Oracle® Retail High-Availability Case Study Real Application Clusters Environment

Release 15.0
2151366.1

July 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 Creating a High-Availability Real Application Clusters (RAC) Environment for Oracle Retail 1

2 Oracle 12cR1 CRS Installation on Linux/x64 2

Node Configuration 2

3 Oracle Clusterware Installation 7

Clusterware Installation Steps 7

4 19Installing the Oracle Database 12c Release 2 (12.1.0.2) on Linux 19

Installing RDBMS 12.1.0.2 Database Software 19

5 Installing the Oracle Database 12c Release 2 (12.1.0.2) on Linux 27

Examples of Oracle Database 12c Release 1 27

JDK 1.7 and Oneoff Patch application 30

6 Create ASM Instances and Diskgroups 31

Create the Raw File Systems 31

Create the ASM Instance 31

7 Configuring SPFILES on Shared Devices with Raw File Systems 37

8 Create Database 39

Create a Container Database Using Database Configuration Assistant Interactively 39

Prerequisites 39

Instance Creation 39

Creating a Container Database Interactively Using the Database Configuration Assistant (DBCA) 39

Post Database Creation 48

Creating a Pluggable Database Interactively Using the Database Configuration Assistant (DBCA) 50

Verify Setup 55

OID Database Creation 55

9 Create the RMS Tablespace 57

Configure the Network Files 57

tnsnames.ora file 57

LISTENER.ora 58

A Appendix: Create the RMS Tablespace 59

1

Creating a High-Availability Real Application Clusters (RAC) Environment for Oracle Retail

Use this case study to create a high availability RAC environment for Oracle Retail release 15.0.

Installation Overview

Oracle Retail applications can be deployed in a grid computing environment, utilizing Oracle Application Server Active-Active clustering and Oracle RAC databases to provide high availability and scalability. This configuration allows lower-cost commodity systems to be combined into a single scalable, fault-tolerant computing environment.

There are many options for configuring an Oracle grid computing environment, and these options can have a significant impact on the way that 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 grid computing environment for Oracle Retail, but to document a specific configuration that was used for verification purposes. As such, the specific installation steps documented in this case study will only apply in their entirety to the exact hardware and software configuration described here.

This case study gives detailed steps that are used to create the high-availability RAC database used for Oracle Retail development:

1. Install Oracle RDBMS RAC environment
2. Install Oracle Cluster Ready Services Software
3. Install Oracle RDBMS Software
4. Install Example CD
5. Create ASM instance and the Diskgroups
6. Create a Container database using DBCA.
7. Create a Pluggable databases using DBCA.
8. Configure the network files.

> **Note:** This case study uses the hostnames dbserver1 and dbserver2 for the different systems in the database cluster. You can substitute to appropriate names of hosts during installation. In a few screenshots, you may see host names as msp00avy and msp00avz, which are the equivalent to dbserver1 and dbserver2.

2

Oracle 12cR1 CRS Installation on Linux/x64

This section describes the Oracle 12cR1 CRS installation on Linux/x86-64.

Node Configuration

To configure the node:

1. Verify that the RPMs and system settings are set up correctly. For more information, see Oracle Database (RDBMS) on Unix AIX, HP-UX, Linux, Mac OS X,Solaris,Tru64 Unix Operating Systems Installation and Configuration Requirements Quick Reference (8.0.5 to 12.1), My Oracle Support Note ID 1587357.1.
   a. Download the Remote Diagnostic Agent (RDA) from My Oracle Support Web site using ID 314422.1.
   b. Extract the RDA on each system of the cluster.
   c. Run /rda.sh –T hcve on each node.
   d. Validate using Oracle Database 12c R1 (12.1.0.2) Preinstall (Linux x86_64) option.
   e. Enter the used Oracle Home path.
   f. Review the output file and resolve any failures that are found:
      i. If the binutils version is earlier than 2.15.92.0.2-13.0.0.0.2, RDA may incorrectly claim that the verification has failed.
      ii. For some packages, both 64-bit and 32-bit RPMs are required, having only one or the other causes RDA to error out.
      iii. Use rpm -qa --qf %{NAME}_%{ARCH}-%{VERSION}-%{RELEASE}\n to list installed RPM names and their architectures (x86-64 or i386).
2. Have the System Administrator configure two gigabit interfaces: one for the public network (eg, eth0), and the other one for the cluster interconnect (e.g. eth1). The virtual IP address should not be assigned to an interface at this point.
3. Have the System Administrator specify the following IP addresses for eth0 and eth1 interfaces for the two cluster nodes. The interconnect IP addresses should be on a separate private subnet that is not registered with DNS. These should be defined in the /etc/hosts file on each node. The public and private interface names must be the same on each node.

   Example:

   ```
   dbserver1 (msp00avy)
   eth0 – ip: 10.177.121.6, netmask: 255.255.252.0 eth1
         – ip: 192.168.1.150, netmask: 255.255.255.0

   dbserver2 (msp00avz)
   eth0 – ip: 10.177.121.7, netmask: 255.255.252.0 eth1
         – ip: 192.168.1.151, netmask: 255.255.255.0
   ```
4. Have the System Administrator request for a Single Client Access Name (SCAN) name that will be used for three IP addresses in a round-robin fashion. The SCAN should be set up in the DNS server, and not in the host file. SCAN is the single point of access for all applications connecting to the database and is responsible for failover and load balancing a new Oracle Database 12c Release 1 feature. VIPs are still used

internally, but the initial connection is made through the SCAN via the remote_listener. You will be prompted for name during the CRS installation. Once the SCAN is setup, it can be verified using nslookup and can be pinged from both the nodes.

```
msp00avy:[sqlnet] /u00/oracle>
--> nslookup msp00avyp-r
Server:         10.141.28.50
Address:        10.141.28.50#53

Non-authoritative answer:
Name:   msp2805-r.us.oracle.com
Address: 10.141.28.30
Name:   msp2805-r.us.oracle.com
Address: 10.141.28.31
```

Server Details

Node1 : dbserver1

```
-->ping msp00avy
PING msp00avy.us.oracle.com (10.141.28.60) 56(84) bytes of data.
64 bytes from msp00avy.us.oracle.com (10.141.28.60): icmp_seq=1 ttl=64
time=0.028 ms
64 bytes from msp00avy.us.oracle.com (10.141.28.60): icmp_seq=2 ttl=64
time=0.064 ms
64 bytes from msp00avy.us.oracle.com (10.141.28.60): icmp_seq=3 ttl=64
time=0.047 ms
```

Node2 : dbserver2

```
--> ping msp00avz
PING msp00avz.us.oracle.com (10.141.28.61) 56(84) bytes of data.
64 bytes from msp00avz.us.oracle.com (10.141.28.61): icmp_seq=1 ttl=64
time=0.196 ms
64 bytes from msp00avz.us.oracle.com (10.141.28.61): icmp_seq=2 ttl=64
time=0.119 ms
```

Update the /etc/hosts file on the cluster nodes. All virtual IP addresses should be listed in the hosts file.

```
Example /etc/host file:
#
# Internet Hosts Table
#
127.0.0.1       localhost.localdomain   loghost
10.141.28.60    msp00avy.us.oracle.com  msp00avy
10.141.28.61    msp00avz.us.oracle.com  msp00avz
10.141.29.18    msp00avy-vip
10.141.29.19    msp00avz-vip
192.168.2.187   msp00avy-p
192.168.2.188   msp00avz-p
```

5. Verify that the eth0 interface is connected to the switch, and communicates with the storage. This can be achieved by pinging the storage:

   Example:

   ```
   $ /usr/sbin/ping mspnas402
   ```

   Note: Run ‘df –k’ to check if the file systems are present.

6. The other interface, eth1 on dbserver1, is able to communicate with its counterpart, eth1 on dbserver2. Use the /usr/sbin/ping command to verify the cluster interconnect from both the cluster nodes using the private names (for example, dbserver1p and dbserver2p).

Example:

$ /usr/sbin/ping dbserver1p

$ /usr/sbin/ping dbserver2p

7. Verify network configuration by pinging dbserver1 from dbserver2 and vice versa.

   Example:

   ping dbserver1

   ping dbserver2

   > **Note:** It is not possible to ping the virtual IP's (dbvip1 and dbvip2) until after the clusterware is installed and running smoothly.

8. Verify the Oracle user and rgbudba group have been created on each host.
9. Determine where the Oracle software is installed.
   - Oracle Base (ORACLE_BASE)
   - Oracle CRS Home (ORA_CRS_HOME)
   - Oracle Home (ORACLE_HOME)

This particular installation uses the following local directories on each node:

ORACLE_BASE	/u00/oracle
ORA_CRS_HOME	/u00/crs/oracle/product/12.1.0.2
ORACLE_HOME	/u00/oracle/product/12.1.0.2
Oracle Inventory	/u00/oraInventory
Oracle User's home directory	/u00/oracle

Create the directories:

a. Oracle Base

   mkdir -p /u00/oracle

b. Oracle CRS Home

   mkdir –p /u00/crs/oracle/product/12.1.0.2

c. Oracle Software Home

   mkdir –p /u00/oracle/product/12.1.0.2

10. Verify the following mount options in the /etc/fstab file. Mount the exported file systems from mspnas402.

    For example:/u02/ocr1, /u02/ocr2, /u02/arch, /u03, /u04, and /u05.

    mspnas402:/export/msp00avy-88/ocr1 /u02/ocr1 nfs hard,nointr,rsize=32768,wsize=32768,tcp,actimeo=0,timeo=600

    mspnas402:/export/msp00avy-88/ocr2 /u02/ocr2 nfs hard,nointr,rsize=32768,wsize=32768,tcp,actimeo=0,timeo=600

    mspnas402:/export/msp00avy-88/ocr3 /u02/ocr3 nfs hard,nointr,rsize=32768,wsize=32768,tcp,actimeo=0,timeo=600

    mspnas402:/export/msp00avy-88/u3 /u03 nfs hard,nointr,rsize=32768,wsize=32768,tcp,actimeo=0,timeo=600

    mspnas402:/export/msp00avy-88/u4 /u04 nfs hard,nointr,rsize=32768,wsize=32768,tcp,actimeo=0,timeo=600

mspnas402:/export/msp00avy-88/u5 /u05 nfs
hard,nointr,rsize=32768,wsize=32768,tcp,actimeo=0,timeo=600

Where:

mspnas402 is the name of the NFS Storage.

/u02/ocr1, /u02/ocr2, /u02 are the mount points on the cluster nodes.

11. Create mount points for the Oracle Cluster Registry (OCR) and Voting Disks as root user on the NFS Storage.

    # mkdir -p /u02/ocr1/oracle

    # mkdir -p /u02/ocr2/oracle

12. After mounting the OCR and Voting Disk directories, grant ownership and read/write permissions to the Oracle user on those mounted directories.

    # chown –R oracle:rgbudba /u02/ocr1

    # chmod –R 775 /u02/ocr1

    # chown –R oracle:rgbudba /u02/ocr2

    # chmod –R 775 /u02/ocr2

13. Change the ownership of the directories above the CRS installation directory so that root owns them.

    # chown root:dba /u00/crs

    # chmod –R 777 /u00/crs

    Note: Changing the file permission to 777 at this point helps you to avoid permission issues during the installation. When you run the root.sh, the script adjusts the permissions at different levels of directories.

14. Configure SSH access between systems. The Oracle Universal Installer uses SSH to copy binaries between RAC hosts during installation. Execute the following steps when logged in directly as 'oracle'.

    a. Create private key on dbserver1.

       $ ssh-keygen –t dsa

       i. Accept the default location for the key.

       ii. Enter the [Return key] twice when prompted for a passphrase.

    b. Create private key on dbserver2.

       $ ssh-keygen –t dsa

       iii. Accept the default location for the key.

       iv. Enter the [Return key] twice when prompted for a passphrase.

    c. Create authorized keys file on dbserver1.

       $ cd ~/.ssh

       $ scp dbserver2:/u00/oracle/.ssh/id_dsa.pub id_dsa.pub2

       $ cat id_dsa.pub id_dsa.pub2 > authorized_keys

    d. Create authorized keys file on dbserver2.

       $ cd ~/.ssh

       $ scp dbserver1:/u00/oracle/.ssh/id_dsa.pub id_dsa.pub2

       $ cat id_dsa.pub id_dsa.pub2 > authorized_keys

e. Make sure /u00/oracle has permissions of 777.

f. Verify SSH access between systems. None of these commands should prompt for a password. Accept if you are prompted to add a known hosts file.

   v. From dbserver1

      $ ssh dbserver2

      $ ssh dbserver1

   vi. From dbserver2

      $ ssh dbserver1

      $ ssh dbserver2

3

Oracle Clusterware Installation

Clusterware Installation Steps

To install the Oracle clusterware:

1. Modify the user profile in oracle home on both nodes adding the following parameters:

   export ORA_CRS_HOME=<CRS_HOME location>

   export NODE_LIST=<Hostnames in cluster, separated by a comma>

   Example:

   export ORA_CRS_HOME=/u00/crs/oracle/product/12.1.0.2 export NODE_LIST=dbserver1,dbserver2

2. As the Oracle UNIX user owns clusterware installation, navigate to the location where the clusterware media is stored.
3. Run clusterware pre-install checks. Resolve any problems, if identified, before continuing with the clusterware installation.

   Navigate to the directory where the CRS software is, and run the runcluvfy.sh to perform pre-installation check.

   a. Execute the following command to perform post-checks for hardware/operating system setup:

      ./runcluvfy.sh stage -pre crsinst -n $NODE_LIST

   b. Execute the following command to check peer compatibility:

      ./runcluvfy.sh comp peer –n all –orainv dba –verbose

   c. Execute the following command to verify node connectivity:

      ./runcluvfy.sh comp nodecon -n $NODE_LIST –verbose

   d. Execute the following command to check system requirements for CRS:

   ./runcluvfy.sh comp sys –n $NODE_LIST –p crs –orainv dba –verbose

   > **Note:** Investigate and resolve any error messages before continuing.

4. Set DISPLAY to IP address.

   export DISPLAY=<ip address>:0.0.

5. Setup environment variables.

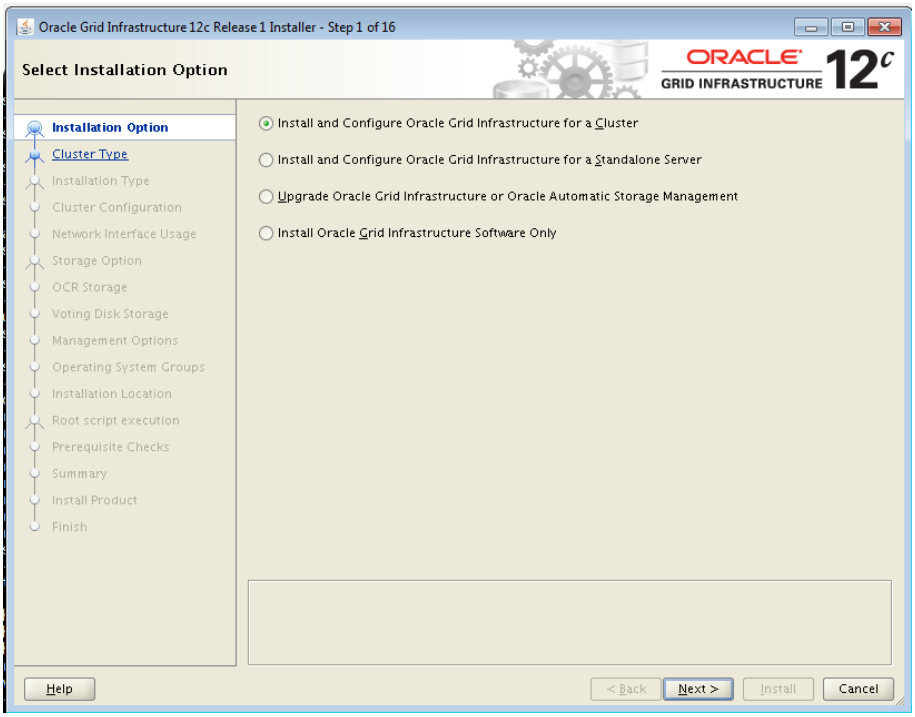
   a. unset `LD_LIBRARY_PATH`.

   b. export `ORACLE_HOME=$ORA_CRS_HOME`

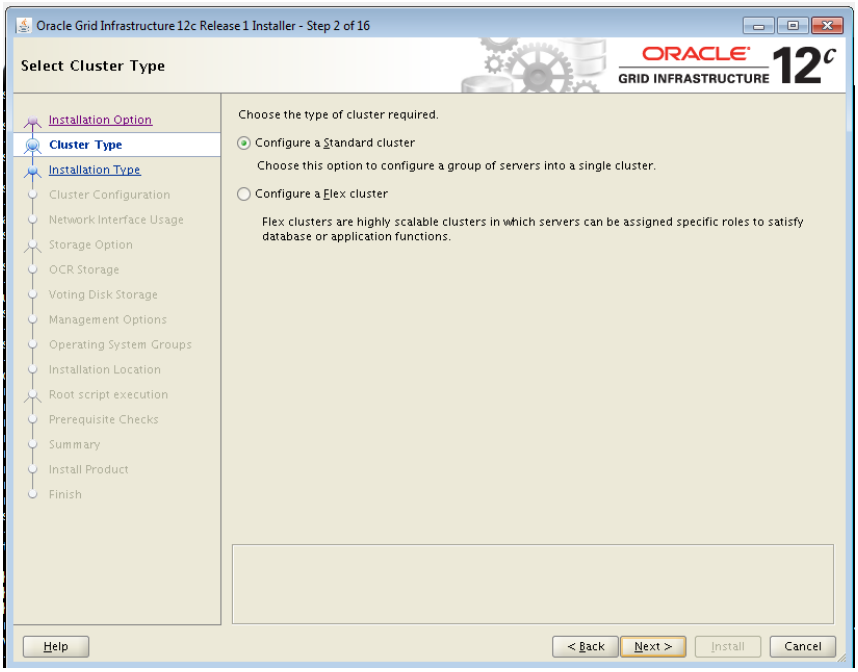
6. Execute the installer. /runInstaller

   > **Note:** This is done on the first node. CRS completes the installation on the second node automatically.

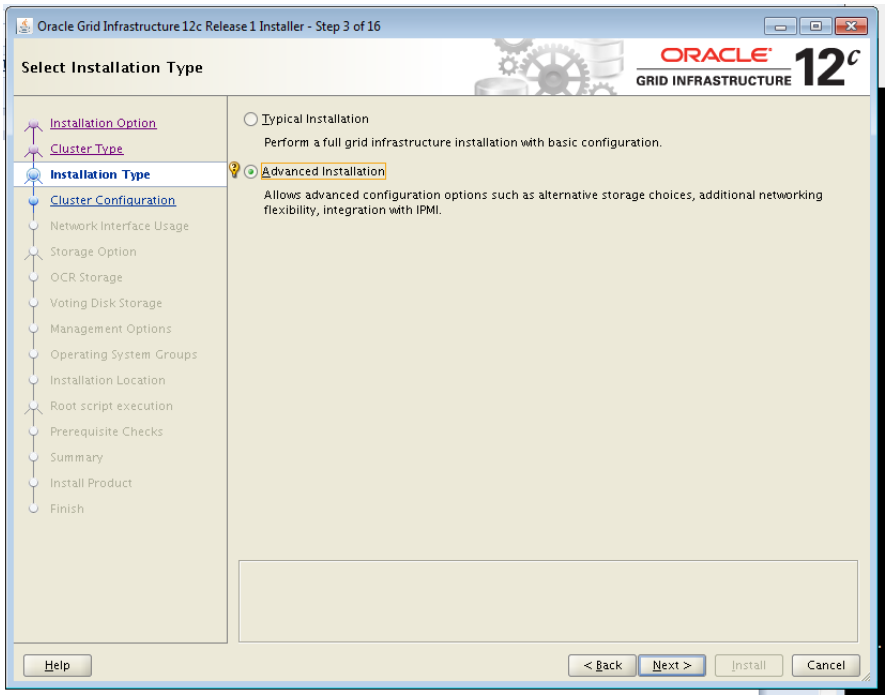
7. The Select Installation Option window opens. Select the **Install and Configure Oracle Grid Infrastructure for a Cluster** check box, and click **Next**.



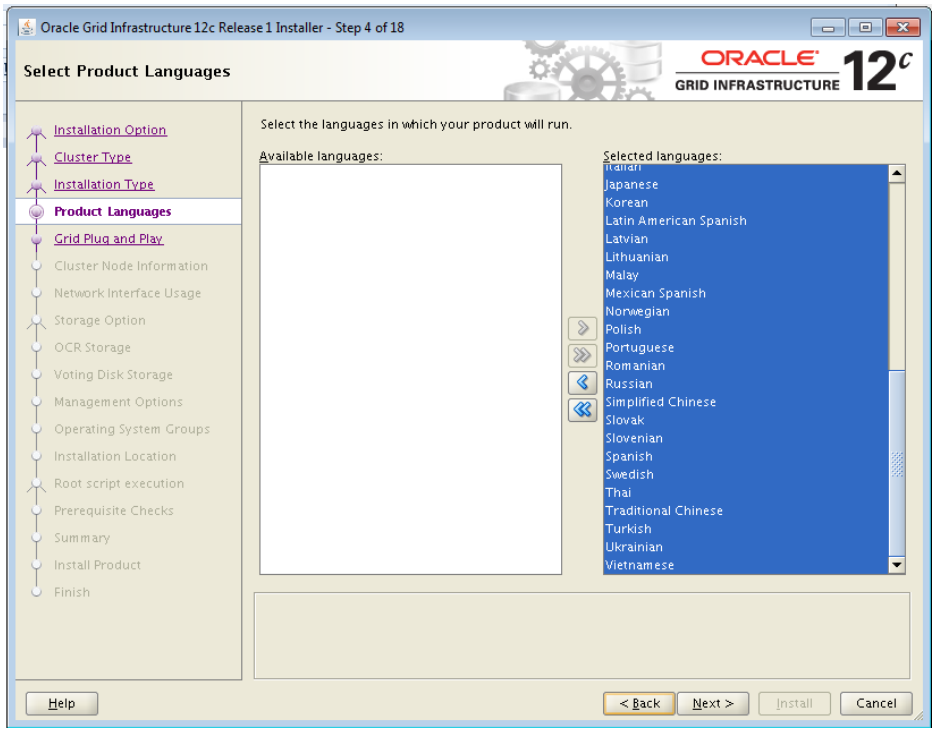
8. The Select Cluster Type window opens. Select the **Configure a Standard Cluster** check box, and click **Next**.



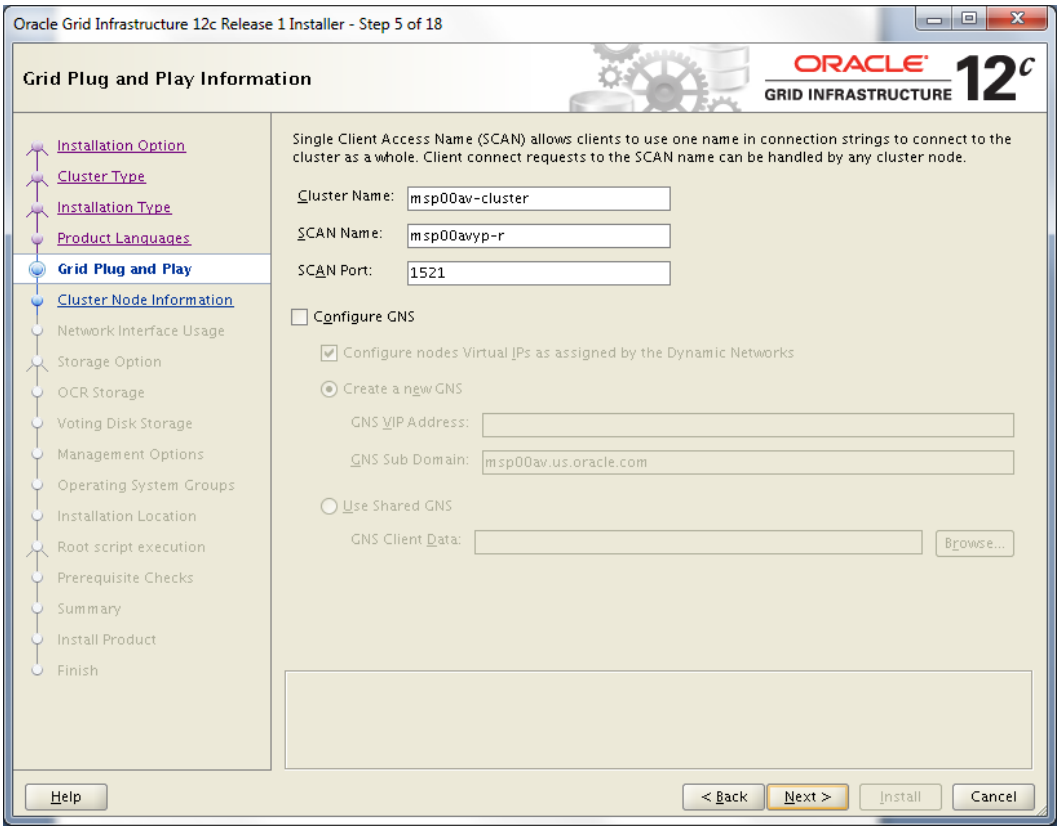
9. The Select Installation Type window opens. Select the **Advanced Installation** check box, and click **Next**.



