



Integrated Cloud Applications & Platform Services

Oracle Database 12c R2: RAC Administration

Activity Guide

D81250GC20

Edition 2.0 | March 2018 | D103168

Learn more from Oracle University at education.oracle.com



ORACLE®

Author

Sean Kim

**Technical Contributors
and Reviewers**

Sean Kim

Jerry Lee

Joel Goodman

Frank Fu

James Womack

Graphic Designers

Seema Bopaiah

Kavya Bellur

Publishers

Raghunath M

Veena Narasimhan

Joseph Fernandez

Copyright © 2018, Oracle and/or its affiliates. All rights reserved.

Disclaimer

This document contains proprietary information and is protected by copyright and other intellectual property laws. You may copy and print this document solely for your own use in an Oracle training course. The document may not be modified or altered in any way. Except where your use constitutes "fair use" under copyright law, you may not use, share, download, upload, copy, print, display, perform, reproduce, publish, license, post, transmit, or distribute this document in whole or in part without the express authorization of Oracle.

The information contained in this document is subject to change without notice. If you find any problems in the document, please report them in writing to: Oracle University, 500 Oracle Parkway, Redwood Shores, California 94065 USA. This document is not warranted to be error-free.

Restricted Rights Notice

If this documentation is delivered to the United States Government or anyone using the documentation on behalf of the United States Government, the following notice is applicable:

U.S. GOVERNMENT RIGHTS

The U.S. Government's rights to use, modify, reproduce, release, perform, display, or disclose these training materials are restricted by the terms of the applicable Oracle license agreement and/or the applicable U.S. Government contract.

Trademark Notice

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

Table of Contents

Course Practice Environment: Security Credentials	I-1
Practices for Lesson 1: Grid Infrastructure Overview	5
Practices for Lesson 1: Overview	6
Practice 1-1: Configuring a Standalone Flex Cluster.....	7
Practices for Lesson 2: RAC Databases Overview & Architecture	49
Practices for Lesson 2	50
Practices for Lesson 3: Grid Infrastructure Overview	51
Practices for Lesson 3: Overview	52
Practice 3-1: Installing RAC Database Software	53
Practice 3-2: Creating a RAC Database	70
Practices for Lesson 4: Oracle RAC Administration	93
Practices for Lesson 4: Overview	94
Practice 4-1: Operating System and Password File Authenticated Connections	95
Practice 4-2: Oracle Database Authenticated Connections	98
Practice 4-3: Stopping a Complete ORACLE_HOME Component Stack	101
Practice 4-4: Configuring Leaf Nodes to run Read-Only Instances.....	104
Practice 4-5: Working with Temporary Tablespaces	119
Practices for Lesson 5: Upgrading and Patching Oracle RAC	127
Practices for Lesson 5	128
Practices for Lesson 6: Managing Backup and Recovery for RAC.....	129
Practices for Lesson 6: Overview	130
Practice 6-1: Configuring Archive Log Mode	131
Practice 6-2: Configuring RMAN and Performing Parallel Backups	135
Practices for Lesson 7: Global Resource Management Concepts.....	145
Practices for Lesson 7	146
Practices for Lesson 8: RAC Database Monitoring and Tuning	147
Practices for Lesson 8: Overview	148
Practice 8-1: ADDM and RAC Part I	149
Practice 8-2: ADDM and RAC Part II	164
Practice 8-3: ADDM and RAC Part III	172
Practices for Lesson 9: Managing High Availability of Services.....	177
Practices for Lesson 9: Overview	178
Practice 9-1: Working with Services	179
Practice 9-2: Monitoring Services	187

Practices for Lesson 10: High Availability for Connections and Applications	195
Practices for Lesson 10: Overview	196
Practice 10-1: Using Application Continuity.....	197
Practices for Lesson 11: Oracle RAC One Node.....	209
Practices for Lesson 11: Overview	210
Practice 11-1: RAC One Node.....	211
Practices for Lesson 12: Oracle Database In-Memory in RAC	217
Practices for Lesson 12: Overview	218
Practice 12-1: Reconfiguring the Environment	219
Practices for Lesson 13: Multitenant Architecture and RAC	237
Practices for Lesson 13: Overview	238
Practice 13-1: Exploring CDB Architecture and Structures in RAC.....	239
Practice 13-2: Cloning a PDB in the RAC CDB	253
Practice 13-3: Affinitizing PDB Services to CDB Instances	258
Practice 13-4: Dropping a PDB.....	269
Practices for Appendix A: Cleanup and Catchup Procedures for Practices	275
Appendix A-1: Cleanup Script.....	276
Appendix A-2: Catchup Script for Practice 1	292
Appendix A-3: Catchup Script for Practice 3	300

