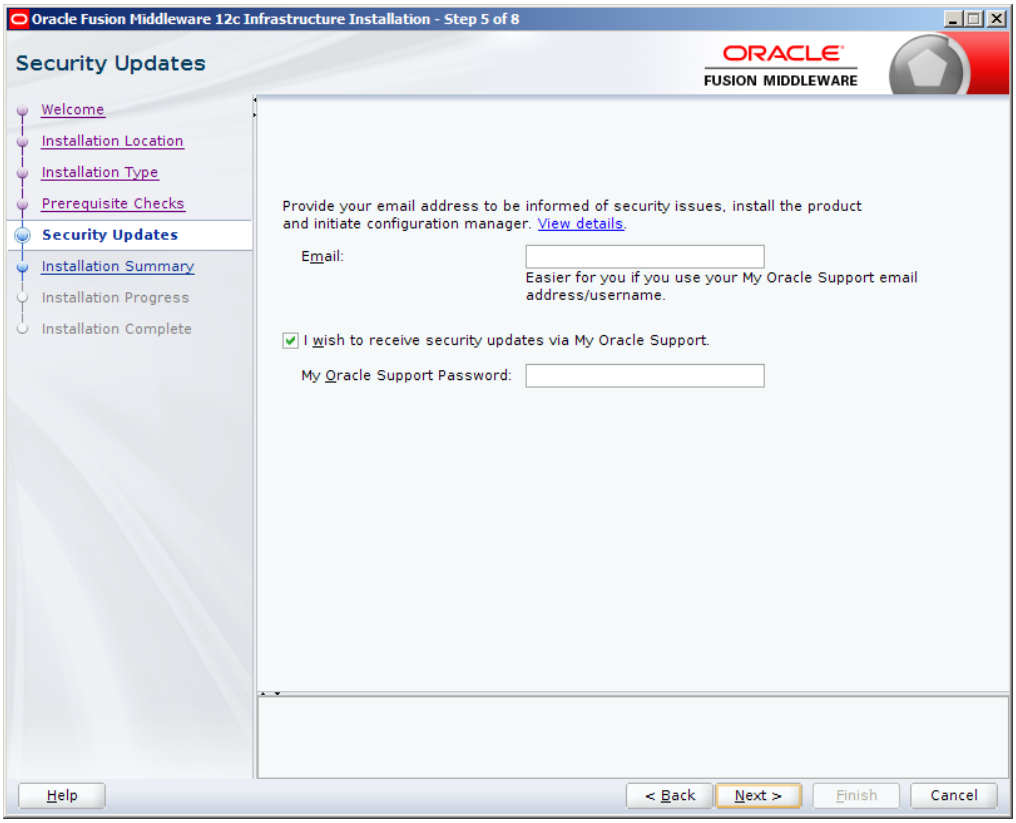
- The Installation Type window opens. Select Fusion Middleware Infrastructure and click **Next**.



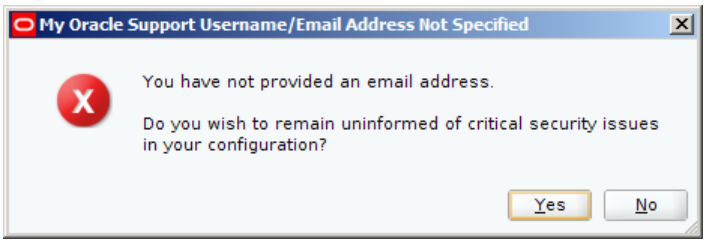
- The Prerequisite Checks window opens. When the checks are completed – click **Next**.



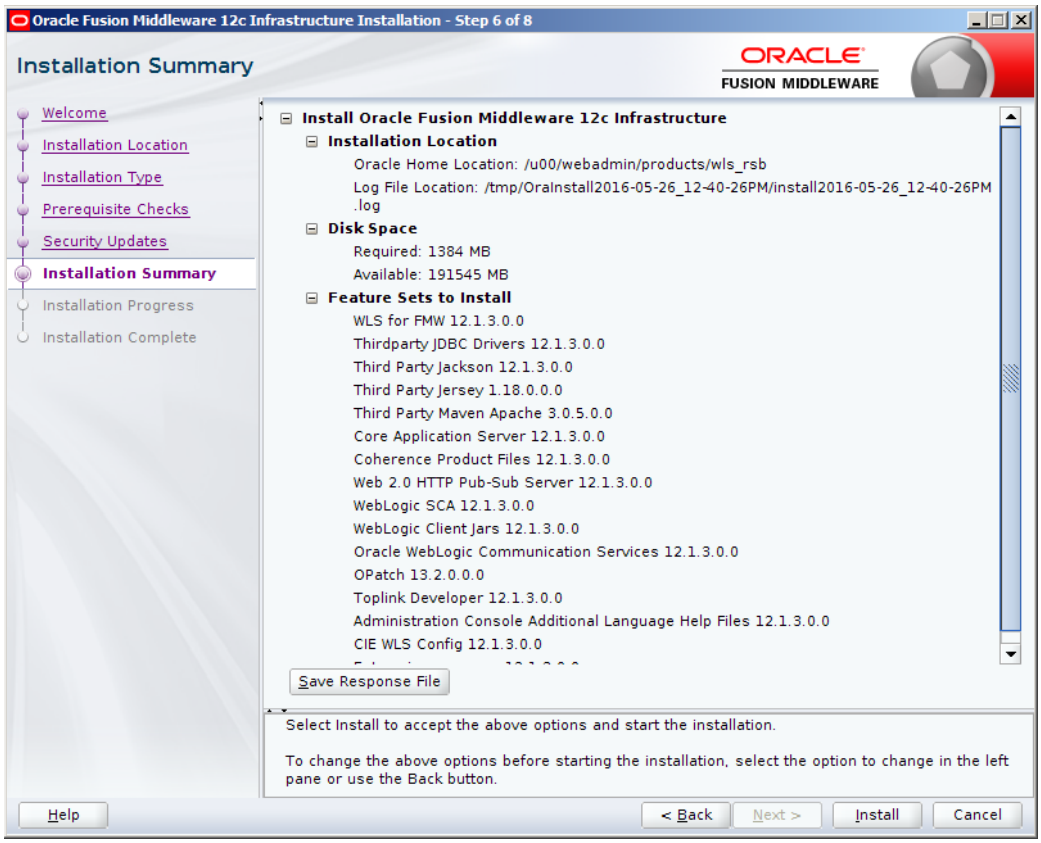
- The Security Updates window opens. Leave the screen blank and click **Next**.



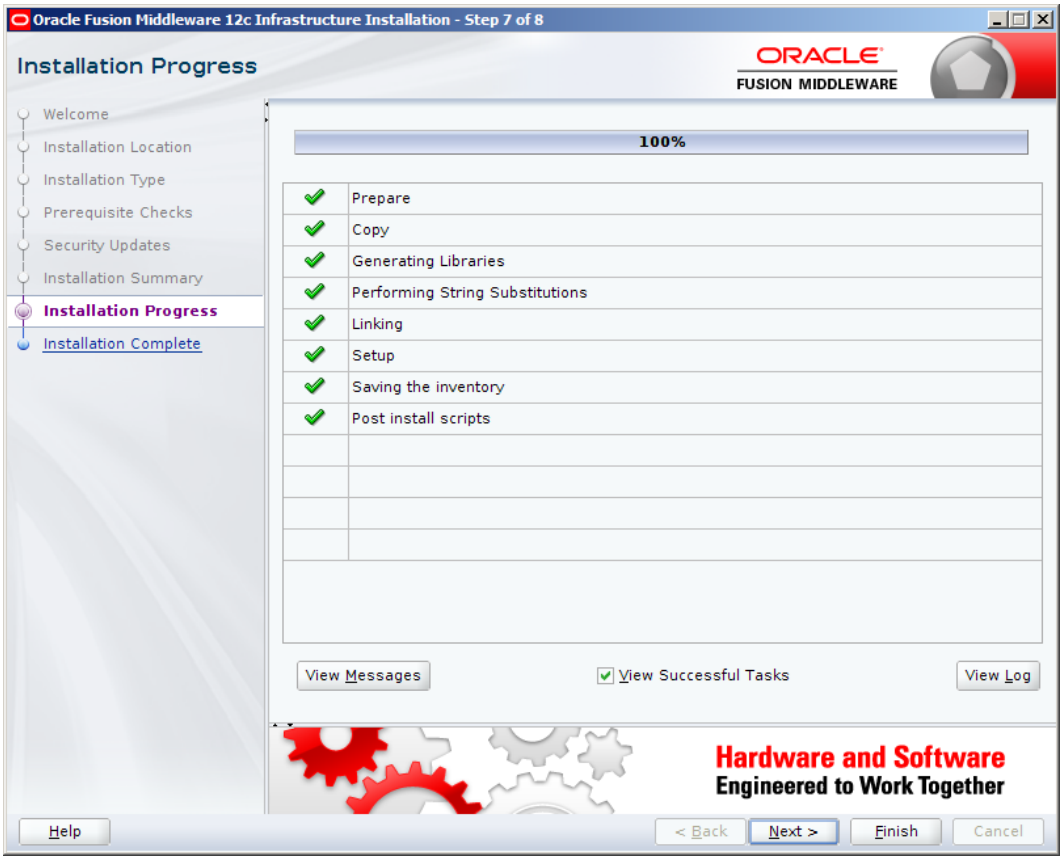
- A confirmation prompt is displayed. Click Yes to continue.



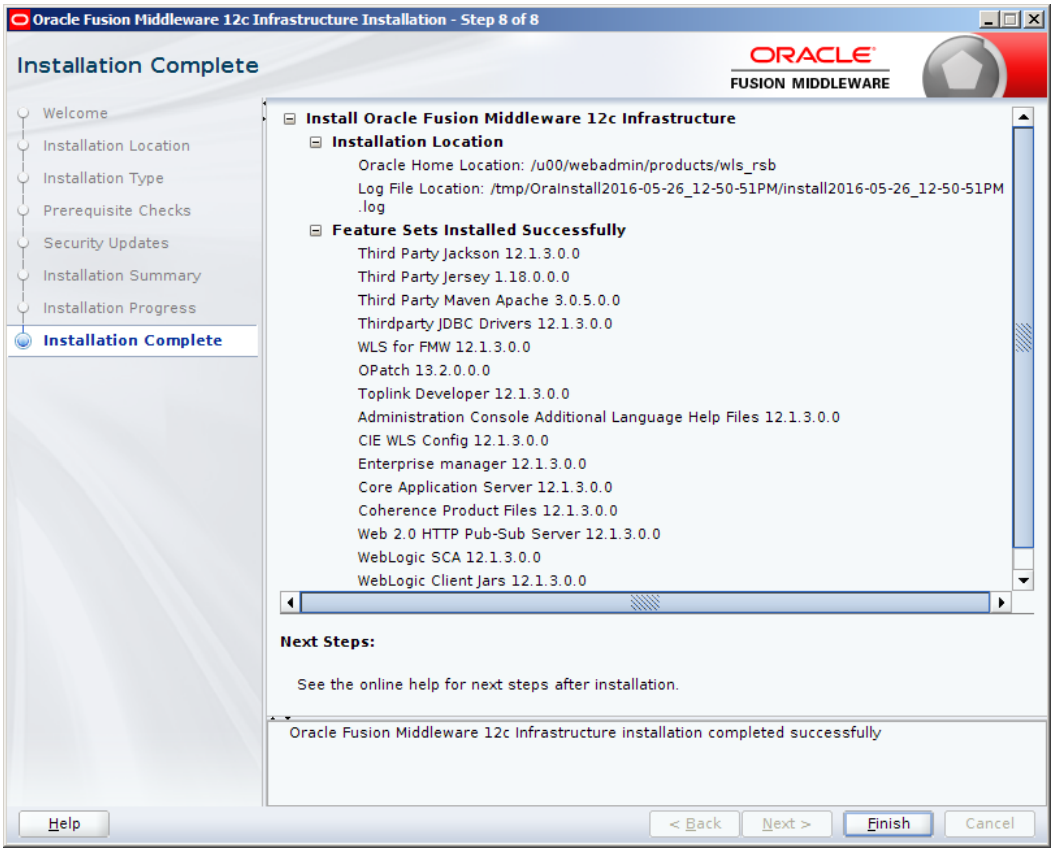
- The Installation Summary window opens. Confirm that Installation Location is correct and click **Next**.



- The Installation Progress window opens. Click **Next** once it has reached 100%.



- The Installation Complete window opens. Click **Finish**.



- Installation of WebLogic 12c is complete.

Install OSB 12.1.3 on rtghost1 and rtghost2

Perform the following steps to install Oracle Service Bus (OSB) 12.1.3 on rtghost1 and rtghost2.

- Download software installation for OSB 12.1.3 and place it in an installation staging directory on rtghost1 and rtghost2.

  Example:
  <INSTALL_DIR>/OSB_12.1.3.

- Set the following variables by exporting them before running the installer so that the JAVA_HOME is added to the PATH variable:
  - JAVA_HOME
  - PATH
  - DISPLAY

  Example:

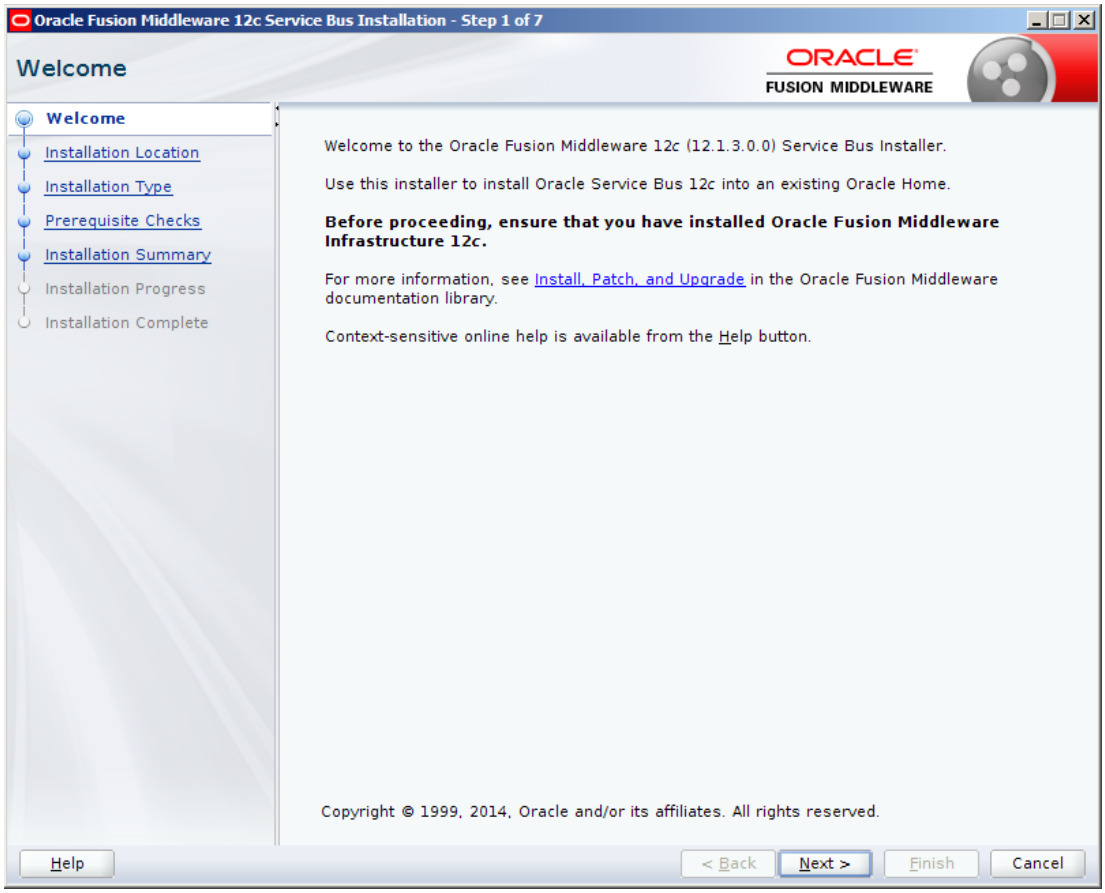
  ```
  $ export JAVA_HOME=/u00/webadmin/products/jdk_java
  $ export PATH=$JAVA_HOME/bin:$PATH
  $ export DISPLAY=<pc_ip_address:0.0>
  ```

- Run the following commands to run the installer:

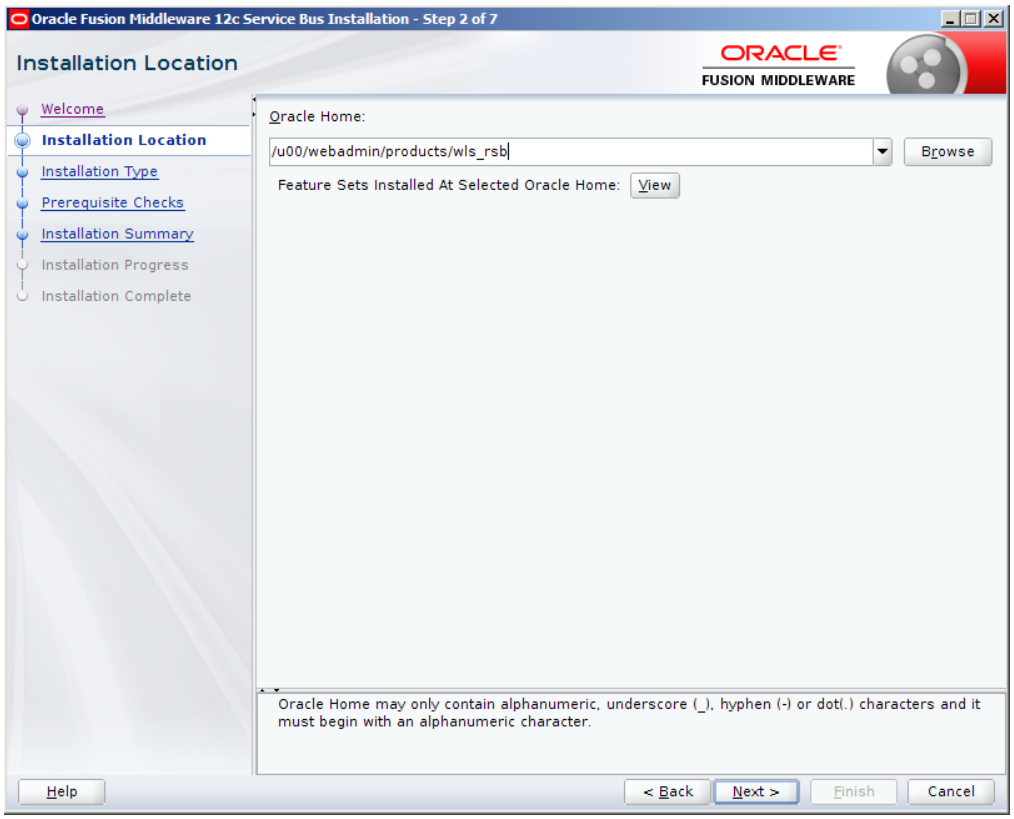
Example:

```
$ cd <INSTALL_DIR>/OSB_12.1.3
$ java -jar fmw_12.1.3.0.0_osb.jar
```

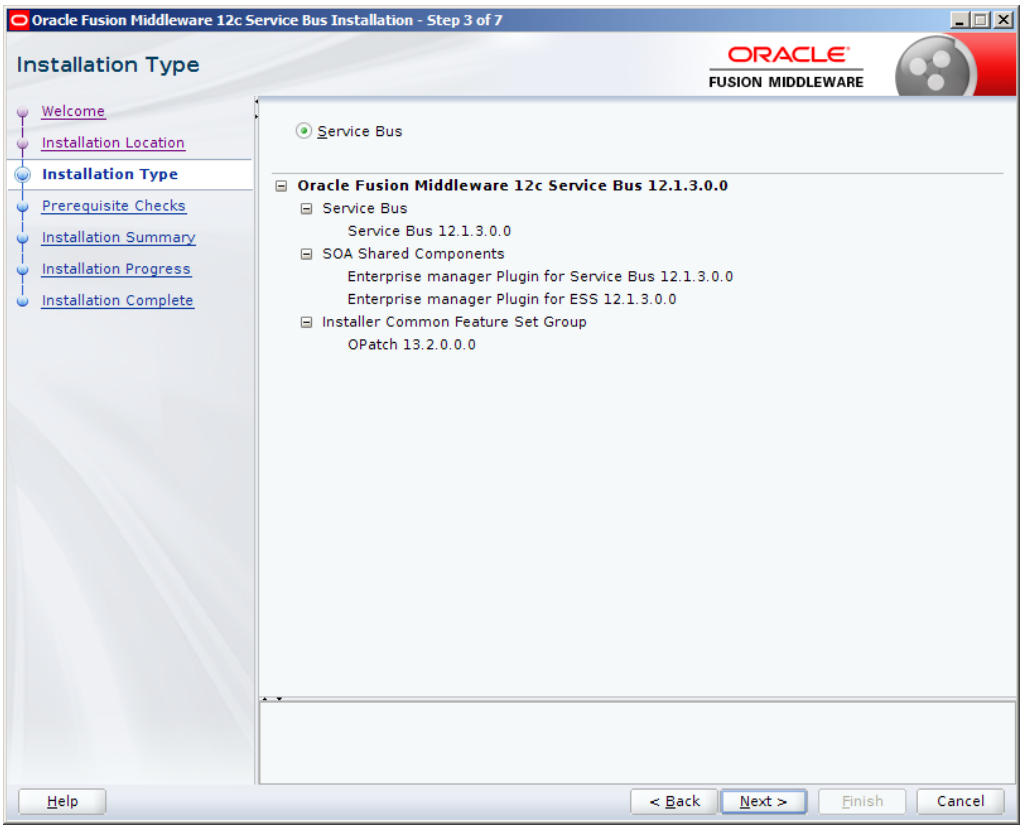
- The Welcome page is displayed. Click **Next**.



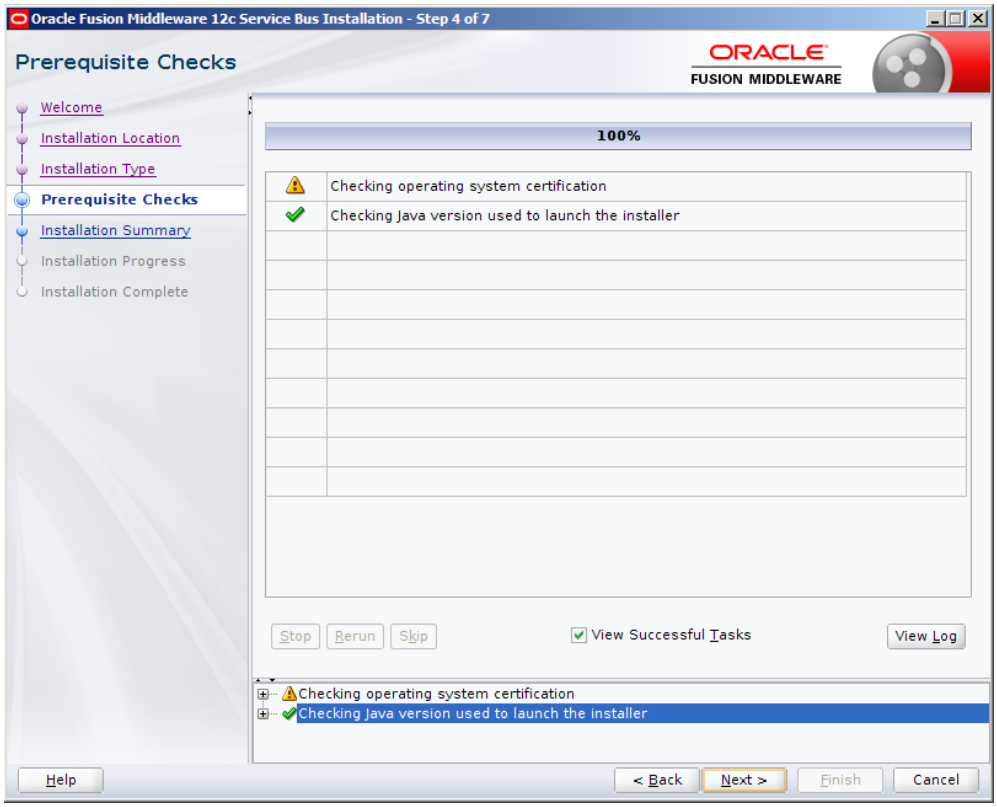
- The Installation Location window opens. Provide the location of where the WebLogic home directory for RSB is located and click **Next**.



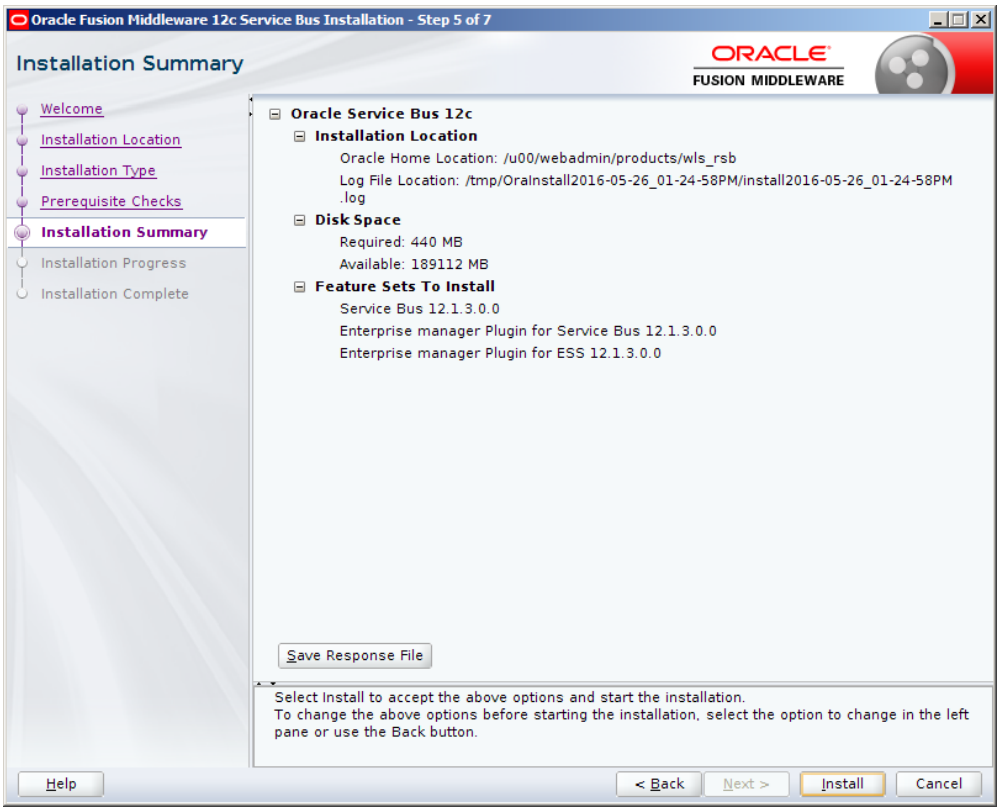
- The Installation Type window opens. Click **Next**.