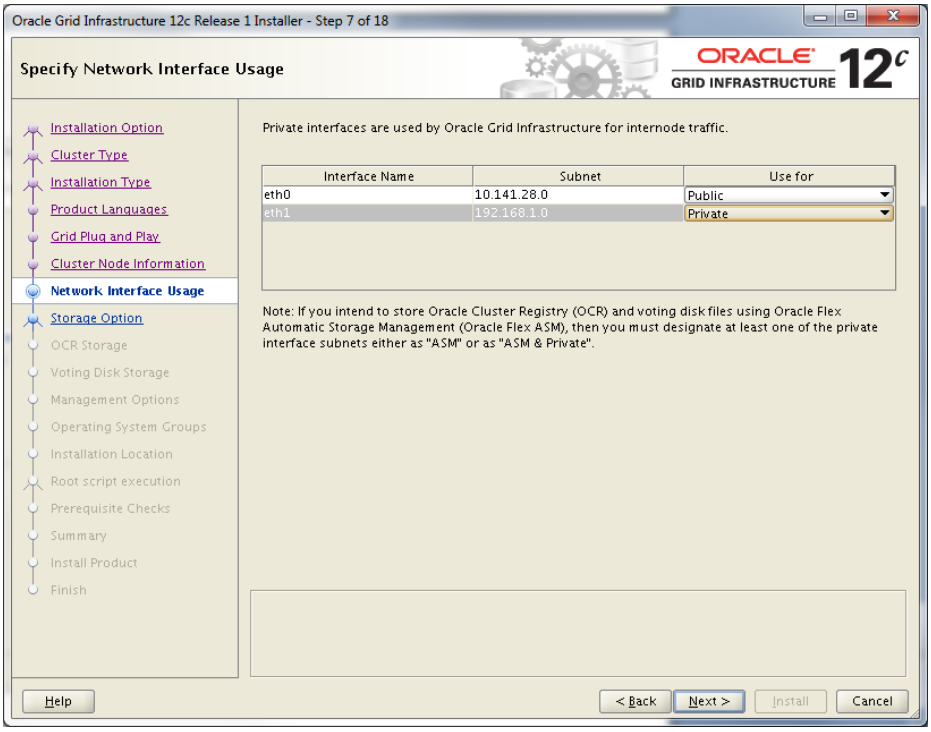
10. The Select Product Languages window opens. Click >> to select all languages, and click **Next**.



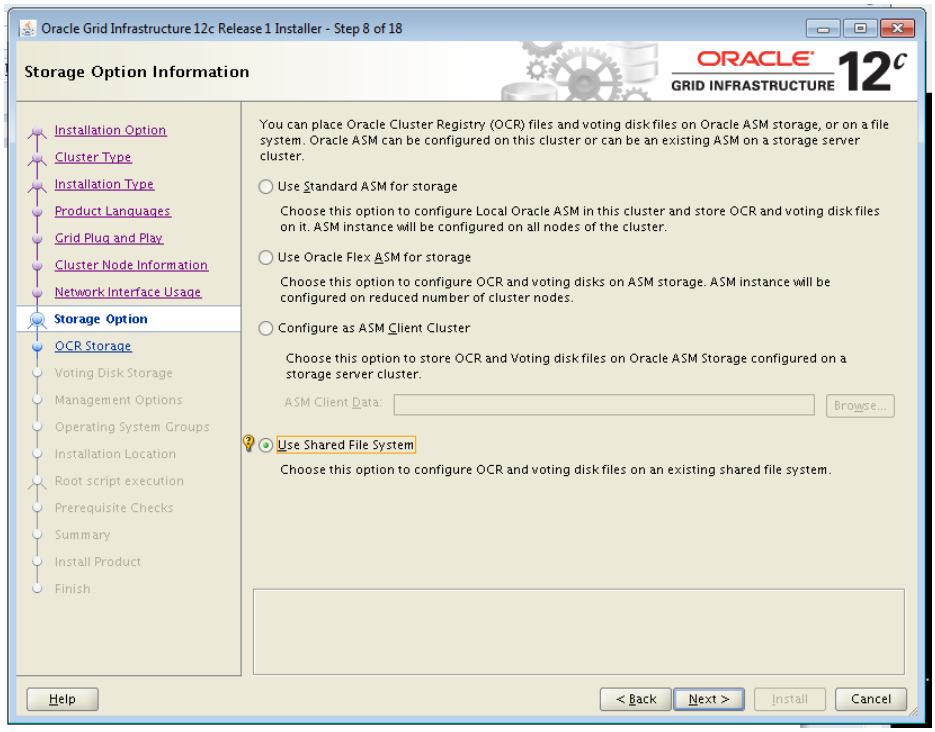
11. The Grid Plug and Play Information window opens. Enter the Cluster Name, Scan Name, and Scan Port, and click **Next**.



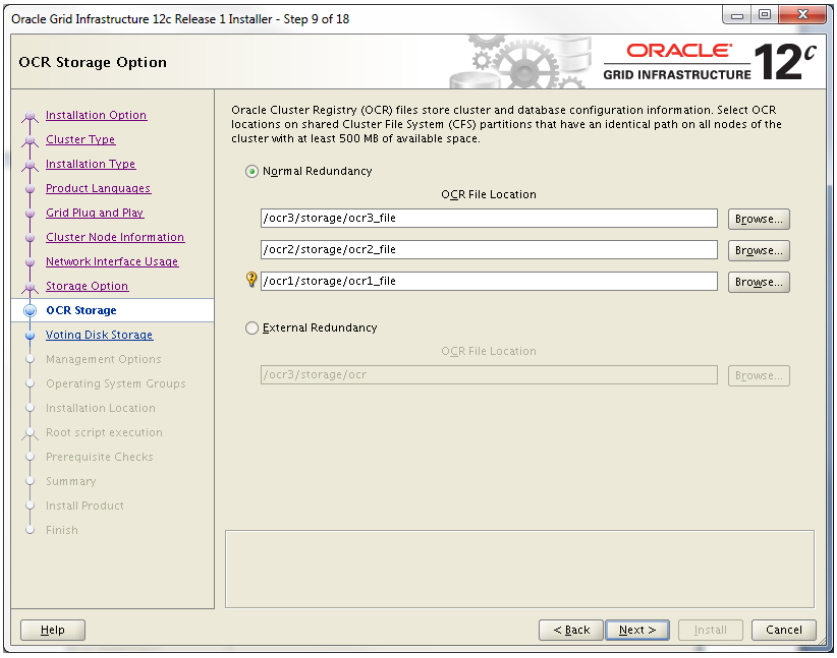
12. The Specify Network Interface Usage window opens. Make sure the following values are set:
    a. eth0 with Subnet 10.141.28.0 is Public.
    b. eth1 with Subnet 192.168.1.is Private.



13. The Storage Option Information window opens. Click the **Shared File System** radio button, and click **Next**.



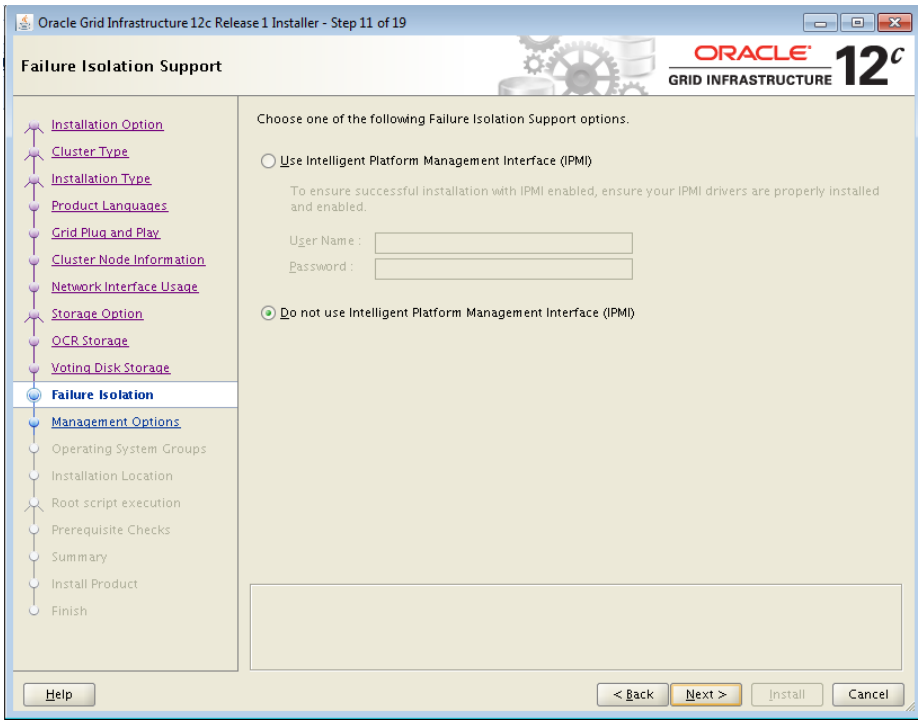
14. The OCR Storage Option window opens. Select the **Normal Redundancy** radio button. Enter the full path of the OCR locations, and click **Next**.



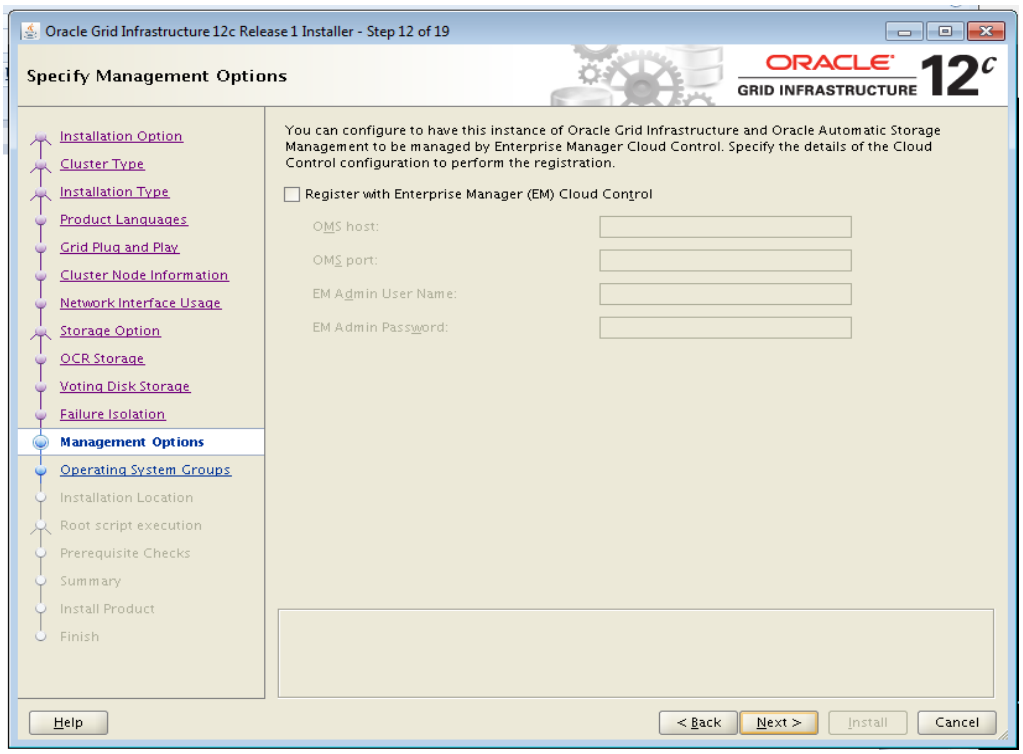
15. The **Voting Disk Storage** window opens. Select the **Normal Redundancy** radio button. Enter the full path of the voting disk locations, and click **Next**.

> **Note:** Make sure the sub-directories /u02/ocr1, /u02/ocr2, and /u02/ocr3 have been created.

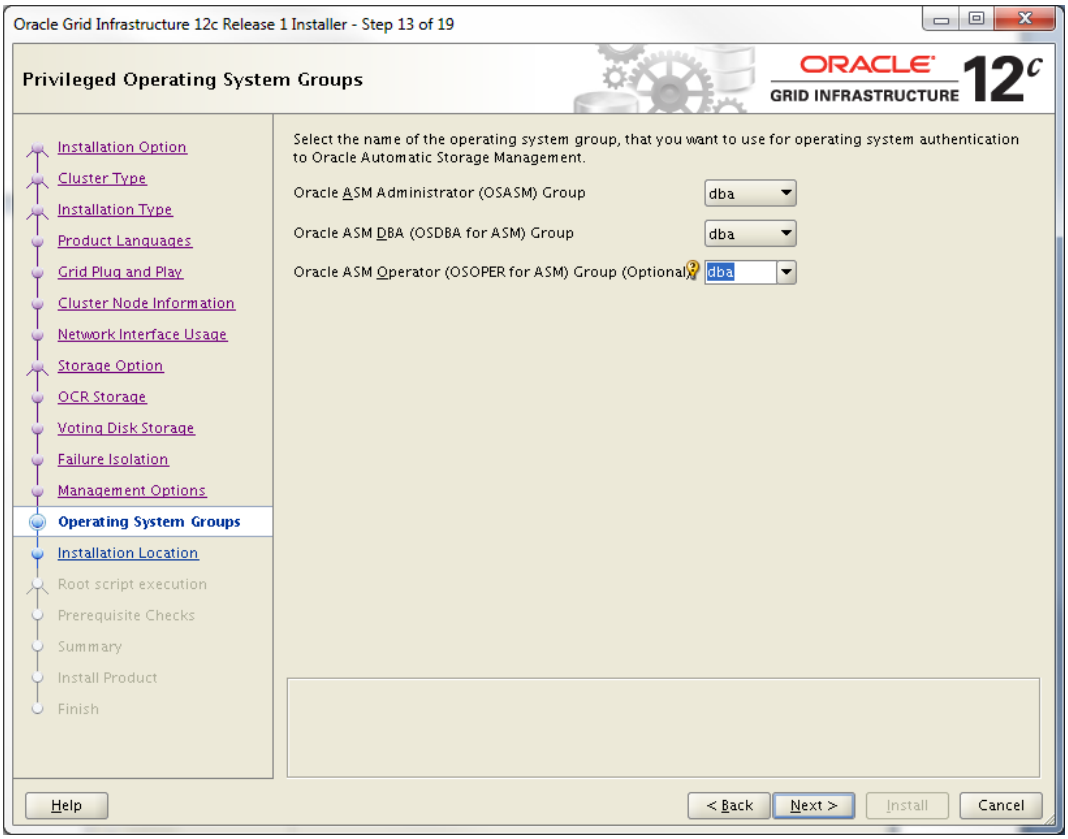
16. The Failure Isolation Support window opens. Select the **Do not use Intelligent Platform Management Interface (IPMI)** radio button, if you do not intend to configure it, and click **Next**.



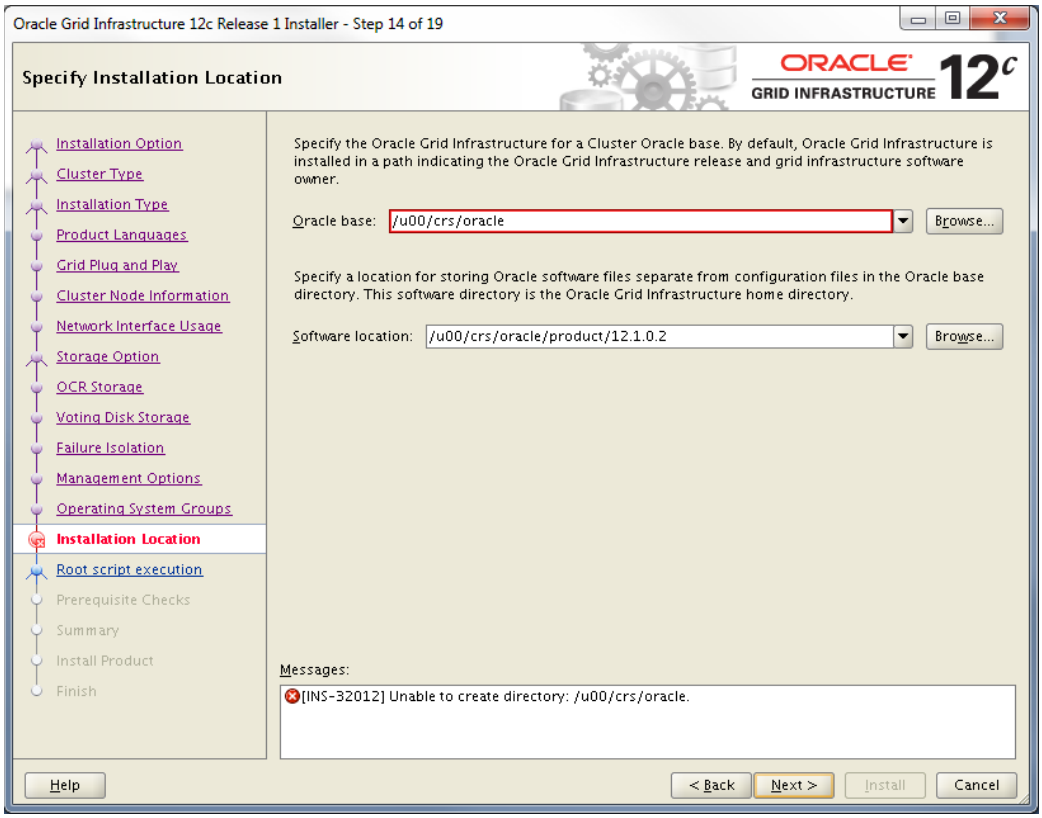
17. The Specify Management Options window opens. Click **Next** if you do not wish to configure Grid to manage the resource on the server.



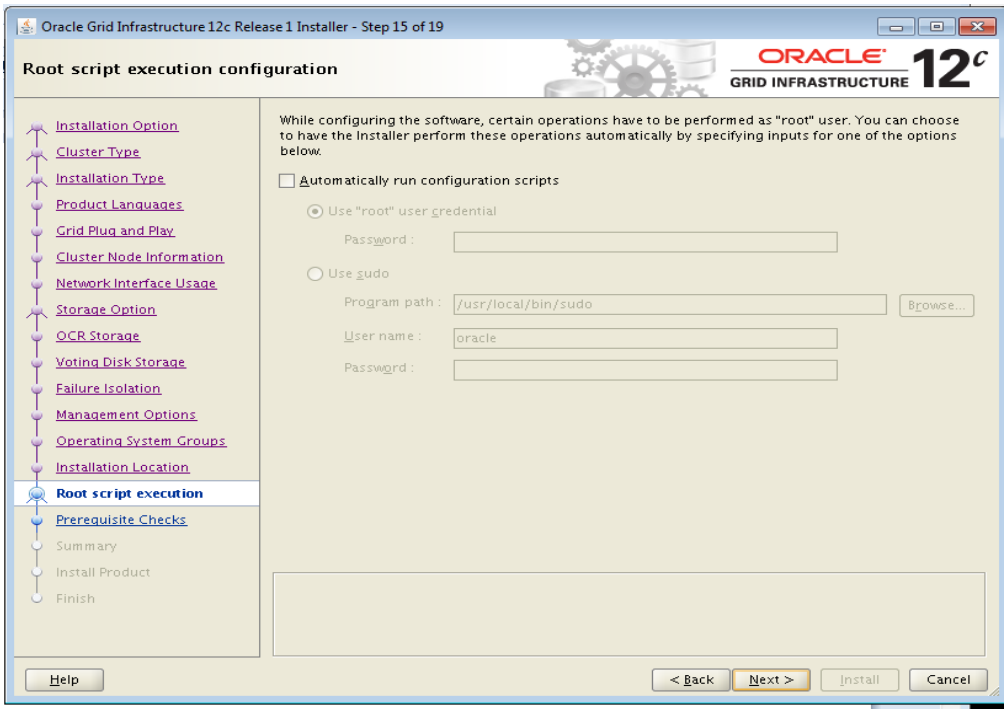
18. The Privileged Operating System Groups window opens. Review if correct values are set for the Operating System Groups, and click **Next.**



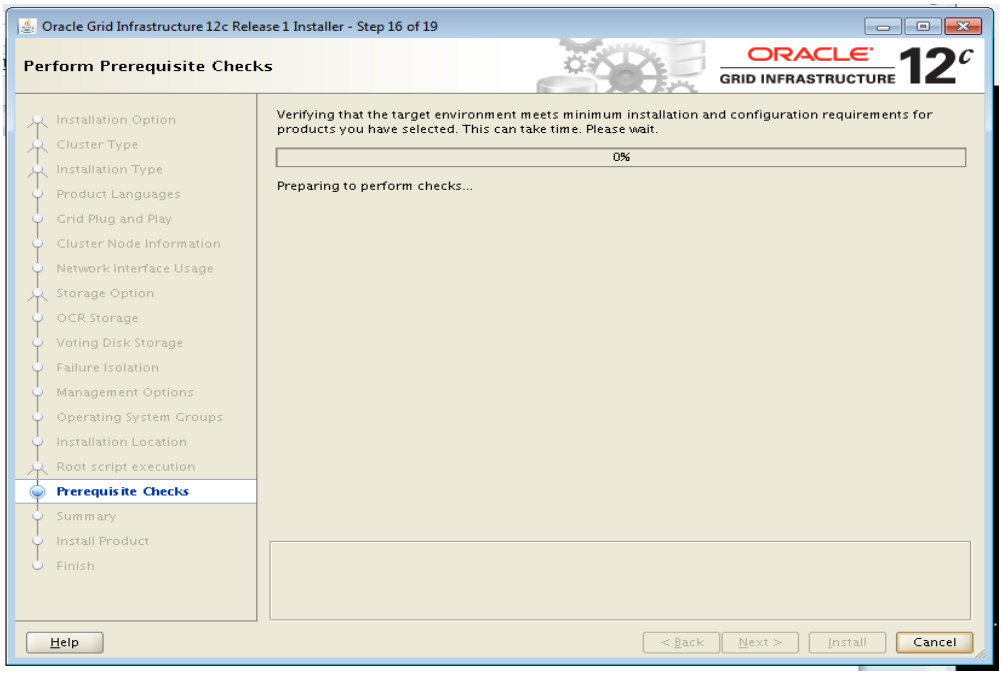
19. The Specify Installation Location window opens. Enter the path in 'Oracle base' and 'Oracle Home' input fields, and click **Next**.



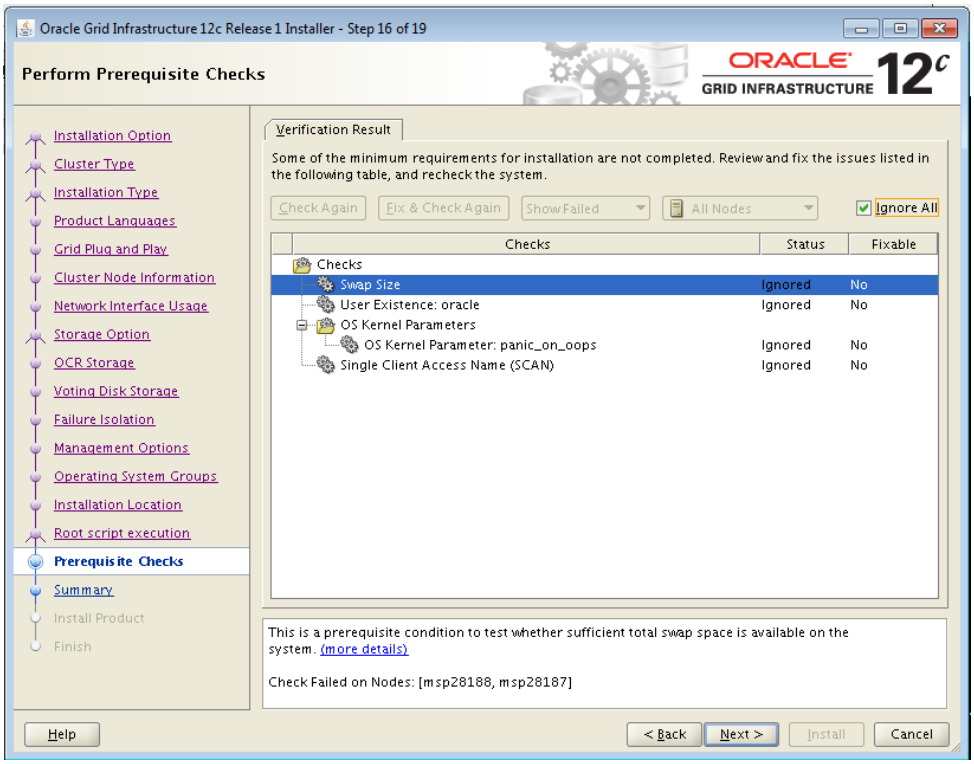
20. The Root script execution configuration window opens. Make sure the 'Automatically run configuration scripts' option is unchecked. This is to make sure that the root.sh file does not execute automatically, by the Installer. Click **Next**.