Course Practice Environment: Security Credentials

For operating system (Linux) usernames and passwords, see the following:

- If you are attending a classroom-based or live virtual class, ask your instructor or LVC producer for OS credential information.
- If you are using a self-study format, refer to the communication that you received from Oracle University for this course.

Operating System Credentials		
Practice Environment	Username	Password
DNS node	root	<i>Read the note above</i>
	oracle	
host01 ~ host05	root	oracle
	grid	oracle
	oracle	oracle

For product-specific credentials used in this course, see the following table:

Product-Specific Credentials		
Product/Application	Username	Password (Case Sensitive)
Enterprise Manager Database Express	sys	oracle_4U
Database (orcl)	sys	oracle_4U
	system	oracle_4U
	dbsnmp	oracle_4U
	hr	hr
	jmw	jmw
ASM Instance	SYS	oracle_4U
	ASMSNMP	oracle_4U

Unauthorized reproduction or distribution prohibited. Copyright© 2019, Oracle and/or its affiliates.

GANG LIU (gangl@baylorhealth.edu) has a non-transferable license
to use this Student Guide.

Practices for Lesson 1: Grid Infrastructure Overview

Practices for Lesson 1: Overview

Overview

In this practice, you will install and configure a standalone Flex Cluster. You will install to five nodes; host01, host02, host03, host04, and host05. You will designate host01~host03 to be HUB nodes and host04~host05 will be designated as LEAF nodes.

Practice 1-1: Configuring a Standalone Flex Cluster

Overview

In this practice, you will install and configure a standalone Flex Cluster. You will install to five nodes; host01, host02, host03, host04, and host05. You will designate host01~host03 to be HUB nodes and host04~host05 will be designated as LEAF nodes.

- From a terminal in the dns node, change to the root account. First, set the time across all nodes using the command shown below. Then restart the NAMED service to ensure viability and availability of the services for the software installation.

```
[oracle@dns ~]$ su -
Password: <Enter the root password at the dns node>

[root@dns ~]# TIME=`date +%T`;for H in host01 host02 host03
host04 host05;do ssh $H date -s $TIME;done

root@host01's password:
Tue Oct 31 10:43:40 UTC 2017
root@host02's password:
Tue Oct 31 10:43:40 UTC 2017
root@host03's password:
Tue Oct 31 10:43:40 UTC 2017
root@host04's password:
Tue Oct 31 10:43:40 UTC 2017
root@host05's password:
Tue Oct 31 10:43:40 UTC 2017

[root@dns ~]# service named restart
Stopping named: [ OK ]
Starting named: [ OK ]
[root@dns ~]#
```

- Review the /etc/hosts file. You should see output similar to the example displayed below. It shows that your environment is configured with six VMs, which you will use to form a cluster in upcoming practices. The dns node in the output is a gateway VM.

```
[root@dns ~]# more /etc/hosts
# Do not remove the following line, or various programs
# that require network functionality will fail.
127.0.0.1      localhost.localdomain localhost
::1            localhost6.localdomain6 localhost6
192.0.2.101    host01.example.com        host01
```

```

192.0.2.102      host02.example.com      host02
192.0.2.103      host03.example.com      host03
192.0.2.104      host04.example.com      host04
192.0.2.105      host05.example.com      host05
192.0.2.1        dns.example.com        dns

[root@dns ~]#

```

- Check that the DHCPD is running on the DNS server. You will configure nodes VIPs using the DHCPD service later.

```

[root@dns ~]# service dhcpcd status
dhcpcd (pid 17024) is running...

[root@dns ~]#