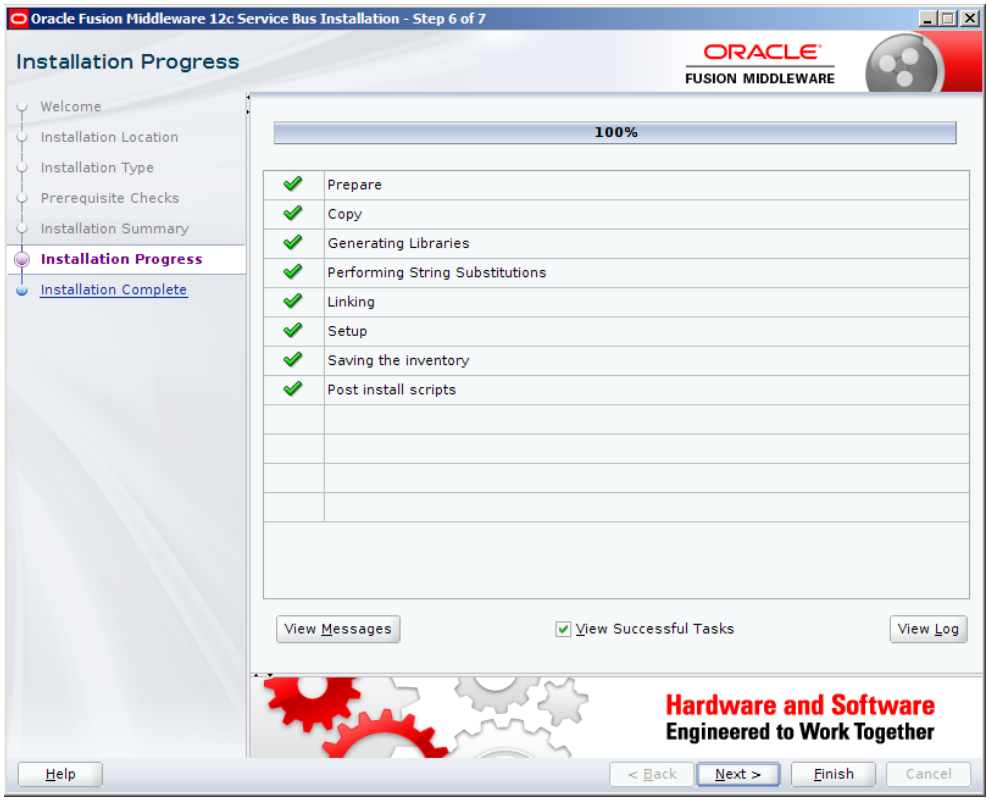
- The Prerequisite Checks window opens. Click **Next**.



- The Installation Summary window opens. Click Install.



- The Installation Progress window opens. When installation progress is 100%, click Finish.



- OSB installation is complete.

Creating Database Repository for RSBDomain

This chapter describes the steps to create the proper schemas required for the managed servers of the RSB and RIC installations. In order to run the Repository Creation Utility (RCU) - It is required to use the database credentials for the SYS user. Perform the following steps to create a database repository using Repository Creation Utility:

- Set the DISPLAY variable before running the installer:

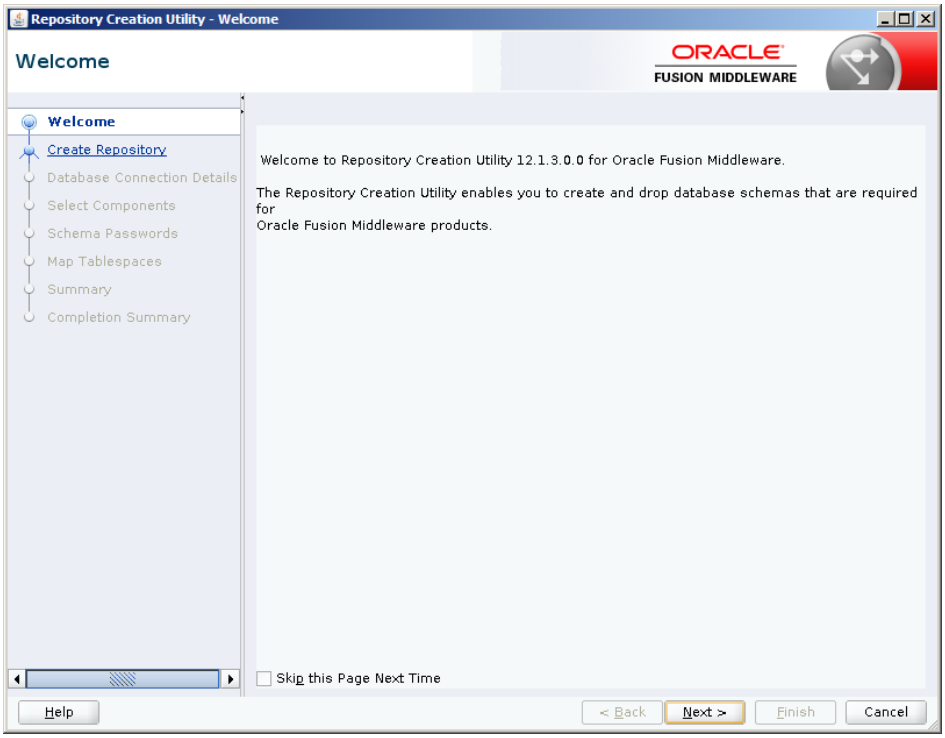
  Example:

  ```
  $ export DISPLAY=<pc_ip_address:0.0>
  ```

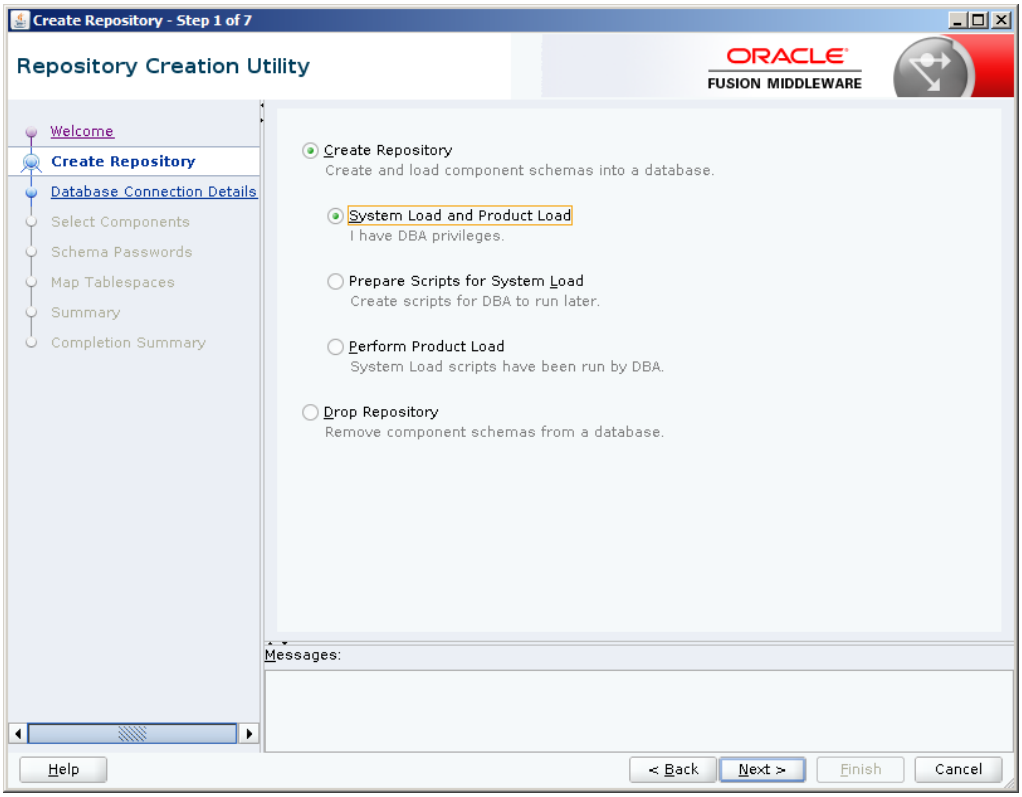
- RCU is packaged with 12c WebLogic installations so it is easy to make sure you are running the correct version of RCU. Navigate to the location of the rcu executable and run the installer by executing the following commands:

  ```
  $ cd /u00/webadmin/products/wls_rsb/oracle_common/bin
  $ ./rcu
  ```

- The Welcome Window displays. Click **Next**.

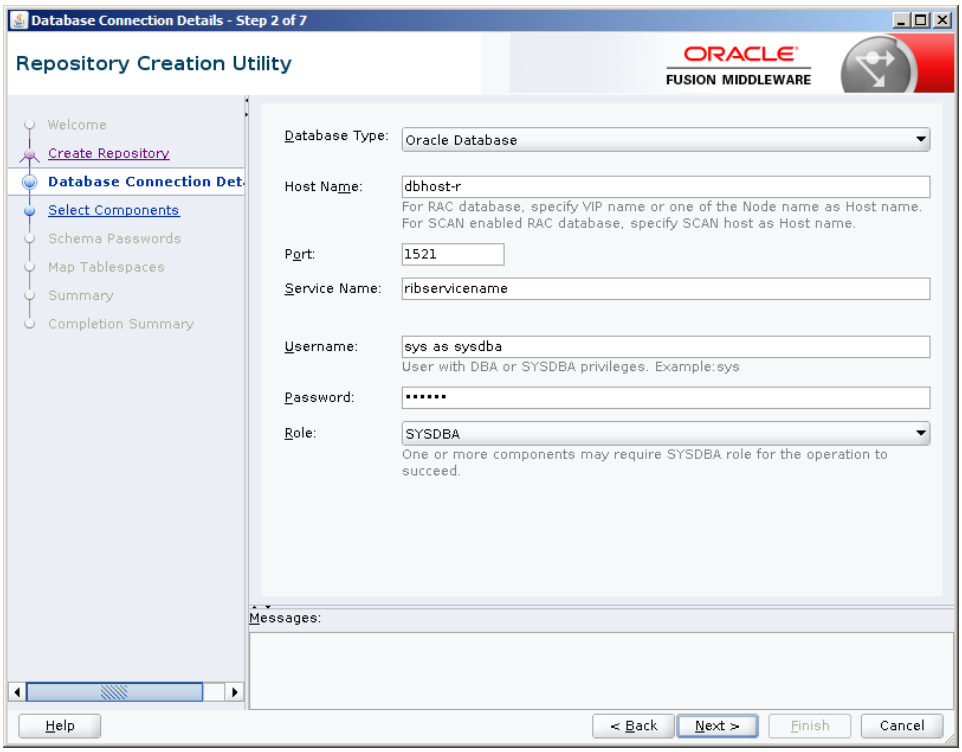


- The Create Repository window displays. Select **Create** and click **Next**.

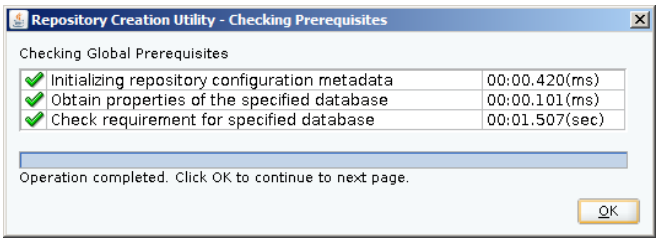


- The Database Connection Details window opens. Provide the database details and click **Next**:
  - Host Name: dbhost-r
  - Port: 1521

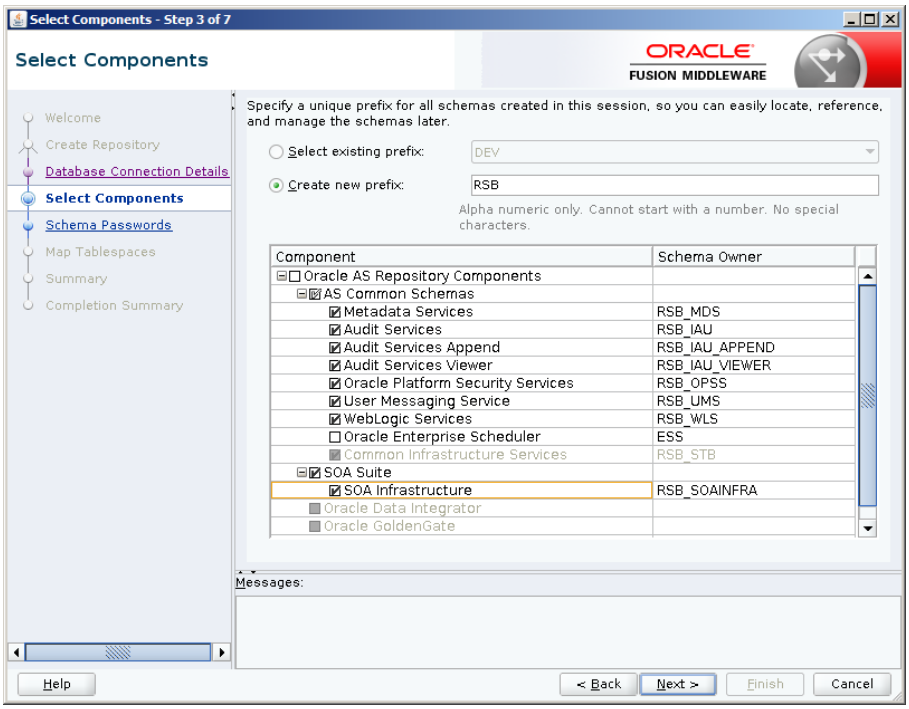
- Service Name: ribservicename
- Username (must have DBA or SYSDBA privileges)
- Password



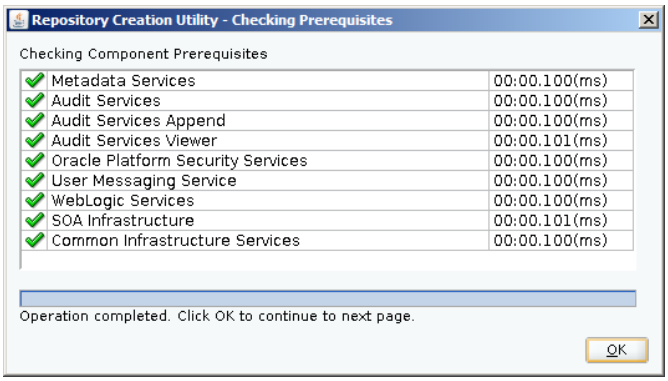
- RCU checks database prerequisites. Click **OK**.



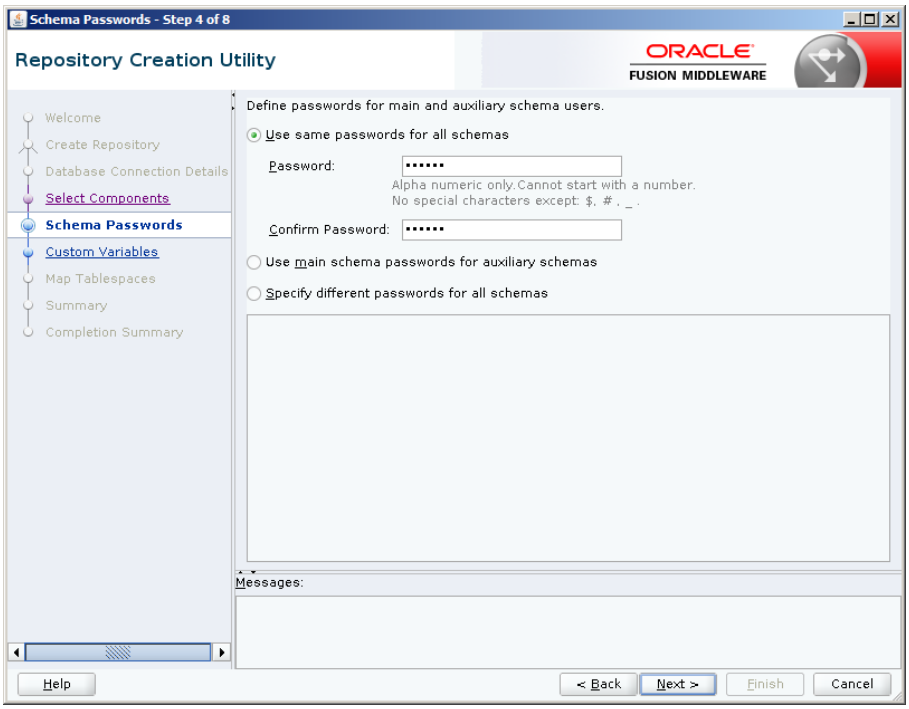
- The Select Components window opens. Select just the following items below and click **Next**.
  - Select Create a new Prefix: RSB
  - Select the following components:
    a. Metadata Services
    b. Audit Services
    c. Audit Services Append
    d. Audit Services Viewer
    e. Oracle Platform Security Services
    f. Common Infrastructure Services
    g. SOA Infrastructure



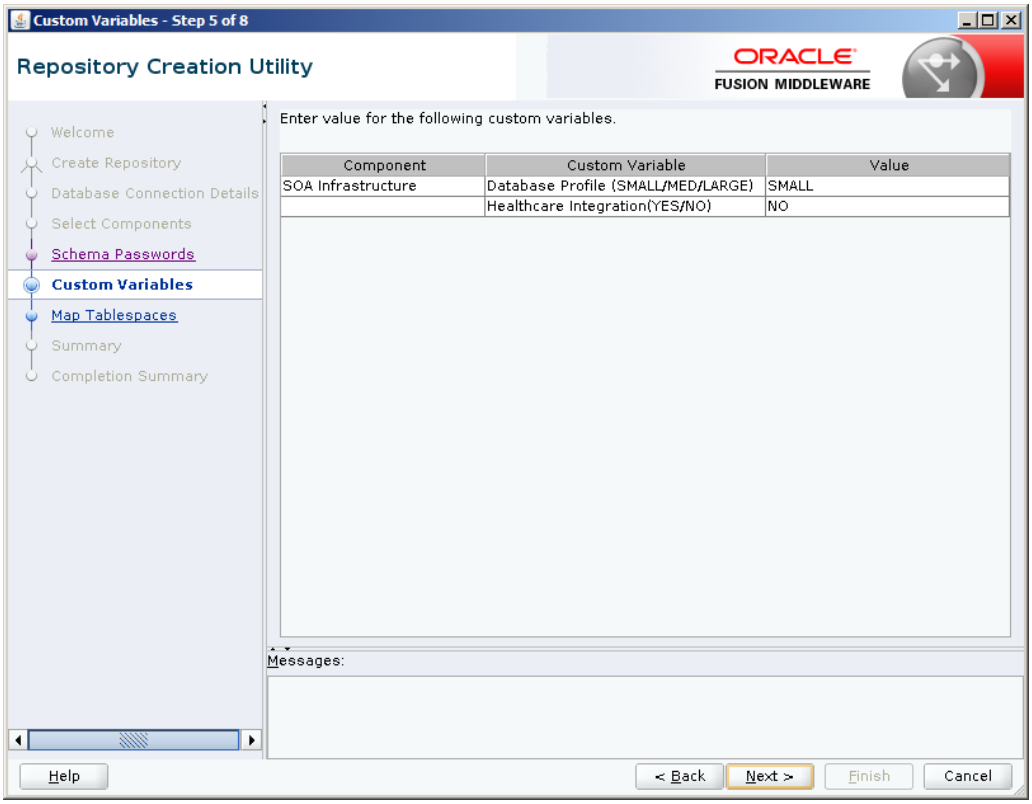
- RCU will check database prerequisites to make sure it can create the selected components. Click **OK**.



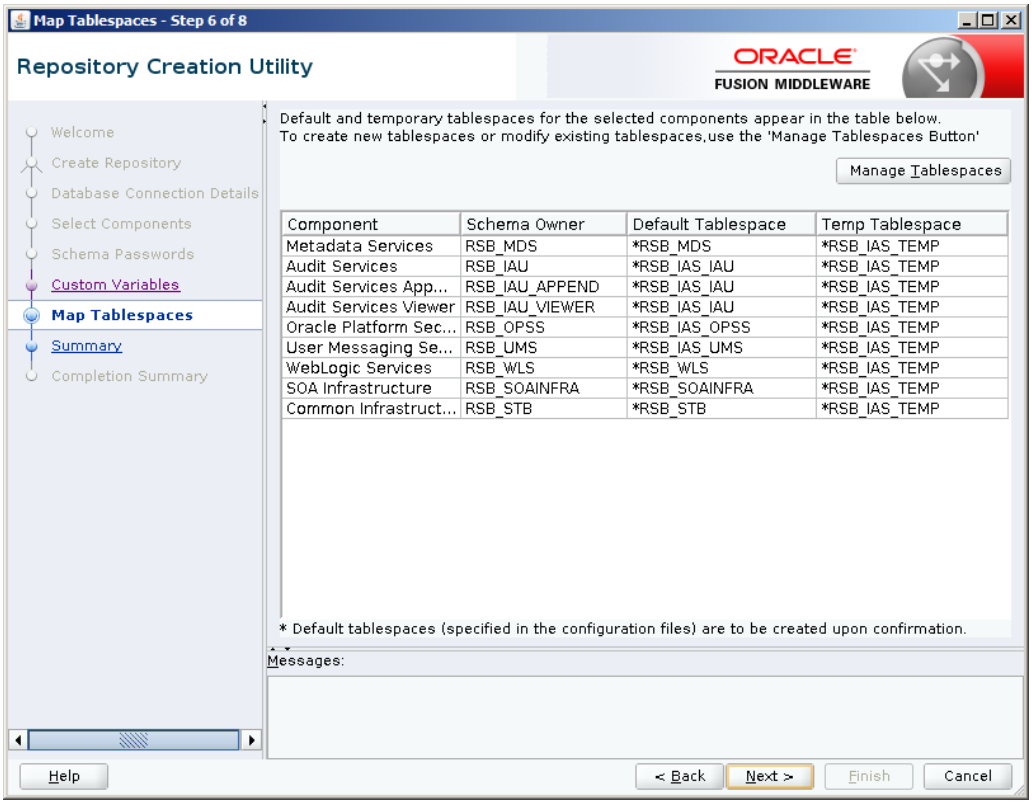
- The Schema Passwords window opens.
  - Select the Use same passwords for all schemas option.
  - Enter in the password.
  - Click **Next**.



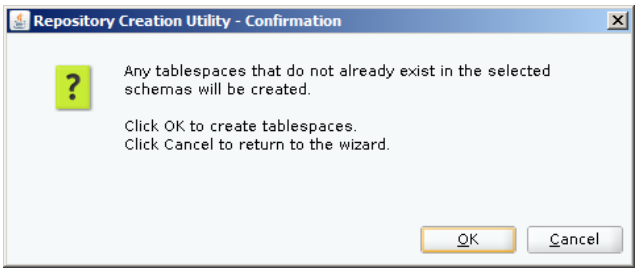
- The Custom Variables window opens. Click **Next**.



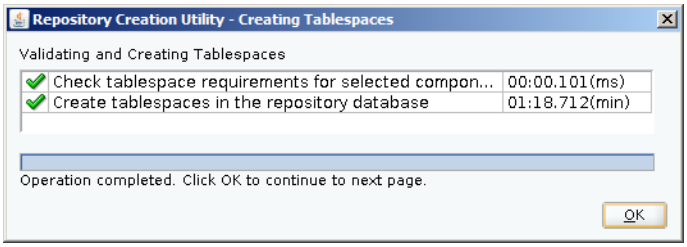
- The Map Tablespaces window opens. Click **Next**.



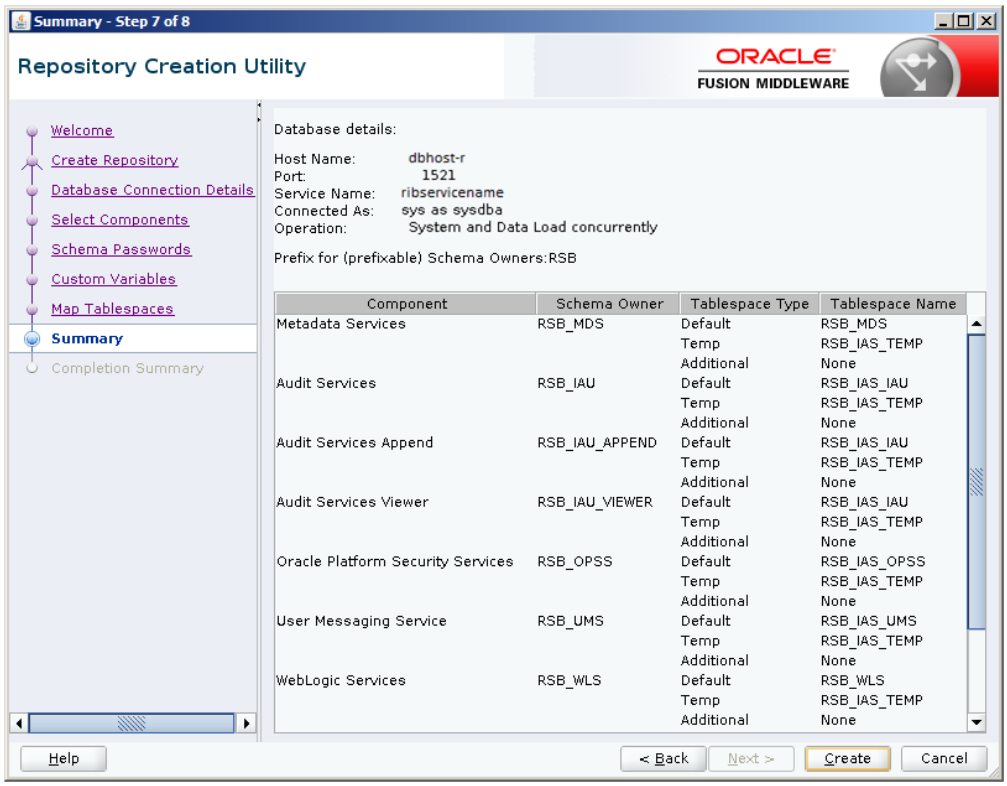
- The Repository Creation Utility - Confirmation window displays. Click **OK**.