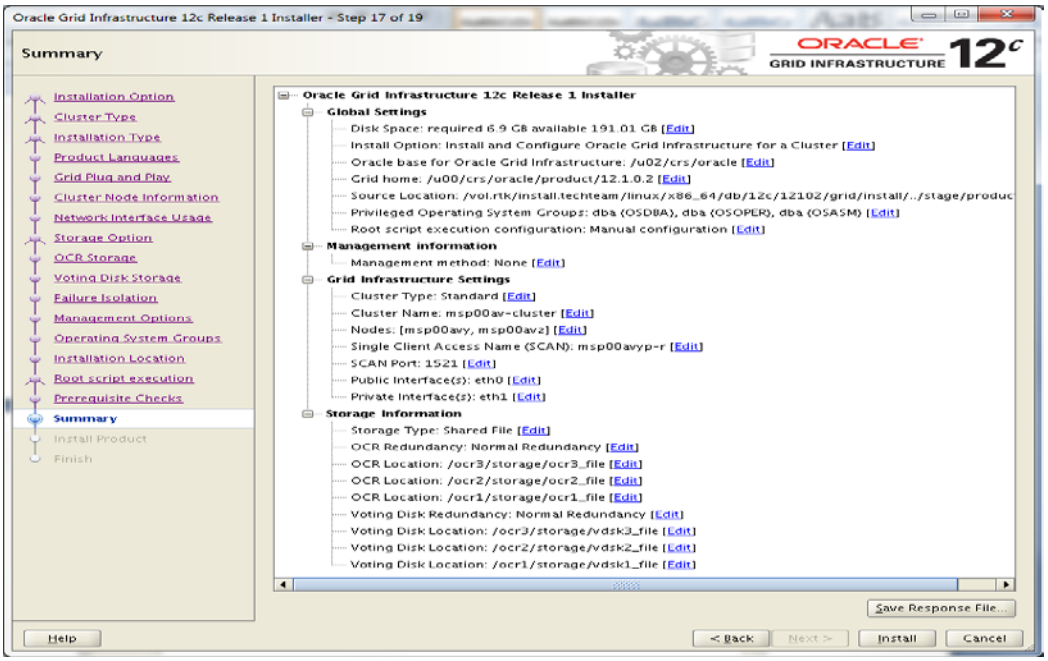
21. The Perform Prerequisite Checks window opens. The progress bar displays the progress of pre-requisite check.



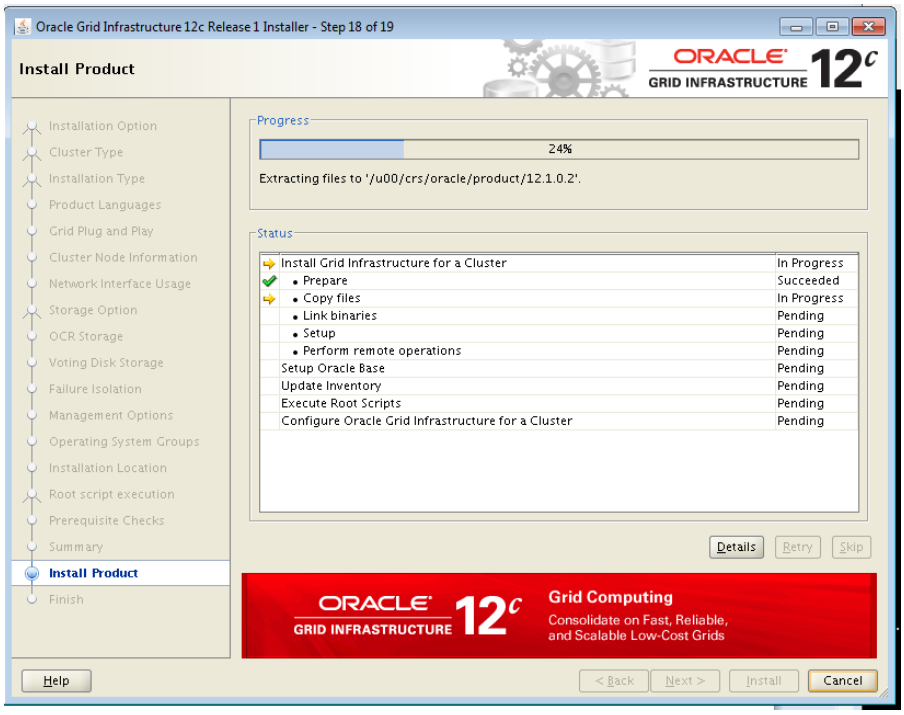
22. The results of the pre-requisite checks are displayed in the **Verification Result** tab. If you see any warnings, click 'Fix & Check Again' to generate a fix-up script for the root or sudo owner. You can alternatively choose to ignore the warnings by selecting the 'Ignore All' check box and click **Next**.



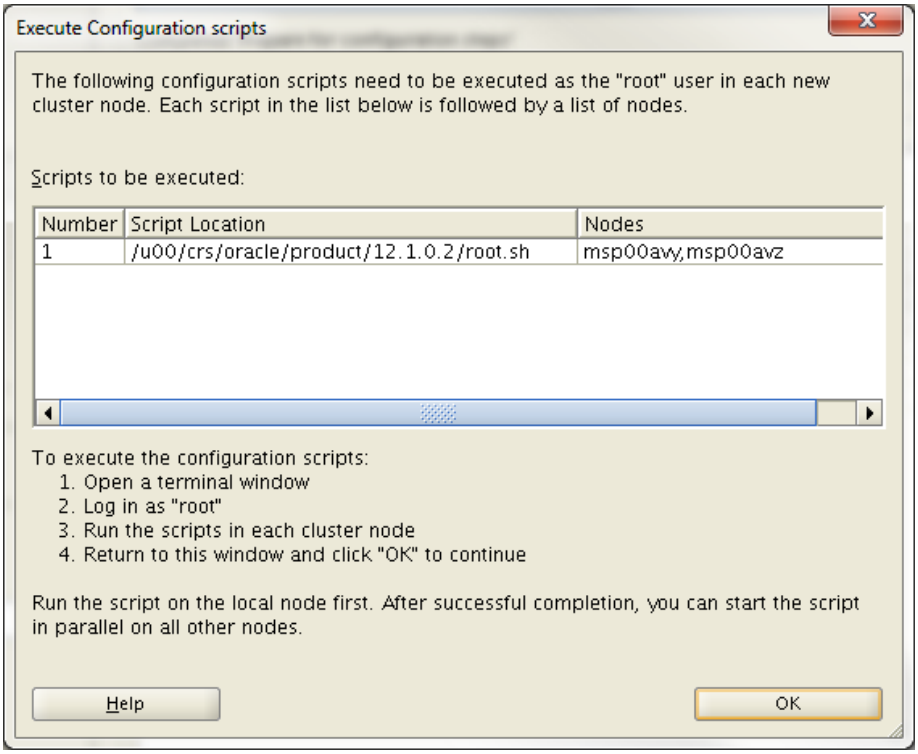
23. The Summary window opens. It displays a summary of items to be installed. Click **Install** to begin the installation.



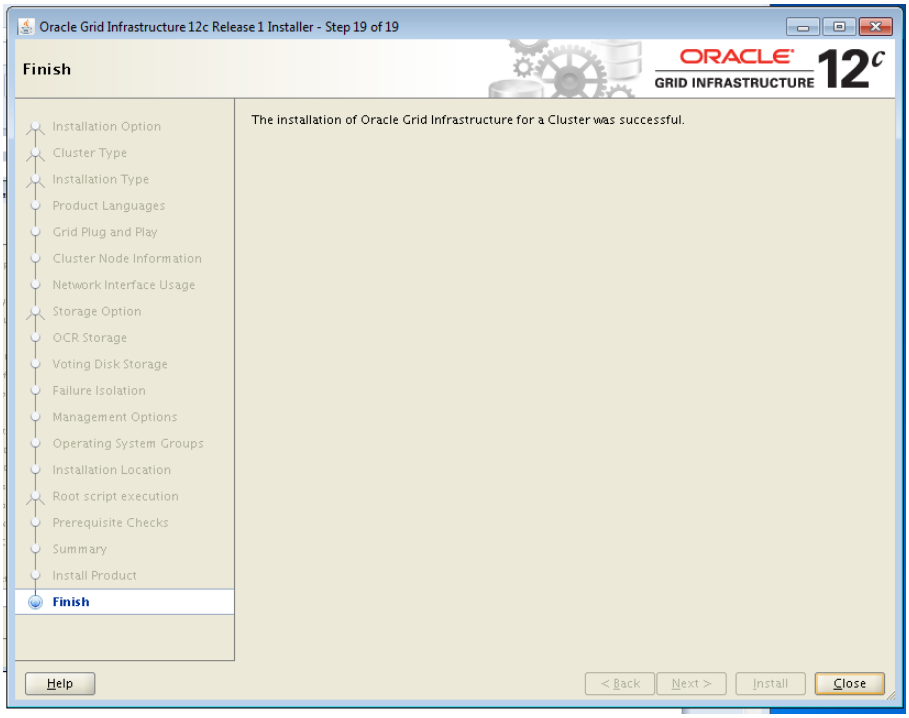
24. The Install Product window opens. The progress of the installation is displayed in the Progress bar.



25. When the installation is complete, the Execute Configuration scripts window opens. On each node, navigate to $ORA_CRS_HOME/bin, and execute root.sh. When root.sh has been executed on all the nodes, click **OK** to return to the Installation screen.


Execute Configuration scripts
The following configuration scripts need to be executed as the "root" user in each new cluster node. Each script in the list below is followed by a list of nodes.
Scripts to be executed:
Number
Script Location
Nodes
1
/u00/crs/oracle/product/12.1.0.2/root.sh
msp00avy,msp00avz
To execute the configuration scripts:
1. Open a terminal window
2. Log in as "root"
3. Run the scripts in each cluster node
4. Return to this window and click "OK" to continue
Run the script on the local node first. After successful completion, you can start the script in parallel on all other nodes.
Help
OK

26. The Finish screen opens. Click **Close** to exit the Installer.



27. Validate CRS. Execute $ORACLE_HOME/bin/crs_stat to verify the status of crs. All the resources that are online are shown.

```
msp00avy.us.oracle.com:[+ASM1] /u00/oracle>
--> $ORACLE_HOME/bin/crs_stat -t
Name           Type           Target    State     Host
------------------------------------------------------------
ora....ER.lsnr ora....er.type ONLINE    ONLINE    msp00avy
ora....N1.lsnr ora....er.type ONLINE    ONLINE    msp00avz
ora....N2.lsnr ora....er.type ONLINE    ONLINE    msp00avy
ora....N3.lsnr ora....er.type ONLINE    ONLINE    msp00avy
ora.MGMTLSNR   ora....nr.type ONLINE    ONLINE    msp00avy
ora.asm        ora.asm.type   OFFLINE   OFFLINE
ora.cvu        ora.cvu.type   ONLINE    ONLINE    msp00avz
ora....SM1.asm application    OFFLINE   OFFLINE
ora....VY.lsnr application    ONLINE    ONLINE    msp00avy
ora....avy.ons application    ONLINE    ONLINE    msp00avy
ora....avy.vip ora....t1.type ONLINE    ONLINE    msp00avy
ora....SM2.asm application    OFFLINE   OFFLINE
ora....VZ.lsnr application    ONLINE    ONLINE    msp00avz
ora....avz.ons application    ONLINE    ONLINE    msp00avz
ora....avz.vip ora....t1.type ONLINE    ONLINE    msp00avz
ora....network ora....rk.type ONLINE    ONLINE    msp00avy
ora.oc4j       ora.oc4j.type  ONLINE    ONLINE    msp00avz
ora.ons        ora.ons.type   ONLINE    ONLINE    msp00avy
ora.scan1.vip  ora....ip.type ONLINE    ONLINE    msp00avz
ora.scan2.vip  ora....ip.type ONLINE    ONLINE    msp00avy
ora.scan3.vip  ora....ip.type ONLINE    ONLINE    msp00avy
```

4

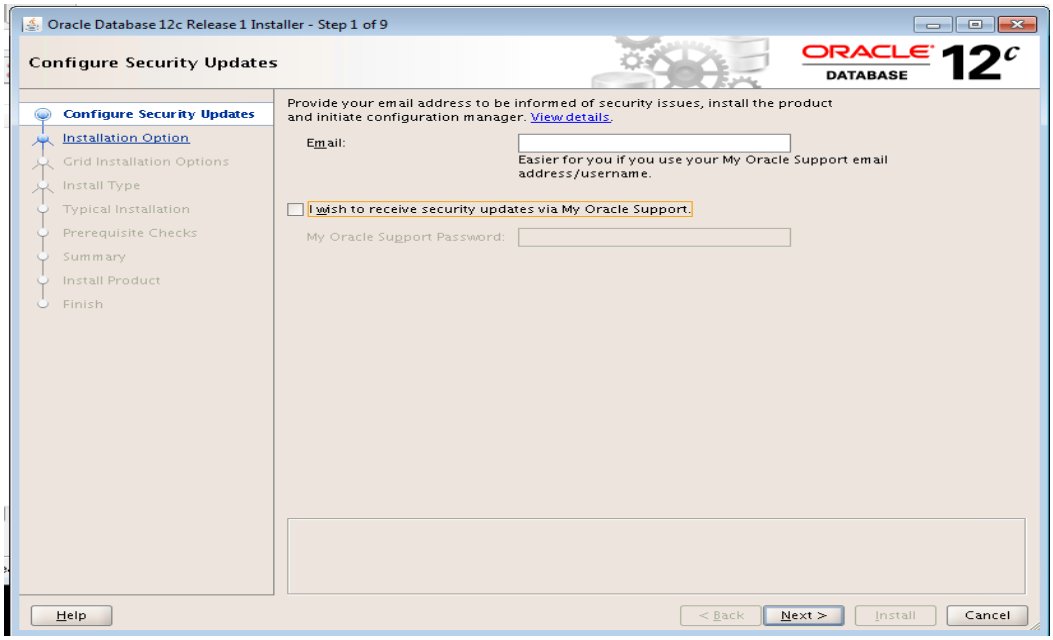
Installing the Oracle Database 12c Release 2 (12.1.0.2) on Linux

This section explains steps to install oracle database 12c Release 1 (12.1.0.2) software on Linux.

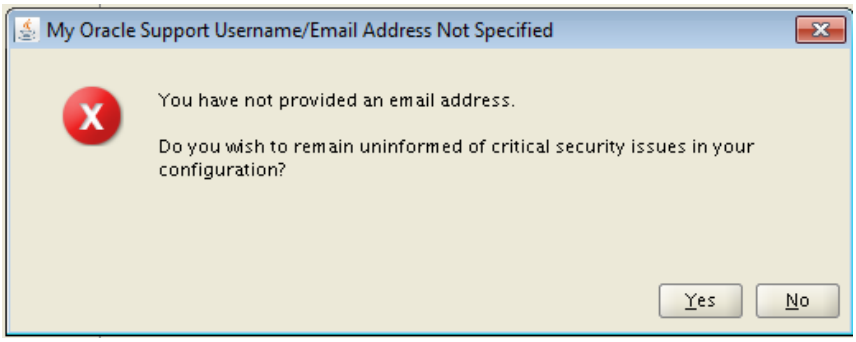
Installing RDBMS 12.1.0.2 Database Software

To install RDBMS 12.1.0.2 database software:

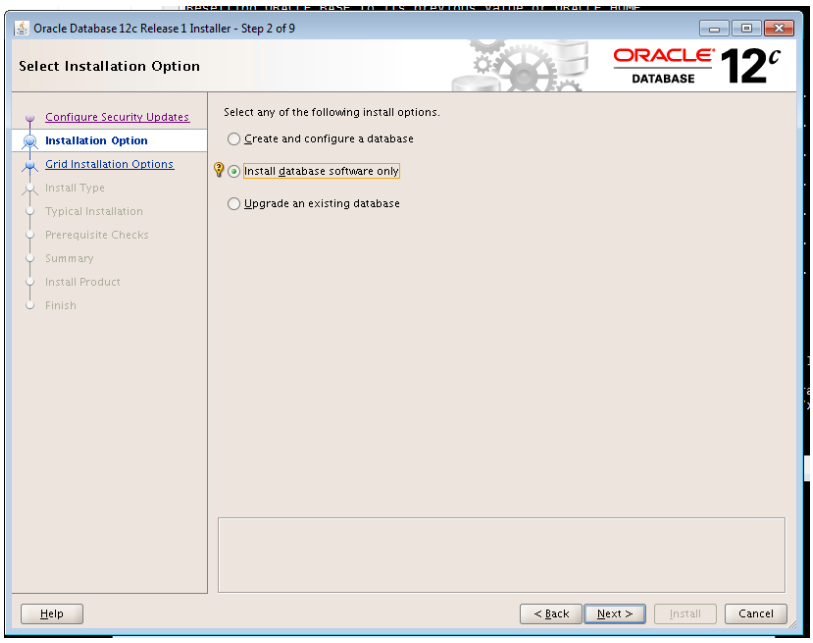
1. On the Dbserver1, change to the directory where the Oracle Database 12.1.0.2 software is located.
2. Set your environment to point to the desired Oracle Database Home.
   a. export ORACLE_HOME=/u00/oracle/product/12.1.0.2
   b. export ORACLE_BASE=/u00/oracle /
3. Navigate to the 12c binary repository to start the installer
   ./runInstaller
4. The Configure Security Updates window opens. If you do not wish to receive security update emails, uncheck the option, and click **Next**.



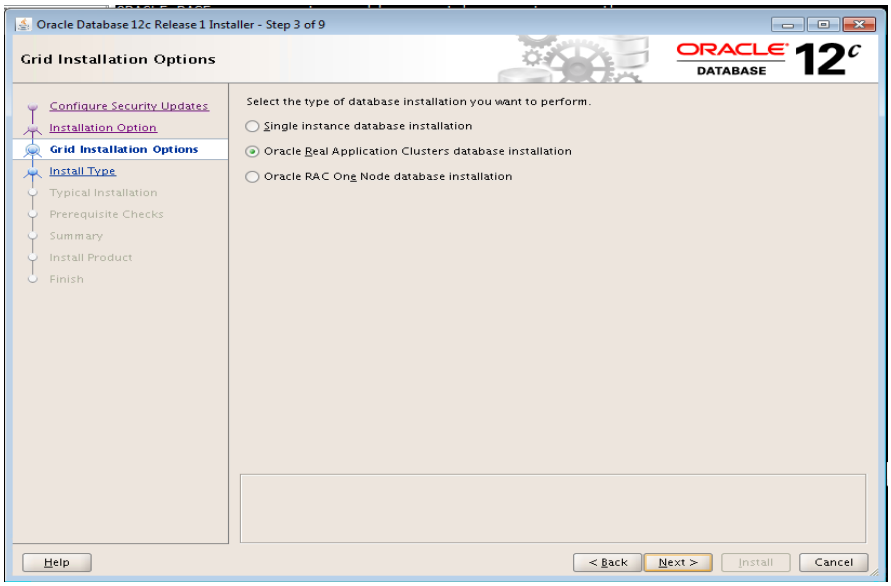
5. The My Oracle Support Username/Email Address Not Specified window opens. Click **Yes** to continue.



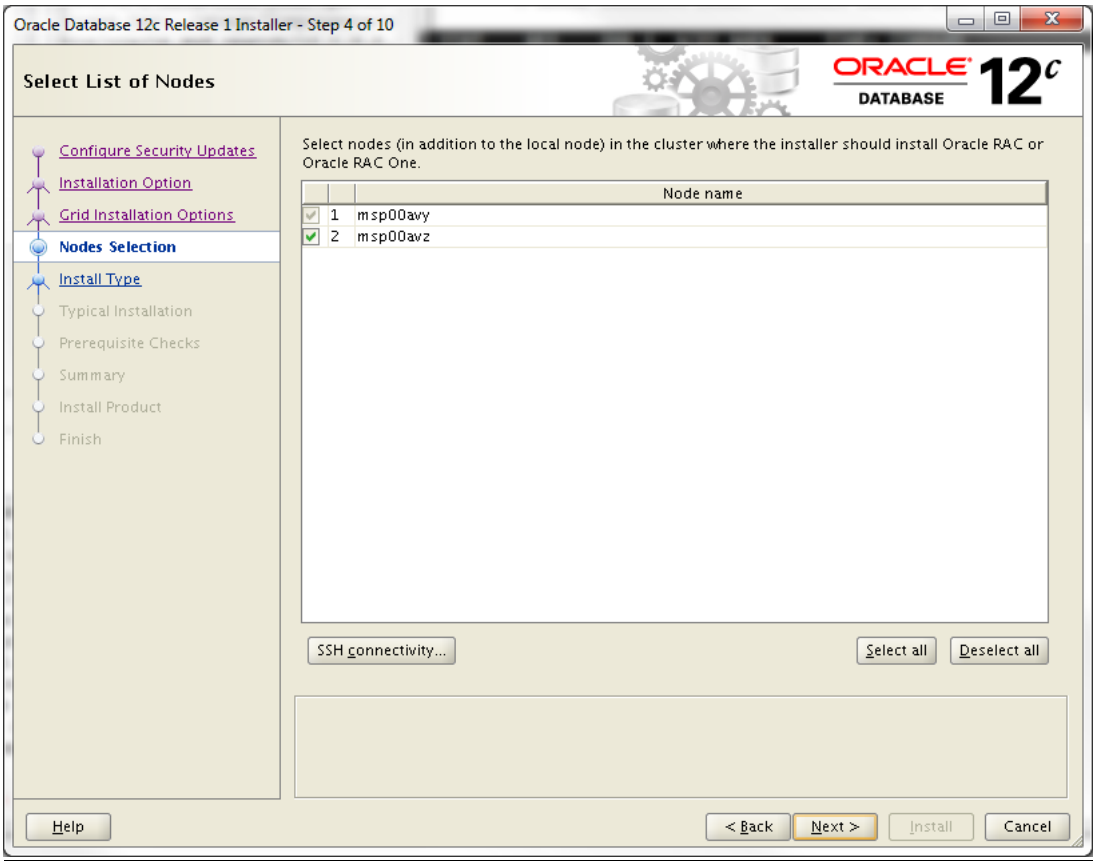
6. The Select Installation Option screen opens. Select the **Install database software only** radio button, and click **Next**.



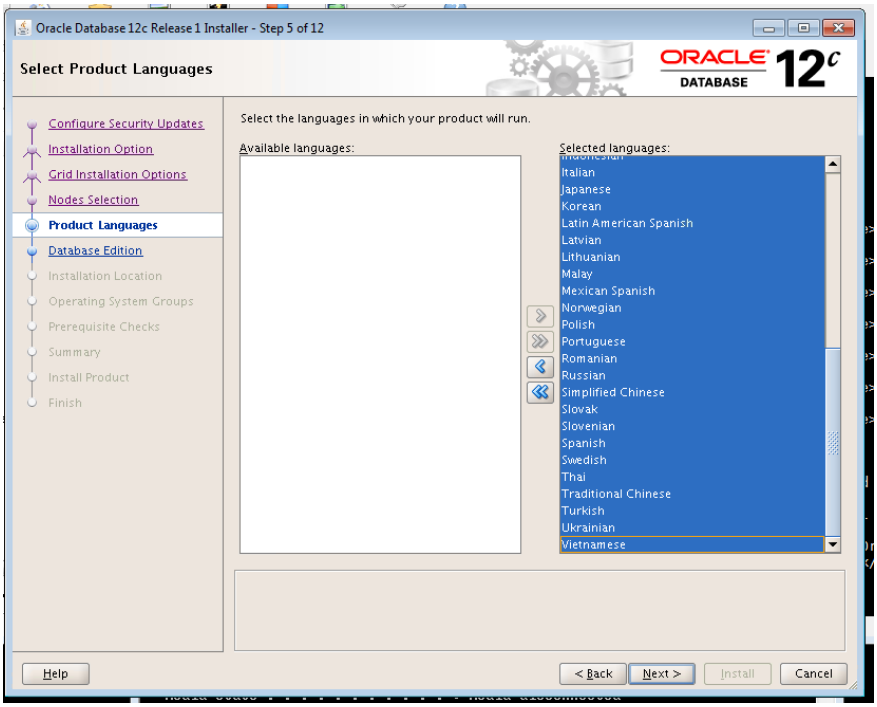
7. The Grid Installation Options window opens. Select **Oracle Real Application Clusters database installation** radio button, and click **Next**.