```

- Using SSH, connect to host01 as root user account. Start the local naming cache daemon on all five cluster nodes with the service nscd start command. To make sure nscd starts at reboot, execute the chkconfig nscd command. **Perform these steps on all five of your nodes.**

```

[root@dns ~]# ssh host01
root@host01's password: <Enter the root password at the host01 node>

[root@host01 ~]# service nscd start
Starting nscd: [ OK ]
[root@host01 ~]# chkconfig nscd on

[root@host01 ~]# ssh host02 service nscd start
Starting nscd: [ OK ]
[root@host01 ~]# ssh host02 chkconfig nscd on

[root@host01 ~]# ssh host03 service nscd start
Starting nscd: [ OK ]
[root@host01 ~]# ssh host03 chkconfig nscd on

[root@host01 ~]# ssh host04 service nscd start
Starting nscd: [ OK ]
[root@host01 ~]# ssh host04 chkconfig nscd on

[root@host01 ~]# ssh host05 service nscd start
Starting nscd: [ OK ]
[root@host01 ~]# ssh host05 chkconfig nscd on

[root@host01 ~]#

```

5. As the root user, run the /stage/RAC/labs/less_01/limits.sh script on host01. This script replaces the profile for the oracle and grid users and replaces /etc/profile. It replaces the /etc/security/limits.conf file with a new one with entries for oracle and grid.

```
[root@host01 ~]# cat /stage/RAC/labs/less_01/bash_profile
# .bash_profile

# Get the aliases and functions
if [ -f ~/.bashrc ]; then
    . ~/.bashrc
fi

# User specific environment and startup programs

PATH=$PATH:$HOME/bin
export PATH

umask 022

[root@host01 ~]# cat /stage/RAC/labs/less_01/profile
# /etc/profile

# System wide environment and startup programs, for login setup
# Functions and aliases go in /etc/bashrc

pathmunge () {
    if ! echo $PATH | /bin/egrep -q "(:$1($|:)" ; then
        if [ "$2" = "after" ] ; then
            PATH=$PATH:$1
        else
            PATH=$1:$PATH
        fi
    fi
}

# ksh workaround
if [ -z "$EUID" -a -x /usr/bin/id ] ; then
    EUID=`id -u`
    UID=`id -ru`
fi

# Path manipulation
if [ "$EUID" = "0" ] ; then
    pathmunge /sbin
    pathmunge /usr/sbin
    pathmunge /usr/local/sbin
fi

# No core files by default
ulimit -S -c 0 > /dev/null 2>&1
```

```

if [ -x /usr/bin/id ]; then
    USER=`id -un`
    LOGNAME=$USER
    MAIL="/var/spool/mail/$USER"

fi

HOSTNAME=`/bin/hostname`
HISTSIZE=1000

if [ -z "$INPUTRC" -a ! -f "$HOME/.inputrc" ]; then
    INPUTRC=/etc/inputrc
fi

export PATH USER LOGNAME MAIL HOSTNAME HISTSIZE INPUTRC

for i in /etc/profile.d/*.sh ; do
    if [ -r "$i" ]; then
        . $i
    fi
done

if [ $USER = "oracle" ] || [ $USER = "grid" ]; then
    umask 022
    if [ $SHELL = "/bin/ksh" ]; then
        ulimit -p 16384
        ulimit -n 65536
        ulimit -s 10240
    else
        ulimit -u 16384 -n 65536 -s 10240
    fi
fi
unset i
unset pathmunge

[root@host01 ~]# cat /stage/RAC/labs/less_01/limits.conf

#      - priority - the priority to run user process with
#      - locks - max number of file locks the user can hold
#      - sigpending - max number of pending signals
#      - msgqueue - max memory used by POSIX message queues
#      - nice - max nice priority allowed to raise to
#      - rtprio - max realtime priority
#<domain>      <type>  <item>          <value>

#*           soft    core        0
#*           hard    rss       10000
#@student    hard    nproc      20
#@faculty   soft    nproc      20
#@faculty   hard    nproc      50
#ftp         hard    nproc       0
#@student    -      maxlogins  4