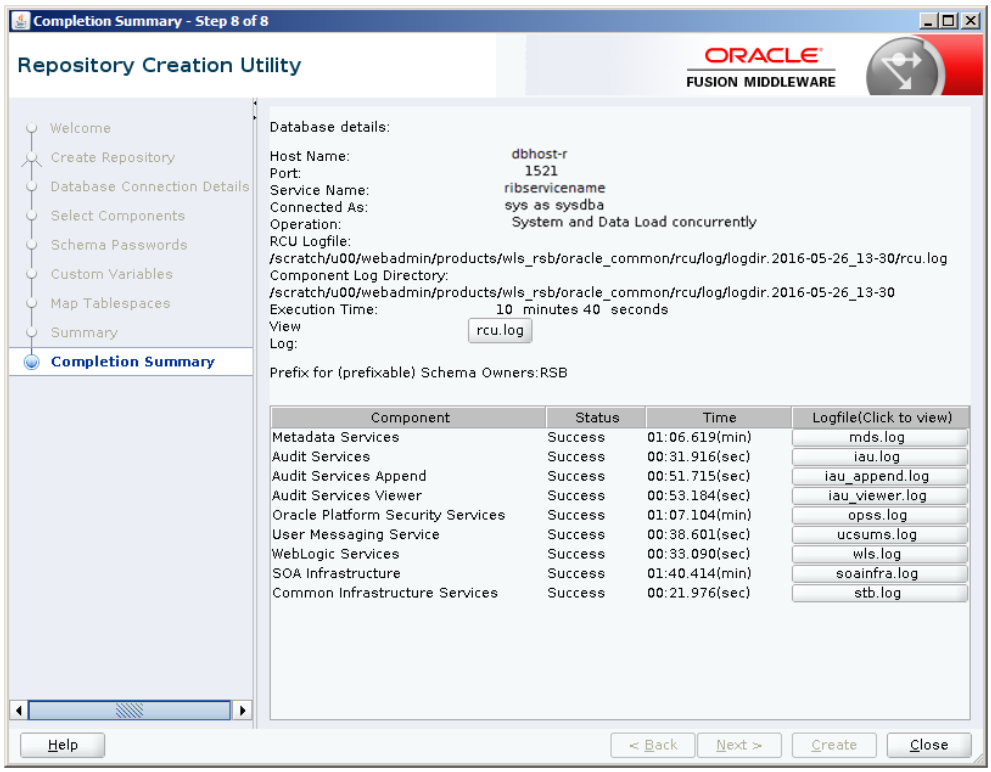
- The **Repository Creation Utility – Creating Tablespaces** window displays. Click **OK**.



- The Summary window opens. Click **Create**.



- The Completion Summary window opens. Click **Close**.



- The RCU schema creations are complete.

Configure RSBDomain on rtghost1

This section describes the steps to configure the admin and managed servers for the RSB and RIC applications. The steps should only be performed on rtghost1. The configuration will then be propagated to rtghost2 once configuration has been completed.

Perform the following steps to configure RSB admin and managed servers on rsbhost1 only.

- Set the following variables by exporting them before running the installer so that the JAVA_HOME is added to the PATH variable:
  - JAVA_HOME
  - PATH
  - DISPLAY

  Example:

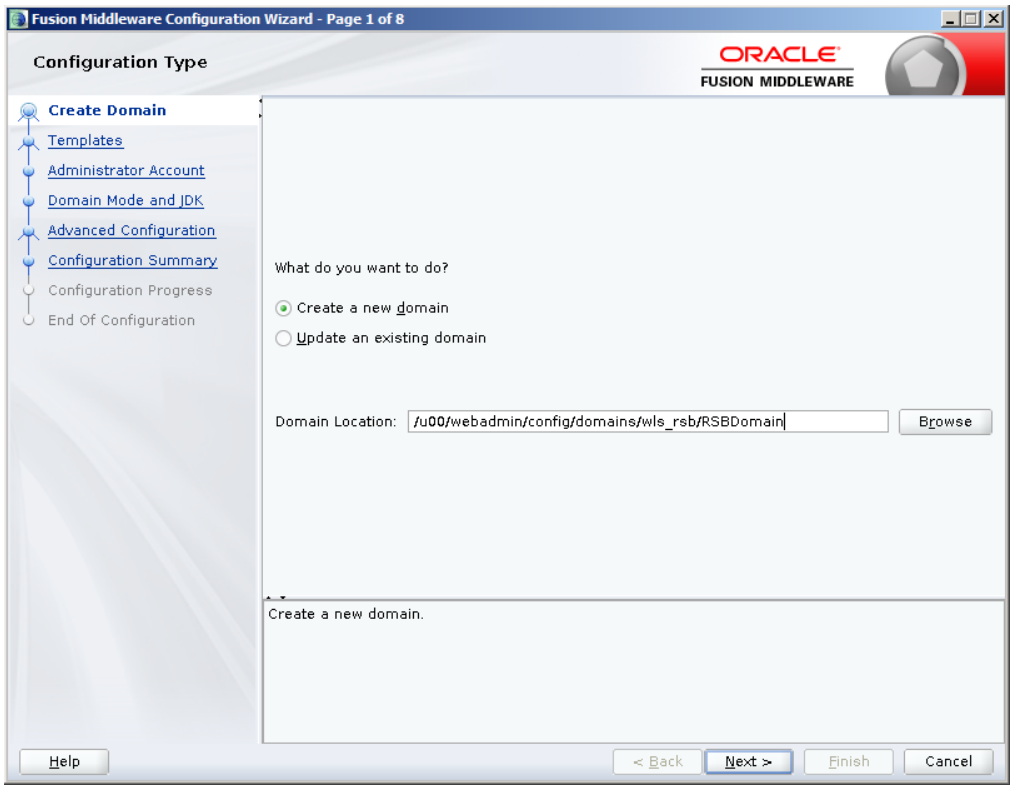
  ```
  $ export JAVA_HOME=/u00/webadmin/products/jdk_java
  $ export PATH=$JAVA_HOME/bin:$PATH
  $ export DISPLAY=<pc_ip_address:0.0>
  ```

- Navigate to the Navigate to <RTG_WEBLOGIC_HOME>/oracle_common/common/bin and run config.sh:

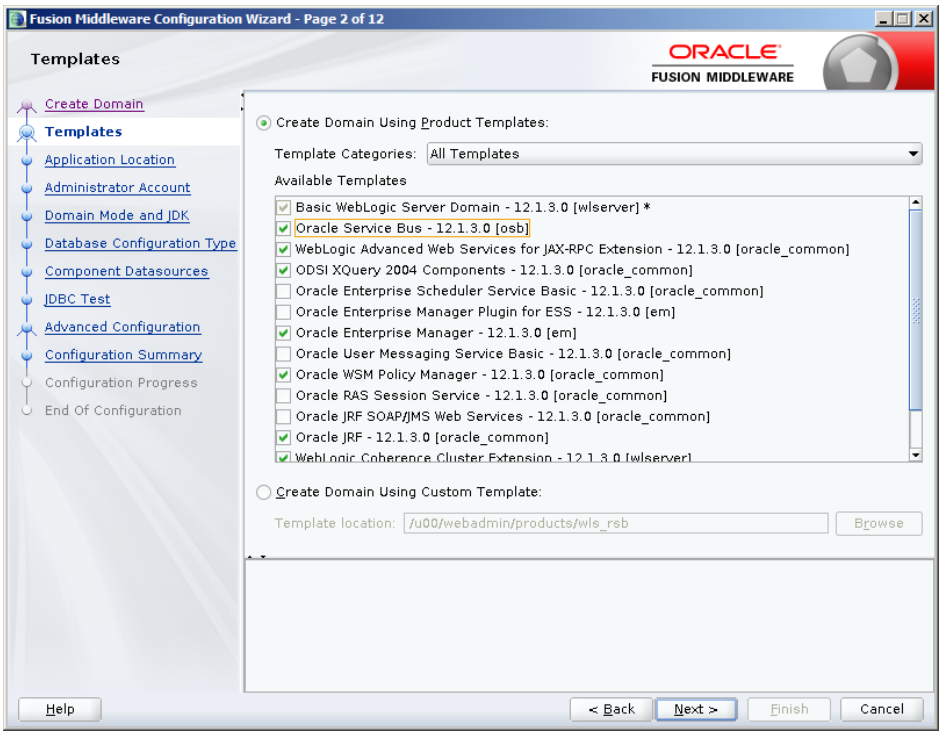
  Example:

  ```
  $ cd /u00/webadmin/products/wls_rsb/oracle_common/common/bin
  $ ./config.sh
  ```

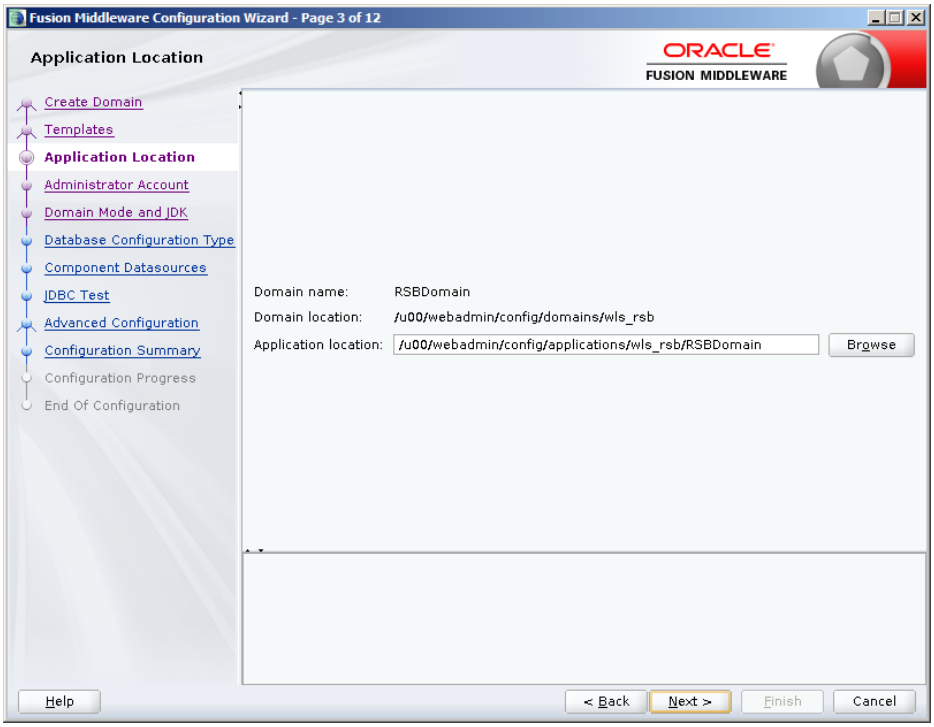
- The Configuration Type window opens. Update the screen as follows and click **Next**.
  - Select **Create a new domain**.
  - Domain Location: /u00/webadmin/config/domains/wls_rsb/RSBDomain



- The Templates window opens. Select the Oracle Service Bus - 12.1.3.0 [osb] option and the following components will automatically be selected. Click **Next**.
  - WebLogic Advanced Web Services for JAX-RPC Extension - 12.1.3 [oracle_common]
  - Oracle Enterprise Manager - 12.1.3.0 [em]
  - Oracle WSM Policy Manager - 12.1.3.0 [oracle_common]
  - Oracle JRF - 12.1.3.0 [oracle_common]
  - WebLogic Coherence Cluster Extension - 12 1.3.0 [wlserver]
  - ODSI XQuery 2004 Components - 12.1.3.0 [oracle_common]



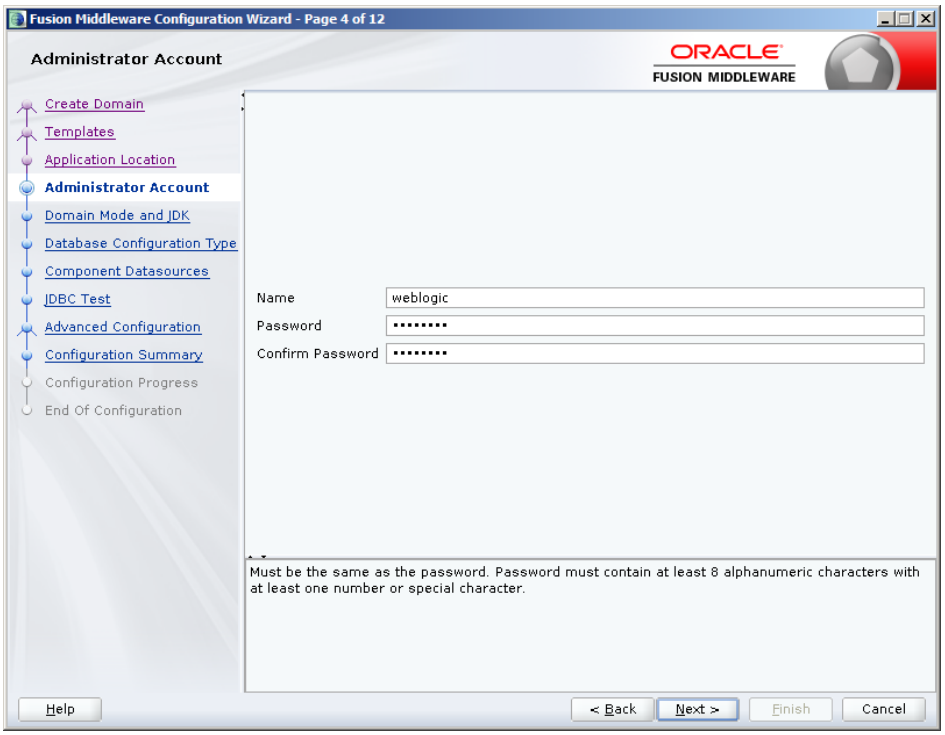
- The Application Location window opens. Update the Application Location as follows and click **Next**.
  - Application Location: /u00/webadmin/config/applications/wls_rsb/RSBDomain



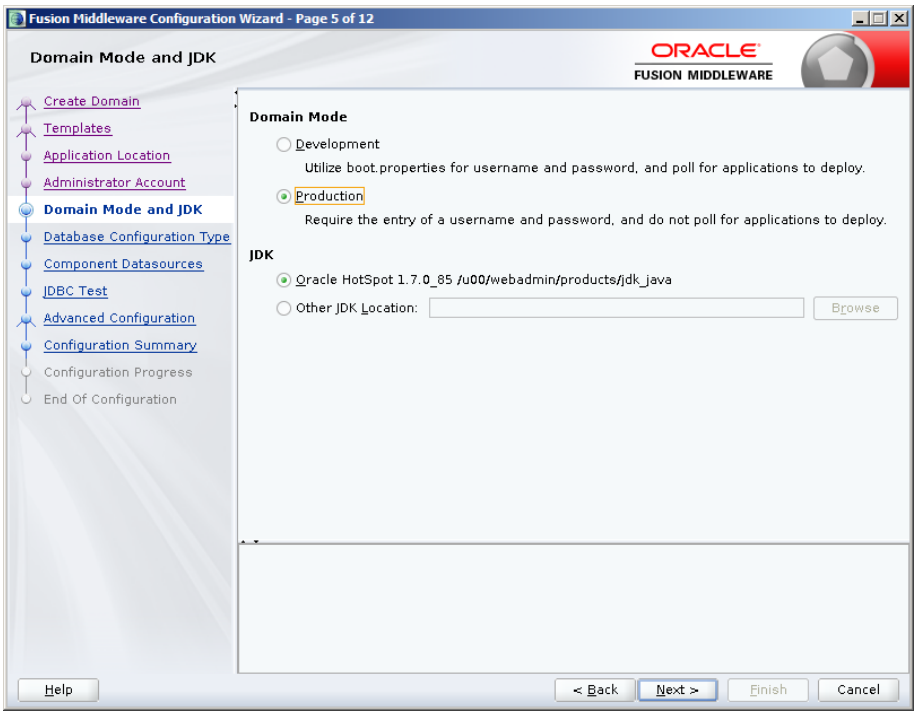
- The Administrator Account window opens. Provide the WebLogic administrator users credentials and click **Next**.

Example:

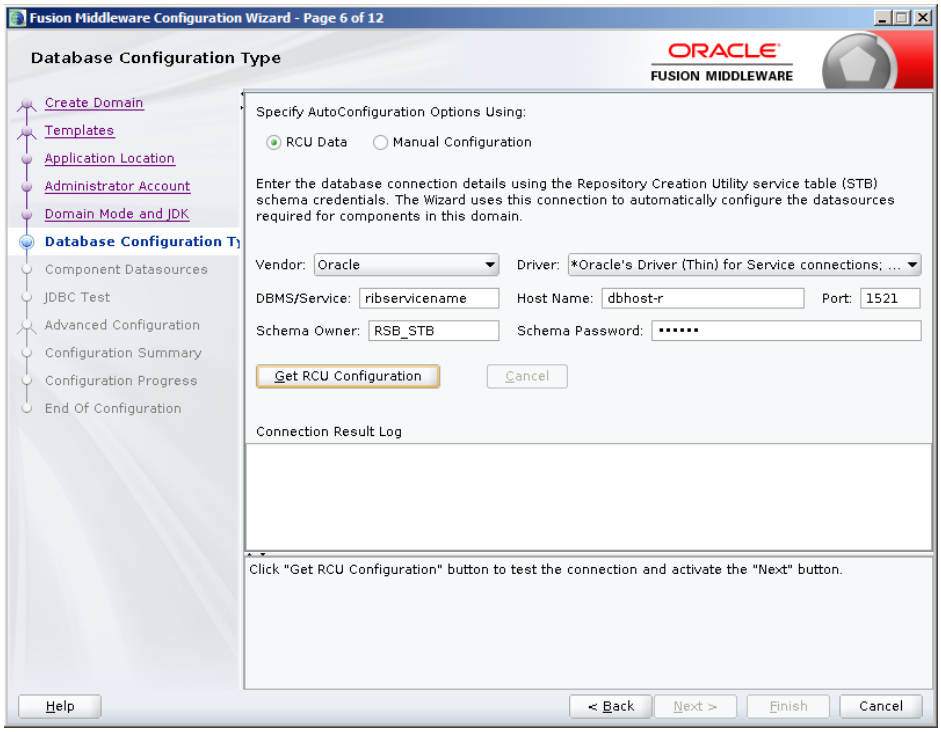
– Name: weblogic
– Password: <password>
– Confirm Password: <password>



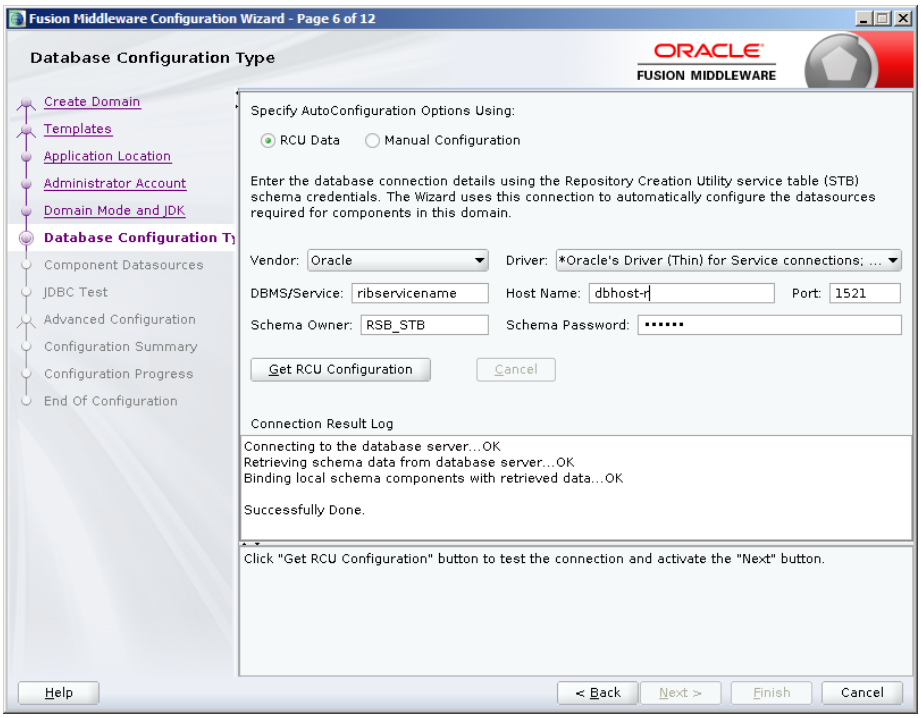
- The Domain Mode and JDK window opens. Select the Production Mode and the appropriate JDK options. Click **Next**.



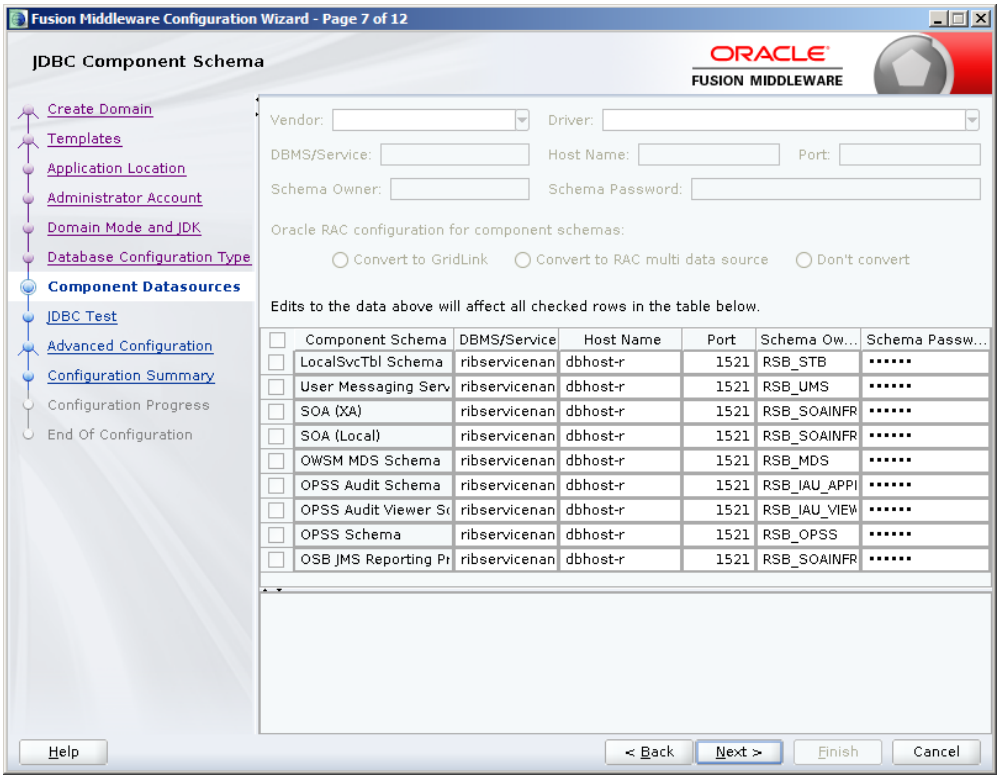
- The Database Configuration Type window opens. Update it as follows and click the Get RCU Configuration button.
  - Driver: *Oracle's Driver (Thin) for Service connections; Versions 9.0.1 and later
  - DBMS/Service: ribservicename
  - Host Name: dbhost-r
  - Port: 1521
  - Schema Owner: RSB_STB
  - Schema Password: <password>



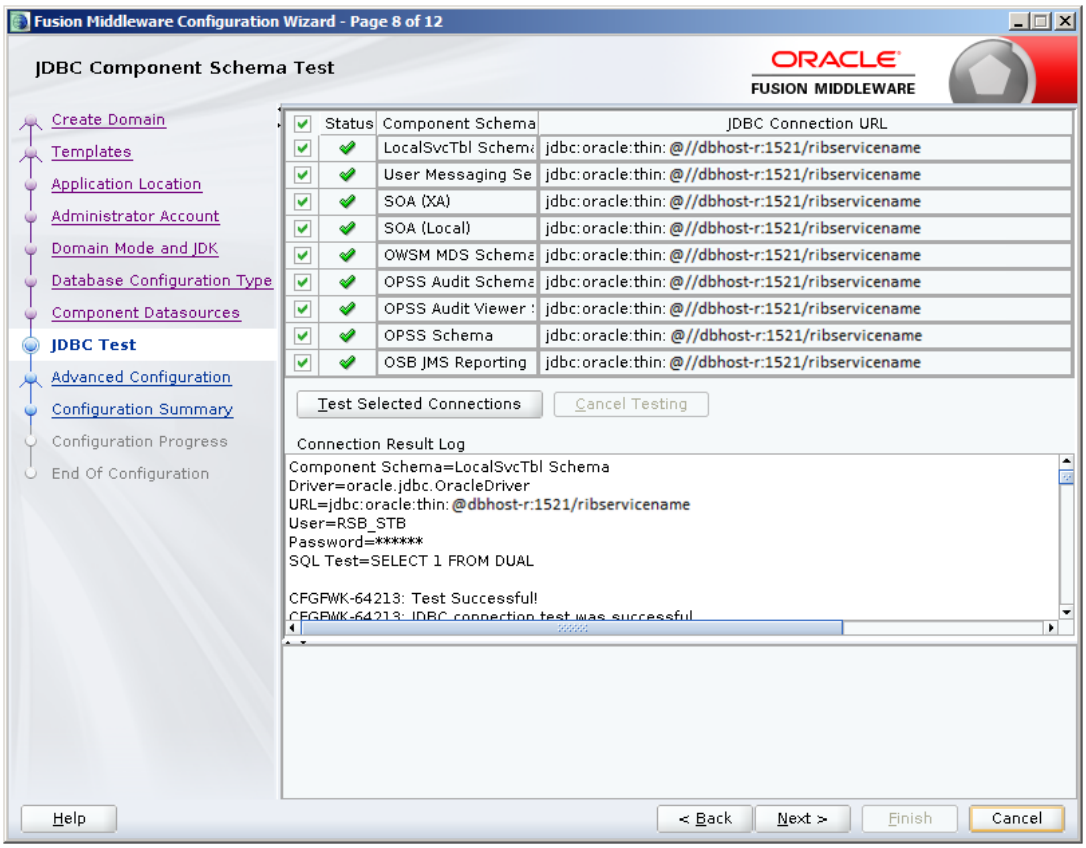
- After the screen validates the entries on the screen the **Next** button is enabled. Click **Next**.