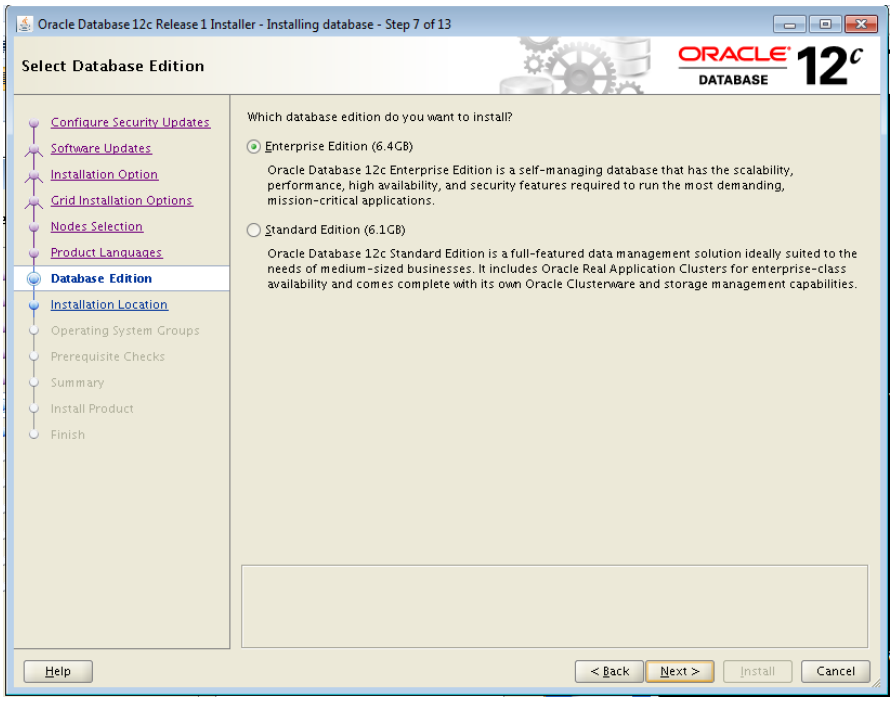
8. The Select List of Nodes window opens. Verify the nodes displayed are correct, and click **Next**.



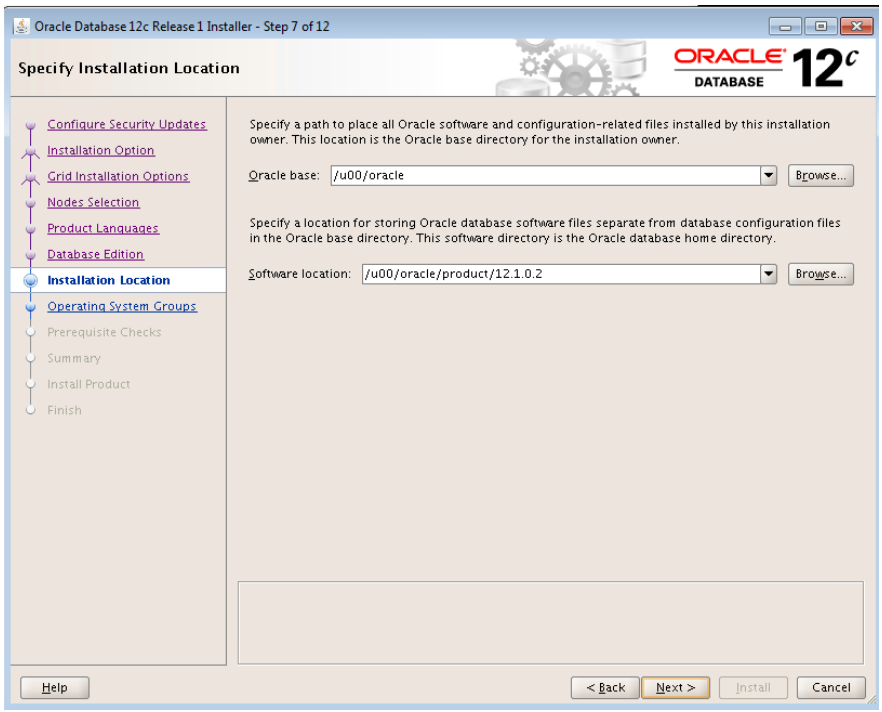
9. The Select Product Languages window opens. Click >> to install all languages, and click **Next**.



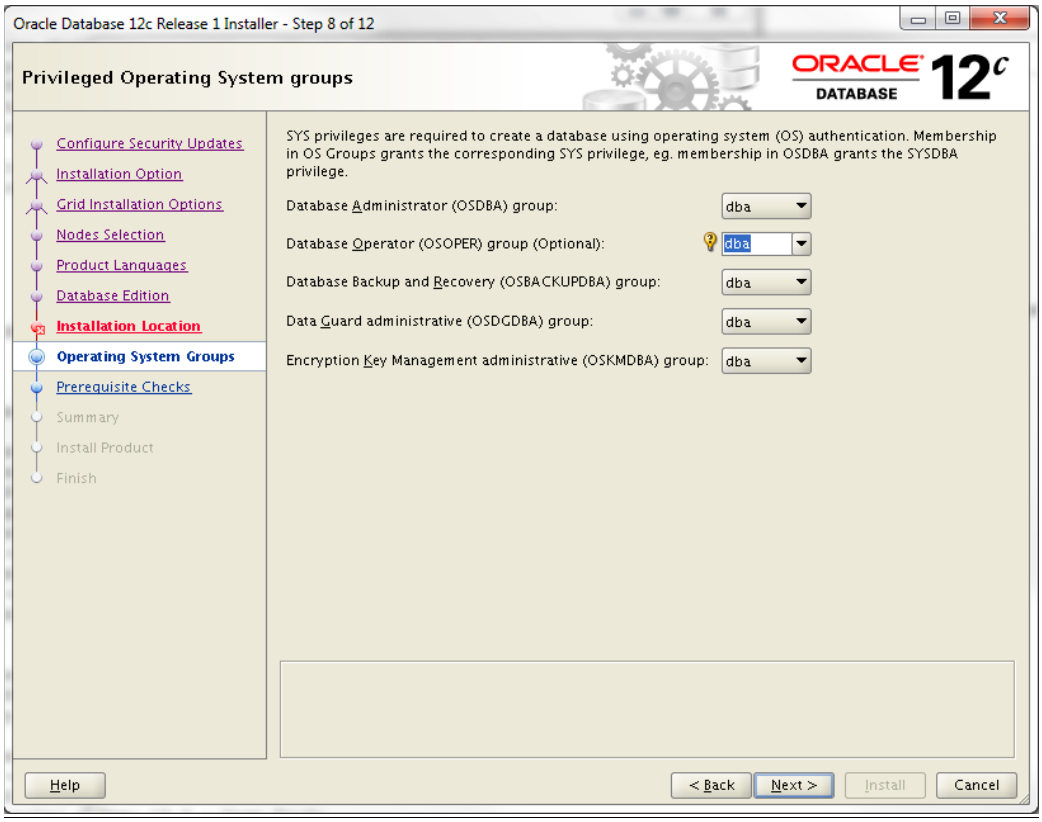
10. The Select Database Edition window opens. Select the **Enterprise Edition** radio button, and click **Next**.



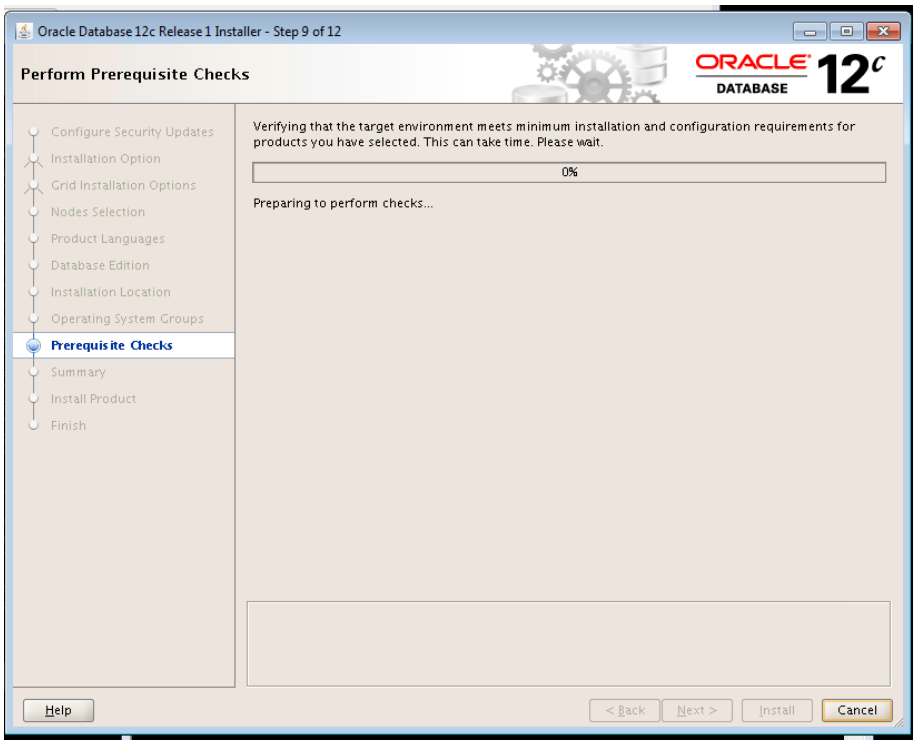
11. The Specify Installation Location window opens. Verify that the path entered in the **Oracle base and Software location** are correct, and click **Next**.



12. The Privileged Operating System Groups window opens. Click the **Database Operator (OSOPER) group (Optional)** drop-down button, and select the correct OS user group for this installation. Click **Next**.

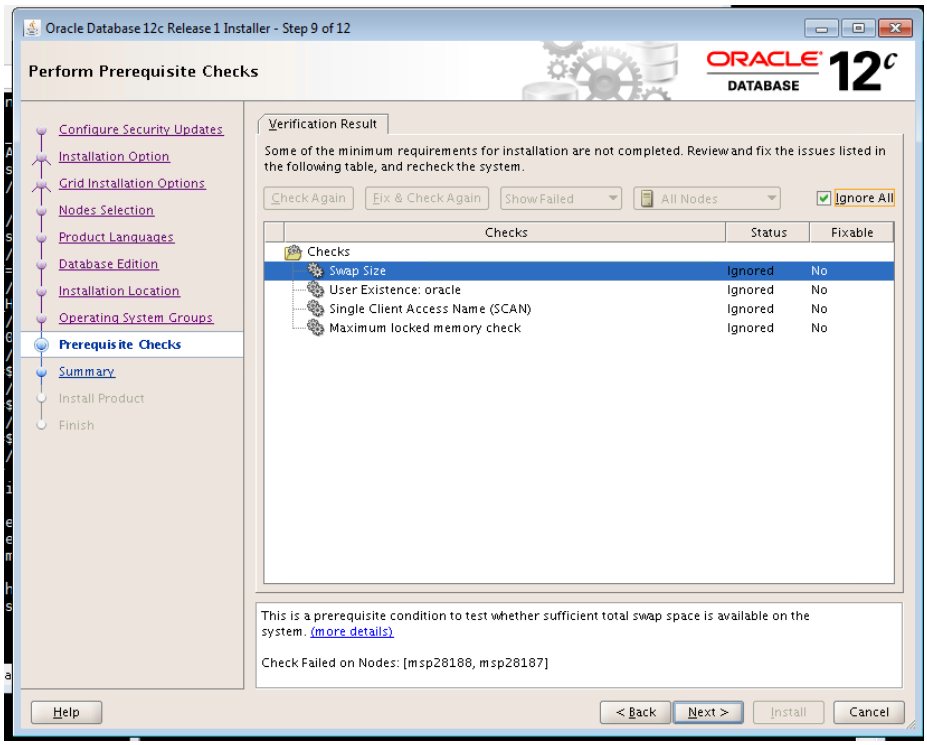


13. The Perform Prerequisite Checks window opens. The progress of the pre-requisite checks is displayed in the Progress bar.

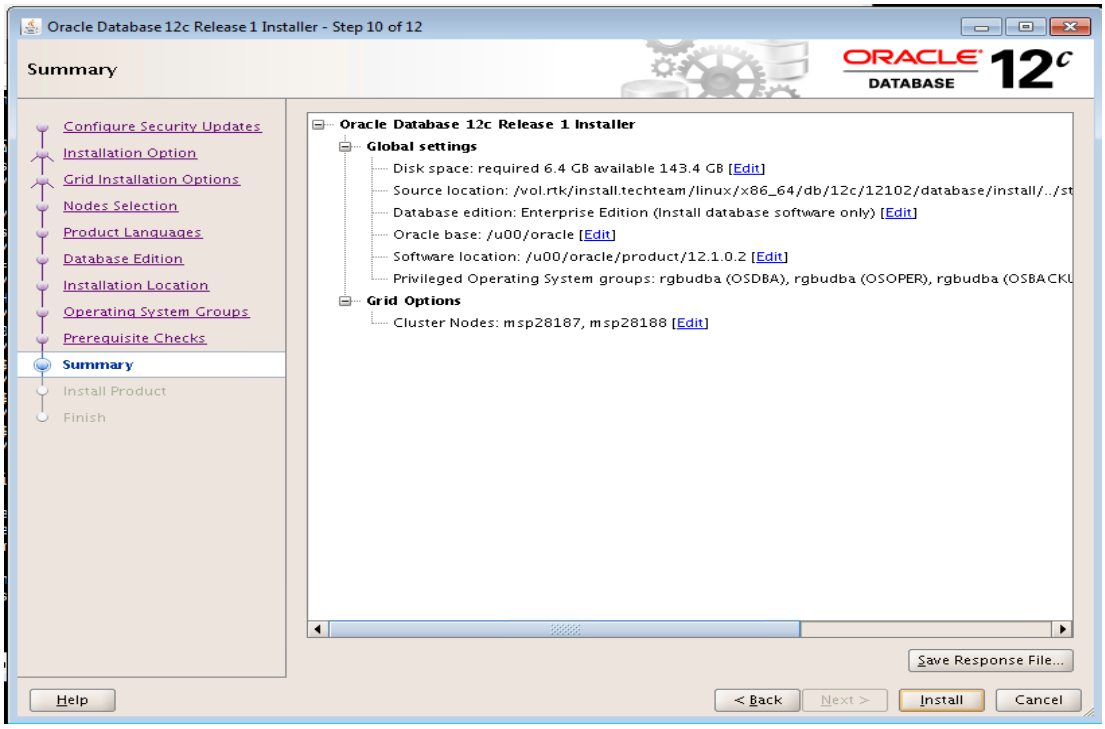


14. The results of the pre-requisite checks are displayed in the **Verification Result** tab. If you see any warnings, click **Fix & Check Again** to generate a fix-up script for root to

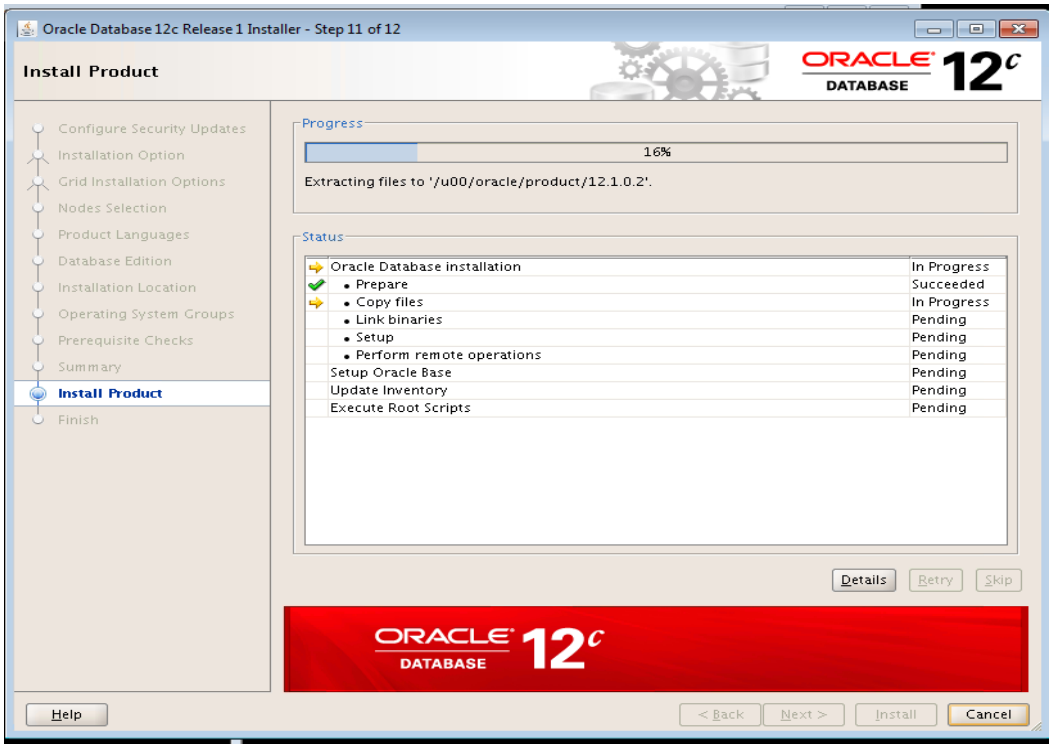
execute to fix the settings. You can alternatively select the **Ignore All** check box, and click **Next**.



15. The Summary window opens. Click **Install**.



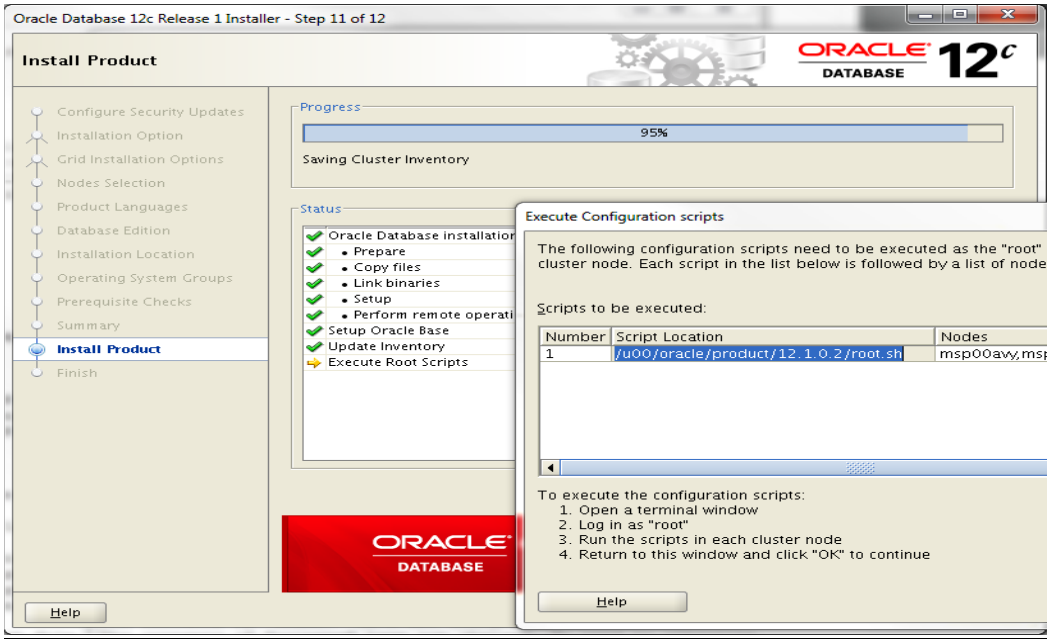
16. The Install Product window. The installation progress is displayed in the Progress bar.



17. The Execute Configuration Scripts pop-up window opens. Towards the end of the installation, the installer prompts you to run the root.sh script. Run the script on one server at a time as root:

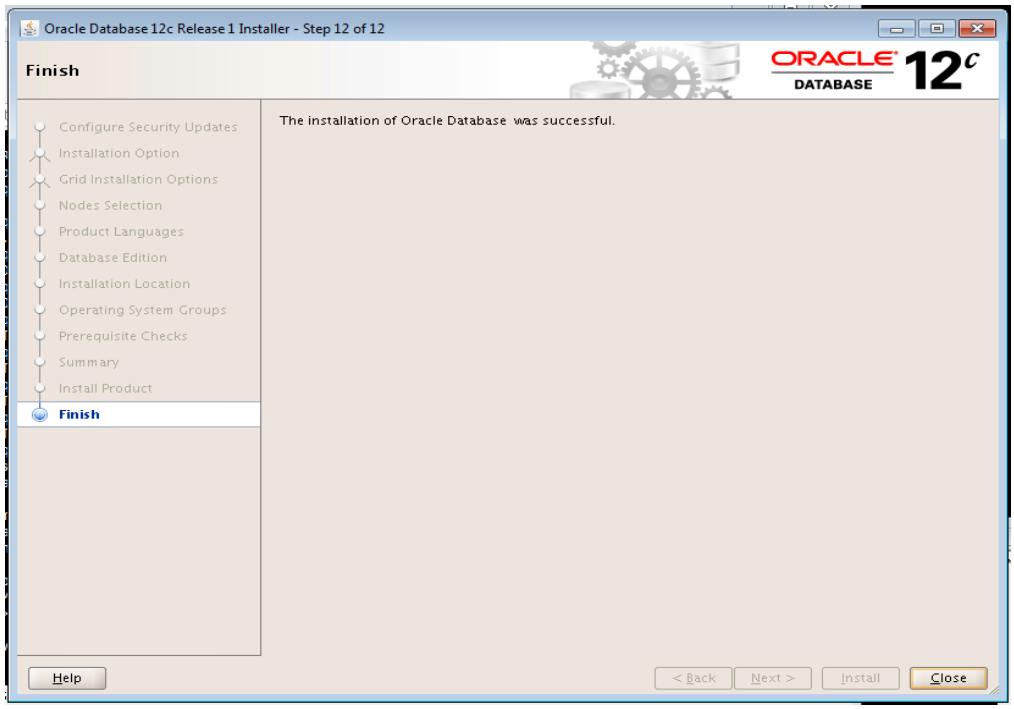
    /u00/oracle/product/12.1.0.2/root.sh

    Click **OK** when the root.sh has been run on all nodes.



Note: Take care not to run root.sh on all nodes simultaneously.

18. The Finish window opens. The installation of RDBMS binary completes on all nodes. Click **Close** to exit the Installer.



5

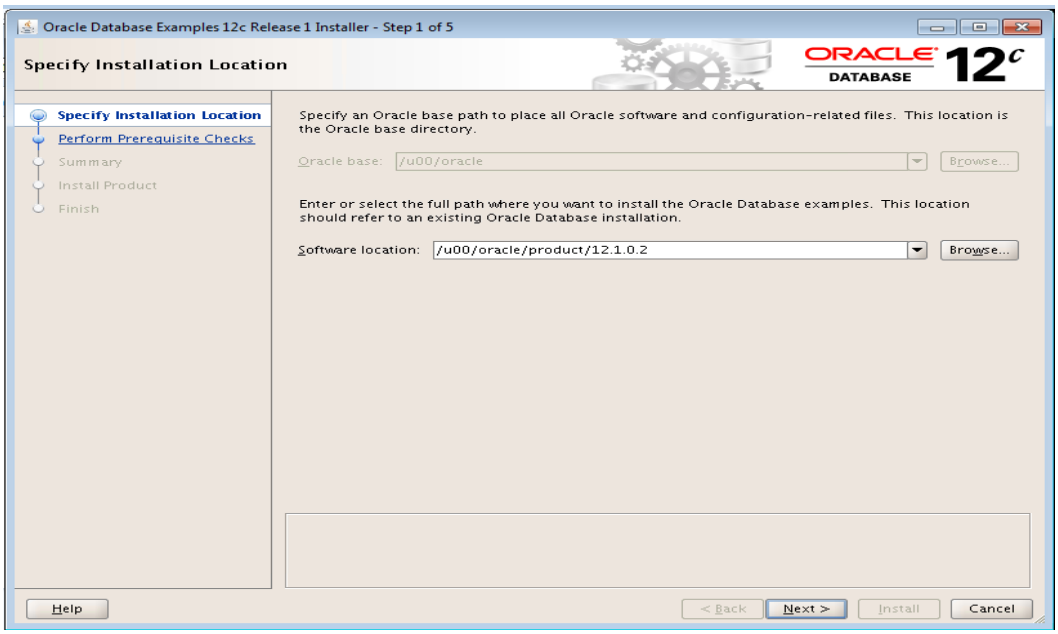
Installing the Oracle Database 12c Release 2 (12.1.0.2) on Linux

Examples of Oracle Database 12c Release 1

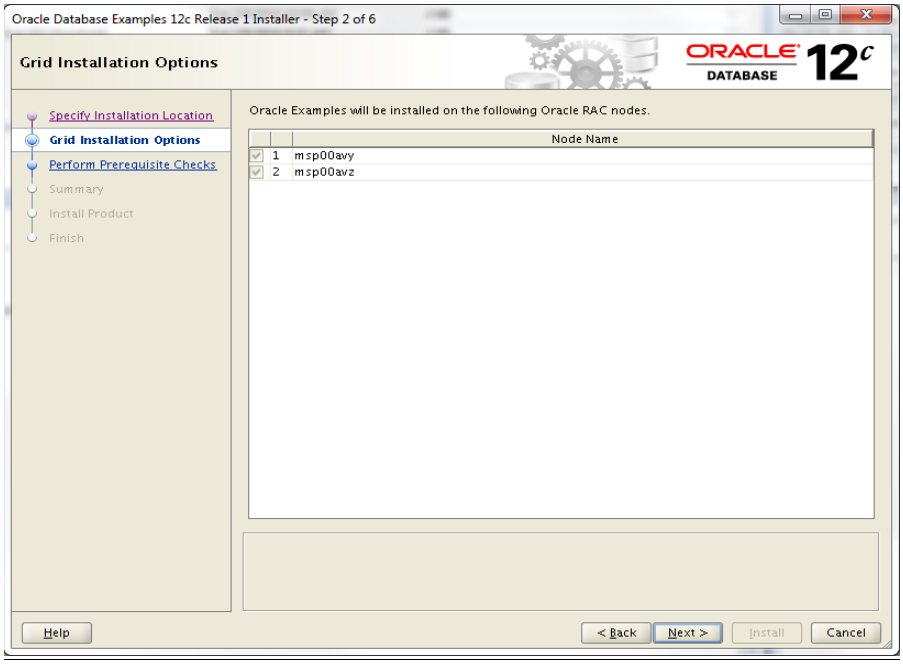
This section describes the Oracle Database 12c Release 1 examples:

1. Navigate to the directory where the Oracle Database Companion software is located.
2. Run the installer.

   ./runInstaller
3. The Specify Installation Location window opens. Verify that the software path entered in the 'Software Location' input field is correct, and click **Next**.