```

```

# End of file
oracle soft nofile      131072
oracle hard nofile      131072
oracle soft nproc 131072
oracle hard nproc 131072
oracle soft core unlimited
oracle hard core unlimited
oracle soft memlock 3500000
oracle hard memlock 3500000
oracle soft stack 10240
oracle hard stack 32768
grid  soft  nofile  131072
grid  hard  nofile  131072
grid  soft   nproc  131072
grid  hard   nproc  131072
grid  soft   core  unlimited
grid  hard   core  unlimited
grid  soft   memlock 3500000
grid  hard   memlock 3500000
grid  soft  stack  10240
grid  hard  stack  32768

# Recommended stack hard limit 32MB for oracle installations
# oracle    hard  stack  32768

[root@host01 ~]# /stage/RAC/labs/less_01/limits.sh
bash_profile                                100% 194      0.2KB/s 00:00
bash_profile                                100% 194      0.2KB/s 00:00
limits.conf                                  100% 1276     1.3KB/s 00:00
profile                                      100% 1279     1.3KB/s 00:00
profile                                      100% 1279     1.3KB/s 00:00
bash_profile                                100% 194      0.2KB/s 00:00
bash_profile                                100% 194      0.2KB/s 00:00
limits.conf                                  100% 1276     1.3KB/s 00:00
profile                                      100% 1279     1.3KB/s 00:00
profile                                      100% 1279     1.3KB/s 00:00
bash_profile                                100% 194      0.2KB/s 00:00
bash_profile                                100% 194      0.2KB/s 00:00
limits.conf                                  100% 1276     1.3KB/s 00:00
profile                                      100% 1279     1.3KB/s 00:00
profile                                      100% 1279     1.3KB/s 00:00
bash_profile                                100% 194      0.2KB/s 00:00
bash_profile                                100% 194      0.2KB/s 00:00
limits.conf                                  100% 1276     1.3KB/s 00:00
profile                                      100% 1279     1.3KB/s 00:00
profile                                      100% 1279     1.3KB/s 00:00
bash_profile                                100% 194      0.2KB/s 00:00
bash_profile                                100% 194      0.2KB/s 00:00
limits.conf                                  100% 1276     1.3KB/s 00:00
profile                                      100% 1279     1.3KB/s 00:00
profile                                      100% 1279     1.3KB/s 00:00
[root@host01 ~]#

```

6. Create the installation directories for both grid and oracle-owned software. Set the ownership and permissions of the directories as shown below: Do this on all five hosts. Use the `/stage/RAC/labs/less_01/cr_dir.sh` script to save time.

```
[root@host01 ~]# cat /stage/RAC/labs/less_01/cr_dir.sh
#!/bin/bash

mkdir -p /u01/app/12.2.0/grid
mkdir -p /u01/app/grid
mkdir -p /u01/app/oracle
chown -R grid:oinstall /u01
chown oracle:oinstall /u01/app/oracle
chmod -R 775 /u01/

ssh host02 mkdir -p /u01/app/12.2.0/grid
ssh host02 mkdir -p /u01/app/grid
ssh host02 mkdir -p /u01/app/oracle
ssh host02 chown -R grid:oinstall /u01
ssh host02 chown oracle:oinstall /u01/app/oracle
ssh host02 chmod -R 775 /u01

ssh host03 mkdir -p /u01/app/12.2.0/grid
ssh host03 mkdir -p /u01/app/grid
ssh host03 mkdir -p /u01/app/oracle
ssh host03 chown -R grid:oinstall /u01
ssh host03 chown oracle:oinstall /u01/app/oracle
ssh host03 chmod -R 775 /u01

ssh host04 mkdir -p /u01/app/12.2.0/grid
ssh host04 mkdir -p /u01/app/grid
ssh host04 mkdir -p /u01/app/oracle
ssh host04 chown -R grid:oinstall /u01
ssh host04 chown oracle:oinstall /u01/app/oracle
ssh host04 chmod -R 775 /u01

ssh host05 mkdir -p /u01/app/12.2.0/grid
ssh host05 mkdir -p /u01/app/grid
ssh host05 mkdir -p /u01/app/oracle
ssh host05 chown -R grid:oinstall /u01
ssh host05 chown oracle:oinstall /u01/app/oracle
ssh host05 chmod -R 775 /u01

[root@host01 ~]# /stage/RAC/labs/less_01/cr_dir.sh
[root@host01 ~]#
```