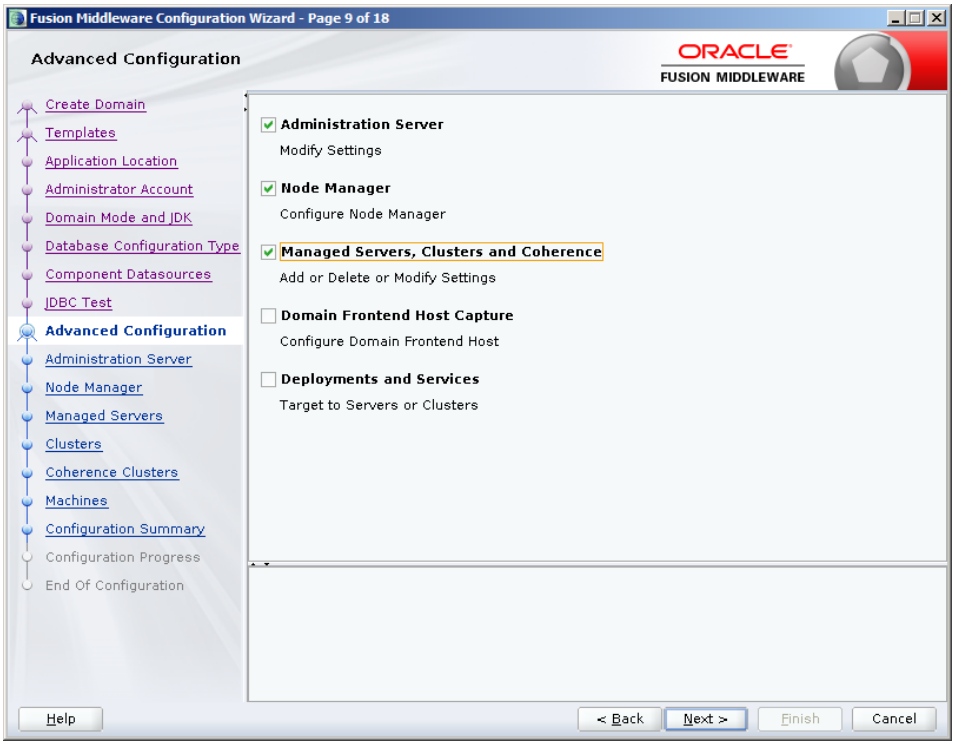
- The JDBC Component Schema is displayed. Review the entries and click **Next**.



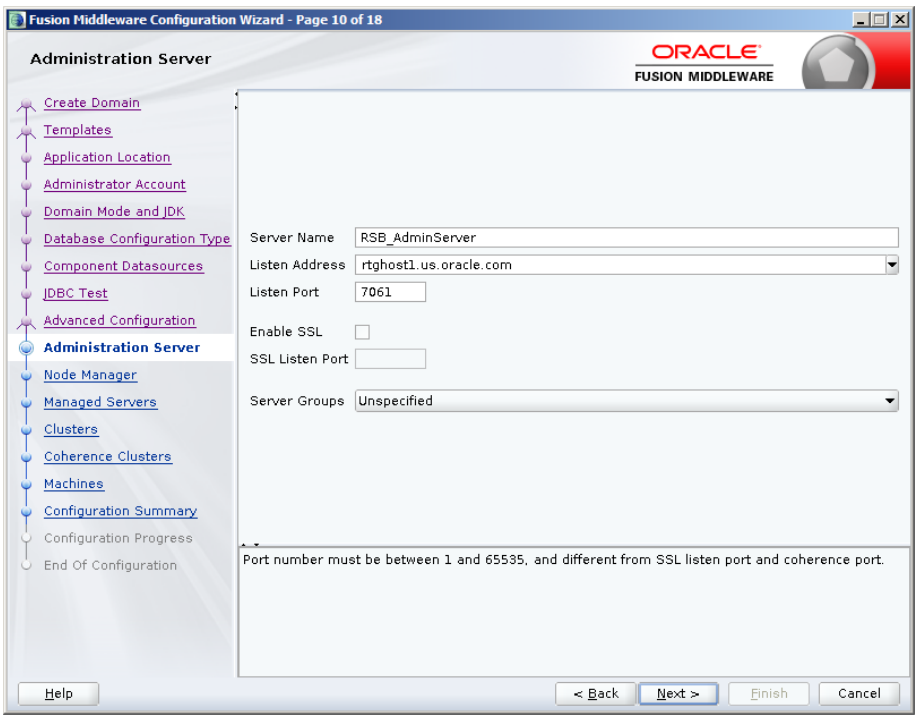
- The JDBC Component Schema Test window opens. Once all the connection tests are successful, click **Next**.



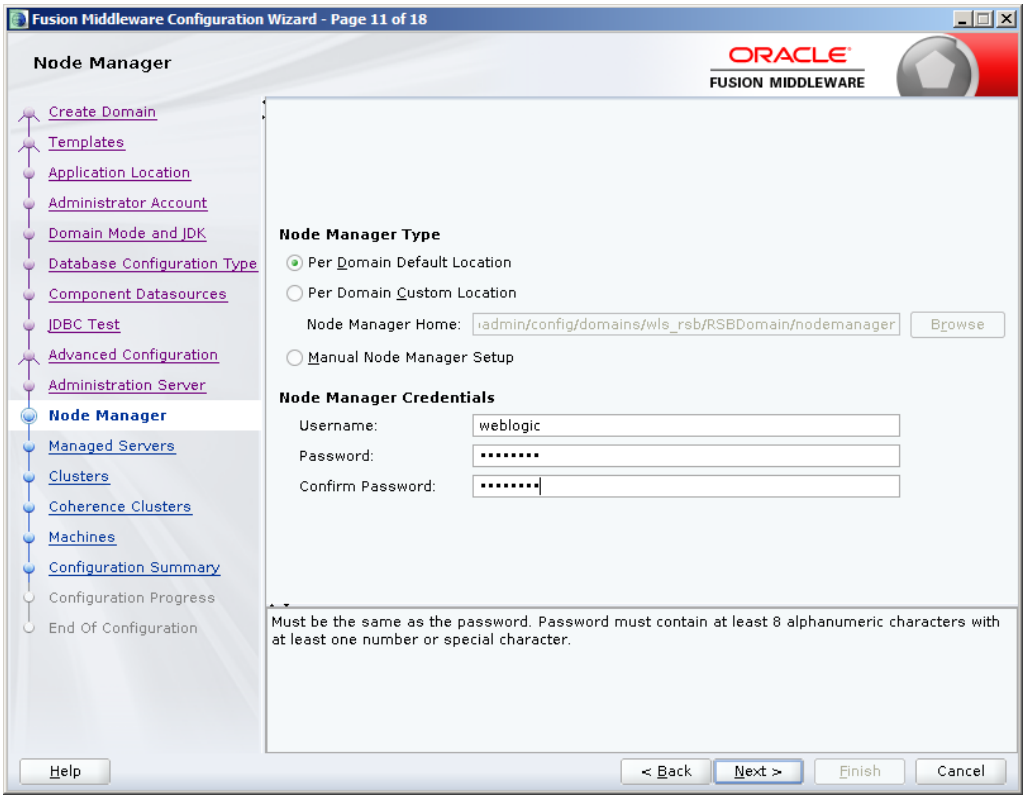
- The Advanced configuration window opens. Select the following and click **Next**.
  - Administration Server
  - Node Manager
  - Managed Servers, Clusters and Coherence



- The Administration Server window opens. Provide the Server Name, Listen Address, and Listen Port. The Listen address should be the rtghost1. See screen shot for examples. Click **Next**.



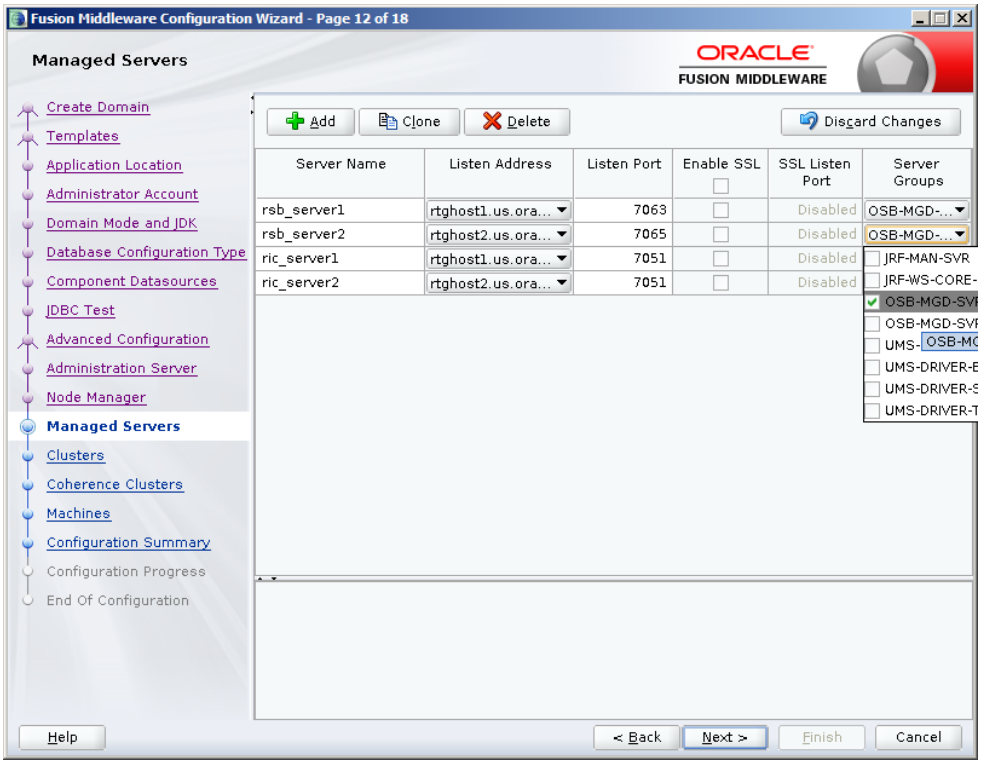
- The Node Manager window opens. Select **Per Domain Default Location** as a Node Manager Type and provide the Node Manager Credentials. Click **Next**.



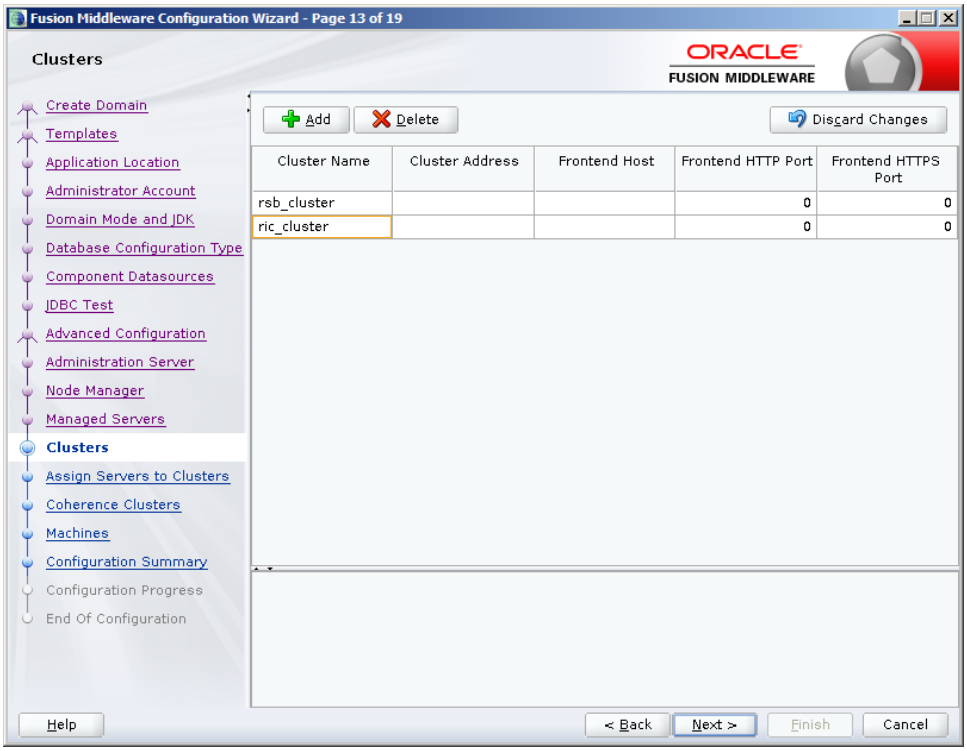
- The Managed Servers window opens. Using the Add button create an entry for each of the managed server and port specifications shown in the screen shot below. Keeping in mind that the managed servers will be created on both rtghost1 and rtghost2 - Make sure to specify the appropriate rtghost for each of the managed servers Listen Address. Select JRF-MAN-SRVS-COMBINED for the ric-server's and select OSB-MGD-SRV for the rsb-server's. See screen shot below for example. Click **Next**.

> **NOTE**: Server ports can be anything but they do have dependencies in other applications so be careful to consider that when creating them. The ports used in this document are in alignment with RSB, RIB and RGBU Application ports used in examples from other 15.0.1 HA documents.

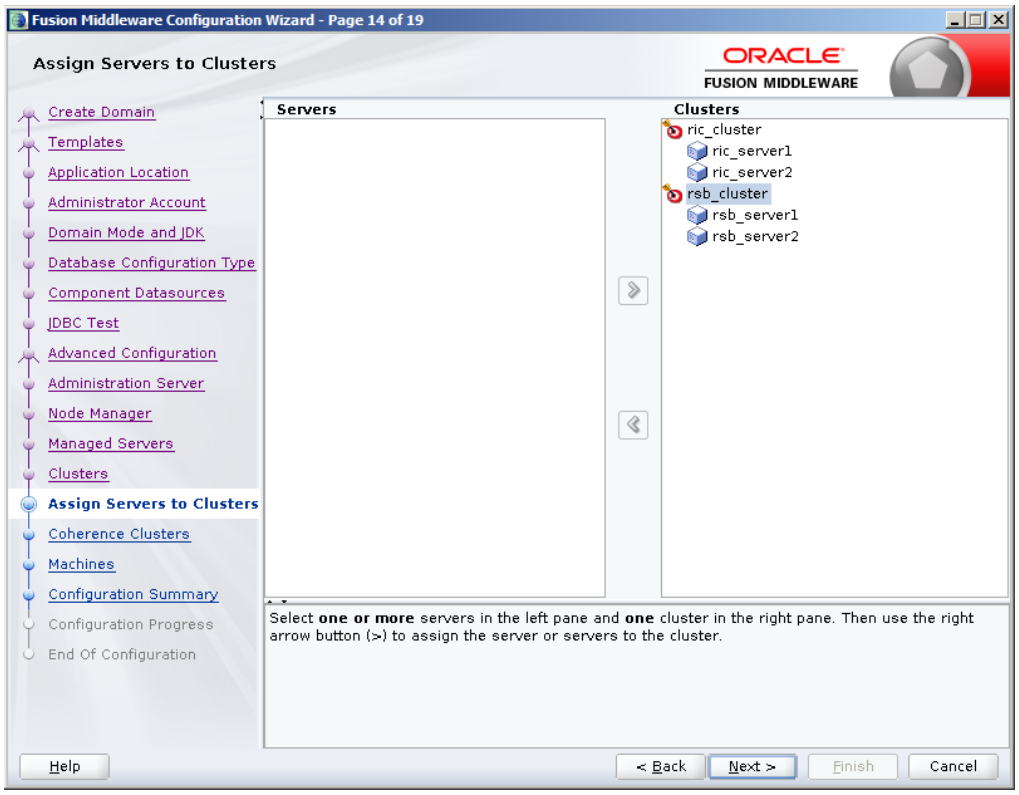
> **NOTE**: The rsb-server1 and rsb-server2 must have different ports assigned for the installation to work.



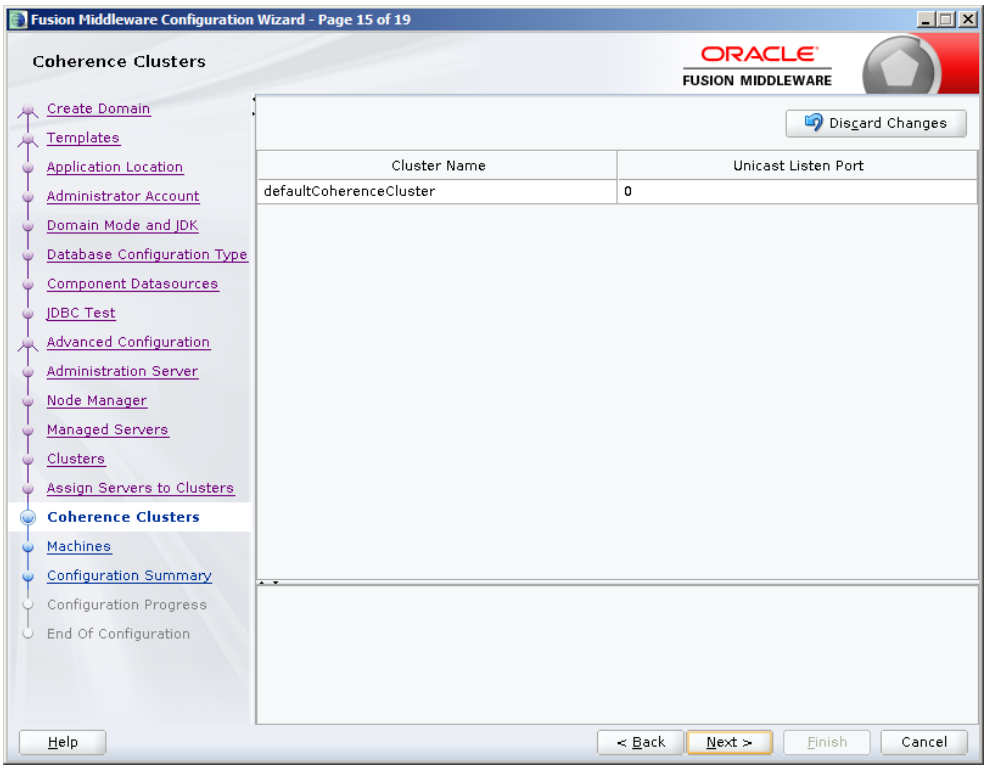
- The Clusters window opens. Using the Add button – create the following clusters and click **Next**.
  - rsb_cluster
  - ric_cluster



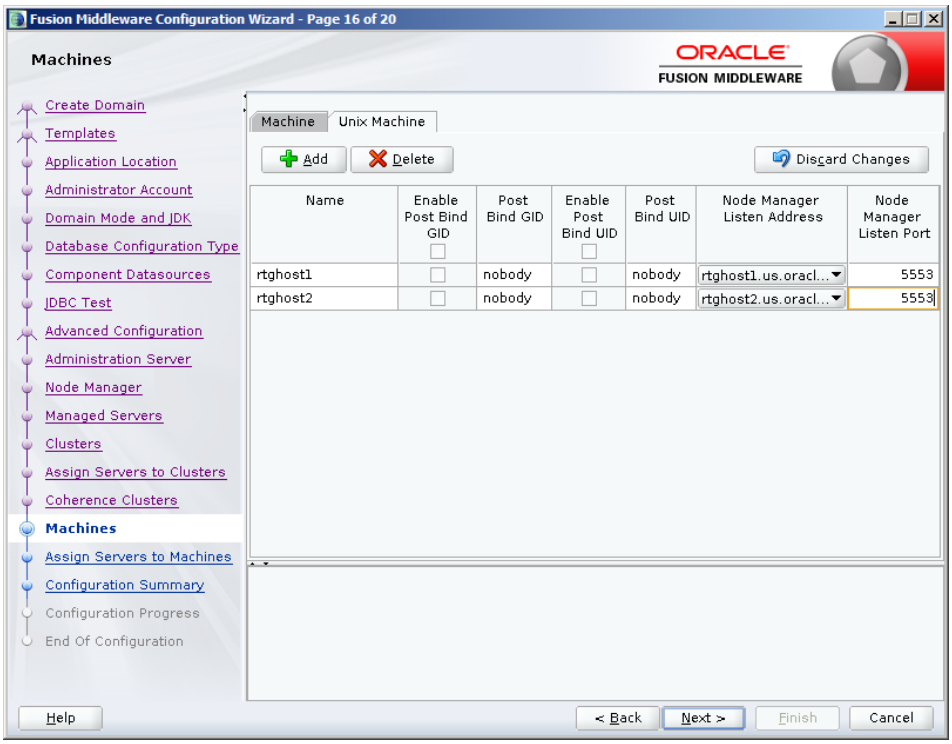
- The Assign Servers to Clusters window opens. Following the examples in the screen shot below assign the managed servers with the appropriate clusters and click **Next**.



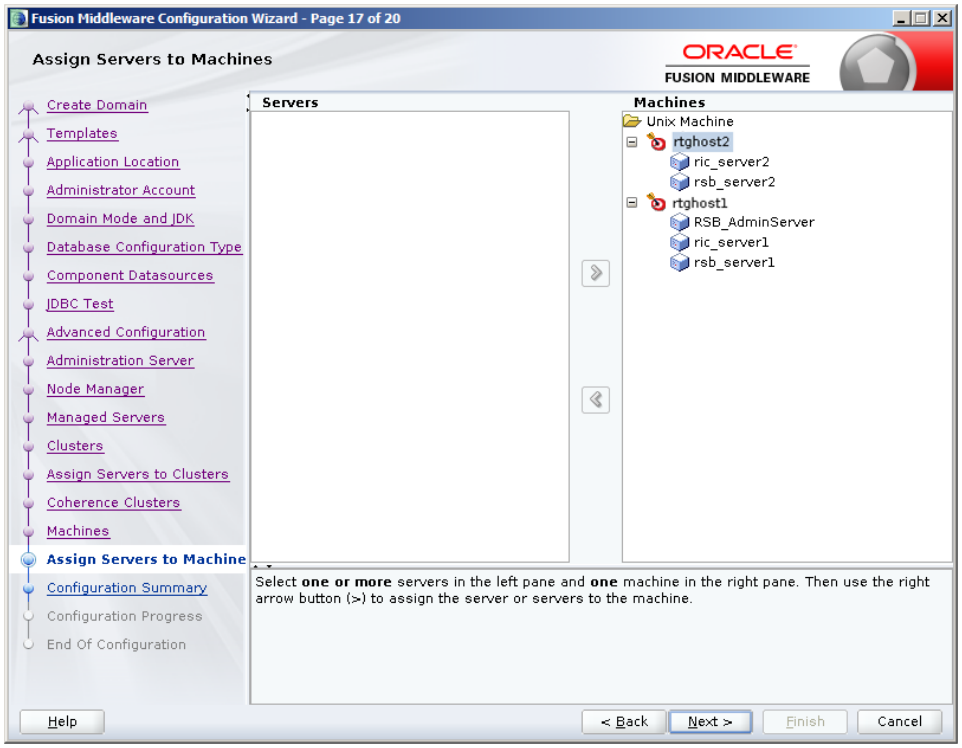
- The Coherence Clusters window opens. Click **Next**.



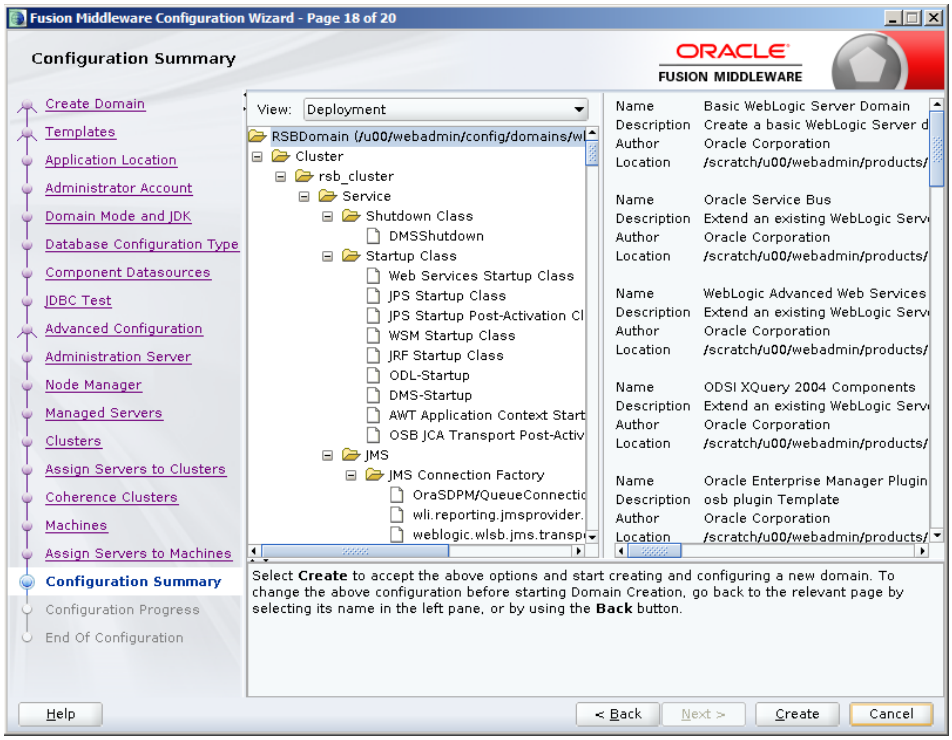
- The Machines window opens. Make sure to click on the Unix Machine tab so the screen looks like the one shown below. Refer the screenshot below, and provide the details to create a Node Manager for each rtghost. Click **Next**.