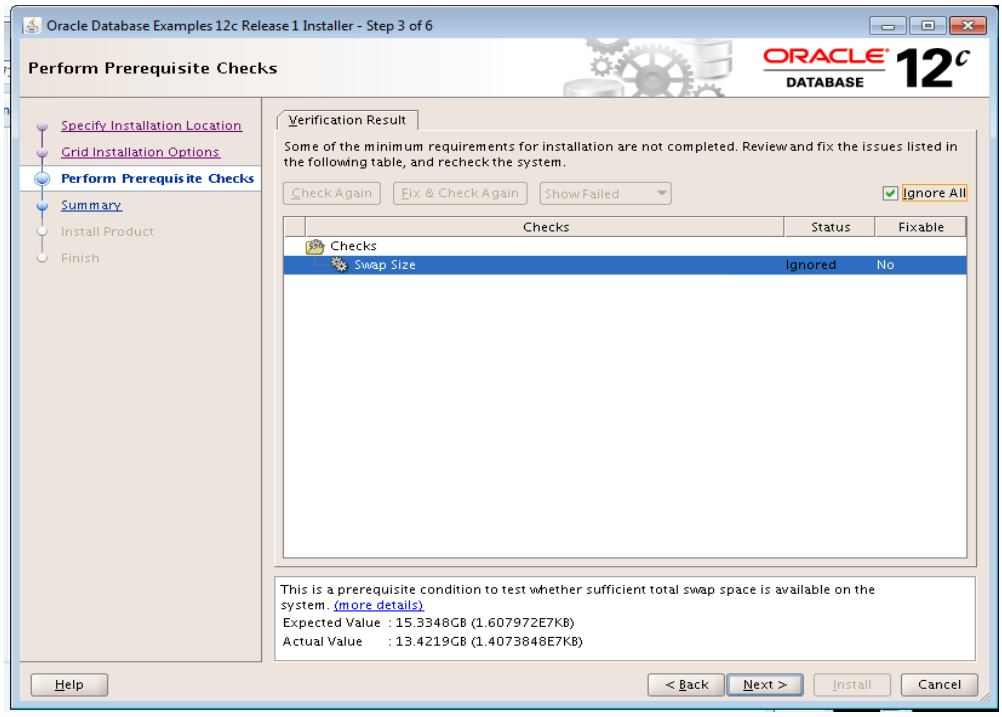


4. The Grid Installation option window opens. Uncheck all servers, and click **Next**.

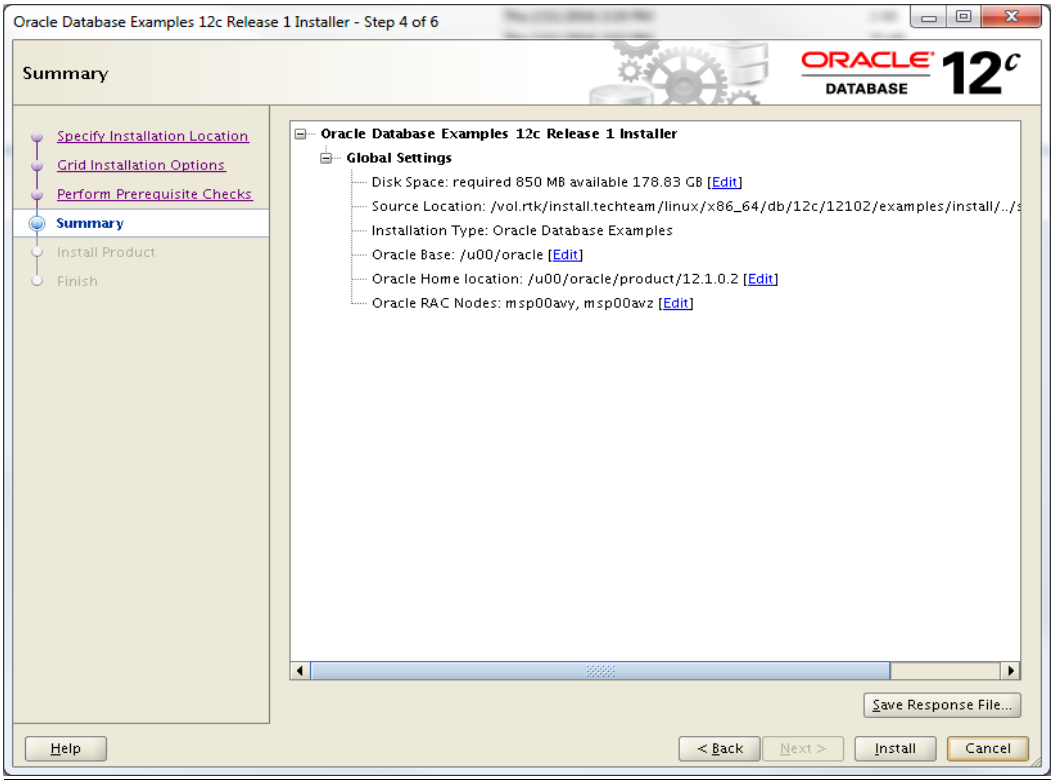


5. The Perform Prerequisite Checks window opens. Evaluate the warnings. If you see any warnings, click the **Fix & Check Again** button to generate a fix-up script. Execute the script, and click **Check Again**.

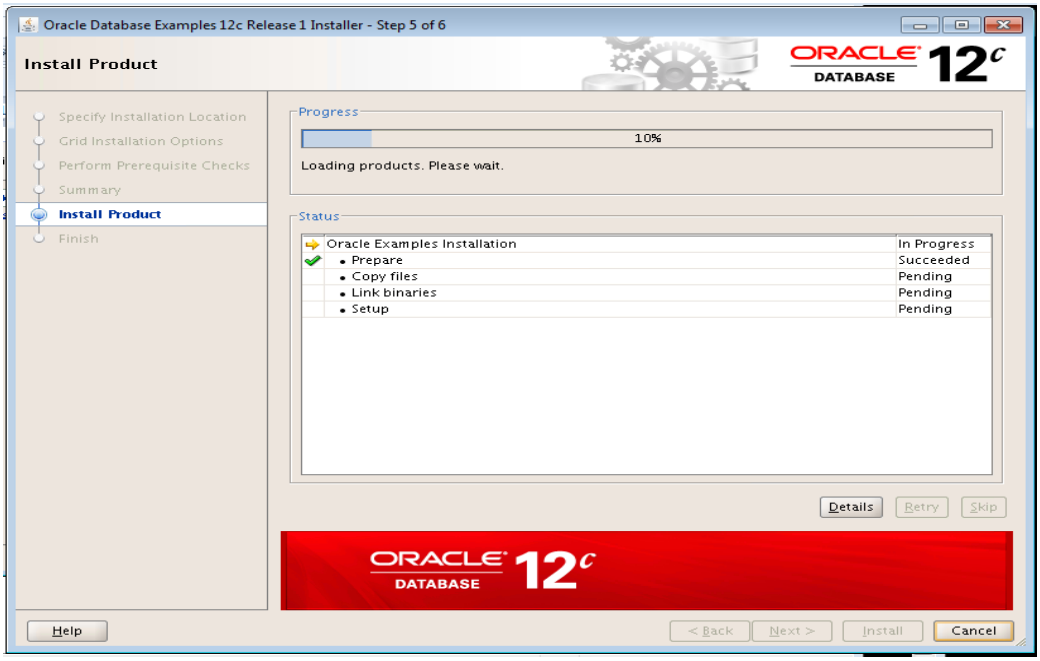
   Alternatively, you can select the 'Ignore All' check box, and click **Next**.



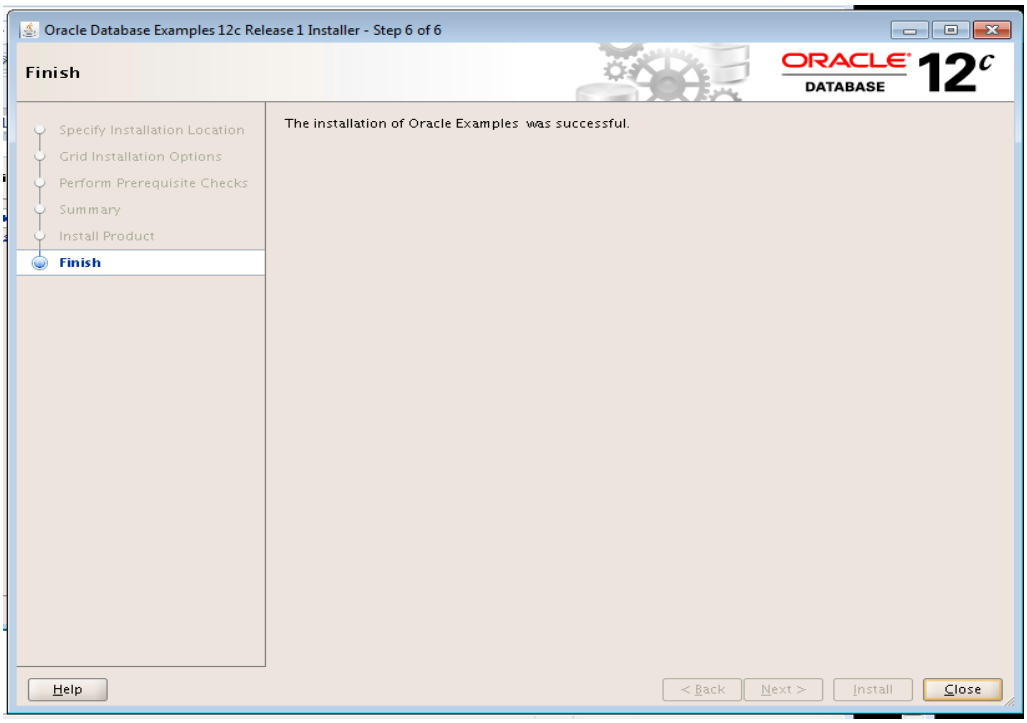
6. The Summary window opens. Click **Install** to begin the installation.



7. The Install Product window opens. The installation progress is displayed in the Progress bar.



8. The Finish window opens, and shows the status of installation. Click **Close** to exit the installer.



JDK 1.7 and Oneoff Patch application

Release 15.0 uses JDK 1.7. Follow the instructions below to configure the database to use JDK1.7.

http://docs.oracle.com/database/121/JJDEV/chone.htm#JJDEV01600

Oneoff Patches:

a. 20846438: ORA-600 [KKPAPXFORMFKK2KEY_1] WITH LIST PARTITION

b. 19623450: MISSING JAVA CLASSES AFTER UPGRADE TO JDK 7

c. 20406840: PROC 12.1.0.2 THROWS ORA-600 [17998] WHEN PRECOMPILING BY 'OTHER' USER

d. 20925154: ORA-39126: WORKER UNEXPECTED FATAL ERROR IN KUPW$WORKER GATHER_PARSE_ITEMS JAVA

RAC Only:

e. 21260431: APPSST 12C : GETTING ORA-4031 AFTER 12C UPGRADE

f. 21373473: INSTANCE TERMINATED AS LMD0 AND LMD2 HUNG FOR MORE THAN 70 SECS

> **Note:** The default version of JDK is 1.6. After installing the 12.1.0.2 binary, apply patch 19623450. Follow the instructions on Oracle Database Java Developer's Guide 12c Release 1 to upgrade JDK to 1.7. The guide is available at: http://docs.oracle.com/database/121/JJDEV/chone.htm#JJDEV01000.

Create ASM Instances and Diskgroups

This section describes how to create ASM instances and diskgroups.

Create the Raw File Systems

Login as "Oracle", and on one of the servers, create the raw partitions that will be used for the ASM diskgroup.

```
dd if=/dev/zero of=/u03/oradata/data1 bs=8192 count=2000000
dd if=/dev/zero of=/u03/oradata/data2 bs=8192 count=2000000
dd if=/dev/zero of=/u04/oradata/data1 bs=8192 count=2000000
dd if=/dev/zero of=/u04/oradata/data2 bs=8192 count=2000000
dd if=/dev/zero of=/u05/oradata/data1 bs=8192 count=2000000
dd if=/dev/zero of=/u05/oradata/data2 bs=8192 count=2000000
```

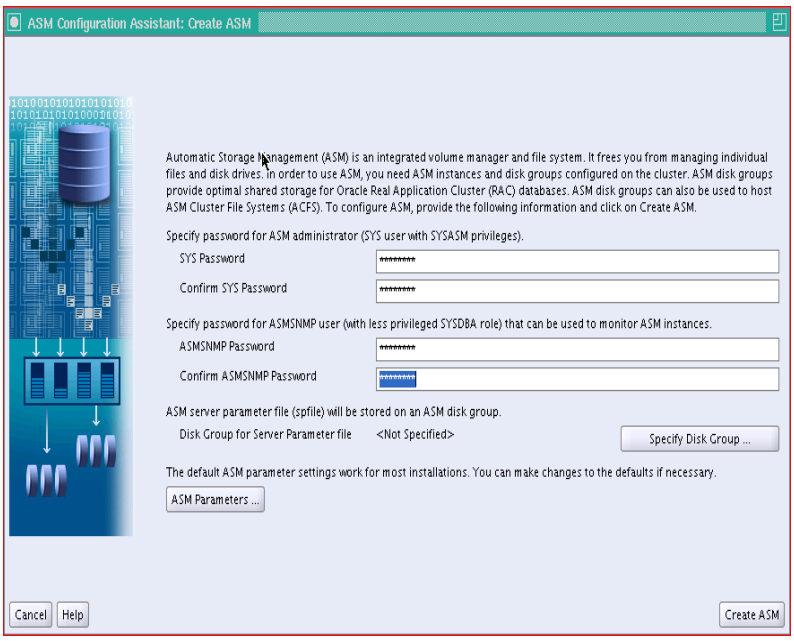
Create the ASM Instance

Navigate to the home directory of CRS, to begin the creation of the ASM instances:

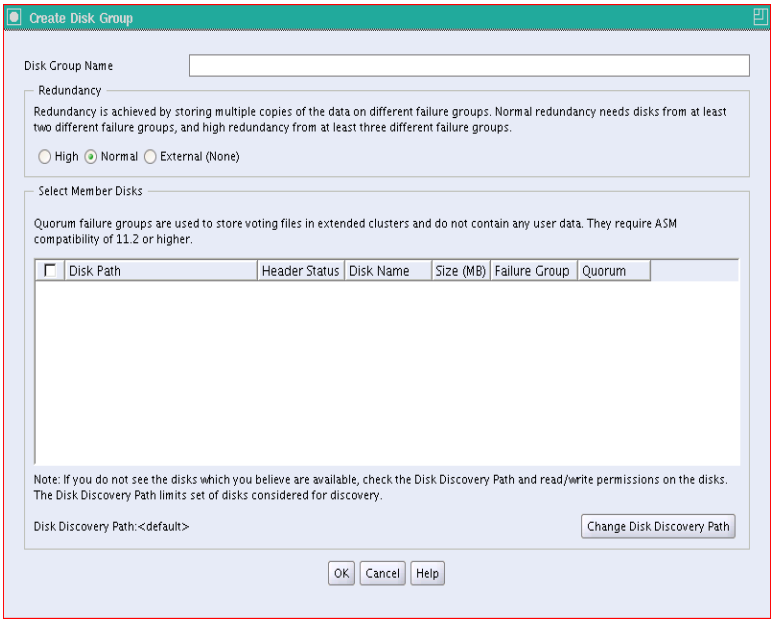
```
export DISPLAY=<ip address:0.0>
export ORA_CRS_HOME=<home_of_crs>
export TMP=<dir_has_enough_space>/install_temp
export TMPDIR=$TMP

$ORA_CRS_HOME/bin/asmca
```

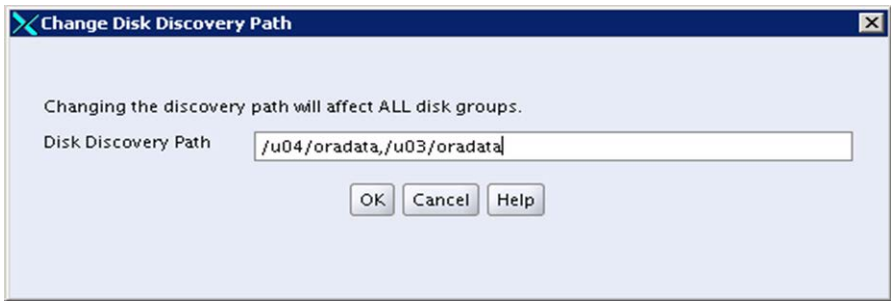
9. The Create ASM window opens. Fill in the 'SYS Password and Confirm SYS Password' input fields. Also, fill in the 'ASMSNMP Password' and 'Confirm ASMSNMP Password' input fields, and click **Specify Disk Group.**



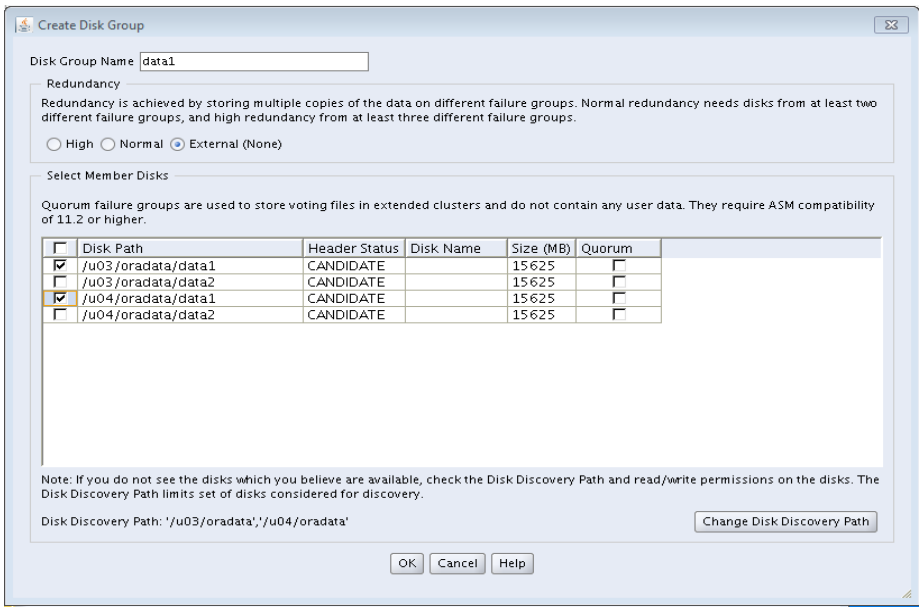
10. The Create Disk Group window opens. Click **Change Disk Discovery Path** to configure disk groups.



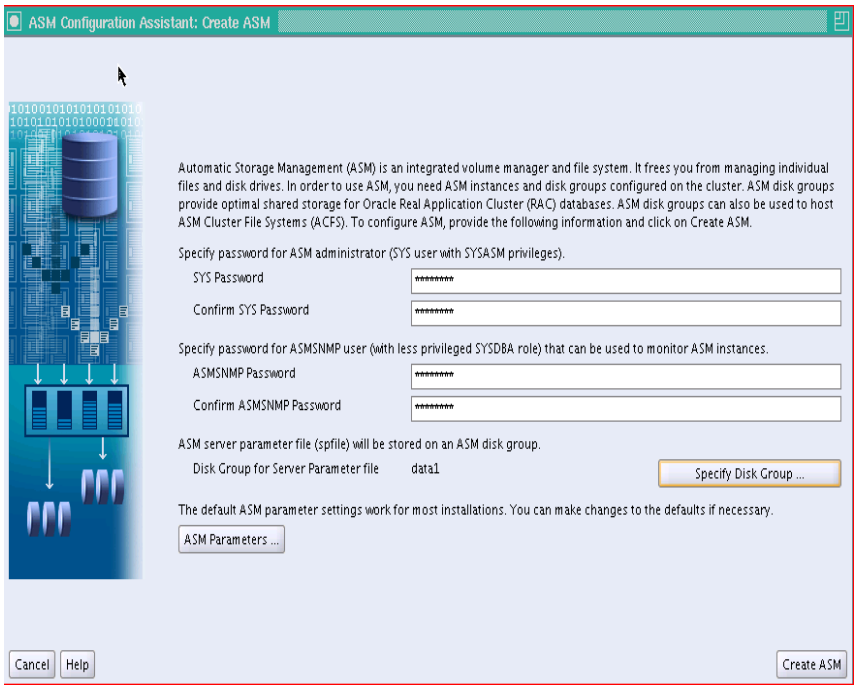
11. The Change Disk Discovery Path window opens. Enter the paths in the **Disk Discovery Path** input field, and Click **OK**. All the raw partitions on the specified paths will be retrieved and displayed on the next screen.



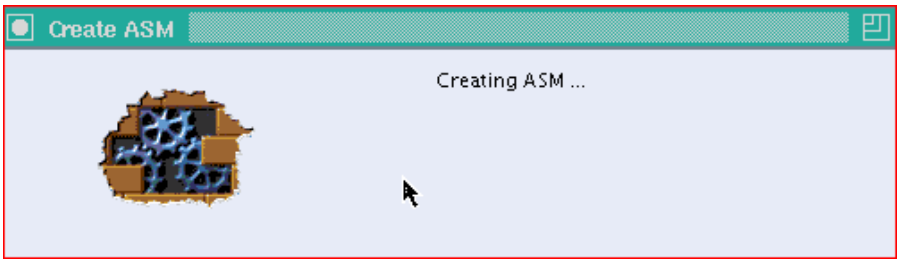
12. The Create Disk Group window opens. Check the disks that you would like to assign to the diskgroup, and click **OK** to return to the previous screen.



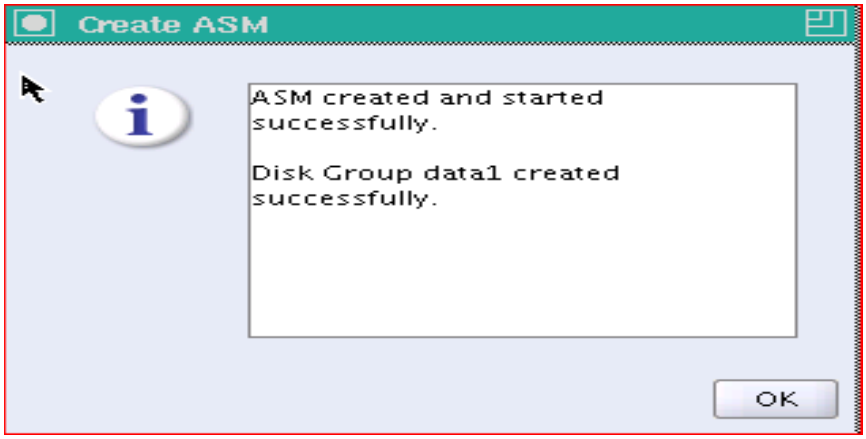
13. The ASM Configuration Assistant: Create ASM window opens. Click **Create ASM** to initiate the creation of the first diskgroup and the ASM instance.



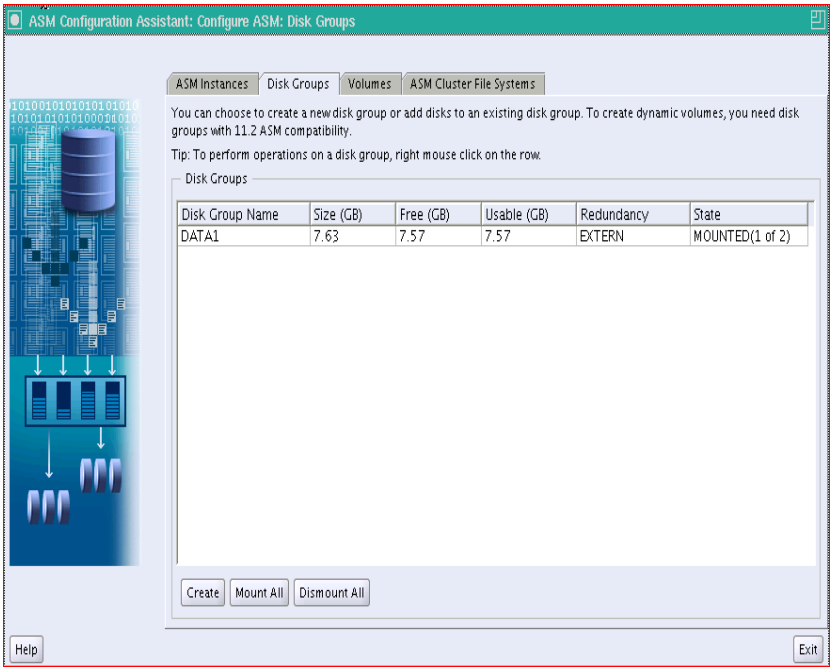
14. The Create ASM window opens displaying the progress of the instance.