7. Open a terminal window on your dns node and ssh to host01 as the grid user. Unzip the /stage/clusterware.zip file to /u01/app/12.2.0/grid on host01.

```
[oracle@dns ~]# ssh -X grid@host01
grid@host01's password: <Enter the grid password at the host01 node>

[grid@host01 ~]$ unzip -d /u01/app/12.2.0/grid
/stage/clusterware.zip
Archive:      /stage/clusterware.zip
...
finishing deferred symbolic links:
/u01/app/12.2.0/grid/bin/lbuilder -> ../nls/lbuilder/lbuilder
...
/u01/app/12.2.0/grid/lib/libclntsh.so -> libclntsh.so.12.1
/u01/app/12.2.0/grid/lib/libocci.so -> libocci.so.12.1

[grid@host01 ~]$
```

8. As the grid user, change your working directory to /u01/app/12.2.0/grid and execute the gridSetup.sh script.

```
[grid@host01 ~]$ cd /u01/app/12.2.0/grid
[grid@host01 grid]$ ./gridSetup.sh
```

9. On the Select Configuration Option page, select “Configure Oracle Grid Infrastructure for a New Cluster and click “Next”.

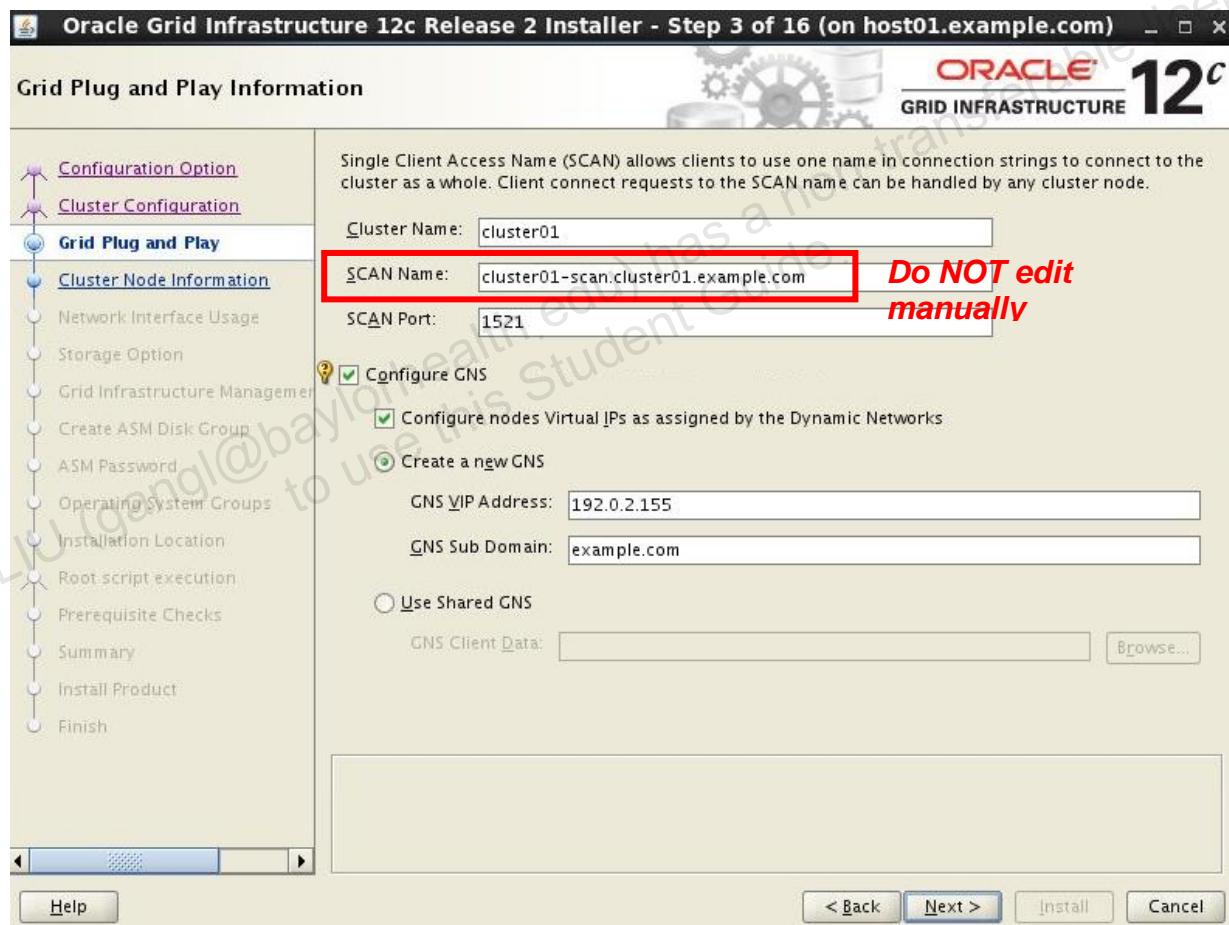


10. On the Select Cluster Configuration page, select “Configure an Oracle Standalone Cluster” and click “Next”.



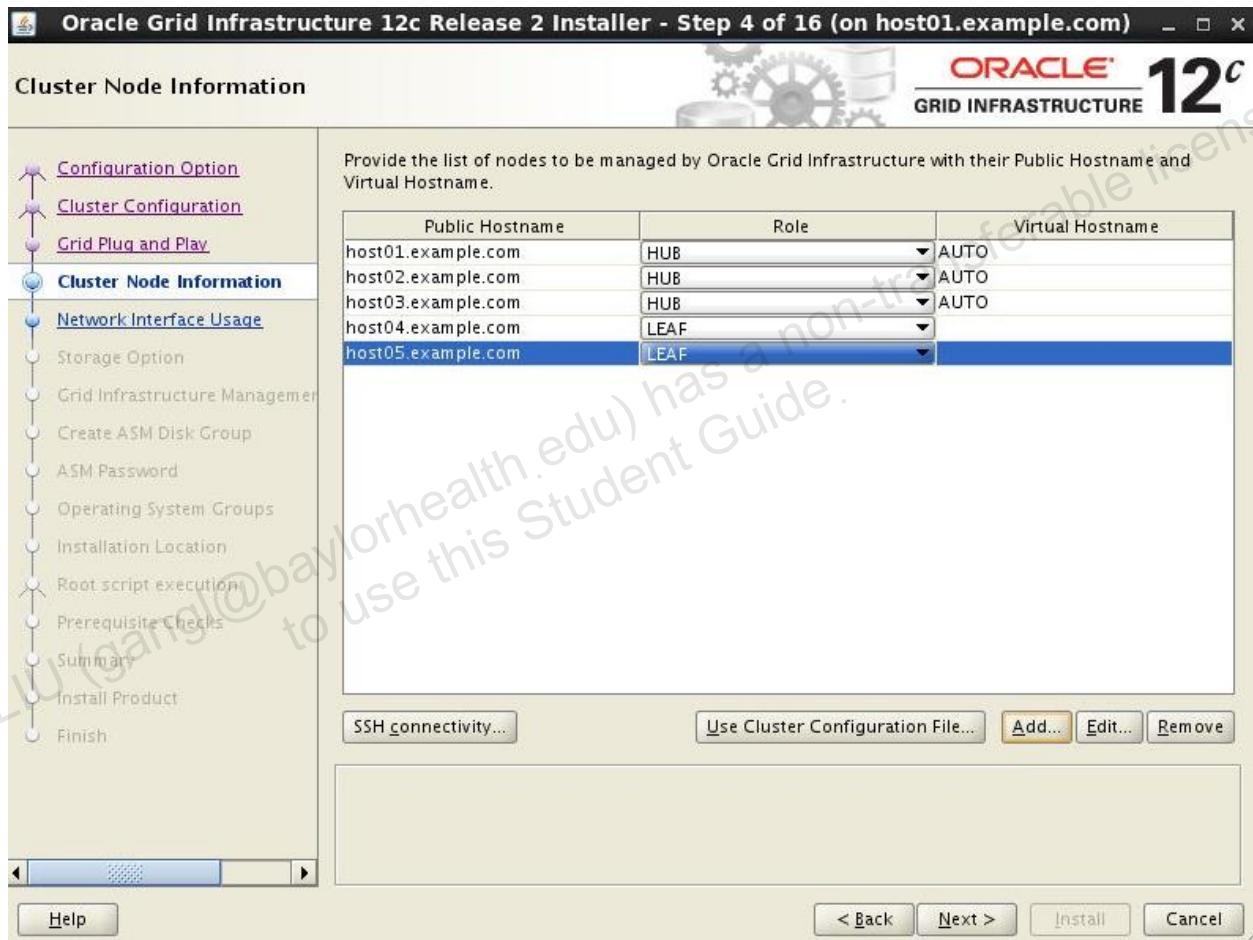
11. On the Grid Plug and Play Information page, enter the following settings:

- Cluster Name: cluster01
- SCAN Name: cluster01-scan.cluster01.example.com (Auto-filled, Do NOT edit it manually).
- SCAN Port: 1521
- Click the “Configure GNS” check box
- Click the “Configure nodes Virtual IPs as assigned by the Dynamic Networks”
- Click the “Create a new GNS” radio button
- GNS VIP Address: 192.0.2.155
- GNS Sub Domain: example.com

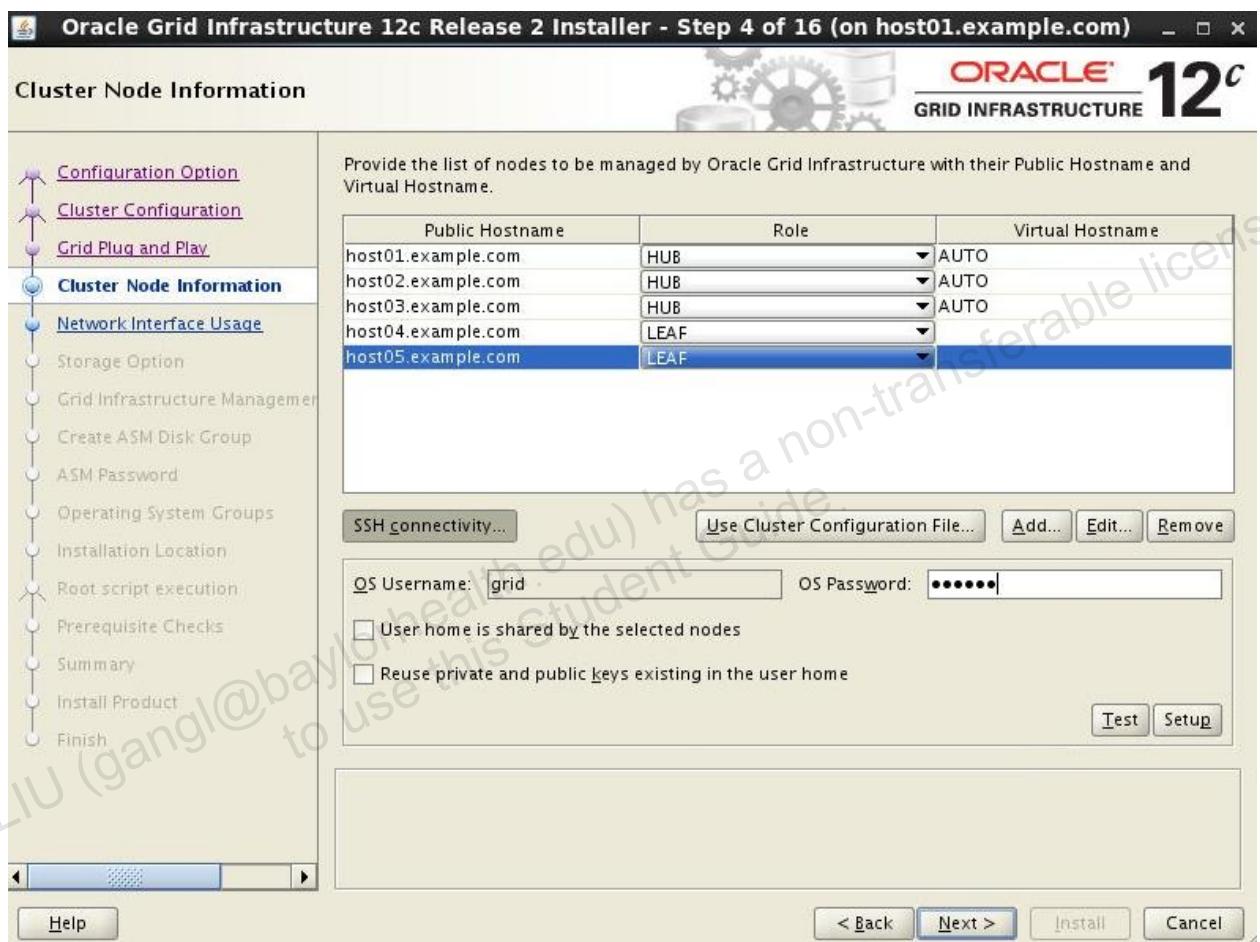


12. On the Cluster Node Information page, click Add to begin the process of specifying additional cluster nodes. Click the Add button and add host02.example.com. Make sure to set Node Role to **HUB**, and click OK. **Repeat for the following nodes.**