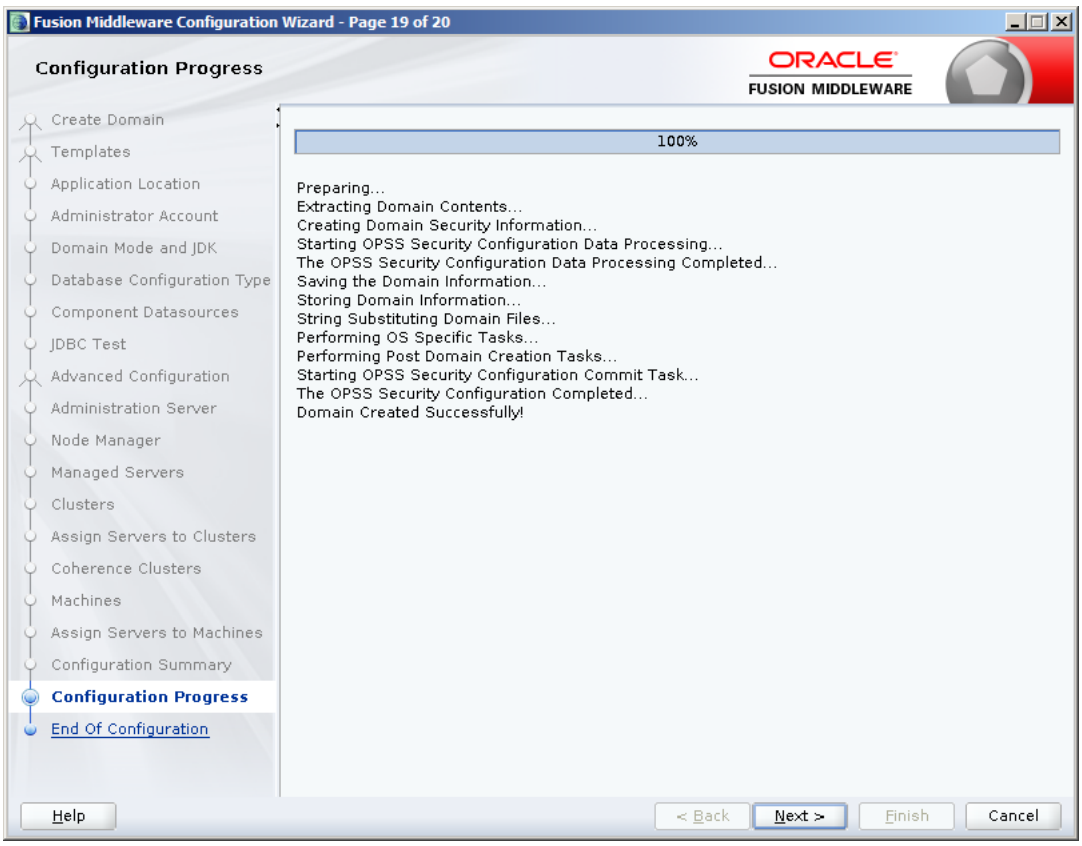
- The Assign Servers to Machines window opens. Organize and assign managed servers to appropriate rtghost and click **Next**.



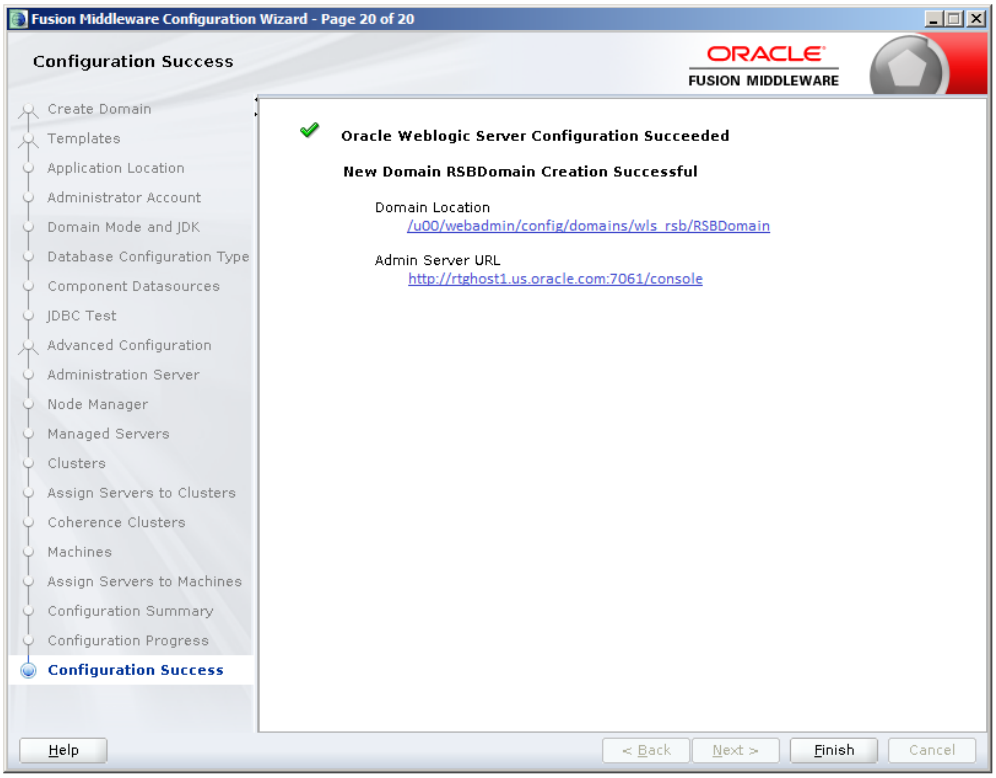
- The Configuration Summary window opens. Click **Create**.



- The Configuration Progress window opens. When progress reaches 100% - click **Next**.



- The Configuration Success window opens. Take note of the Admin Server URL and click **Finish**.



- RSBDomain configuration setup is complete.

Start WLS and Managed Servers - Create boot.properties for RSBDomain

Start the Node Manager, WebLogic admin server and create WebLogic boot.properties file with username and password so that the startup script does not prompt or require user input to start the admin server.

- Navigate to <RSB_DOMAIN_HOME>/bin and run the startNodeManager.sh script to start the Node Manager.

  Example:

  ```
  $ cd /u00/webadmin/config/domains/wls_rsb/RSBDomain/bin
  $ nohup ./startNodeManager.sh &
  ```

Create WebLogic boot.properties as follows:

- On rtghost1 - navigate to <RSB_DOMAIN_HOME>/bin and run the startWebLogic.sh script. Enter in the admin server name user and password when prompted

  Example:

  ```
  $ cd /u00/webadmin/config/domains/wls_rsb/RSBDomain/bin
  $ ./startWebLogic.sh
  username=weblogic
  password=<password>
  ```

- Shut down the admin server.
- Navigate to <RSB_DOMAIN_HOME>/servers/RSB_AdminServer/security. If the security directory does not yet exist – create it.
- Create a new file called boot.properties and add the WLS admin username and password credentials.

  Example:
  ```
  $ cd
  /u00/webadmin/config/domains/wls_rsb/RSBDomain/servers/RSB_AdminServer/sec
  urity
  $ vi boot.properties

  username=weblogic
  password=<password>
  ```

- Start the WebLogic admin server.

  Example:
  ```
  $ cd /u00/webadmin/config/domains/wls_rsb/RSBDomain/bin
  $ nohup ./startWebLogic.sh &
  ```

- Log onto the WebLogic admin server and start the managed servers that have a listen address of rtghost1 using the following URL: http://rtghost1:7061/console

Propagate RSBDomain from rthost1 to rtghost2

This section includes the instructions to be performed that will propagate the configuration of the RSBDomain configuration from rtghost1 to rtghost2.

- Shut down the admin and managed servers for the RSBDomain on rtghost1. This will make sure there are no changes that can take place and cause the hosts to be out of sync while copying the RSBDomain from rtghost1 to rtghost2.

- Add the path of the WebLogic pack.sh script to the PATH variable on rtghost1.

  Example:

  ```
  $ export
  PATH=/u00/webadmin/products/wls_rsb/oracle_common/common/bin:$PATH
  ```

- Run the pack.sh script on rtghost1 to pack up the RSBDomain and place it in a jar file called RSBDomain.jar. Following the examples provided for these steps place it in the same directory as the RSBDomain.

  Example:
  ```
  $ pack.sh -domain=/u00/webadmin/config/domains/wls_rsb/RSBDomain -
  template=/u00/webadmin/config/domains/wls_rsb/RSBDomain.jar -
  template_name=RSBDomain -managed=true
  ```

  Output:
  ```
  << read domain from /u00/webadmin/config/domains/wls_rsb/RSBDomain"
  >>  succeed: read domain from
  "/u00/webadmin/config/domains/wls_rsb/RSBDomain"
  << set config option Managed to "true"
  ```

```
>>  succeed: set config option Managed to "true"
<< write template to "/u00/webadmin/config/domains/wls_rsb/RSBDomain.jar"
.......................................................................
.........................
>>  succeed: write template to
"/u00/webadmin/config/domains/wls_rsb/RSBDomain.jar"
<< close template
>>  succeed: close template
```

- Log onto rtghost2 – Then create a directory like the one that the RSBDomain.jar exists in on rtghost1 and then copy the RSBDomain.jar there.

  Example:

  ```
  $ mkdir -p /u00/webadmin/config/domains/wls_rsb
  $ cd /u00/webadmin/config/domains/wls_rsb
  $ scp -r
  webadmin@rtghost1:/u00/webadmin/config/domains/wls_rsb/RSBDomain.jar .
  ```

- Still on rtghost2 - Add the path of the WebLogic pack.sh script to the PATH variable.

  Example:

  ```
  $ export
  PATH=/u00/webadmin/products/wls_rsb/oracle_common/common/bin:$PATH
  ```

- Un-pack the RSBDomain on rtghost2 using unpack.sh script.

  Example:
  ```
  $ unpack.sh -domain=/u00/webadmin/config/domains/wls_rsb/RSBDomain -
  template=/u00/webadmin/config/domains/wls_rsb/RSBDomain.jar -
  overwrite_domain=true -
  app_dir=/u00/webadmin/config/applications/wls_rsb/RSBDomain
  ```

  Output:
  ```
  << read template from "/u00/webadmin/config/domains/wls_rsb/RSBDomain.jar"
  >>  succeed: read template from
  "/u00/webadmin/config/domains/wls_rsb/RSBDomain.jar"
  << set config option OverwriteDomain to "true"
  >>  succeed: set config option OverwriteDomain to "true"
  << set config option AppDir to
  "/u00/webadmin/config/applications/wls_rsb/RSBDomain"
  >>  succeed: set config option AppDir to
  "/u00/webadmin/config/applications/wls_rsb/RSBDomain"
  << set config option DomainName to "RSBDomain"
  >>  succeed: set config option DomainName to "RSBDomain"
  << write Domain to "/u00/webadmin/config/domains/wls_rsb/RSBDomain"
  .......................................................................
  ................
  >>  succeed: write Domain to
  "/u00/webadmin/config/domains/wls_rsb/RSBDomain"
  << close template
  >>  succeed: close template
  ```

- Start the Node manager on rtghost2 by navigating to <RSB_DOMAIN_HOME>/bin and running the startNodeManager.sh script.

  Example:
  ```
  $ cd /u00/webadmin/config/domains/wls_rsb/RSBDomain/bin
  $ nohup ./startNodeManager.sh &
  ```

- On rtghost1 restart the WebLogic admin server.

  Example:
  ```
  $ cd /u00/webadmin/config/domains/wls_rsb/RSBDomain/bin
  $ nohup ./startWebLogic.sh &
  ```
- Log onto the WebLogic Administration Console and start all of the managed servers for RSB on both rtghost1 and rtghost2.
- Propagating the RSBDomain from rtghost1 to rtghost2 is complete.

Configure the RSBDomain Memory Allocation Arguments

This section explains how to configure the memory allocation for the managed servers in the RSBDomain. For these changes to take effect the server will have to be restarted after the changes have been made.

- Shutdown the WebLogic admin and managed servers if they are not already down.
- Take a back up of the <RSB_DOMAIN_HOME>/bin/ setDomainEnv.sh script.

  Example:
  ```
  $ cd /u00/webadmin/config/domains/wls_rsb/RSBDomain/bin
  $ cp setDomainEnv.sh setDomainEnv.shbk1
  ```
- Edit the <RSB_DOMAIN_HOME>/bin/setDomainEnv.sh script to include the following:
  - USER_MEM_ARGS="-Xms1024m -Xmx2048m -XX:MaxPermSize=1024m"

    Example:
    ```
    ...
    # IF USER_MEM_ARGS the environment variable is set, use it to override
    ALL MEM_ARGS values

    USER_MEM_ARGS="-Xms1024m -Xmx2048m -XX:MaxPermSize=1024m"

    if [ "${USER_MEM_ARGS}" != "" ] ; then
        MEM_ARGS="${USER_MEM_ARGS}"
        export MEM_ARGS
    fi
    ...
    ```
  - “-Djava.awt.headless=true"

    Example:

    ```
    ...
    if [ "${WEBLOGIC_EXTENSION_DIRS}" != "" ] ; then
        JAVA_OPTIONS="${JAVA_OPTIONS} -
    Dweblogic.ext.dirs=${WEBLOGIC_EXTENSION_DIRS}"
        export JAVA_OPTIONS
    fi

    JAVA_OPTIONS="${JAVA_OPTIONS} -Djava.awt.headless=true"

    JAVA_OPTIONS="${JAVA_OPTIONS}"
    export JAVA_OPTIONS

    # SET THE CLASSPATH
    ...
    ```

- Do not restart the RSB WebLogic admin and managed servers – the **Next** section requires them to be shut down.

Configure weblogic.policy for RSB Application

This section explains how to setup the security permission grants required for RSB deployment that will be installed in order for them to be accessible. For these changes to take effect the server will have to be restarted after the changes have been made.

- Shutdown the WebLogic admin and managed servers if they are not already down.

- Take a back up of the <RSB_WEBLOGIC_HOME>/wlserver/server/lib/weblogic.policy.

  Example:

  ```
  $ cd (/u00/webadmin/products/wls_rsb/wlserver/server/lib
  $ cp weblogic.policy weblogic.policybk1
  ```

- Edit the <RSB_WEBLOGIC_HOME>/wlserver/server/lib/weblogic.policy to include the following at the end of the file.

  NOTE: If copying the following text from this guide to UNIX, ensure that it is properly formatted in UNIX. Each line entry beginning with ”permission” must terminate on the same line with a semicolon.

  Example:

  ```
  grant codeBase "file:/u00/webadmin/config/domains/wls_rsb/RSBDomain/-" {
  permission java.security.AllPermission;
  permission
  oracle.security.jps.service.credstore.CredentialAccessPermission
  "credstoressp.credstore", "read,write,update,delete";
  permission
  oracle.security.jps.service.credstore.CredentialAccessPermission
  "credstoressp.credstore.*", "read,write,update,delete";
  };
  ```

Prepare for RSB and RIC Installations

This section covers some preliminary steps that will properly prepare for the RSB and RIC applications to be installed. The intent is to organize and create a RSB_HOME and RIC_HOME directories with the proper files placed in specific directories in order to install the applications correctly.

- On rthghost1 and rtghost2 - Restart the Node Manager as well as the WebLogic admin **server**.

  Example:

  ```
  $ cd /u00/webadmin/config/domains/wls_rsb/RSBDomain/bin
  $ nohup ./startNodeManager.sh &
  $ nohup ./startWebLogic.sh &
  ```

- Log onto the WebLogic Administration Console and start all of the managed servers for RSB on both rtghost1 and rtghost2 in the RSBDomain.
- Create directories on ribhost1 that can be used as a staging area to create the RSB_HOME and RIC_HOME directories.

Example:

```
$ mkdir -p /u00/webadmin/RSB_INSTALL
$ mkdir -p /u00/webadmin/RIC_INSTALL
```

- As a part of the 'Install RSE Appllication' steps in the previous chapter, a staging directory for RSB <RSB_STAGING> was created on rtghost1 and the RSB application distribution should have been placed in there and extracted. If this step has not been completed – it will need to be done before continuing. Going forward this document will refer to this directory as <RSB_STAGING> and the examples in this document will show it as:

  Example:

  ```
  /u00/webadmin/media/rsb/
  ```

- From the <RSB_STAGING>/CDROM/rsb150a/CDROM directory - copy the RsbKernel15.0.0ForAll15.x.xApps_eng_ga.zip into the RSB_INSTALL directory and extract its contents. The rsb-home will be created inside this directory.

  Example:

  ```
  $ cd /u00/webadmin/media/rsb/15.0/CDROM/rsb150a/CDROM
  $ cp RsbKernel15.0.0ForAll15.x.xApps_eng_ga.zip /u00/webadmin/RSB_INSTALL
  $ cd /u00/webadmin/RSB_INSTALL
  $ unzip RsbKernel15.0.0ForAll15.x.xApps_eng_ga.zip
  ```

- Copy all of the RsbAppServiceDecoratorPak15.0.0For<app>15.0.0_eng_ga.zip to rsb-home/download-home/all-app-service-decorator directory. Do not extract the files.

  Example:

  ```
  $ cd /u00/webadmin/media/rsb/15.0/CDROM/rsb150a/CDROM
  $ cp RsbAppServiceDecoratorPak*
  /u00/webadmin/RSB_INSTALL/Rsb1500ForAll15xxApps/rsb-home/download-
  home/all-app-service-decorator
  ```

- Copy the all RsbServiceIntegrationFlowPak15.0.0ForRibOmsToRsbOmsRouting_eng_ga.zip rsb-home/download-home/all-functional-service-int-flow directory. Do not extract the file.

  Example:

  ```
  $ cd /u00/webadmin/media/rsb/15.0/CDROM/rsb150a/CDROM
  $ cp RsbServiceIntegrationFlowPak*
  /u00/webadmin/RSB_INSTALL/Rsb1500ForAll15xxApps/rsb-home/download-
  home/all-functional-service-int-flow
  ```

- From the <RSB_STAGING>/CDROM/rsb150a/CDROM directory - copy the RicKernel15.0.0ForAll15.x.xApps_eng_ga.zip into the RIC_INSTALL directory and extract its contents. The ric_home will be created inside this directory.

  Example:

  ```
  $ cd /u00/webadmin/media/rsb/15.0/CDROM/rsb150a/CDROM
  $ cp RicKernel15.0.0ForAll15.x.xApps_eng_ga.zip /u00/webadmin/RIC_INSTALL
  $ cd /u00/webadmin/RIC_INSTALL
  $ unzip RicKernel15.0.0ForAll15.x.xApps_eng_ga.zip
  ```

- Give execute permissions to RSB_HOME and RIC_HOME directories:

  Example:

  ```
  $ chmod -R 755 /u00/webadmin/RSB_INSTALL
  $ chmod -R 755 /u00/webadmin/RIC_INSTALL
  ```

- Preparation for RSB and RIC complete.

Install RSB Application

Perform the following steps on rtghost1 to install the RSB application.

- Set the following variables by exporting them before running the installer so that the JAVA_HOME is added to the PATH variable:
  - JAVA_HOME
  - PATH

  Example:

  ```
  $ export JAVA_HOME=/u00/webadmin/products/jdk_java
  $ export PATH=$JAVA_HOME/bin:$PATH
  ```

- Run rsb-home/download-home/bin/check-version-and-unpack.sh script. This will verify the versions of the kernel and downloaded decorators and extract them in respective folders.

  Example:

  ```
  $ cd /u00/webadmin/RSB_INSTALL/Rsb1500ForAll15xxApps/rsb-home/download-home/bin
  $ ./check-version-and-unpack.sh
  ```

- Navigate to <RSB_HOME>/toolsdeployment-home/conf/ take a backup and then edit the rsb-deployment-env-info.properties file. Using Appendix I – rsb-deployment-env-info.properties as an example and considering the properties listed below containing <> tags indicate a variable inside the <> tags, Update the following properties with the environment information:
  - JAVA_HOME
  - rsb-deployment-env-info.service-provider-app-in-scope-for-integration
  - rsb-deployment-env-info.service-requester-app-in-scope-for-integration
  - rsb-osb-container.domain-name
  - rsb-osb-container.<domain-name>.home
  - rsb-osb-container.<domain-name>.cluster-name
  - rsb-osb-container.<domain-name>.<cluster-name>.http-url (Cluster port is the port of http proxy server)
  - rsb-osb-container.<domain-name>.admin-server-name
  - rsb-osb-container.<domain-name>.admin-server-http-url
  - rsb-osb-container.<domain-name>.admin-server-connection-url
  - rsb-osb-container.<domain-name>.<cluster-name>.managed-servers: It is a comma-separated list of managed servers in the cluster, excluding the http proxy managed server.
  - rsb-osb-container.<domain-name>.<cluster-name>.<managed-server>.managed-server-connection-url: Repeat this property for all the managed servers in the cluster.
  - service-infrastructure-db.jdbc-url
  - edge-app-container.<app>.connection-url: The host:port of the edge-application.
  - rib.home.path

- Navigate to <RSB_HOME>/service-assembly-home/bin and run the rsb-compiler.sh to setup RSB credentials and compile objects prior to deployment. The rsb-compiler script will prompt the user for input regarding the following credentials: (Answer each prompt and press enter.)
  - sidb-jdbc-user-alias: This is the soa_infra datasource created by the RCU for RSB.
  - admin-server-user-alias: This is the WebLogic admin user.
  - rsb-admin-user-alias: This will create a user that can be used to log onto the RSB Console.

  Example:

  ```
  $ cd /u00/webadmin/RSB_INSTALL/Rsb1500ForAll15xxApps/rsb-home/service-assembly-home/bin
  $ rsb-compiler.sh -setup-security-credential
  ```

  Output:

  ```
  replace ../../integration-lib/weblogic-12.1.3.jar? [y]es, [n]o, [A]ll, [N]one, [r]ename: y

  Enter User Name for Alias[sidb-jdbc-user-alias]: rsb_soainfra
  Enter password for Alias[sidb-jdbc-user-alias]: <password>

  Enter User Name for Alias[admin-server-user-alias]: weblogic
  Enter password for Alias[admin-server-user-alias]: <password>

  Enter User Name for Alias[rsb-admin-user-alias]: rsbadmin
  Enter password for Alias[rsb-admin-user-alias]: <password>
  ```

- Review the log files located in <RSB_HOME>/service-assembly-home/log/rsb-admin-assembly log. Fix any errors that are encountered before continuing.
- Navigate to <RSB_HOME>/deployment-home/bin and run the rsb-deployer.sh to deploy RSB.

  Example:

  ```
  $ cd /u00/webadmin/RSB_INSTALL/Rsb1500ForAll15xxApps/rsb-home/deployment-home/bin
  $ rsb-deployer.sh -prepare-wls
  ```

- Review the log files located in <RSB_HOME>/deployment-home/log/configure-wls-for-rsb.log. Fix any errors that are encountered before continuing.
- Navigate to <RSB_HOME>/deployment-home/bin and run the rsb-deployer.sh to deploy RSB service decorators.

  Example:

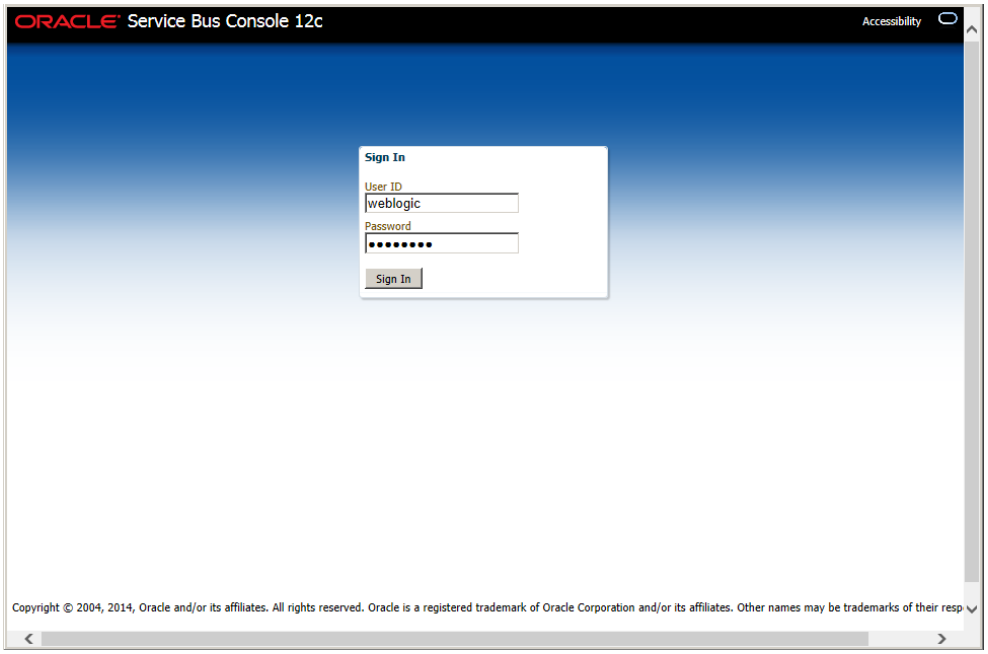
  ```
  $ cd /u00/webadmin/RSB_INSTALL/Rsb1500ForAll15xxApps/rsb-home/deployment-home/bin
  $ rsb-deployer.sh -deploy-all-rsb-service
  ```

- Review the rsb-config-agent-deployer.log and rsb-service-deployer.log log files located in the <RSB_HOME>/deployment-home/log/ directory to validate no errors were encountered during deployment.
- RSB Installation complete.