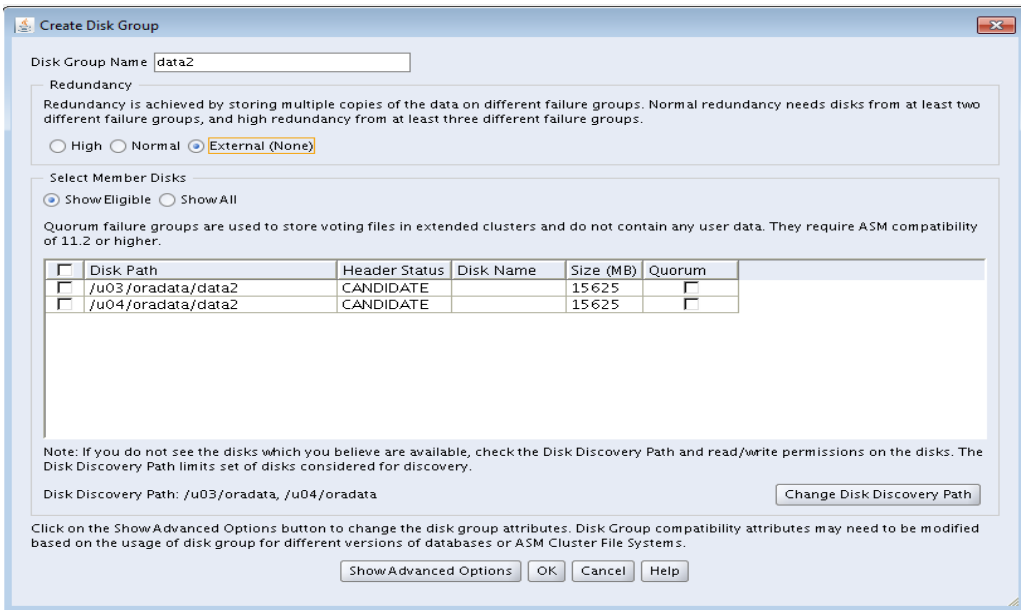
15. When complete, the completion status is displayed in a pop-up window. Click **OK** to proceed to the next screen.



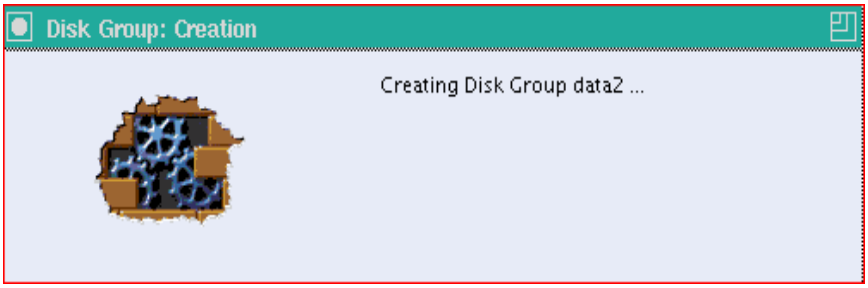
16. The ASM Configuration Assistant: Configure ASM: Disk Groups window opens. The newly created disk group is displayed in a table. To create additional disk groups, click **Create** from the **Disk Groups** tab.



17. The Create Disk Group window opens. Enter the name of the disk group in the **Disk Group Name** input field. Select **High/Normal/External (None)** for Redundancy. Check the disks that you wish to assign to this group and click **OK** to start the creation of the disk group.

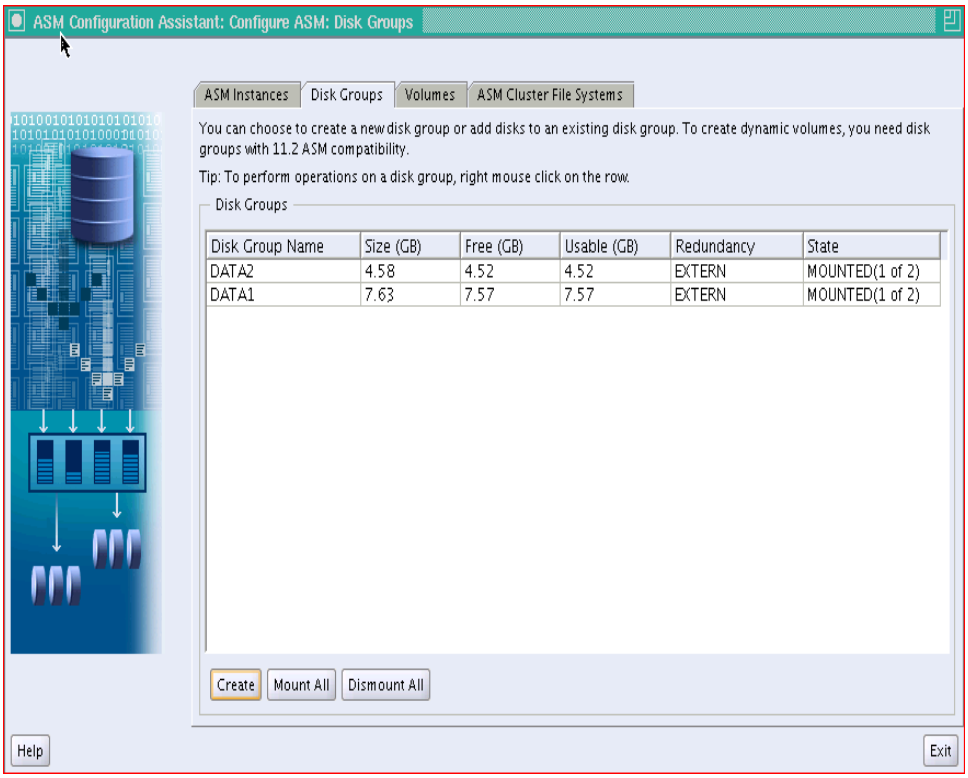


18. The Disk Group: Creation window opens. The status of creation of the Diskgroup data2 is displayed in the Progress bar.



19. The completion status is displayed in a pop-up window. Click **OK** to return to the ASM Configuration Assistant: Configure: ASM Disk Groups window.

20. The created disk groups are displayed in a table. The installation is now complete. Click **Exit** to close this window.



7

Configuring SPFILES on Shared Devices with Raw File Systems

The ASM configuration assistant creates a default spfile. We will create a pfile from the spfile and create an alias for the spfile which resides on the shared device with a raw file system.

1. Create an alias for the default SPFILE created in the device with the raw file system.

   With the SID pointing to the ASM instance, find the location of the default SPFILE created by ASM.

   a. Login to sqlplus as sysdba.

   b. Enter: show parameter spfile

   c. The details are displayed as follows:

```
NAME         TYPE                VALUE

                                 msp-RND-Cluster/ASMPARA
spfile       string              METERFILE/registry.253.870871607
```

   d. Create an alias for the SPFILE using the ASM command line language.

```
--> asmcmd
ASMCMD> cd +DATA1/msp-RND-Cluster
ASMCMD> mkalias +DATA1/ msp00av-
Cluster/ASMPARAMETERFILE/registry.253.870871607 spfileasm.ora
ASMCMD> ls -l
Type              Redund  Striped  Time             Sys  Name
                                                    Y    ASMPARAMETERFILE/
ASMPARAMETERFILE  UNPROT  COARSE   FEB 05 12:00:00  N    spfileasm.ora =>
+DATA1/msp00av-Cluster/ASMPARAMETERFILE/REGISTRY.253.870871607
```

2. Log in as sysdba user, and prepare a backup pfile with the necessary parameters for the instances:

```
-> create pfile='/u00/crs/oracle/product/12.1.0.2/dbs/init$ORACLE_SID.ora'
from spfile';
```

   a. Ensure the PFILE has values for:

```
asm_diskgroups='diskgroup names separated by comma'
asm_disstring='disk_string_names separated by comma'
instance_type='asm'
```

   b. Rename this file as a backup pfile.

```
-> mv init$ORACLE_SID.ora init$ORACLE_SID.ora.bak
```

3. Create another PFILE for the instance that contains one line pointing to the SPFILE on the raw device as shown below:

```
--> cat /u00/crs/oracle/product/12.1.0.2/dbs/init+ASM1.ora
spfile='+DATA1/msp00av-Cluster/spfileasm.ora'
```

> **Note:** Instead of pointing to the name REGISTRY.253.870871607, it points to a more readable alias filename.

4. Log on as a sysasm user, and then restart the first instance to ensure the correct SPFILE is used:

```
shutdown immediate
startup
show parameter diskgroup
show parameter instance
```

5. Repeat steps 3 and 4 to create a more readable pfile for the remaining instances.

8

Create Database

Create a Container Database Using Database Configuration Assistant Interactively

This section illustrates the creation of a retail container database. Both the retail and the OID container databases are created in the same manner. Each container database has two pluggable databases which can be named according to their purposes. For our example, we will create a retail database with two pluggable databases.

This section assumes the following environments are used for the RAC database:

```
Database Name: retaildb
Container: true
Datafile paths: +data1/*.dbf, +data2/*.dbf, +data3/*.dbf
Archivelog path: /arch/retaildb
Host1 Name: dbserver1
Instance1 Name: retaildb1
pdb name: retailapp, retailrib
Host2 Name: dbserver2
Instance2 Name: retaildb2
```

Prerequisites

Oracle 12.1.0.2 binary with the required patched must be already installed in the predefined $ORACLE_HOME.

Instance Creation

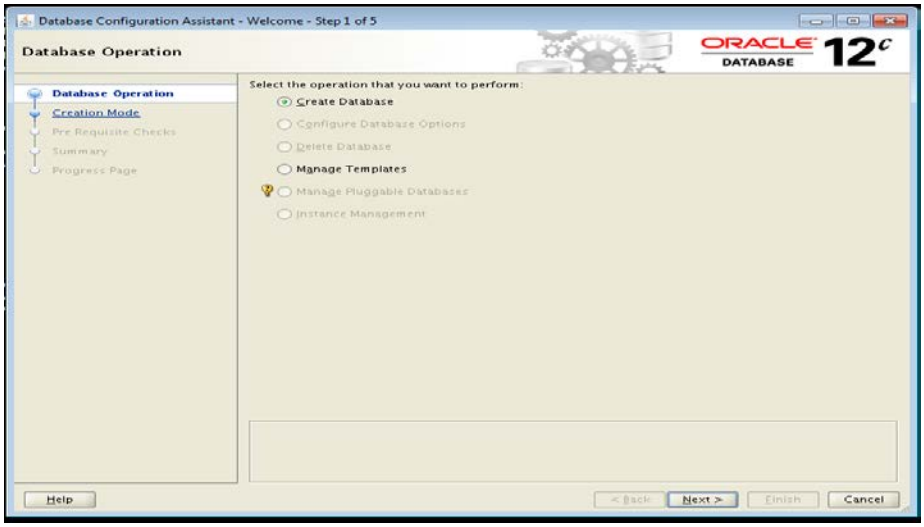
Log into UNIX server as 'oracle' user, or any other account, who owns the Oracle Database software.

Creating a Container Database Interactively Using the Database Configuration Assistant (DBCA)

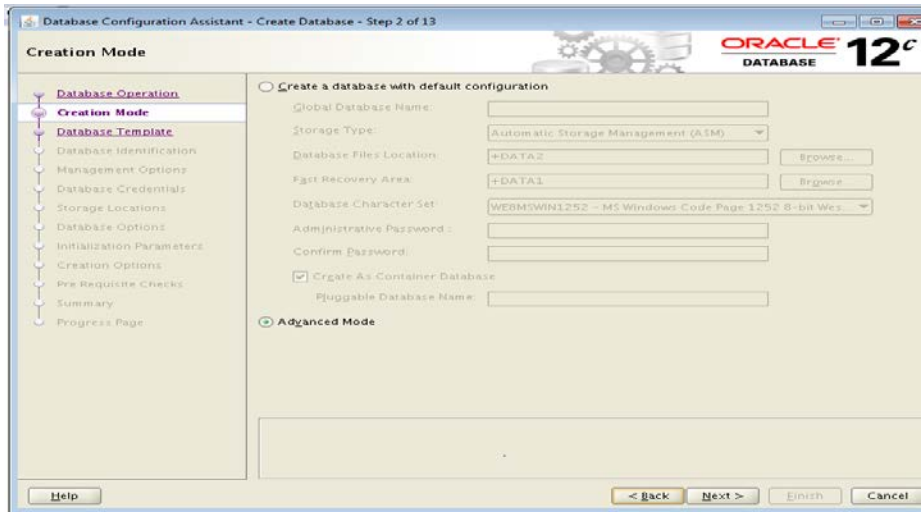
To install interactively using Database Configuration Assistant (DBCA), launch the DBCA using the following commands:

```
$> cd $ORACLE_HOME/bin
$> export DISPLAY=<set DISPLAY>
$> ./dbca
```

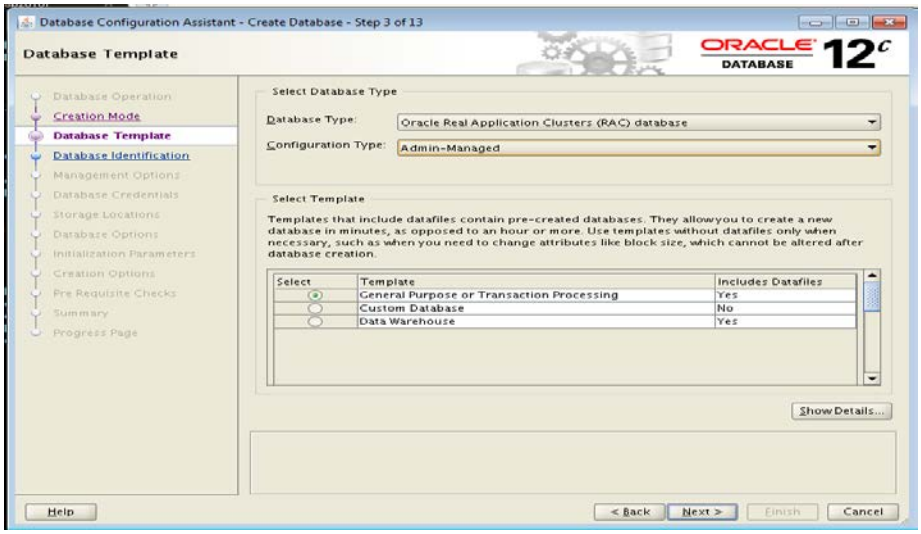
6. The Database Operations window opens. Select 'Create Database', and click **Next**.



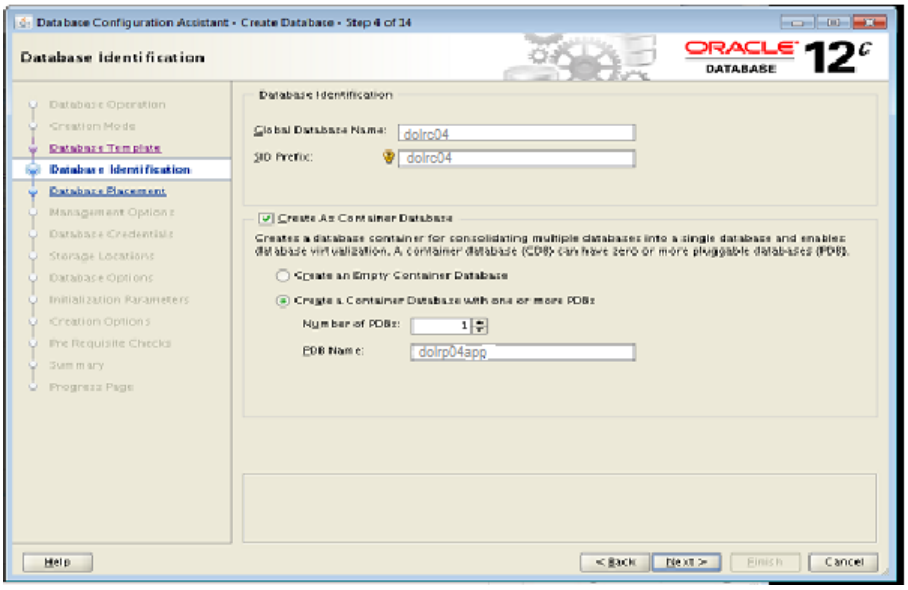
7. The Creation Mode window opens. Select 'Advanced Mode', **and** click **Next**.



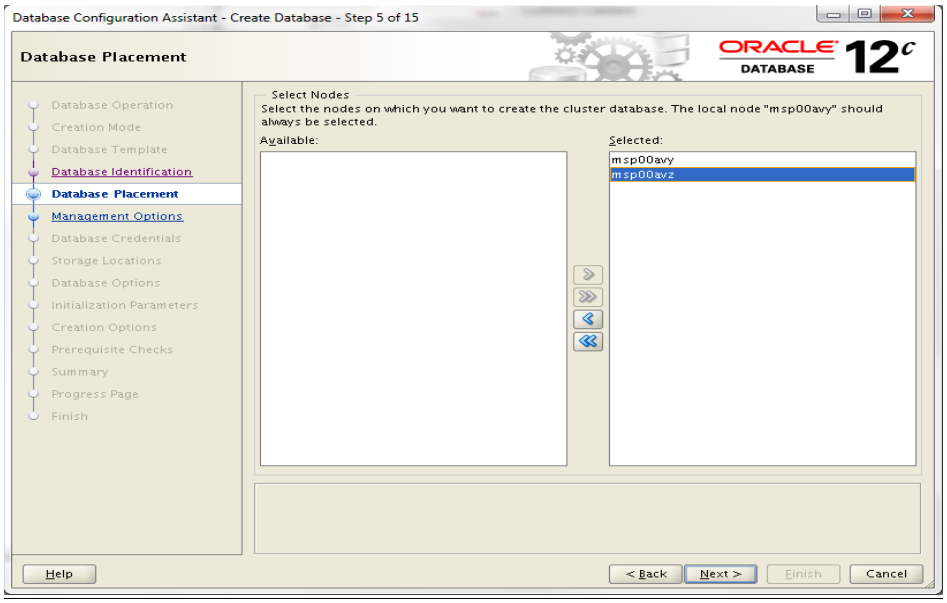
8. The Database Templates window opens. Select 'Admin-Managed' from the 'Configuration Type' drop-down list. Select the 'General Purpose or Transaction Processing' radio button from the table, and click **Next**.



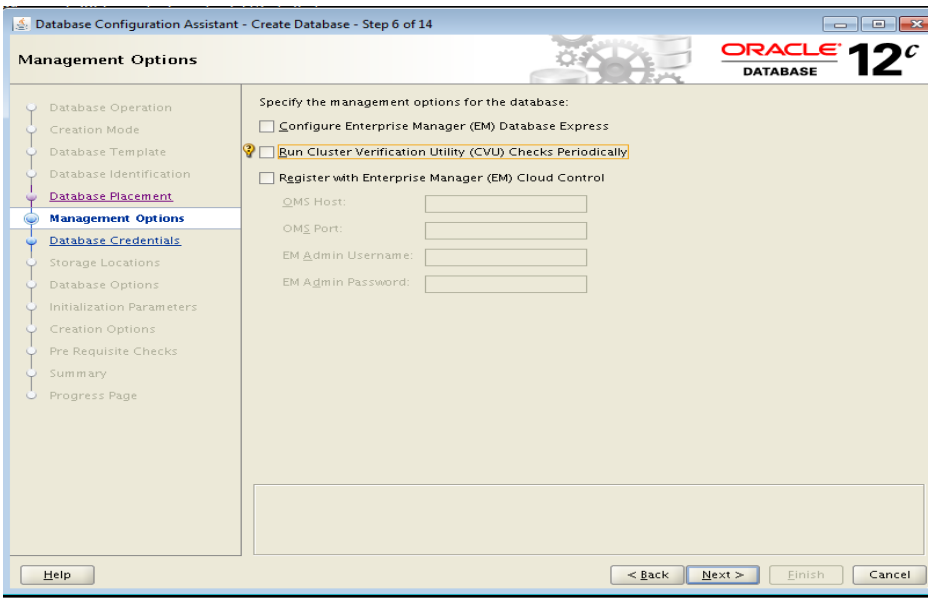
9. The Database Identification window opens. Enter the database name in the **Global Database Name** input field. Select the **Create a Container Database with one or more PDB?** Radio button, and enter the pluggable database name in the **PDB Name** input field, and click **Next**.



10. The Database Placement window opens. Ensure all servers have been selected, and click **Next**.

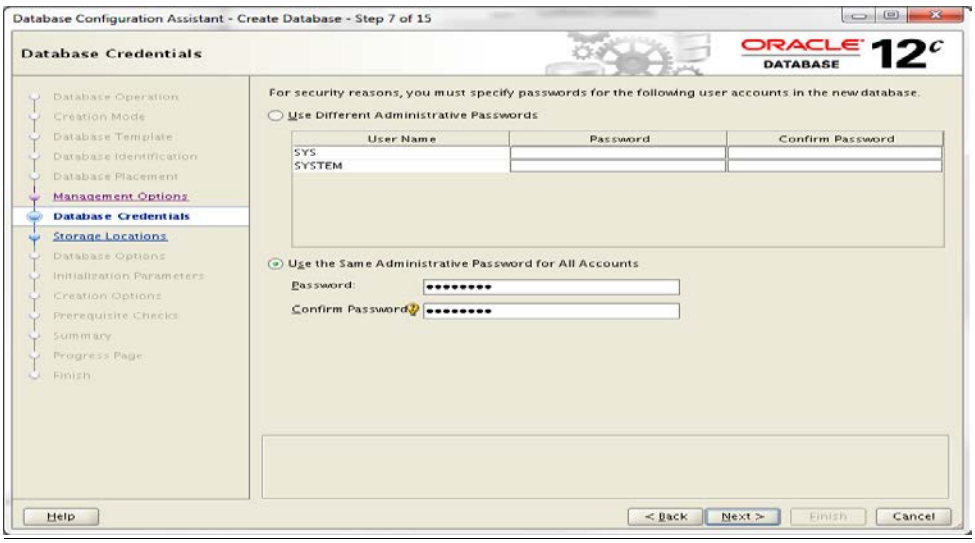


11. The Management Options window opens. For this example, do not check any options, and click **Next** to continue.

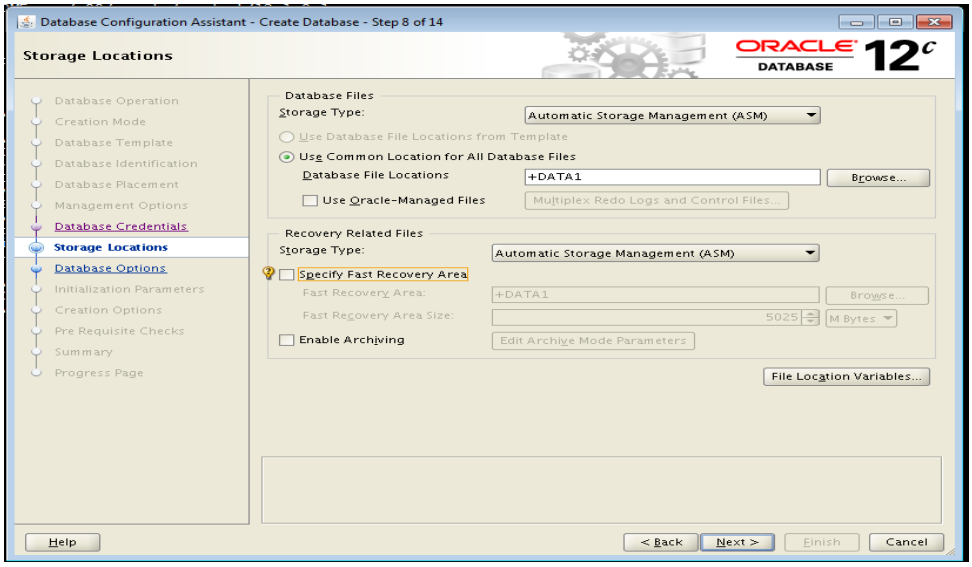


12. The Database Credentials window opens.

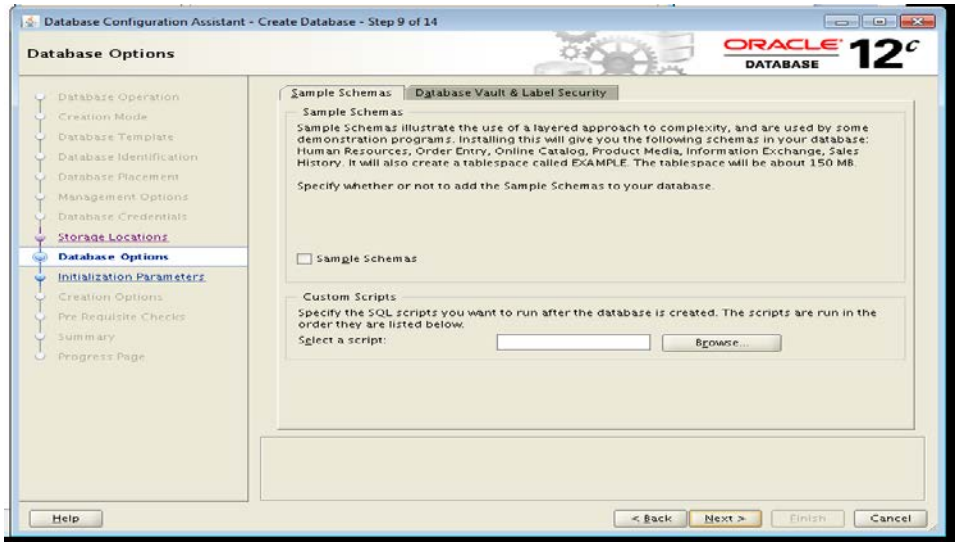
- **a.** Select the 'Use the Same Administrative Passwords for All Accounts' radio button
- **b.** Enter password in the 'Password' input field.
- **c.** Enter the password again in 'Confirm Password' input field, and click **Next**.