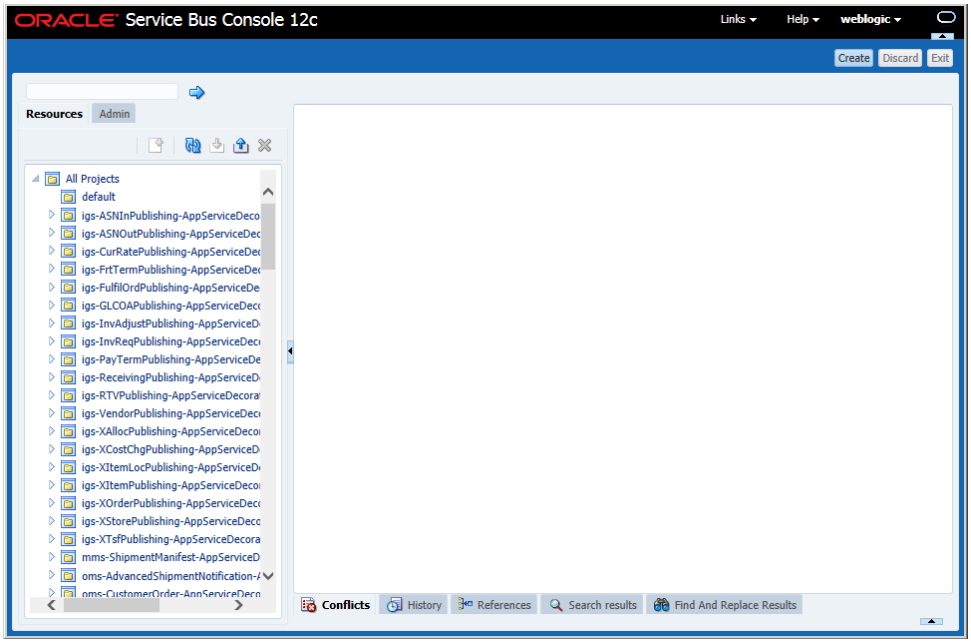
Verification - Verify RSB Installation

Perform the following steps to validate the RSB installation by logging onto the application.

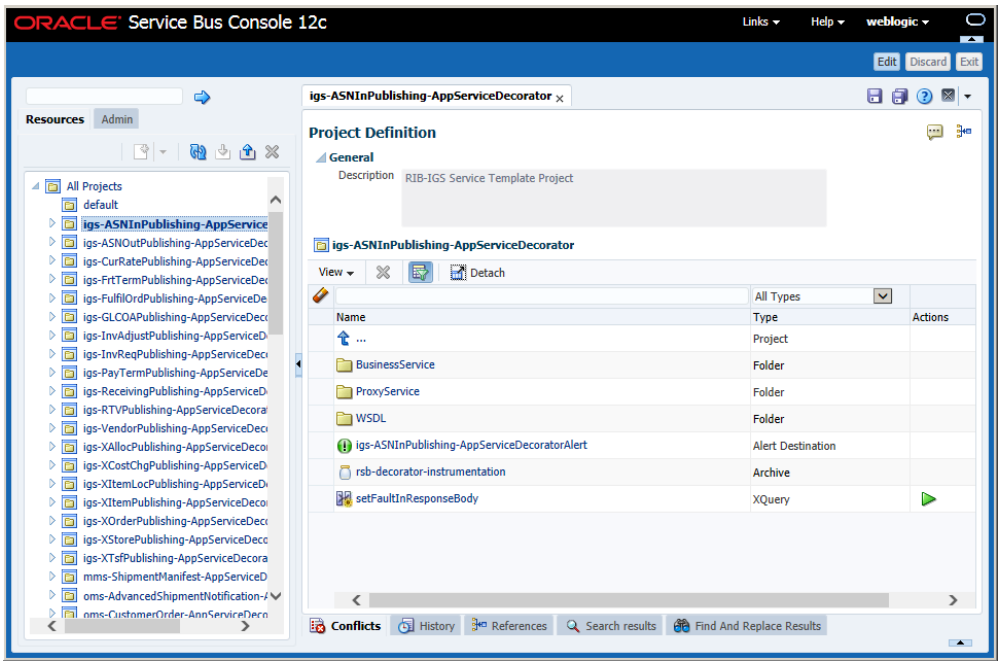
- Launch the following URL to make sure the Service Bus Console is running. The RSB installer deploys the Service Bus Console to the admin server so it will only run from rtghost1.
  - http://rtghost1.us.oracle.com:7061/sbconsole



- The Service Bus Console window opens. Click on one of the links in the navigation pane on the left side of the screen.



- The screen will be populated with details regarding the link that was clicked. See example below.



- Test complete.

Install RIC Application

Perform the following steps on rtghost1 to install the RIC application.

- Set the following variables by exporting them before running the installer so that the JAVA_HOME is added to the PATH variable:
  - JAVA_HOME
  - PATH

    Example:

```
$ export JAVA_HOME=/u00/webadmin/products/jdk_java
$ export PATH=$JAVA_HOME/bin:$PATH
```

- Navigate to <RIC_HOME>/conf/ take a backup and then edit the ric-deployment-env-info.json file. Using Appendix II – ric-deployment-env-info.json as an example. The following properties will be updated:
  - DataSourceDef
    a. jdbcUrl
  - MiddlewareServerDef
    a. weblogicDomainName
    b. weblogicDomainHome
    c. weblogicDomainAdminServerUrl
    d. weblogicDomainAdminServerHost
    e. weblogicDomainAdminServerPort
    f. weblogicDomainTargetManagedServerName
    g. ricUiUrl
    h. ricUiUserGroup

  - IntegrationProduct
    a. ribHome
    b. rsbHome
    c. ddiHome
- Navigate to <RIC_HOME>/bin and run the ric-app-compiler.sh to setup the RIC ear file and setup credentials. The ric-app-compiler script will prompt the user for input regarding the following credentials: (Answer each prompt and press enter.)
  - ricAppServerAdminServerUserAlias: This is the WebLogic admin user.
  - ricAdminUiUserAlias: This will create a user that can be used to log onto the RIC Console.
  - ricDataSourceUserAlias: This is the soa_infra datasource created by the RCU for RSB.

    Example:

```
$ cd /u00/webadmin/RIC_INSTALL/ric-home/bin
$ ./ric-app-compiler.sh -setup-credentials
```

    Output:

```
Credential required for
weblogicDomainAdminServerHost(rtghost1.us.oracle.com)
weblogicDomainAdminServerPort(7061):
Enter username for alias (ricAppServerAdminServerUserAlias):weblogic
Enter Password: <password>

Credential required for ricUiUrl(http://rtghost1.us.oracle.com:7061/rsb-
admin):
Enter username for alias (ricAdminUiUserAlias):rsbadmin
Enter Password: <password>

-- side note created during RCU schema creation time.
Credential required for dataSource(jdbc/RicDataSource)
jdbcUrl(jdbc:oracle:thin:@//msp00avyp-r:1521/DOLRP04RIB):
Enter username for alias (ricDataSourceUserAlias):rsb_soainfra
Enter Password: <password>
```

- Navigate to <RIB_HOME>/bin and run the ric-app-deployer.sh to deploy RIC and create the user and group on the RSB WebLogic server for the RIC admin user(ricAdminUiUserAlias value).

  Example:

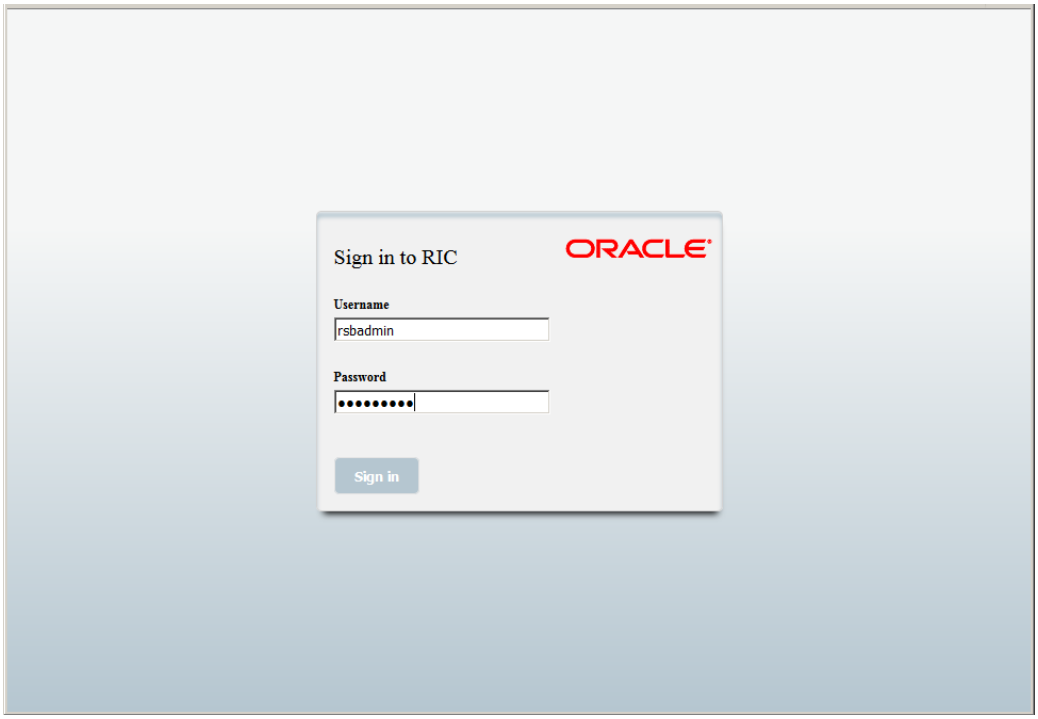
```
$ cd /u00/webadmin/RIC_INSTALL/ric-home/bin
$ ./ric-app-deployer.sh -deploy-ric-app
```

- RIC Installation complete.

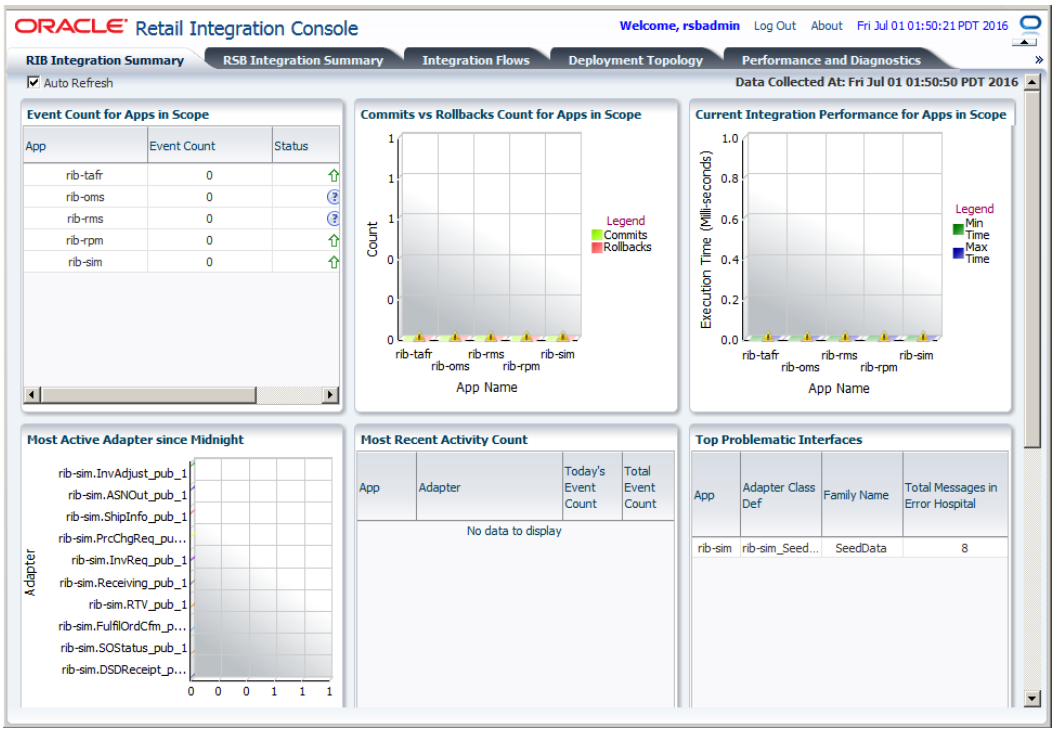
Verification - Verify RIC Installation

Perform the following steps to validate the RIC installation by logging onto the application.

- Launch the following URL to make sure the Retail Integration Console is running. The RIC installer deploys the application to the admin server so it will only run from rtghost1.
  - http://rtghost1.us.oracle.com:7061/rsb-admin/



- The RIB integration Summary page will be displayed with details showing the current status and statistics for the applications that are installed and configured to use the RIB.



- Test is complete.

6

OHS Configuration for RIB and RTG Applications

In Part 1 Web Tier was installed on the RTG and RIB hosts. This section covers the final steps of configuring Web Tier in order to setup the RIB, RSB, and RTG applications to be highly available and ensuring that users will be able to access applications running on multiple hosts using the same URL.

Keep in mind that RIB and its components are installed on a shared mount in an active/passive cluster. This means only have to make updates on 1 server for RIB. The RTG applications are active/active so all nodes in the cluster will need to be updated.

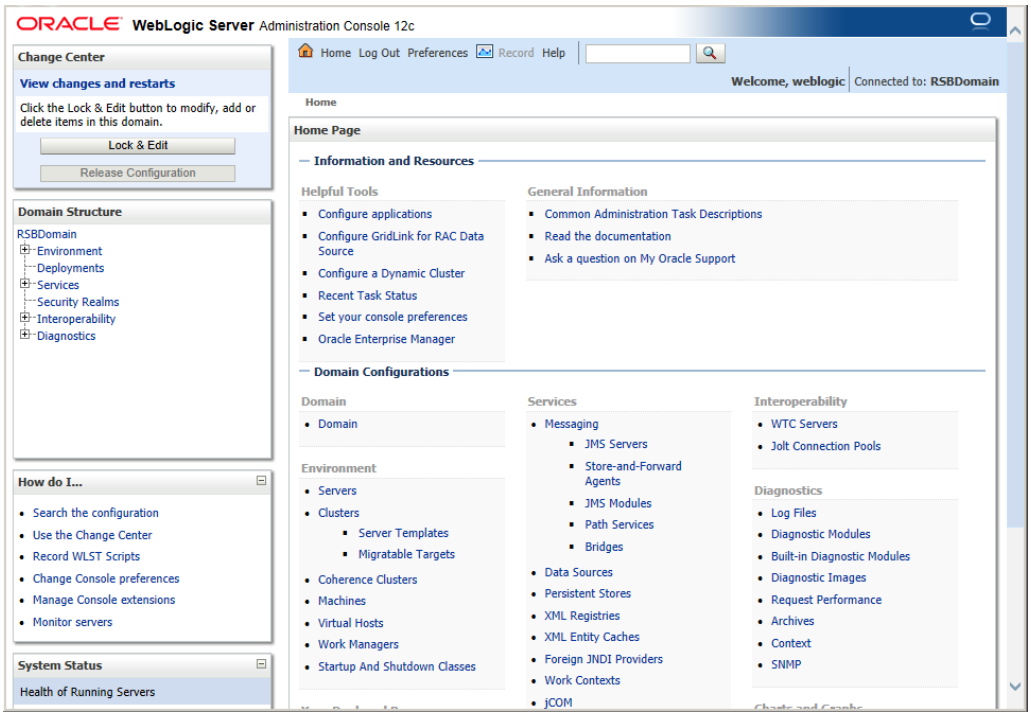
Enable Tunneling for Servers in RIB, RTG, and RSB Domains

Perform the following steps to enable tunneling for each of the admin and managed servers in the RIB, RTG, and RSB domains using the WebLogic Administration Console.

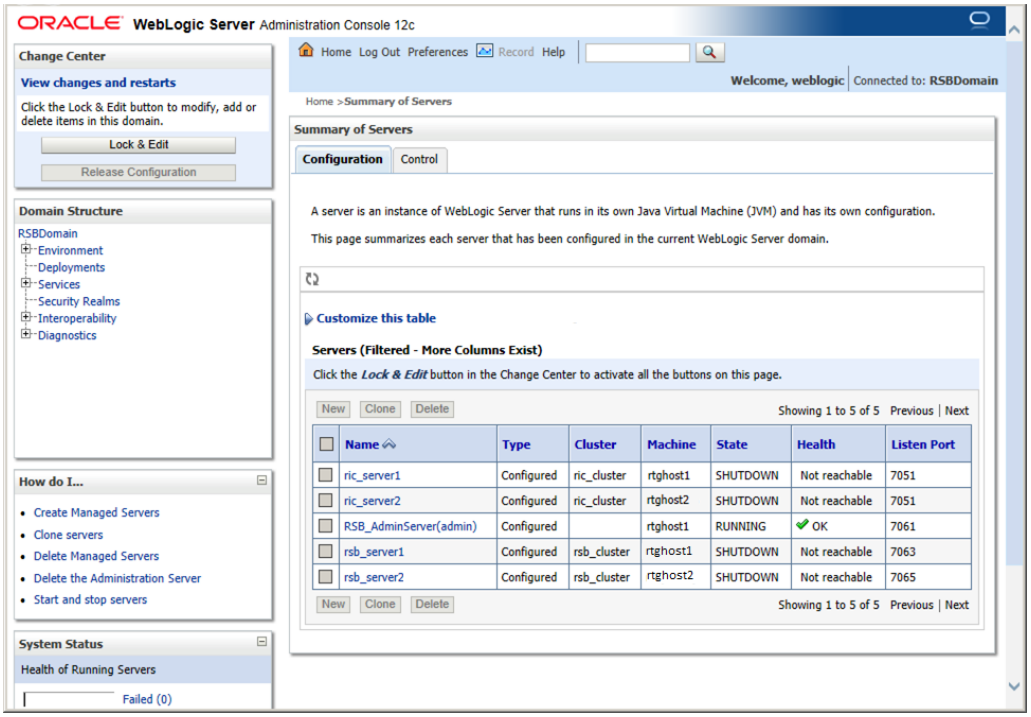
The steps below are for enabling tunneling on the admin and managed server in the RSBDomain. These steps should be repeated for both the RIBDomain and RTGDomain domains when finished.

- Log onto the RSB WebLogic Administration Console.

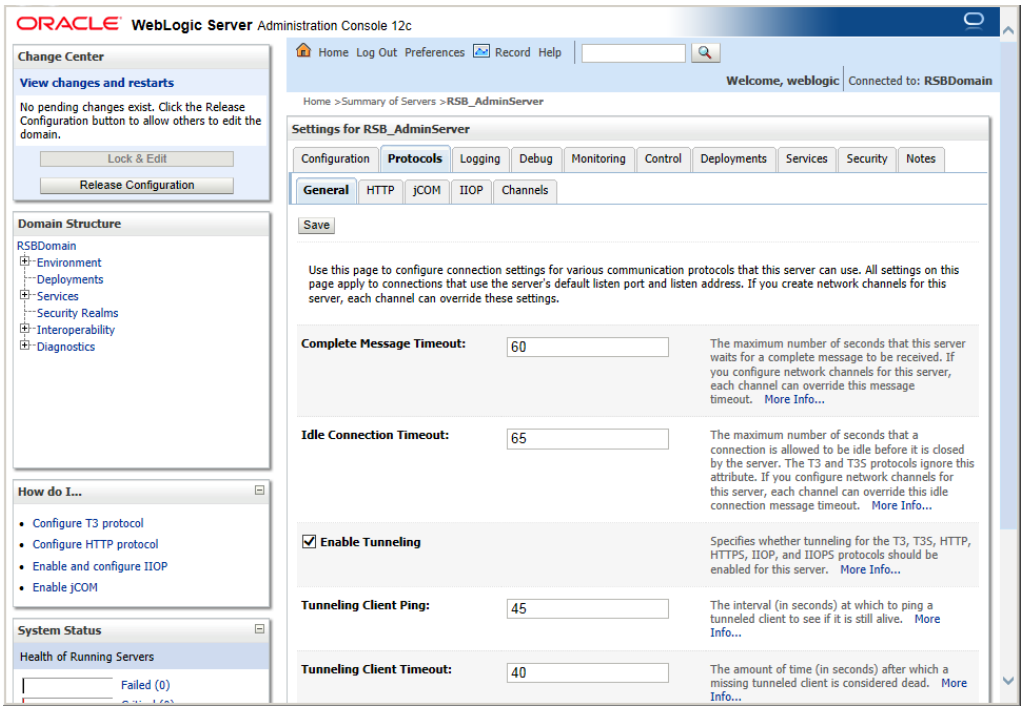
- The Home Page window opens. Click **Servers**.



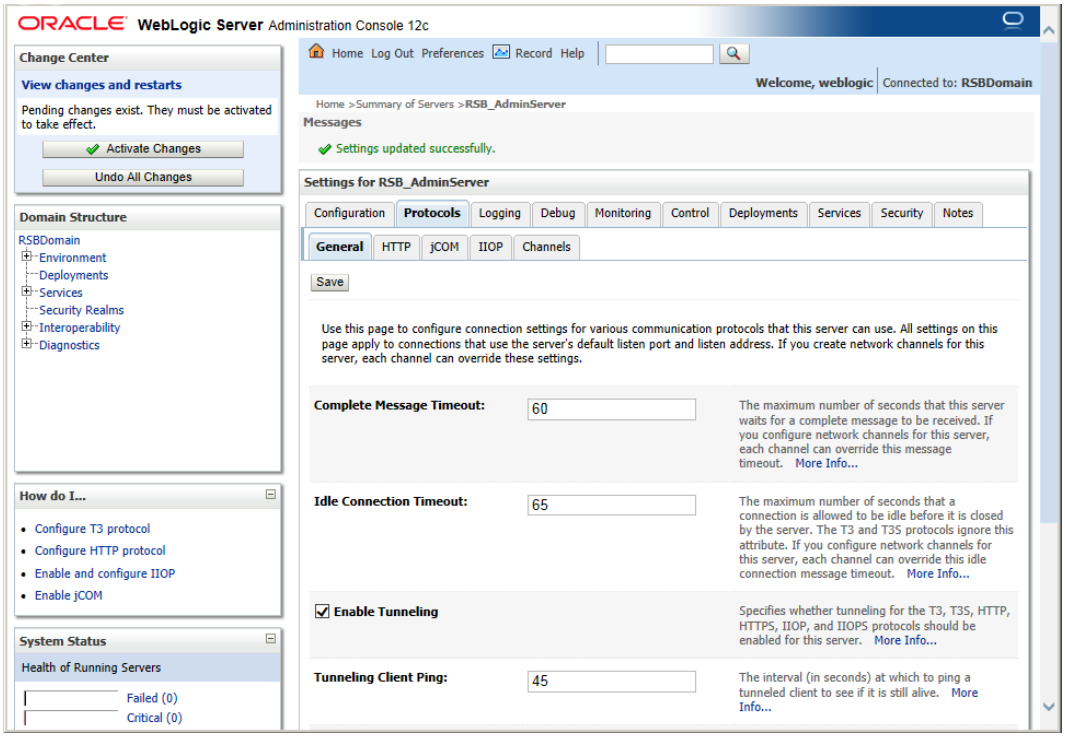
- The Summary of Servers window opens. Click the **RSB_AdminServer(admin)** link.



- The Settings for RSB_AdminServer window opens. Navigate to the Protocols-> General tab and **Click Lock & Edit**. The screen becomes available for updates. Select the 'Enable Tunneling' option and click **Save**.



- The **Activate Changes** button becomes enabled. Click to activate the changes and then repeat the above steps for each of the managed servers.



Update Web Tier for RIB Applications

Perform the following steps to configure the Web Tier to manage the RIB applications' connections.

- On ribhost1 – set the ORACLE_INSTANCE variable so that the Web Tier opmnctl commands can be used to start and stop Web Tier's opmn processes.

  Example:

  ```
  $ export
  ORACLE_INSTANCE=/u00/webadmin/products/wls_ohs/Oracle_WT1/instances/instan
  ce1/
  ```

- Navigate to <WEB_TIER_WLS_HOME>/ Oracle_WT1/opmn/bin and stop the Web Tier opmn processes.

  Example:

  ```
  $ cd /u00/webadmin/products/wls_ohs/Oracle_WT1/opmn/bin
  $ opmnctl stopall
  ```

- Navigate to the <ORACLE_INSTANCE>/config/OHS/ohs1 directory for Web Tier, then create a backup and update the mod_wl_ohs.conf file to include tunneling and location URL redirects for the RIB application. Make sure to use the floating IP address in place of hostnames since this is for active/passive host configuration.

  Example:

  ```
  $ cd
  /u00/webadmin/products/wls_ohs/Oracle_WT1/instances/instance1/config/OHS/o
  hs1
  $ vi mod_wl_ohs.conf
  ```

  See Appendix III RIB mod_wl_ohs.conf for example of entries.

- Navigate to <WEB_TIER_WLS_HOME>/ Oracle_WT1/opmn/bin and start the Web Tier opmn processes.

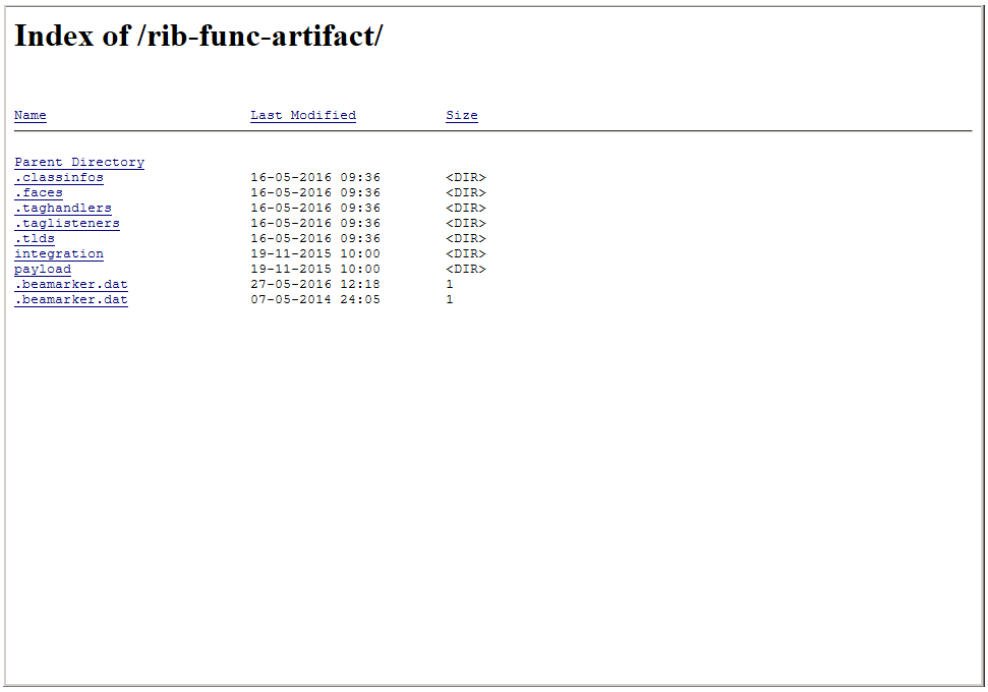
  Example:

  ```
  $ cd /u00/webadmin/products/wls_ohs/Oracle_WT1/opmn/bin
  $ opmnctl startall
  ```

Verification - Verify Web Tier Configuration for RIB

Perform the following steps to validate that the RIB applications can be accessed via URL's made up of the virtual host alias (VIP) and Web Tier port.