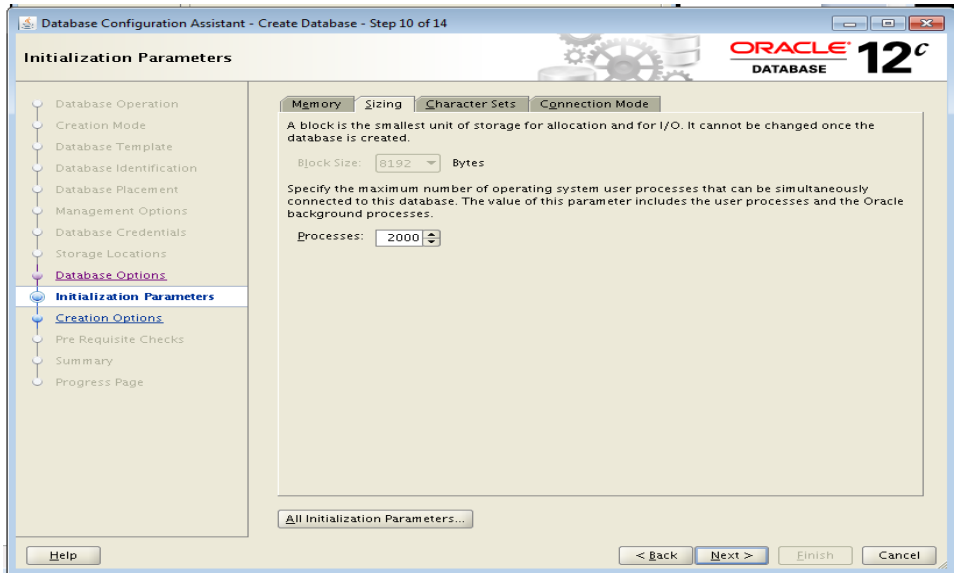
13. The Storage Location window opens.
    a. Select **Automatic Storage Management (ASM) from the Storage Type** drop-down list.
    b. Select the **Use Common Location for All Database Files** radio button. Click Browse, select +DATA1 for Database File Locations, and click **Next**.



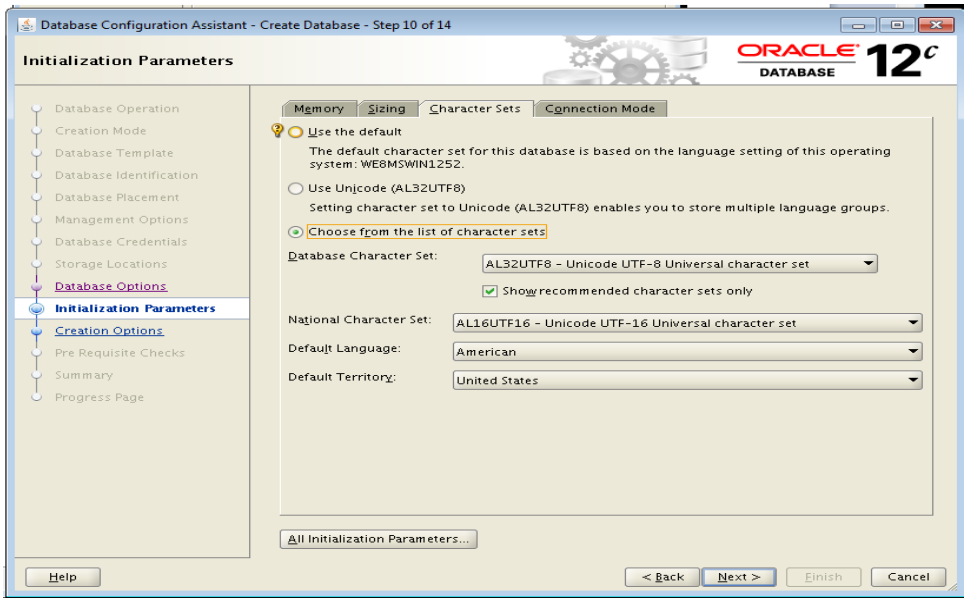
14. On the **Sample Schemas** tab, we neither need an example schema, nor want to use the database vault, so click Next.



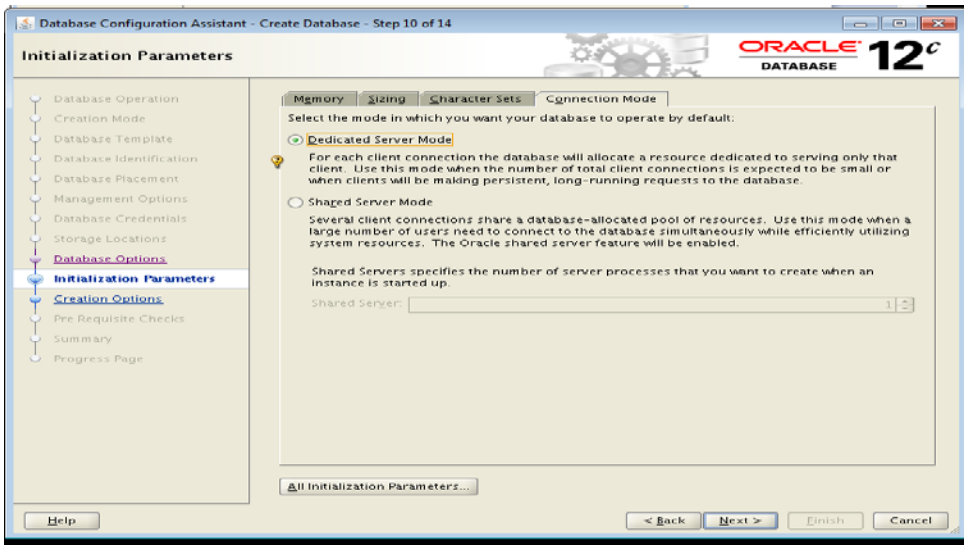
15. On the **Sizing** tab, ensure the number of Processes is 2000, and then click the **Character Sets** tab.



16. On the **Character Sets** tab, select the Choose from the list of character sets radio button.
    a. Select AL32UTF8 from the Database Character set drop-down list.
    b. Select the Show recommended character sets only check box, to expand the list of available character sets.
    c. Select the **Connection Mode** tab.

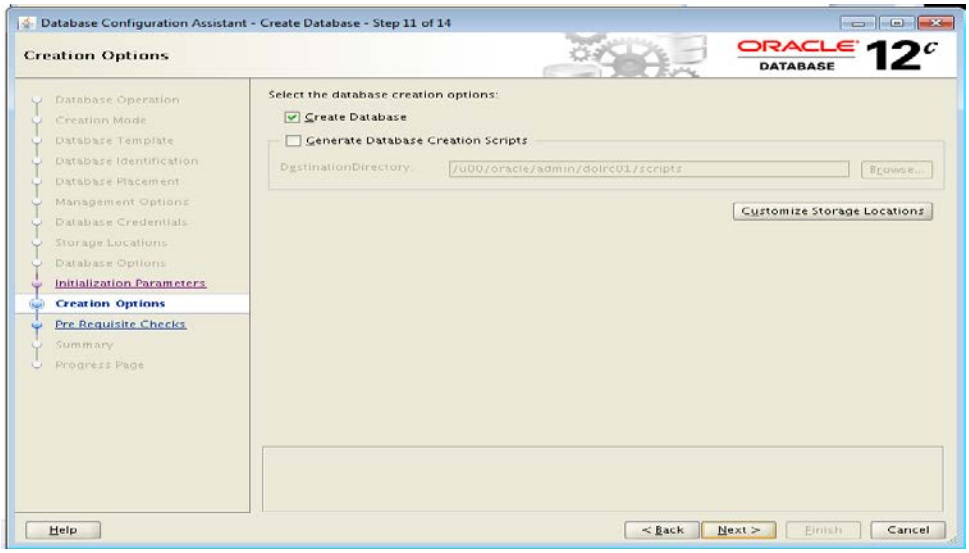


17. On the **Connection Mode** tab, ensure Dedicated Server Mode is checked, and click **Next**.

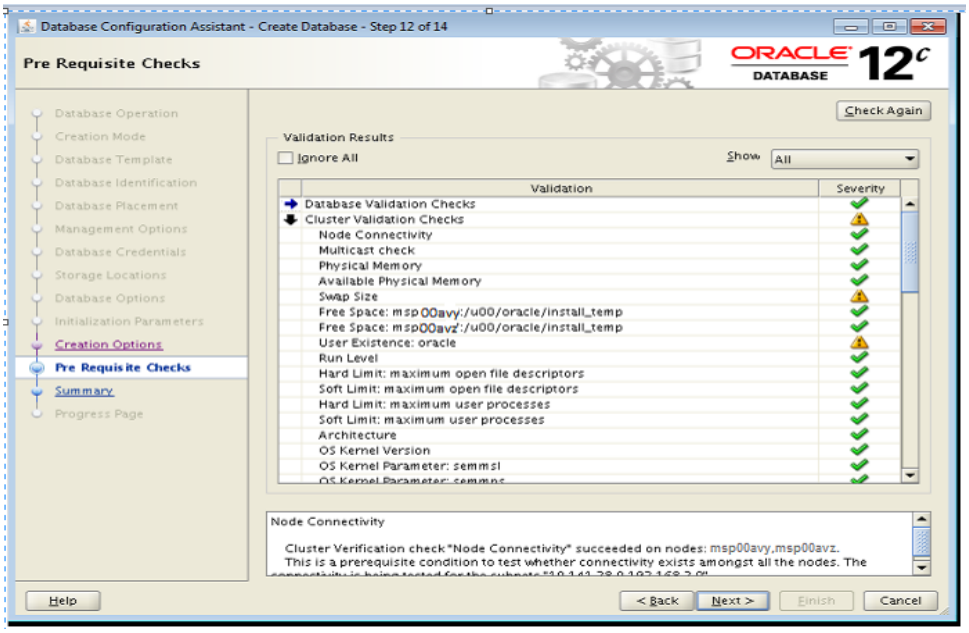


18. The Creation Options window opens. Select the 'Create Database' radio button, and click **Next.**

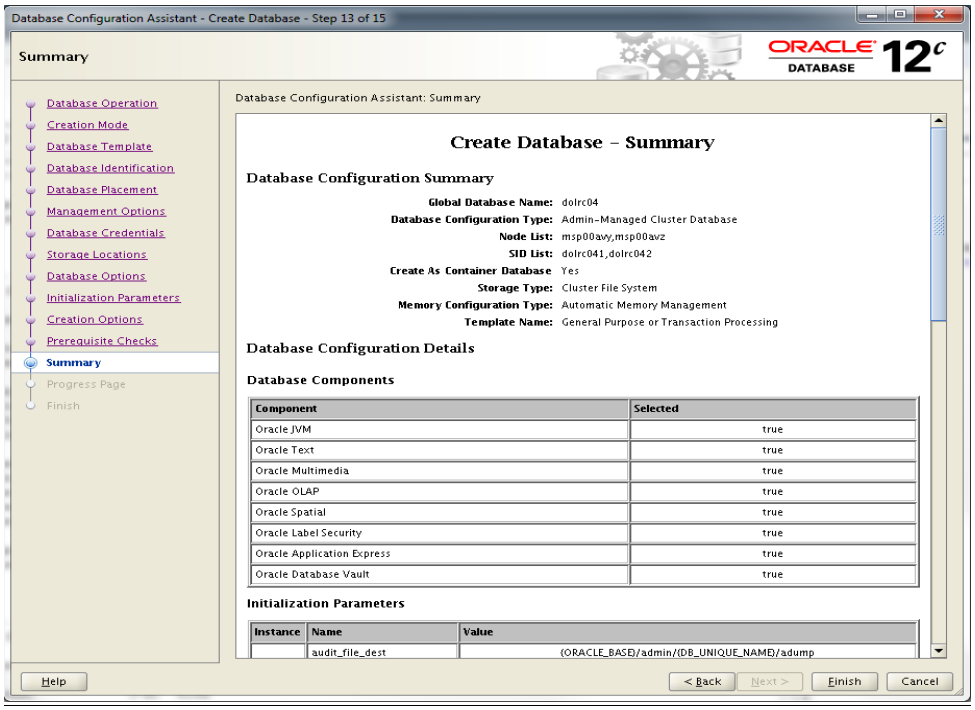
> **Note:** If you wish DBCA to generate the database creation scripts, then select 'Generate Database Creation Scripts'.



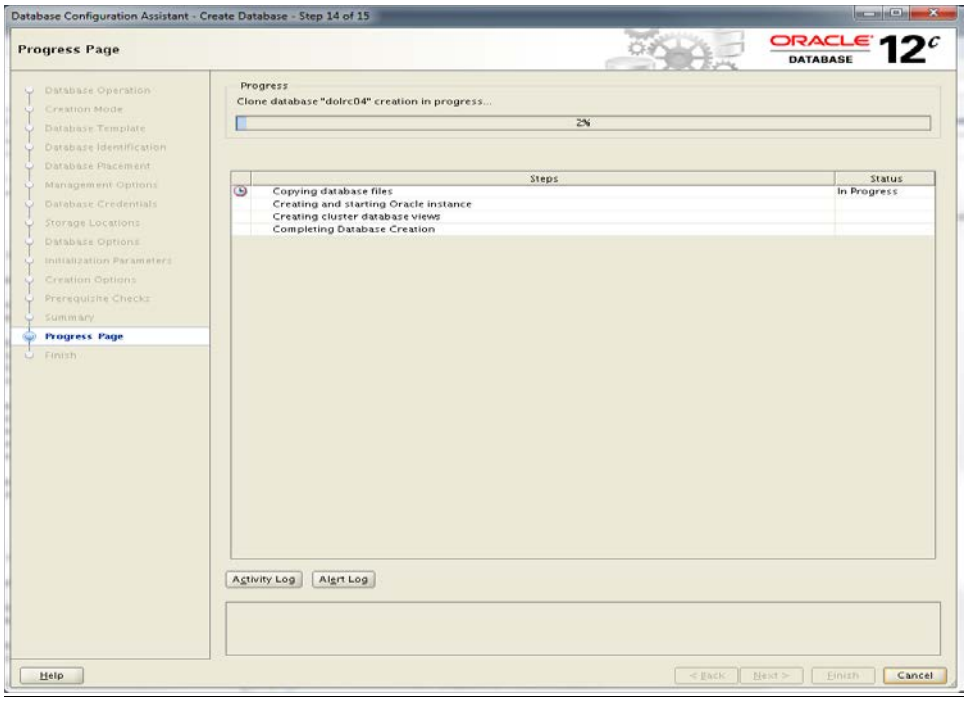
19. The Pre Requisite Checks window opens. Click **Next**.



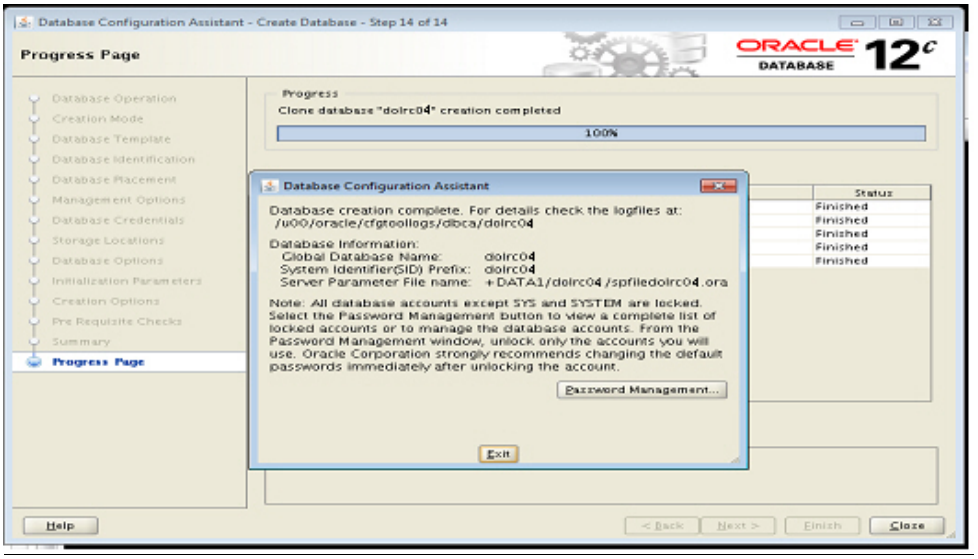
20. The Summary window opens. Click **Next**.



21. The Creation Options window opens. The progress of database creation is displayed in the Progress bar.



22. The creation of a container database with one pluggable database is complete.



Post Database Creation

Controlfile multiplexing

If the controlfiles are not multiplexed, follow the steps below to multiplex at least one.

Log into sqlplus as sys and execute the following statement:

select name from v$controlfile;

```
NAME
---------------------------------------------
+DATA1/dolrc04/control01.ctl
+DATA1/dolrc04/control02.ctl
```

Execute the following command to update the controlfile location in the spfile

```
alter system set control_files='+DATA1/dolrc04/control01.ctl','+data2'
scope=spfile sid='*';
```

Shutdown the RAC database and open in nomount mode:

```
srvctl stop database –d dolrc04
srvctl start database -d dolrc04 -o nomount
```

Log into rman to copy the controlfile to a new destination:

```
rman target /
restore controlfile to '+data2' from '+DATA1/dolrc04/control01.ctl';;
```

A copy of the controlfile would have been copied into +data2.

Log into ASM disk using the asmcmd, to find out the exact name and location of the file.

Log into sqlplus as sys again, to update the full name of the newly copied controlfile.

```
alter system set
control_files='+DATA1/dolrc04/control01.ctl','+data2/dolrc04/CONTROLFILE/current.2
83. current.283.871999251’  scope=spfile sid='*';
```

Bounce the RAC database:

```
srvctl stop database -d dolrc04
srvctl start database -d dolrc04
```

When the RAC database restarts, check to ensure the location is correct:

```
SQL> select name from v$controlfile;
NAME
--------------------------------------------------------------------------------
+DATA1/dolrc04/control01.ctl
```

```
+DATA2/dolrc04/CONTROLFILE/current.283.871999251
```

Redo log multiplexing

If the redo logs are not multiplexed, follow the instructions below to create a member for each group. Log into sqlplus as sys and execute the following statement:

```
select group#, substr(member, 1, 50) member, status
from v$logfile order by group#;
    1 +DATA1/dolrc04/redo01a.log
    2 +DATA1/dolrc04/redo02a.log
    3 +DATA1/dolrc04/redo03a.log
    4 +DATA1/dolrc04/redo04a.log
```

There is only one member in each group, and we need to have at least 2 members. Execute the statement below to add a new member to each group:

```
ALTER DATABASE ADD LOGFILE MEMBER '+data2' TO GROUP 2;
```

Create Pluggable Database Auto-Open Trigger

After the container database is bounced, the pluggable database is not open by default. As sysdba, create a trigger to open the pluggable database at database startup time.

```
CREATE or REPLACE trigger OPEN_ALL_PLUGGABLES
   after startup
   on  database
BEGIN
   execute immediate 'alter pluggable database all open';
END open_all_pdbs;
/
```

Adjust Database Parameters

DBCA creates databases with a set of predefined parameters. Customize these parameters so they are more suitable for your environment.

1. Create a temporary pfile in the multi-instance format for editing. Execute these commands using sqlplus against instance 1:

   ```
   create spfile='/tmp/spfileretaildb1.ora' from pfile;
   create pfile='/tmp/initretaildb1.ora' from spfile='/tmp/spfileretaildb1.ora';
   ```

2. Edit the temporary pfile and change parameters for instance 1.

Change these parameters:

```
*.open_cursors=300
*.pga_aggregate_target=300m
dolrc041.sga_target=<dbca pre-defined value>
dolrc042.sga_target=<dbca pre-defined value>
```

To these values:

```
*.open_cursors=900
*.pga_aggregate_target=<appropriate value for your environment>
dolrc041.memory_target=<appropriate value for your environment>
dolrc042.memory_target=<appropriate value for your environment>
```

Add these parameters:

```
*.streams_pool_size=<appropriate value for your environment>
dolrc041.local_listener='(DESCRIPTION=(ADDRESS_LIST=(ADDRESS=(PROTOCOL=TCPS)(H
OST=msp00avy-vip)(PORT=2484))(ADDRESS=(PROTOCOL=TCP)(HOST=msp00avy-
vip)(PORT=1521))))'
```

```
dolrc042.local_listener='(DESCRIPTION=(ADDRESS_LIST=(ADDRESS=(PROTOCOL=TCPS)(H
OST=msp00avz-vip)(PORT=2484))(ADDRESS=(PROTOCOL=TCP)(HOST=msp00avz-
vip)(PORT=1521))))'
*.log_archive_format='dolrc04%r_%s_%t.log'
*.remote_listener=’msp00avyp-r:1521'
```

Create the spfile on the shared device by executing these commands using sqlplus against instance 1:

```
create spfile='+data3/spfile.ora' from pfile='/tmp/initretaildb1.ora';
```

Test the correctness of the parameter setting by bouncing the database using the new spfile. Either start one instance at a time or use srvctl.

```
srvctl stop database –d dolrc04
srvctl start database –d dolrc04
```

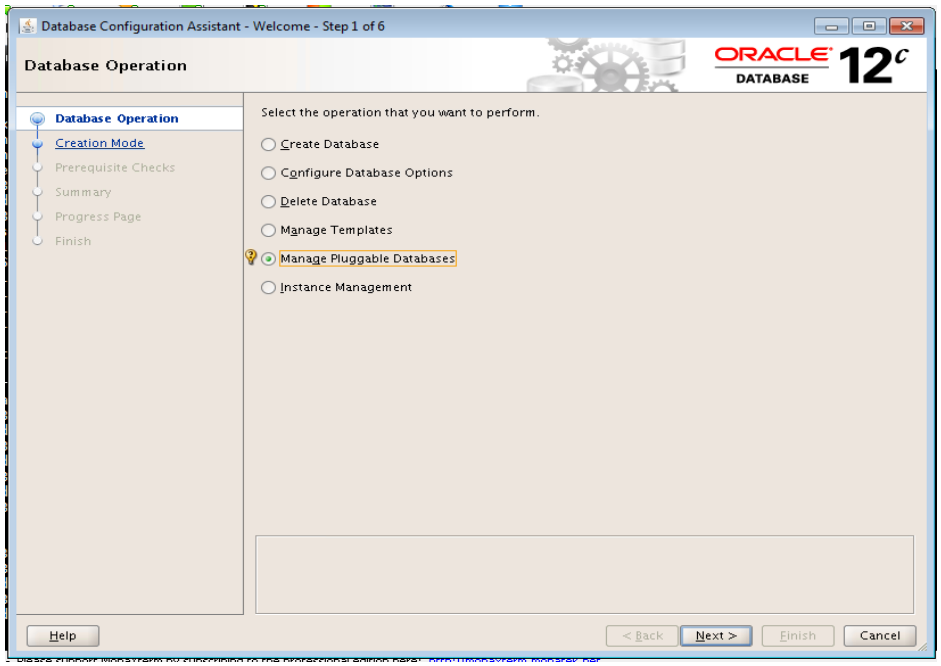
Creating a Pluggable Database Interactively Using the Database Configuration Assistant (DBCA)

To create interactively using Database Configuration Assistant (DBCA), launch the DBCA using the following commands:

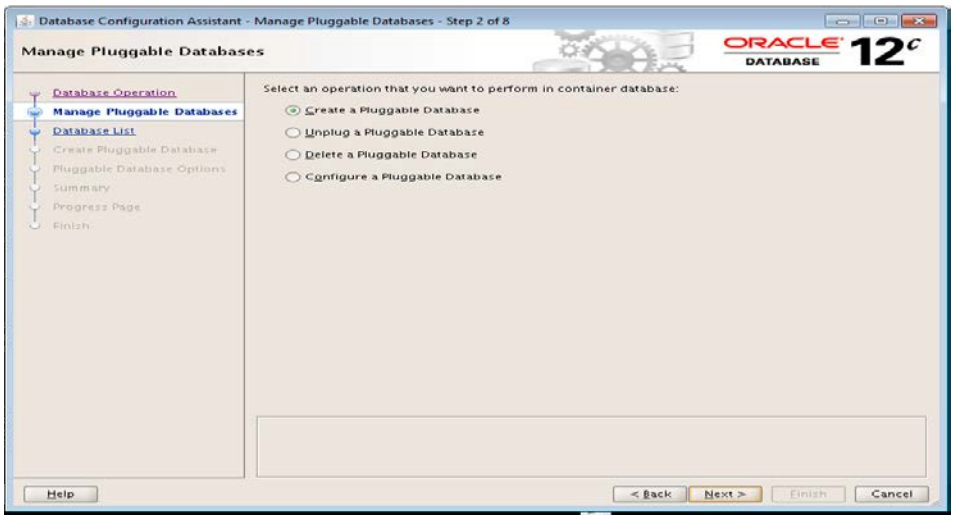
```
$> cd $ORACLE_HOME/bin
$> export DISPLAY=<set DISPLAY>

$> ./dbca
```

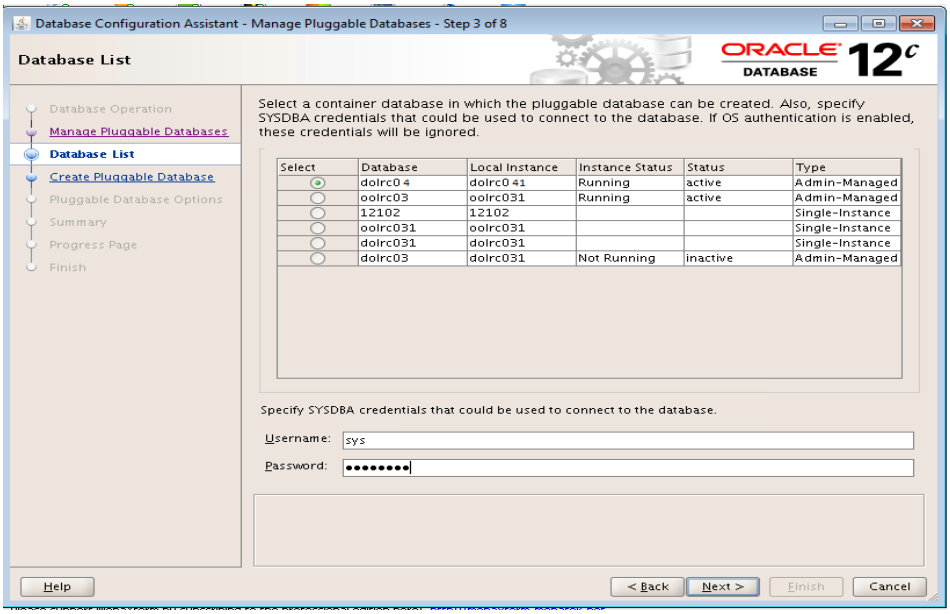
1. The Database Operation window opens. Select Manage Pluggable Databases, and click **Next**.