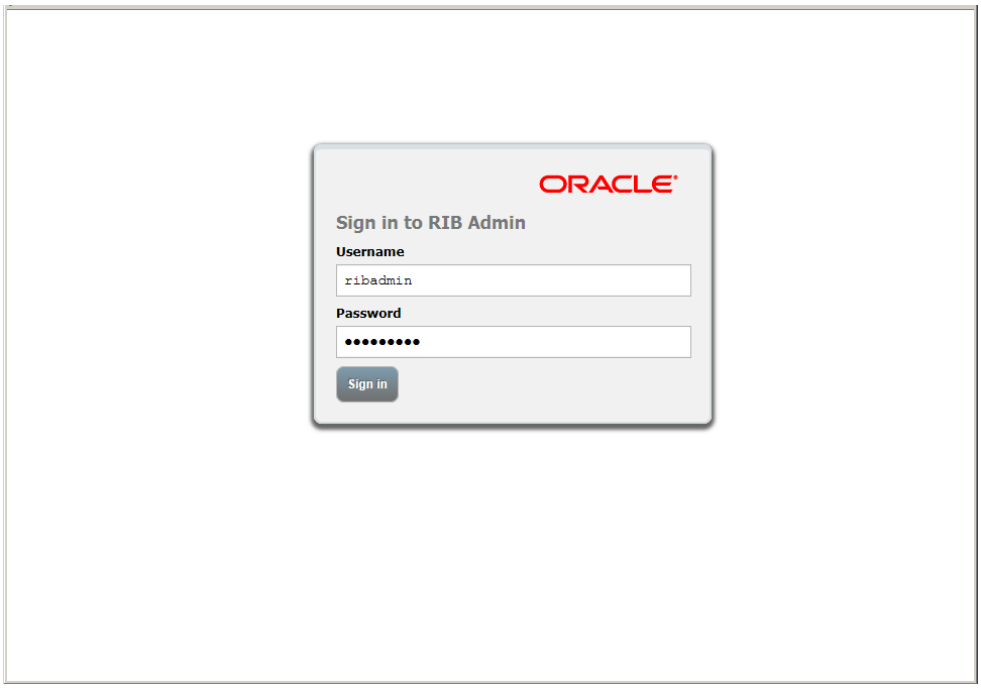
- Launch the following URL's to make sure the Index of rib-func-artifact page is accessible.
  - http://ribhost-vip.us.oracle.com:7777/rib-func-artifact



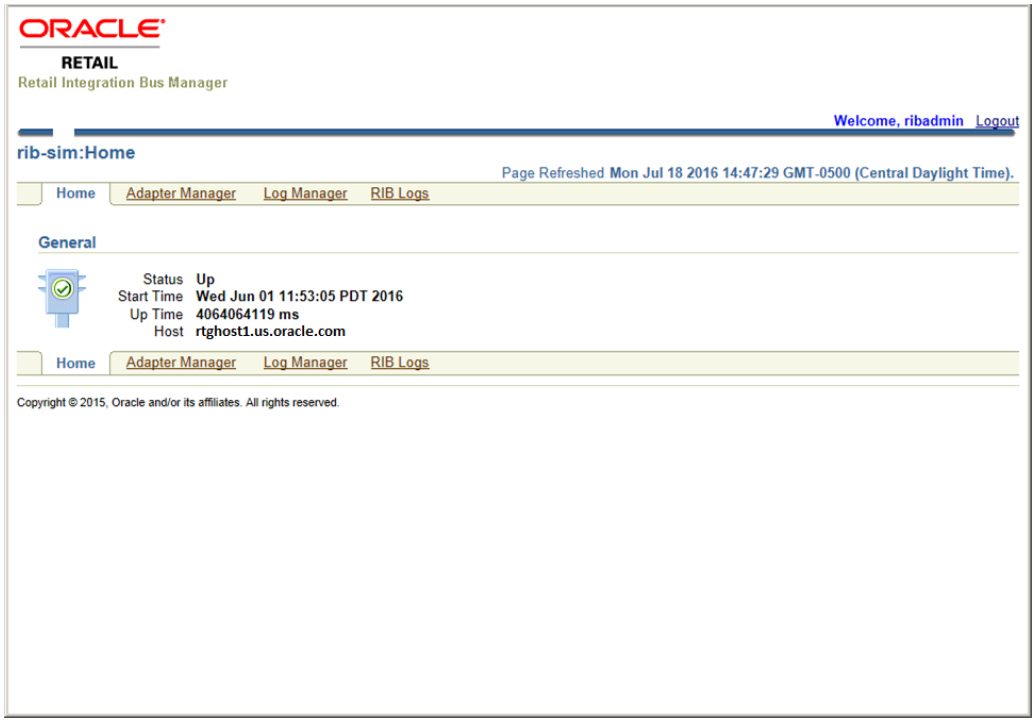
Index of /rib-func-artifact/

Name	Last Modified	Size
Parent Directory		
.classinfos	16-05-2016 09:36	<DIR>
.faces	16-05-2016 09:36	<DIR>
.taghandlers	16-05-2016 09:36	<DIR>
.taglisteners	16-05-2016 09:36	<DIR>
.tlds	16-05-2016 09:36	<DIR>
integration	19-11-2015 10:00	<DIR>
payload	19-11-2015 10:00	<DIR>
.beamarker.dat	27-05-2016 12:18	1
.beamarker.dat	07-05-2014 24:05	1

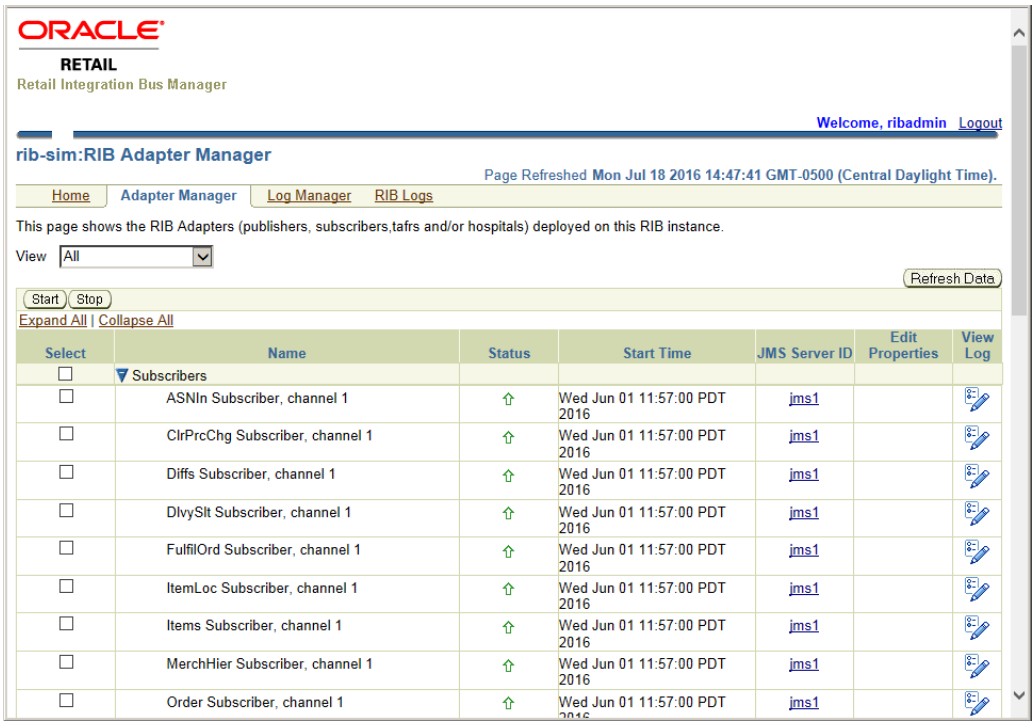
- Launch each of the following URL's to make sure the RIB applications are accessible and log into the RIB Admin GUI screens.
  - http://ribhost-vip.us.oracle.com:7777/rib-rms-admin-gui/index.jsp
  - http://ribhost-vip.us.oracle.com:7777/rib-rpm-admin-gui/index.jsp
  - http://ribhost-vip.us.oracle.com:7777/rib-tafr-admin-gui/index.jsp
  - http://ribhost-vip.us.oracle.com:7777/rib-sim-admin-gui/index.jsp
  - http://ribhost-vip.us.oracle.com:7777/rib-oms-admin-gui/index.jsp



- The Home screen of the RIB is displayed. Click the **Adapter Manager** link to see if the adapters are up.



- The Adapter Manager window opens.



- Test is complete.

Update Web Tier for RTG and RSB Applications

- Perform the following steps to configure the Web Tier to manage the RTG and RIB applications connections.
  On rtghost1 – set the ORACLE_INSTANCE variable so that the Web Tier opmnctl commands can be used to start and stop Web Tier's opmn processes.

  Example:

  ```
  $ export
  ORACLE_INSTANCE=/u00/webadmin/products/wls_ohs/Oracle_WT1/instances/instan
  ce1/
  ```

- On rtghost2 – set the ORACLE_INSTANCE variable so that the Web Tier opmnctl commands can be used to start and stop Web Tier's opmn processes.

  Example:

  ```
  $ export
  ORACLE_INSTANCE=/u00/webadmin/products/wls_ohs/Oracle_WT1/instances/instan
  ce2/
  ```

- Stop the opmn processes for Web Tier on both rtghost1 and rtghost2 by navigating to the <WEB_TIER_WLS_HOME>/ Oracle_WT1/opmn/bin directory and running the opmnctl stopall command.

  Example:

  ```
  $ cd /u00/webadmin/products/wls_ohs/Oracle_WT1/opmn/bin
  $ opmnctl stopall
  ```

- Update the mod_wl_ohs.conf file to include location URL redirects for the RTG and RSB applications. The examples show the location of the mod_wl_ohs.conf file on each server. Entries will need to be made for the following: (See Appendix IV RSB and RTG mod_wl_ohs.conf for example of entries.)
  - RSB Console
  - RSB Application Decorators
  - RIC Console
  - JMS Console
  - RSE
  - RMS Webservices
  - IGS Webservices

  Example rtghost1 :

  ```
  $ cd
  /u00/webadmin/products/wls_ohs/Oracle_WT1/instances/instance1/config/OHS/o
  hs1
  $ vi mod_wl_ohs.conf
  ```

  Example rtghost2 :

  ```
  $ cd
  /u00/webadmin/products/wls_ohs/Oracle_WT1/instances/instance2/config/OHS/o
  hs2
  $ vi mod_wl_ohs.conf
  ```

- Restart the Web Tier opmn processes on rtghost1 and rtghost2.

  Example:

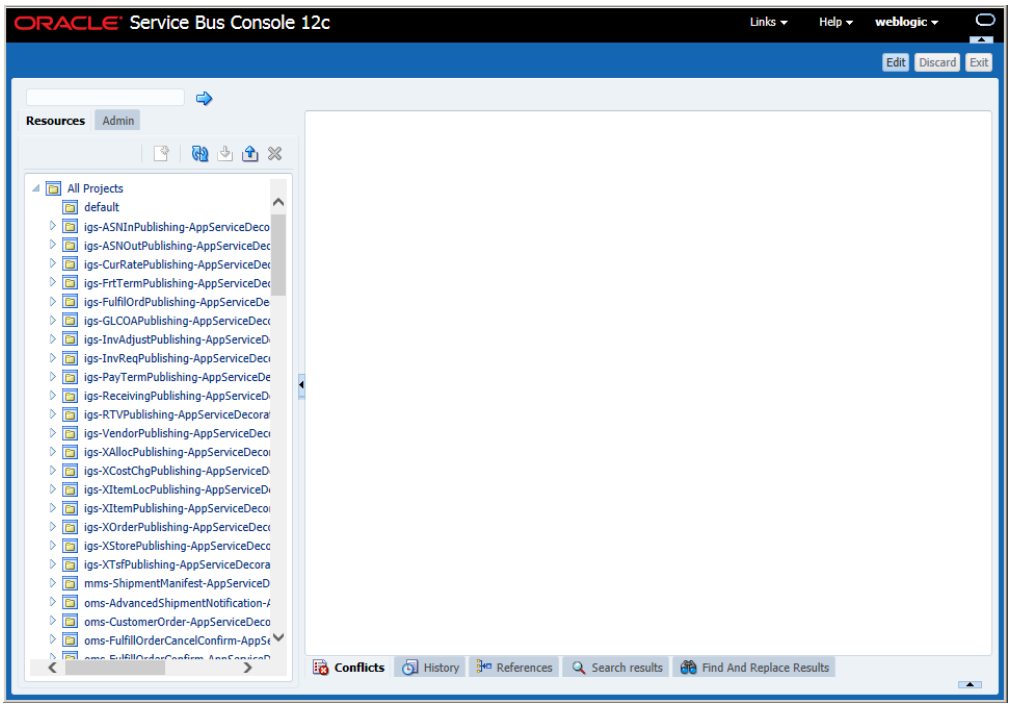
  ```
  $ cd /u00/webadmin/products/wls_ohs/Oracle_WT1/opmn/bin
  $ opmnctl startall
  ```

Verification - Verify Web Tier Configuration for RSB

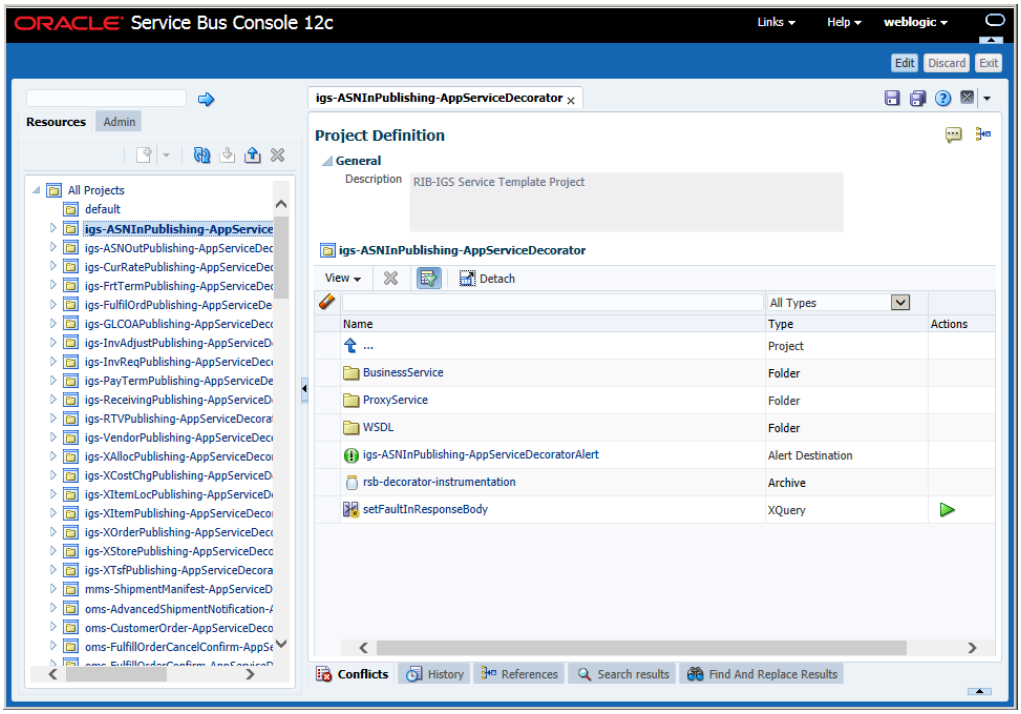
Perform the following steps to validate that the RSB Console can be accessed via URL's made up of the virtual host alias (VIP) and Web Tier port.

- Launch the following URL in a web browser to validate the RSB Console and AppServiceDecorator configuration in Web Tier.
  - http://rtghost-vip.us.oracle.com:7777/sbconsole

- The Service Bus Console is displayed. Click on the igs-ASNInPublishing-AppServiceDecorator link located in the navigation pane on the left side of the screen.



- The screen should be displayed with details similar to the screen shot below.

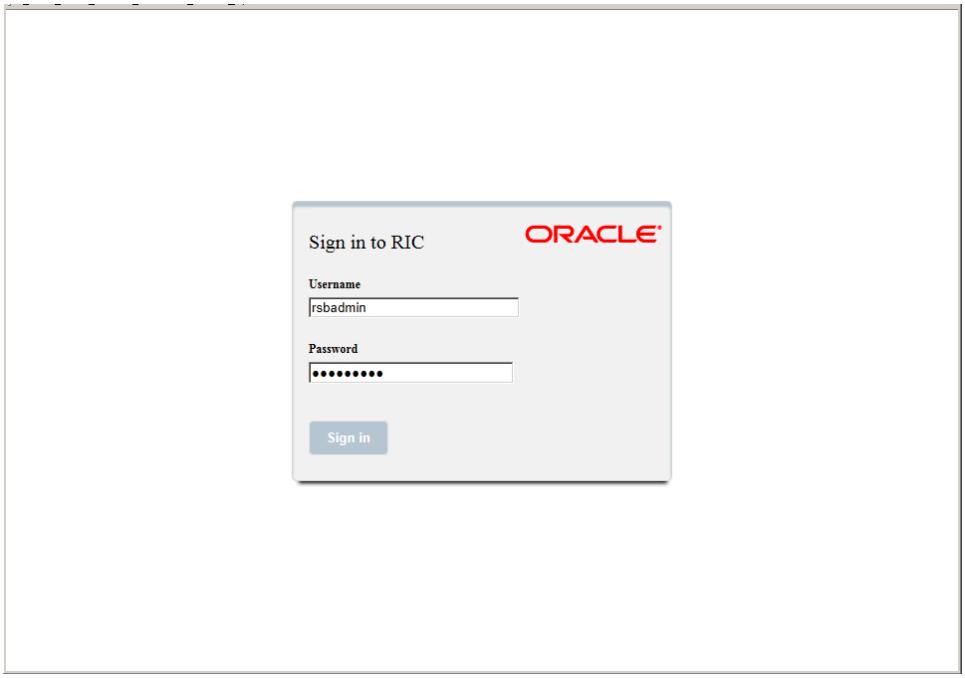


- Test is complete.

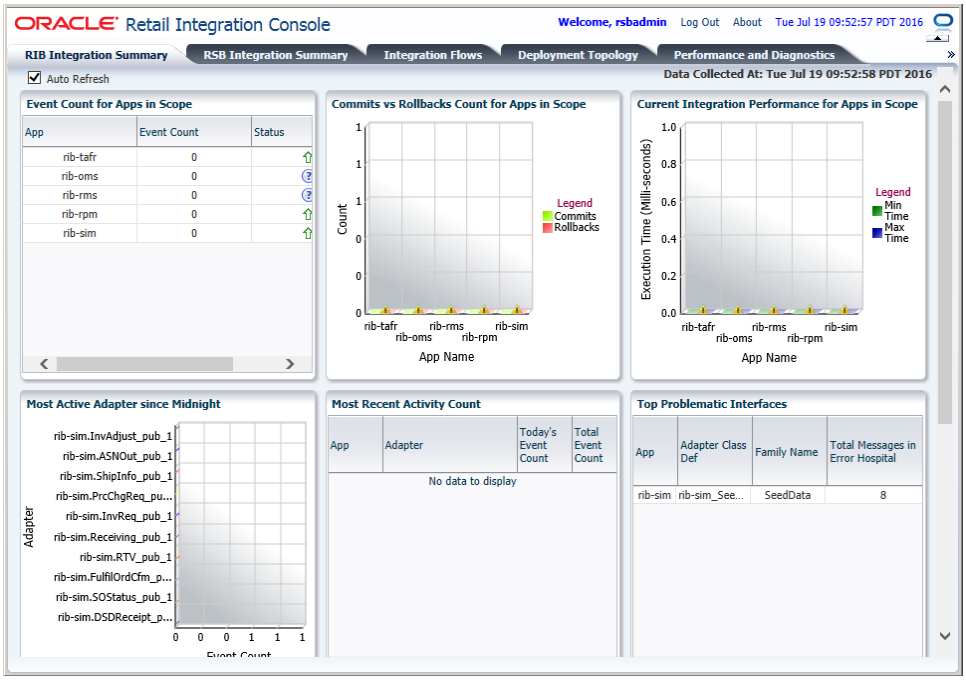
Verification - Verify Web Tier Configuration for RIC

Perform the following steps to validate that the RIC Console can be accessed via URL's made up of the virtual host alias (VIP) and Web Tier port.

- Launch the following URL in a web browser to validate the RIC Console configuration in Web Tier.
  - http://rtghost-vip.us.oracle.com:7777/rsb-admin



- The Retail Integration Console is displayed and looks similar to the screen shot below with details pertaining to the RIB and RSB installation configurations.



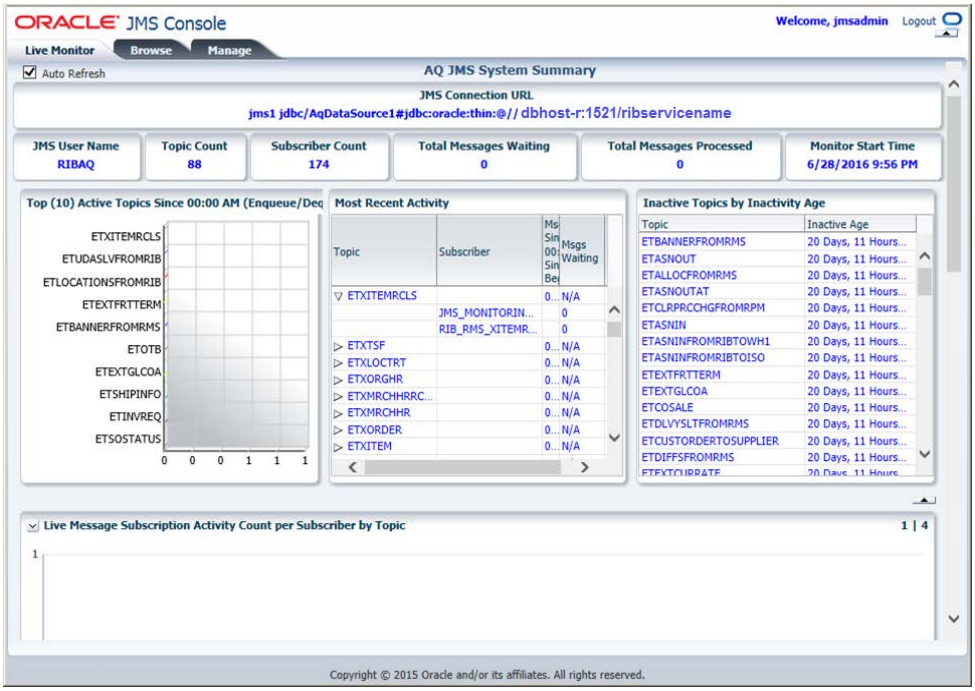
- Test is complete.

Verification - Verify Web Tier Configuration for JMS

Perform the following steps to validate that the JMS Console can be accessed via URL's made up of the virtual host alias (VIP) and Web Tier port.

- Launch the following URL in a web browser to validate the JMS Console configuration in Web Tier.
  - http://rtghost-vip.us.oracle.com:7777/jms-console

- The JMS Console is displayed and looks similar to the screen shot.



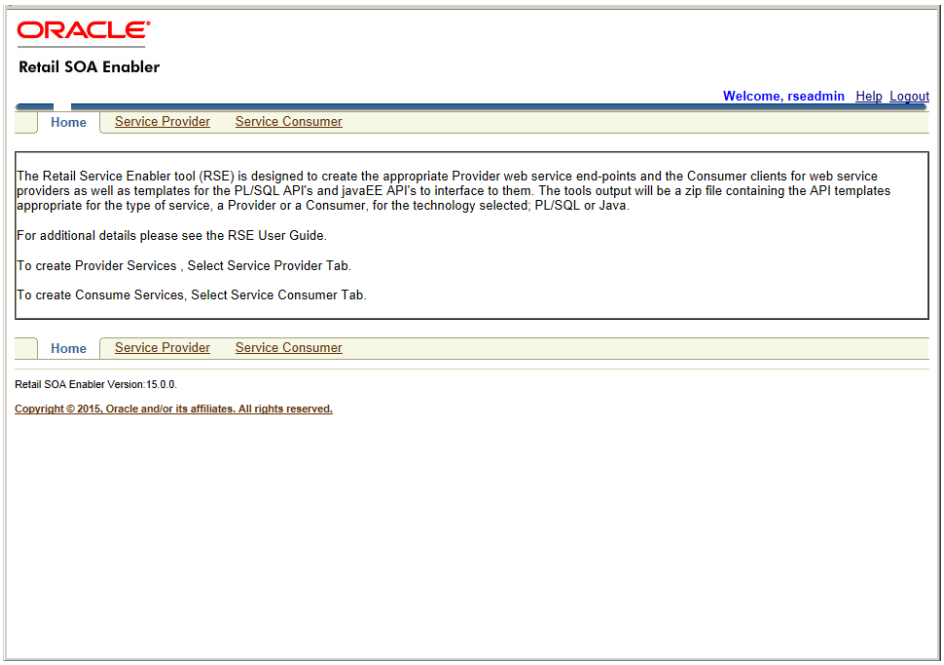
- Test is complete.