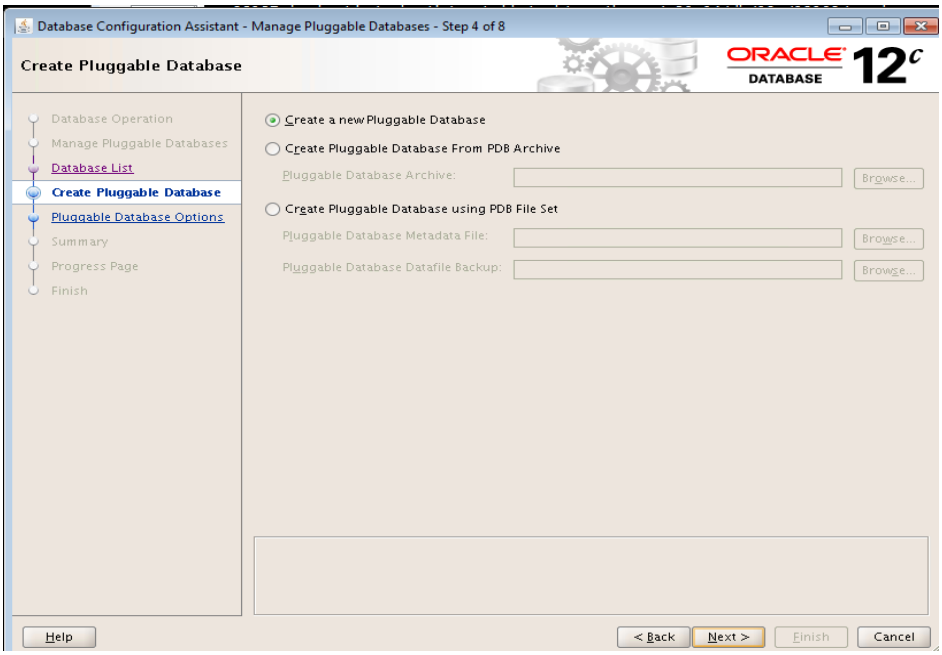
2. The Manage Pluggable Databases window opens. Select the Create a Pluggable Database radio button, and click **Next**.



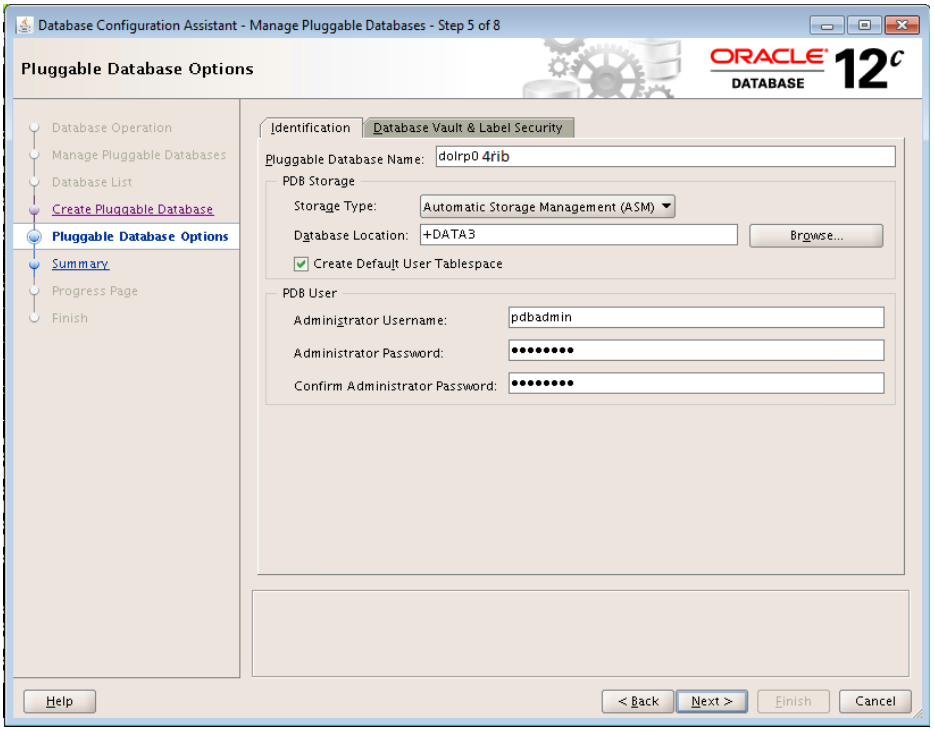
3. The Database List window opens. Check the container database where you wish to create the pluggable database. Enter the user with sysdba privilege and provide a password. Click **Next**.



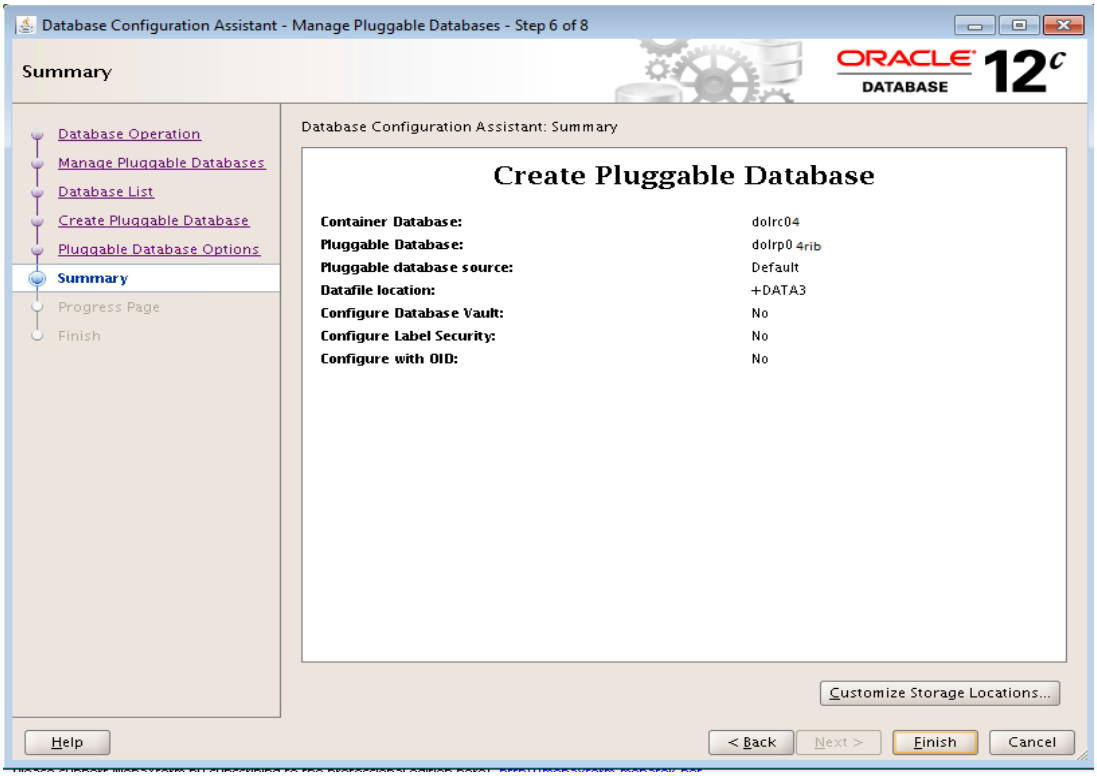
4. The Create Pluggable Database window opens. Check the **Create a new Pluggable Database** radio button, and click **Next**.



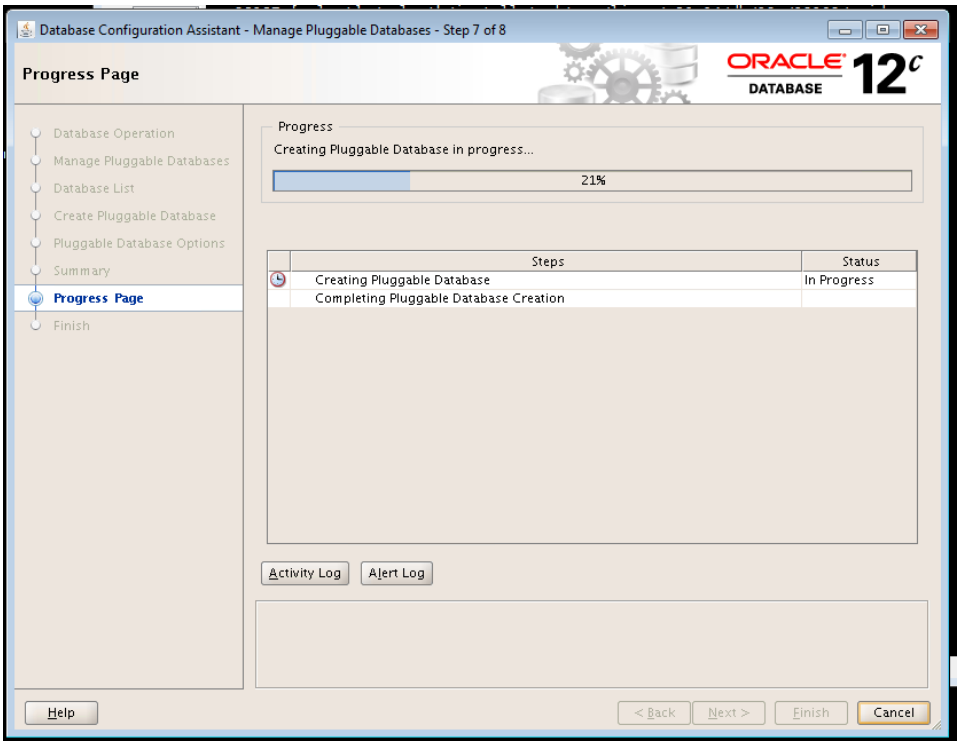
5. The Pluggable Database Options window opens.
   a. Enter a pluggable database name.
   b. Enter a PDB User account name and the password.
   c. Select the options pertaining to your environment, and click **Next**.



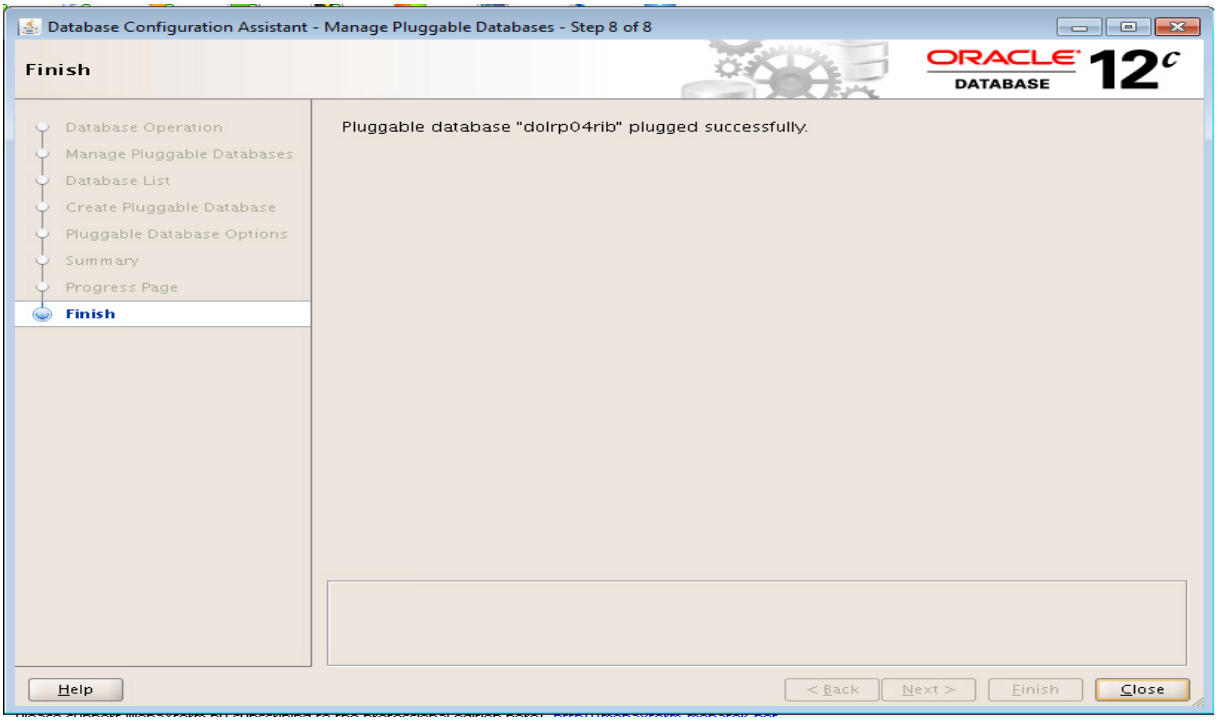
6. The Summary window opens. Click **Finish** to start the creation of the pluggable database.



7. The Progress Page window opens. The database creation progress is displayed in the Progress bar.



8. The Finish window opens. Click **Close** to exit from the installer.


Database Configuration Assistant - Manage Pluggable Databases - Step 8 of 8
Finish
ORACLE DATABASE 12c
Database Operation
Manage Pluggable Databases
Database List
Create Pluggable Database
Pluggable Database Options
Summary
Progress Page
Finish
Pluggable database "dolrp04rib" plugged successfully.
Help
< Back
Next >
Finish
Close

Verify Setup

1. Verify if the v$active_instances show both instances. Execute the following query in sqlplus against each instance:
   `select * from v$active_instances;`
2. Verify the container database and the pluggables:

```
SQL> SELECT NAME, OPEN_MODE, RESTRICTED, OPEN_TIME FROM V$PDBS
  2  /

NAME            OPEN_MODE  RESTRICTED OPEN_TIME
--------------- ---------- ---------- -----------------------------------
PDB$SEED        READ ONLY  NO         20-FEB-16 08.56.33.905 PM -08:00
DOLRP04APP      READ WRITE NO         20-FEB-16 08.56.51.431 PM -08:00
DOLRP04RIB      READ WRITE NO         20-FEB-16 08.56.51.433 PM -08:00
```

3. Verify the database and its instances have been added to CRS. Execute the following command as 'oracle' on either host. You should be able to see the database and the two instances from the output.

```
--> srvctl config database -d dolrc04
Database unique name: dolrc04
Database name: dolrc04
Oracle home: /u00/oracle/product/12.1.0.2
Oracle user: rgbuora
Spfile: +DATA3/DOLRC04/PARAMETERFILE/spfile.253.870871607
Password file: +DATA3//DOLRC04/orapwdolrc04.253.872001301
Domain:
Start options: open
Stop options: immediate
Database role: PRIMARY
Management policy: AUTOMATIC
Server pools:
Disk Groups: DATA3,DATA2
Mount point paths:
Services:
Type: RAC
Start concurrency:
Stop concurrency:
OSDBA group: rgbudba
OSOPER group: rgbudba
Database instances: dolrc041,dolrc042
Configured nodes: msp00avy,msp00avz
Database is administrator managed
msp00avy.us.oracle.com:[sqlnet] /u00/oracle>
-->
```

OID Database Creation

Please proceed to create an OID database in the same fashion, with the following exception:

1. The Processes parameter does not need to be 2000, because OID typically does not have many connections.
2. Memory settings can be smaller for the same reason that ODI databases are not expected to have many connections.
3. No RMS tablespaces need to be created, as all the tablespaces will be created while the applications are being installed via RCU.

9

Create the RMS Tablespace

To create the RMS tablespace, refer Appendix A. Connect to the retail Pluggable Databases and execute create_rms_tablespaces.sql to create the tablespaces via sqlplus in the following manner:

sqlplus rms01@tnsnames

Configure the Network Files

tnsnames.ora file

For container databases, tns entries need to be configured for both the container and each of the pluggable databases created. The pluggable database will be created as a service name, rather than an instance.

1. The following example tnsnames entries can be used to connect to the RAC database (load-balanced across both instances).

```
retaildb =
DESCRIPTION =
(ADDRESS_LIST =
(ADDRESS = (PROTOCOL = TCP)(HOST = scan.domain)(PORT = 1521))
)
(CONNECT_DATA =(SERVICE_NAME = retaildb))
)
```

2. The following example tnsnames entries can be used to connect to a specific instance.

```
retaildb1 =
(DESCRIPTION =
    (ADDRESS_LIST =
    (ADDRESS = (PROTOCOL = TCP)(HOST = scan.domain)(PORT = 1521)))
    (CONNECT_DATA = (SERVICE_NAME = retaildb) (INSTANCE_NAME=retaildb1))
)

retaildb2 =
(DESCRIPTION =
    (ADDRESS_LIST =
    (ADDRESS = (PROTOCOL = TCP)(HOST = scan.domain)(PORT = 1521)))
    (CONNECT_DATA = (SERVICE_NAME = retaildb) (INSTANCE_NAME=retaildb2))
)
--
-- Pluggables ---
retaildbapp =
(DESCRIPTION =
    (ADDRESS_LIST =
    (ADDRESS = (PROTOCOL = TCP)(HOST = scan.domain)(PORT = 1521)))
    (CONNECT_DATA = (SERVICE_NAME = retaildbapp))
)
retaildbrib =
(DESCRIPTION =
    (ADDRESS_LIST =
    (ADDRESS = (PROTOCOL = TCP)(HOST = scan.domain)(PORT = 1521)))
    (CONNECT_DATA = (SERVICE_NAME = retaildbrib))
)
--
```

```
-- If secure listener is used ---
--
retaildbapp_secure =
(DESCRIPTION =
    (ADDRESS_LIST =
    (ADDRESS = (PROTOCOL = TCP)(HOST = scan.domain)(PORT = 2484)))
    (CONNECT_DATA = (SERVICE_NAME = retaildbapp))
)

retaildbrib_secure =
(DESCRIPTION =
    (ADDRESS_LIST =
    (ADDRESS = (PROTOCOL = TCP)(HOST = scan.domain)(PORT = 2484)))
    (CONNECT_DATA = (SERVICE_NAME = retaildbrib))
)
```

LISTENER.ora

For Oracle Database 12g Release 1, there are two listener files:

1. listener.ora
2. endpoints_listener.ora.

The files are located in $ORA_CRS_HOME/network/admin.

3. The following example entries are setup in the listener.ora file of one node.

```
LISTENER_SCAN2=(DESCRIPTION=(ADDRESS_LIST=(ADDRESS=(PROTOCOL=IPC)(KEY=LISTENER
_SCAN2))))
# line added by Agent

SID_LIST_LISTENER=
(SID_LIST =
(SID_DESC = (PROGRAM = extproc)
(SID_NAME = extproc_agent_1521)
(ENVS='EXTPROC_DLLS=ANY')
)
)
LISTENER=
  (DESCRIPTION_LIST =
    (DESCRIPTION =
      (ADDRESS = (PROTOCOL = TCP)(HOST = node1-vip)(PORT = 1521)(IP = FIRST))
      (ADDRESS = (PROTOCOL = TCP)(HOST = node1)(PORT = 1521)(IP = FIRST))
      (ADDRESS = (PROTOCOL = TCP)(HOST = node1-vip)(PORT = 2484)(IP = FIRST))
      (ADDRESS = (PROTOCOL = TCP)(HOST = node1)(PORT = 2484)(IP = FIRST))
      (ADDRESS = (PROTOCOL = IPC)(KEY = extproc_key_1521)))
)
ENABLE_GLOBAL_DYNAMIC_ENDPOINT_LISTENER=ON # line added by Agent
ENABLE_GLOBAL_DYNAMIC_ENDPOINT_LISTENER_SCAN2=ON # line added by Agent
```

The following example shows the setup in the endpoints_listener.ora file of one node. The file is created during CRS installation.

```
LISTENER_dbserver1=(DESCRIPTION=(ADDRESS_LIST=(ADDRESS=(PROTOCOL=TCP)(HOST=
dbvip1)(PORT=1521))(ADDRESS=(PROTOCOL=TCP)(HOST=
dbserver1_IP)(PORT=1521)(IP=FIRST)))) # line added by Agent
```

A

Appendix: Create the RMS Tablespace

This section lists the contents of the create_rms_tablespaces.sql SQL script file that enables you to create the RMS tablespace.

```
-------------------------------------------------------------------------------
Script: create_rms_tablespaces.sql
Execute as: sysdba
Note: Before running this script:
Modify <datafile_path> values.
Modify datafile storage parameters and sizes based on partitioning strategy.
-------------------------------------------------------------------------------
spool create_rms_tablespaces.log
CREATE TABLESPACE RETAIL_INDEX DATAFILE
    '+data1/ retek_index01.dbf'  SIZE 500M
    AUTOEXTEND ON NEXT 100M MAXSIZE 2000M
    EXTENT MANAGEMENT LOCAL
    SEGMENT SPACE MANAGEMENT AUTO
;
CREATE TABLESPACE RETAIL_DATA DATAFILE
    '+data2/retek_data01.dbf'  SIZE 500M
    AUTOEXTEND ON NEXT 100M MAXSIZE 2000M
    EXTENT MANAGEMENT LOCAL
    SEGMENT SPACE MANAGEMENT AUTO
;
CREATE TABLESPACE USERS DATAFILE
    '+data1/users01.dbf'  SIZE 100M
    AUTOEXTEND ON NEXT 100M MAXSIZE 2000M
    EXTENT MANAGEMENT LOCAL
    SEGMENT SPACE MANAGEMENT AUTO
;
CREATE TABLESPACE LOB_DATA DATAFILE
    '+data2/lob_data01.dbf' SIZE 50M
    AUTOEXTEND ON NEXT 100M MAXSIZE 2000M
    EXTENT MANAGEMENT LOCAL
    SEGMENT SPACE MANAGEMENT AUTO
;
ALTER TABLESPACE RETAIL_INDEX
ADD DATAFILE '+data2/retek_index02.dbf' SIZE 500M AUTOEXTEND ON MAXSIZE 2000M;

CREATE TABLESPACE ENCRYPTED_RETAIL_INDEX DATAFILE
'+data3/encrypted_retail_index01.dbf'
SIZE 100M AUTOEXTEND ON MAXSIZE 6000M
SEGMENT SPACE MANAGEMENT AUTO
;

CREATE TABLESPACE ENCRYPTED_RETAIL_DATA DATAFILE
'+data3/encrypted_retail_data01.dbf'
SIZE 100M AUTOEXTEND ON MAXSIZE 5000M
SEGMENT SPACE MANAGEMENT AUTO
;
spool off
exit
```

Note: For more information on ENCRYPTED tablespace, see the *Oracle Merchandising Security* guide.