Verification - Verify Web Tier Configuration for RSE

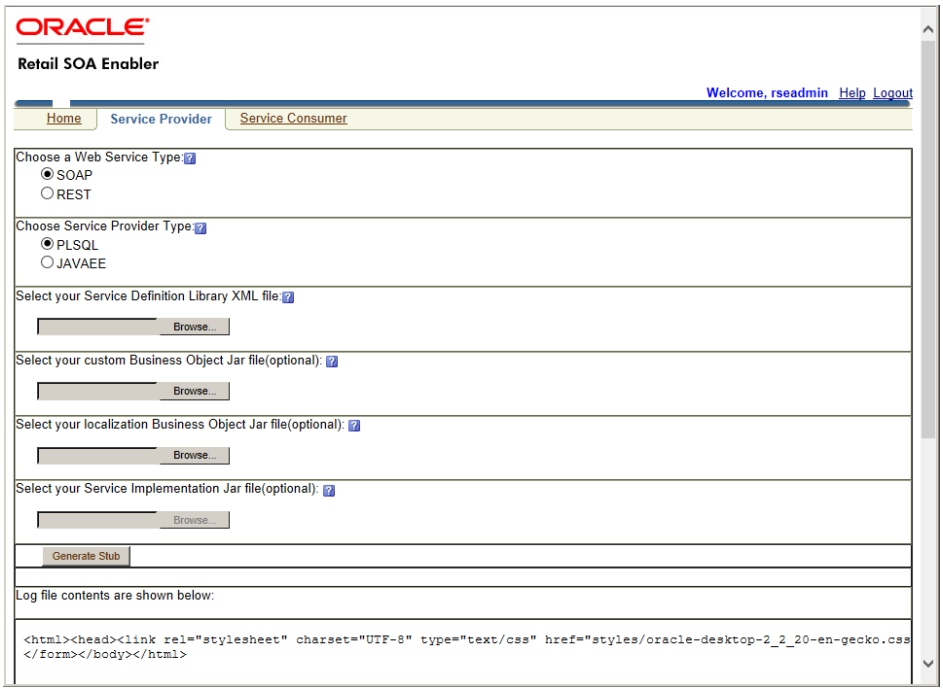
Perform the following steps to validate that the Retail SOA Enabler (RSE) can be accessed via URL's made up of the virtual host alias (VIP) and Web Tier port.

- Launch the following URL in a web browser to validate the RSE configuration in Web Tier.
  - http://rtghost-vip.us.oracle.com:7777/retail-soa-enabler-gui-15.0.0/index.jsp

- The Home page for RSE is displayed. Click on the Service Provider tab.



- The RSE Service Provider screen is displayed and looks similar to the screen shot below.



- Test is complete.

Verification - Verify Web Tier Configuration for IGS Web Services

Perform the following steps to validate that IGS WSDL's can be accessed via URL's made up of the virtual host alias (VIP) and Web Tier port.

- Launch the following URL in a web browser to validate the IGS Web Service configuration in Web Tier.
  - http://rtghost-vip.us.oracle.com:7777/ASNInPublishingBean/ASNInPublishingService?WSDL

- Test complete.

Verification - Verify Web Tier Configuration for RMS Web Services

Perform the following steps to validate that RMS WSDL's can be accessed via URL's made up of the virtual host alias (VIP) and Web Tier port.

- Launch the following URL in a web browser to validate the RMS Web Service configuration in Web Tier.
  - http://rtghost-vip.us.oracle.com:7777/SupplierBean/SupplierService?WSDL


```
<?xml version="1.0" encoding="UTF-8"?>
<!-- Published by JAX-WS RI (http://jax-ws.java.net). RI's version is JAX-WS RI 2.2.10-b140319.1121 svn-
revision#7b34660d6bc70e587e8ec81a17e4d76af68816a6. -->
<!-- Generated by JAX-WS RI (http://jax-ws.java.net). RI's version is JAX-WS RI 2.2.10-b140319.1121 svn-
revision#7b34660d6bc70e587e8ec81a17e4d76af68816a6. -->
+ <definitions xmlns="http://schemas.xmlsoap.org/wsdl/" name="SupplierService"
targetNamespace="http://www.oracle.com/retail/rms/integration/services/SupplierService/v1"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:tns="http://www.oracle.com/retail/rms/integration/services/SupplierService/v1"
xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/" xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"
xmlns:wsam="http://www.w3.org/2007/05/addressing/metadata"
xmlns:wsp1_2="http://schemas.xmlsoap.org/ws/2004/09/policy" xmlns:wsp="http://www.w3.org/ns/ws-policy"
xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd">
```


- Test is complete.

A

Appendix: rsb-development-env-info.properties

```
##############################################################################
#
# Set your java home.
#
##############################################################################
JAVA_HOME=/u00/webadmin/products/jdk_java
##############################################################################
#
# Define the apps that are in scope for this installation. The service
# provider and service consumer apps together makes up the final apps in scope.
#
##############################################################################
rsb-deployment-env-info.service-provider-app-in-scope-for-
integration=rms,igs,sim,oms,mms,rpm
rsb-deployment-env-info.service-requester-app-in-scope-for-
integration=sim,rpm,mms,reim,oms,rms,rib4oms


##############################################################################
#
# Define your rsb weblogic and osb server information. This weblogic info must
# be the one that is running the osb container.
#
##############################################################################

# Domain level info.
rsb-osb-container.domain-name=RSBDomain
rsb-osb-container.RSBDomain.home= /u00/webadmin/config/domains/wls_rsb/RSBDomain

# Cluster level info
rsb-osb-container.RSBDomain.cluster-name=rsb_cluster
rsb-osb-container.RSBDomain.rsb_cluster.http-
url=http://rtghost1.us.oracle.com:7777



##############################################################################
#################################
# Define the HTTPS URL if this installation would use POLICY-A (SSL) security
configuration.
# This is a "Required" field for POLICY-A configuration, For non-POLICY-A setup,
this is "Optional".
# If POLICY-A is not being used in this installation, then you may skip this
property
#
#
# rsb-osb-container.rsb_domain.c1.https-url=https://<your-host>:<your-cluster-
https-port>
# Eg:rsb-osb-container.rsb_domain.c1.https-url=https://rsbhost.example.com:19708
```

```
##################################################################################
#################################


# AdminServer level info
rsb-osb-container.RSBDomain.admin-server-name=RSB_AdminServer
rsb-osb-container.RSBDomain.admin-server-http-
url=http://rtghost1.us.oracle.com:7061
rsb-osb-container.RSBDomain.admin-server-connection-
url=t3://rtghost1.us.oracle.com:7061

# Managed server level info
rsb-osb-container.RSBDomain.rsb_cluster.managed-servers=rsb_server1,rsb_server2
rsb-osb-container.RSBDomain.rsb_cluster.rsb_server1.managed-server-connection-
url=t3://rtghost1.us.oracle.com:7063
rsb-osb-container.RSBDomain.rsb_cluster.rsb_server2.managed-server-connection-
url=t3://rtghost2.us.oracle.com:7065

##################################################################################
#
# Define your rsb database server information. This db is the one that rsb
# uses to keep its internal states.
#
# E.g. For Oracle database
#   service-infrastructure-db.jdbc-
url=jdbc:oracle:thin:@dbhost.example.com:1521:orcl
#
#      For Derby database
#   service-infrastructure-db.jdbc-url=jdbc:derby:rsb-
sidb;create=true;databaseName=rsb-sidb
#
##################################################################################
service-infrastructure-db.jdbc-url=jdbc:oracle:thin:@(DESCRIPTION =(ADDRESS =
(PROTOCOL = TCP)(HOST = dbhost-r)(PORT = 1521))(CONNECT_DATA =(SERVICE_NAME =
ribservicename)))


##################################################################################
#
# Define application server info where application services are running. These
# servers are where your retail business apps are installed and not where rsb/osb
# is installed.
#
# The pattern of this property is
#      edge-app-container.<app>.connection-url=http://<host>:<port>
#    where:
#      <app> - is the retail app name. E.g. rms, sim etc.
#      <host>- is the host where retail apps are running.
#      <port>- is the t3 protocol port where retail apps are running.
#
##################################################################################
edge-app-container.sim.connection-
url=t3://apphost1.us.oracle.com,apphost2.us.oracle.com:7143

edge-app-container.igs.connection-
url=t3://rtghost1.us.oracle.com,rtghost2.us.oracle.com:7033

edge-app-container.rms.connection-
url=t3://apphost1.us.oracle.com,apphost2.us.oracle.com:7123

edge-app-container.rpm.connection-
url=t3://apphost1.us.oracle.com,apphost2.us.oracle.com:7133
```

```
#edge-app-container.cm.connection-url=t3://rsbhost.example.com:7001

edge-app-container.mms.connection-
url=t3://rtghost1.us.oracle.com,rtghost2.us.oracle.com:7777

#edge-app-container.ooc.connection-url=t3://rsbhost.example.com:7001

#edge-app-container.pos.connection-url=t3://rsbhost.example.com:7001

edge-app-container.reim.connection-
url=t3://apphost1.us.oracle.com,apphost2.us.oracle.com:7093

edge-app-container.oms.connection-
url=t3://rtghost1.us.oracle.com,rtghost2.us.oracle.com:7777

#edge-app-container.fin.connection-url=t3://rsbhost.example.com:7001

#edge-app-container.rpm.connection-url=t3://rsbhost.example.com:7001

#edge-app-container.rwms.connection-url=t3://rsbhost.example.com:7001

#edge-app-container.rm.connection-url=t3://rsbhost.example.com:7001

##################################################################################
##################################
#
# The app service endpoint is derived by looking at this global endpoint url
# pattern property. The <HTTP_HOSTNAME>, <HTTP_PORT> and <SERVICE_NAME> gets
# replaced with actual host, port values from property
# edge-app-container.<app>.connection-url=t3://<host>:<port> defined above.
#
##################################################################################
##################################
global.app-service-end-point-url-
pattern=http://<HTTP_HOSTNAME>:<HTTP_PORT>/<SERVICE_NAME>Bean/<SERVICE_NAME>Servic
e


##################################################################################
##################################
# If you want to override app service endpoint url that gets generated at global
# level use the following url pattern.
# <app>.app-service-end-point-url-
pattern=http://<HTTP_HOSTNAME>:<HTTP_PORT>/<SERVICE_NAME>Bean/<SERVICE_NAME>Servic
e
#
# This is an optional property, define only if you have to override.
#
##################################################################################
##################################


##################################################################################
##################################
# If you want to override individule service endpoint url that gets generated at
global
# level and app level use the following url pattern.
# <decorator>.app-service-end-point-
url=http://rsbhost.example.com:7001/SupplierBean/SupplierService
#   E.g. rms-Supplier-AppServiceDecorator.app-service-end-point-
url=http://rsbhost.example.com:7001/SupplierHOHOBean/SupplierService
#
# This is an optional property, define only if you have to override.
```

```
#
##################################################################################
#################################


##################################################################################
#################################
#
# If you want to make rsb aware of rib you will have to point this(rib.home.path)
# property to a valid rib-home folder.
#
# This is an optional property, if you don't define it rib will be regarded out
# of scope.
# EX:
rib.home.path=webadmin@ribhost.example.com:/home/webadmin/RIB_INSTALL/Rib1400ForAl
l14xxApps/rib-home

##################################################################################
#################################
rib.home.path=webadmin@rtghost1.us.oracle.com:/u00/webadmin/RIB_INSTALL/Rib1500For
All15xxApps/rib-home
```

B

Appendix: ric-deployment-env-info.json

```
{
    "RicDeploymentEnvInfo":{
        "DataSourceDef":{
            "RicDataSource":{
                "dataSourceName":"RicDataSource",
                "dataSourceClass":"oracle.jdbc.pool.OracleDataSource",
                "dataSourceJndiName":"jdbc/RicDataSource",
                "jdbcUrl":"jdbc:oracle:thin:@//dbhost-r:1521/DOLRP04RIB",
                "jdbcUserAlias":"ricDataSourceUserAlias",
                "jdbcUser":"GET_FROM_WALLET",
                "jdbcPassword":"GET_FROM_WALLET"
            }

        },
        "MiddlewareServerDef":{
            "RicAppServer":{

                "weblogicDomainName":"RSBDomain",

"weblogicDomainHome":"/u00/webadmin/config/domains/wls_rsb/RSBDomain",
                "weblogicDomainAdminServerUrl":"t3://rtghost1.us.oracle.com:7061",
                "weblogicDomainAdminServerProtocol":"t3",
                "weblogicDomainAdminServerHost":"rtghost1.us.oracle.com",
                "weblogicDomainAdminServerPort":"7061",

"weblogicDomainAdminServerUserAlias":"ricAppServerAdminServerUserAlias",
                "weblogicDomainTargetManagedServerName":"RSB_AdminServer",

                "ricUiUrl":"http://rtghost1.us.oracle.com:7061/rsb-admin",
                "ricUiUserGroup":"RicAdminGroup",
                "ricUiUserAlias":"ricAdminUiUserAlias",
                "ricUiUser":"GET_FROM_WALLET",
                "ricUiPassword":"GET_FROM_WALLET",

            }
        },
        "IntegrationProduct":{
            "ribEnable":"true",
            "rsbEnable":"true",
            "ddiEnable":"true",

"ribHome":"webadmin@rtghost1.us.oracle.com:/u00/webadmin/RIB_INSTALL/Rib1500ForAll
15xxApps/rib-home",
            "rsbHome":"/u00/webadmin/RSB_INSTALL/Rsb1500ForAll15xxApps/rsb-home",
            "ddiHome":"/u00/webadmin/RSB_INSTALL/Rsb1500ForAll15xxApps/rsb-home"
        },
        "RicApplication":{
            "ricAppUses":[
                "RicDataSource",
                "RicAppServer"
            ]
        }
    }
}
```

C

Appendix: RIB mod_wl_ohs.conf

```
# NOTE : This is a template to configure mod_weblogic.

LoadModule weblogic_module   "${ORACLE_HOME}/ohs/modules/mod_wl_ohs.so"

# This empty block is needed to save mod_wl related configuration from EM to this
file when changes are made at the Base Virtual Host Level
<IfModule weblogic_module>
#      WebLogicHost <WEBLOGIC_HOST>
#      WebLogicPort <WEBLOGIC_PORT>
#      Debug ON
#      WLLogFile /tmp/weblogic.log
#      MatchExpression *.jsp
</IfModule>

# <Location /weblogic>
#      SetHandler weblogic-handler
#      PathTrim /weblogic
#      ErrorPage  http:/WEBLOGIC_HOME:WEBLOGIC_PORT/
#  </Location>


<Location /rib-func-artifact>
SetHandler weblogic-handler
WebLogicHost 10.141.32.77
WebLogicPort 7003
</Location>

<Location /rib-oms-admin-gui>
SetHandler WebLogic-handler
WebLogicHost 10.141.32.77
WebLogicPort 7019
</Location>

<Location /rib-rms-admin-gui>
SetHandler weblogic-handler
WebLogicHost 10.141.32.77
WebLogicPort 7005
</Location>

<Location /rib-sim-admin-gui>
 SetHandler weblogic-handler
 WebLogicHost 10.141.32.77
 WebLogicPort 7013
</Location>


<Location /rib-rpm-admin-gui>
SetHandler weblogic-handler
WebLogicHost 10.141.32.77
WebLogicPort 7009
</Location>

<Location /rib-tafr-admin-gui>
SetHandler WebLogic-handler
```

```
WebLogicHost 10.141.32.77
WebLogicPort 7017
</Location>

<LocationMatch ^/bea_wls_internal/>
SetHandler weblogic-handler
WebLogicHost 10.141.32.77
WebLogicPort 7003
</LocationMatch>

<LocationMatch ^/bea_wls_internal/>
SetHandler weblogic-handler
WebLogicHost 10.141.32.77
WebLogicPort 7019
</LocationMatch>

<LocationMatch ^/bea_wls_internal/>
SetHandler weblogic-handler
WebLogicHost 10.141.32.77
WebLogicPort 7005
</LocationMatch>

<LocationMatch ^/bea_wls_internal/>
SetHandler weblogic-handler
WebLogicHost 10.141.32.77
WebLogicPort 7013
</LocationMatch>

<LocationMatch ^/bea_wls_internal/>
SetHandler weblogic-handler
WebLogicHost 10.141.32.77
WebLogicPort 7009
</LocationMatch>

<LocationMatch ^/bea_wls_internal/>
SetHandler weblogic-handler
WebLogicHost 10.141.32.77
WebLogicPort 7017
</LocationMatch>
```

D

Appendix: RSB and RTG mod_wl_ohs.conf

```
# NOTE : This is a template to configure mod_weblogic.

LoadModule weblogic_module   "${ORACLE_HOME}/ohs/modules/mod_wl_ohs.so"
#MatchExpression "*wsdl" WebLogicHost=rtghost1.us.oracle.com|WebLogicPort 7067
# This empty block is needed to save mod_wl related configuration from EM to this
file when changes are made at the Base Virtual Host Level
<IfModule weblogic_module>
#      WebLogicHost <WEBLOGIC_HOST>
#      WebLogicPort <WEBLOGIC_PORT>
#      Debug ON
#      WLLogFile /tmp/weblogic.log
#      MatchExpression *.jsp
</IfModule>

# <Location /weblogic>
#      SetHandler weblogic-handler
#      PathTrim /weblogic
#      ErrorPage  http:/WEBLOGIC_HOME:WEBLOGIC_PORT/
#  </Location>

####### RSB Applications #######
<LocationMatch [*AppServiceDecorator*] >
SetHandler weblogic-handler
WebLogicCluster rtghost1.us.oracle.com:7063,rtghost2.us.oracle.com:7065
</LocationMatch>

<Location /sbconsole >
SetHandler weblogic-handler
WebLogicHost rtghost1.us.oracle.com
WebLogicPort 7061
</Location>

<Location /servicebus>
SetHandler weblogic-handler
WebLogicHost rtghost1.us.oracle.com
WebLogicPort 7061
</Location>

####### RIC Application #######
<Location /rsb-admin >
SetHandler weblogic-handler
WebLogicHost rtghost1.us.oracle.com
WebLogicPort 7061
</Location>

####### JMS Application #######
<LocationMatch /jms-console >
SetHandler weblogic-handler
WebLogicCluster rtghost1.us.oracle.com:7047,rtghost2.us.oracle.com:7047
</LocationMatch>

####### RSE Application #######
<LocationMatch /retail-soa-enabler-gui-15.0.0 >
SetHandler weblogic-handler
WebLogicCluster rtghost1.us.oracle.com:7035,rtghost2.us.oracle.com:7035
```

```
</LocationMatch>

####### IGS WSDL #######
<LocationMatch /ASNInPublishingBean >
SetHandler weblogic-handler
WebLogicCluster rtghost1.us.oracle.com:7033,rtghost2.us.oracle.com:7033
</LocationMatch>

<LocationMatch /ASNOutPublishingBean >
SetHandler weblogic-handler
WebLogicCluster rtghost1.us.oracle.com:7033,rtghost2.us.oracle.com:7033
</LocationMatch>

<LocationMatch /CurRatePublishingBean >
SetHandler weblogic-handler
WebLogicCluster rtghost1.us.oracle.com:7033,rtghost2.us.oracle.com:7033
</LocationMatch>

<LocationMatch /FrtTermPublishingBean >
SetHandler weblogic-handler
WebLogicCluster rtghost1.us.oracle.com:7033,rtghost2.us.oracle.com:7033
</LocationMatch>

<LocationMatch /FulfilOrdPublishingBean >
SetHandler weblogic-handler
WebLogicCluster rtghost1.us.oracle.com:7033,rtghost2.us.oracle.com:7033
</LocationMatch>

<LocationMatch /GLCOAPublishingBean >
SetHandler weblogic-handler
WebLogicCluster rtghost1.us.oracle.com:7033,rtghost2.us.oracle.com:7033
</LocationMatch>

<LocationMatch /IgsInfrastructureManagerBean >
SetHandler weblogic-handler
WebLogicCluster rtghost1.us.oracle.com:7033,rtghost2.us.oracle.com:7033
</LocationMatch>

<LocationMatch /InvAdjustPublishingBean >
SetHandler weblogic-handler
WebLogicCluster rtghost1.us.oracle.com:7033,rtghost2.us.oracle.com:7033
</LocationMatch>

<LocationMatch /InvReqPublishingBean >
SetHandler weblogic-handler
WebLogicCluster rtghost1.us.oracle.com:7033,rtghost2.us.oracle.com:7033
</LocationMatch>

<LocationMatch /PayTermPublishingBean >
SetHandler weblogic-handler
WebLogicCluster rtghost1.us.oracle.com:7033,rtghost2.us.oracle.com:7033
</LocationMatch>

<LocationMatch /ReceivingPublishingBean >
SetHandler weblogic-handler
WebLogicCluster rtghost1.us.oracle.com:7033,rtghost2.us.oracle.com:7033
</LocationMatch>

<LocationMatch /RTVPublishingBean >
SetHandler weblogic-handler
WebLogicCluster rtghost1.us.oracle.com:7033,rtghost2.us.oracle.com:7033
</LocationMatch>
```

```
<LocationMatch /VendorPublishingBean >
SetHandler weblogic-handler
WebLogicCluster rtghost1.us.oracle.com:7033,rtghost2.us.oracle.com:7033
</LocationMatch>

<LocationMatch /XAllocPublishingBean >
SetHandler weblogic-handler
WebLogicCluster rtghost1.us.oracle.com:7033,rtghost2.us.oracle.com:7033
</LocationMatch>

<LocationMatch /XCostChgPublishingBean >
SetHandler weblogic-handler
WebLogicCluster rtghost1.us.oracle.com:7033,rtghost2.us.oracle.com:7033
</LocationMatch>

<LocationMatch /XItemLocPublishingBean >
SetHandler weblogic-handler
WebLogicCluster rtghost1.us.oracle.com:7033,rtghost2.us.oracle.com:7033
</LocationMatch>

<LocationMatch /XItemPublishingBean >
SetHandler weblogic-handler
WebLogicCluster rtghost1.us.oracle.com:7033,rtghost2.us.oracle.com:7033
</LocationMatch>

<LocationMatch /XOrderPublishingBean >
SetHandler weblogic-handler
WebLogicCluster rtghost1.us.oracle.com:7033,rtghost2.us.oracle.com:7033
</LocationMatch>

<LocationMatch /XStorePublishingBean >
SetHandler weblogic-handler
WebLogicCluster rtghost1.us.oracle.com:7033,rtghost2.us.oracle.com:7033
</LocationMatch>

<LocationMatch /XTsfPublishingBean >
SetHandler weblogic-handler
WebLogicCluster rtghost1.us.oracle.com:7033,rtghost2.us.oracle.com:7033
</LocationMatch>

####### RMS WSDL #######
<LocationMatch /CustomerCreditCheckBean >
SetHandler weblogic-handler
WebLogicCluster rtghost1.us.oracle.com:7037,rtghost2.us.oracle.com:7037
</LocationMatch>

<LocationMatch /CustOrdSubstituteBean >
SetHandler weblogic-handler
WebLogicCluster rtghost1.us.oracle.com:7037,rtghost2.us.oracle.com:7037
</LocationMatch>

<LocationMatch /FulfillOrderBean >
SetHandler weblogic-handler
WebLogicCluster rtghost1.us.oracle.com:7037,rtghost2.us.oracle.com:7037
</LocationMatch>

<LocationMatch /InventoryBackOrderBean >
SetHandler weblogic-handler
WebLogicCluster rtghost1.us.oracle.com:7037,rtghost2.us.oracle.com:7037
</LocationMatch>

<LocationMatch /InventoryDetailBean >
SetHandler weblogic-handler
```

```
WebLogicCluster rtghost1.us.oracle.com:7037,rtghost2.us.oracle.com:7037
</LocationMatch>

<LocationMatch /ItemManagementBean >
SetHandler weblogic-handler
WebLogicCluster rtghost1.us.oracle.com:7037,rtghost2.us.oracle.com:7037
</LocationMatch>

<LocationMatch /PayTermBean >
SetHandler weblogic-handler
WebLogicCluster rtghost1.us.oracle.com:7037,rtghost2.us.oracle.com:7037
</LocationMatch>

<LocationMatch /PricingCostBean >
SetHandler weblogic-handler
WebLogicCluster rtghost1.us.oracle.com:7037,rtghost2.us.oracle.com:7037
</LocationMatch>

<LocationMatch /PurchaseOrderManagementBean >
SetHandler weblogic-handler
WebLogicCluster rtghost1.us.oracle.com:7037,rtghost2.us.oracle.com:7037
</LocationMatch>

<LocationMatch /ReportLocatorBean >
SetHandler weblogic-handler
WebLogicCluster rtghost1.us.oracle.com:7037,rtghost2.us.oracle.com:7037
</LocationMatch>

<LocationMatch /RmsInfrastructureManagerBean >
SetHandler weblogic-handler
WebLogicCluster rtghost1.us.oracle.com:7037,rtghost2.us.oracle.com:7037
</LocationMatch>

<LocationMatch /StoreOrderBean >
SetHandler weblogic-handler
WebLogicCluster rtghost1.us.oracle.com:7037,rtghost2.us.oracle.com:7037
</LocationMatch>

<LocationMatch /SupplierBean >
SetHandler weblogic-handler
WebLogicCluster rtghost1.us.oracle.com:7037,rtghost2.us.oracle.com:7037
</LocationMatch>
```

Oracle Corporation
World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065
U.S.A.

Worldwide Inquiries:
Phone: +1.650.506.7000
Fax: +1.650.506.7200
oracle.com

Copyright © 2016, Oracle. All rights reserved.
All rights reserved. This document is provided for information purposes only, and the contents hereof are subject to change without notice. This document is not warranted to be error-free, nor subject to any other warranties or conditions, whether expressed orally or implied in law, including implied warranties and conditions of merchantability or fitness for a particular purpose. We specifically disclaim any liability with respect to this document, and no contractual obligations are formed either directly or indirectly by this document. This document may not be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without our prior written permission.

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

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.