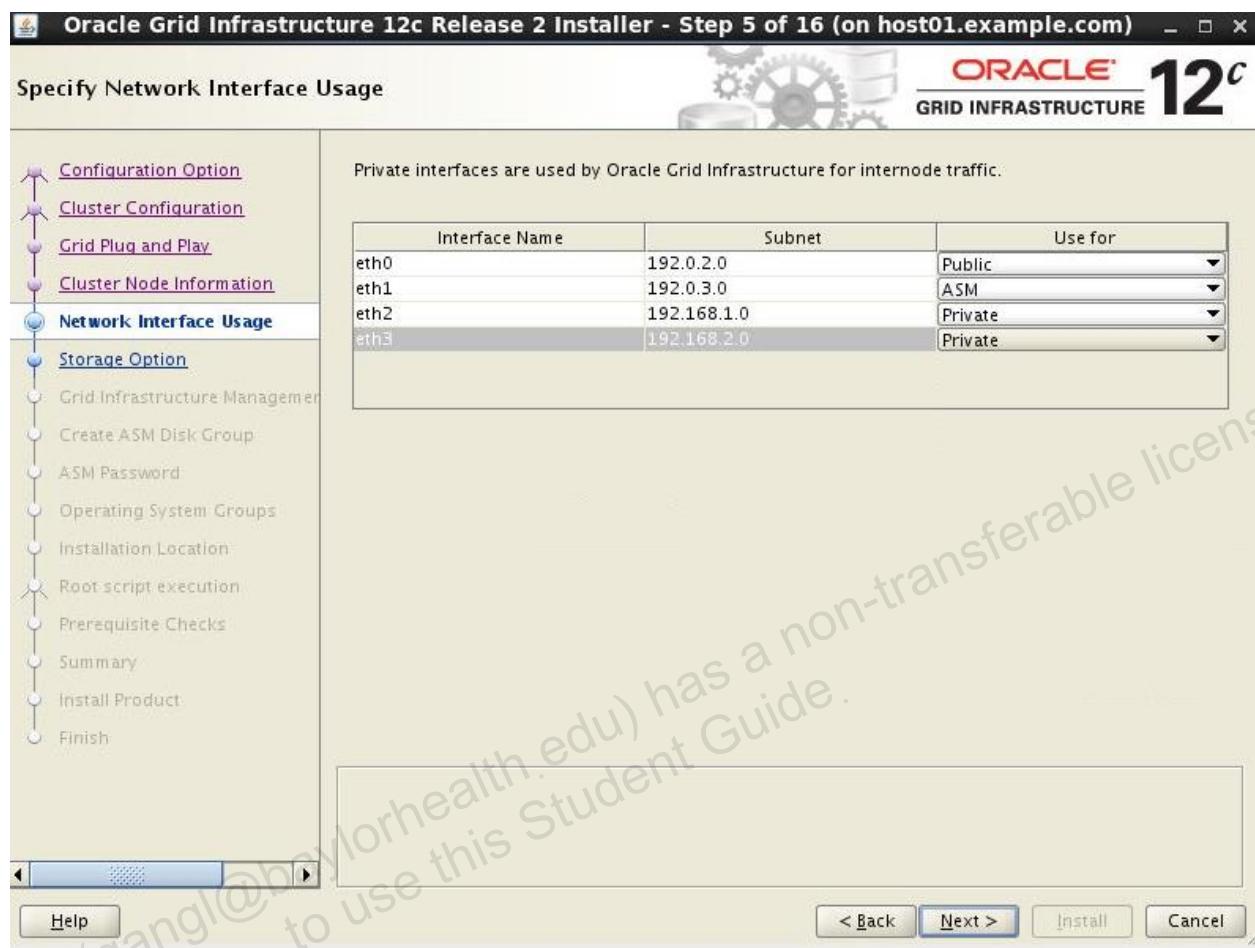
- host01.example.com (**HUB**)
- host02.example.com (**HUB**)
- host03.example.com (**HUB**)
- host04.example.com (**LEAF**)
- host05.example.com (**LEAF**)



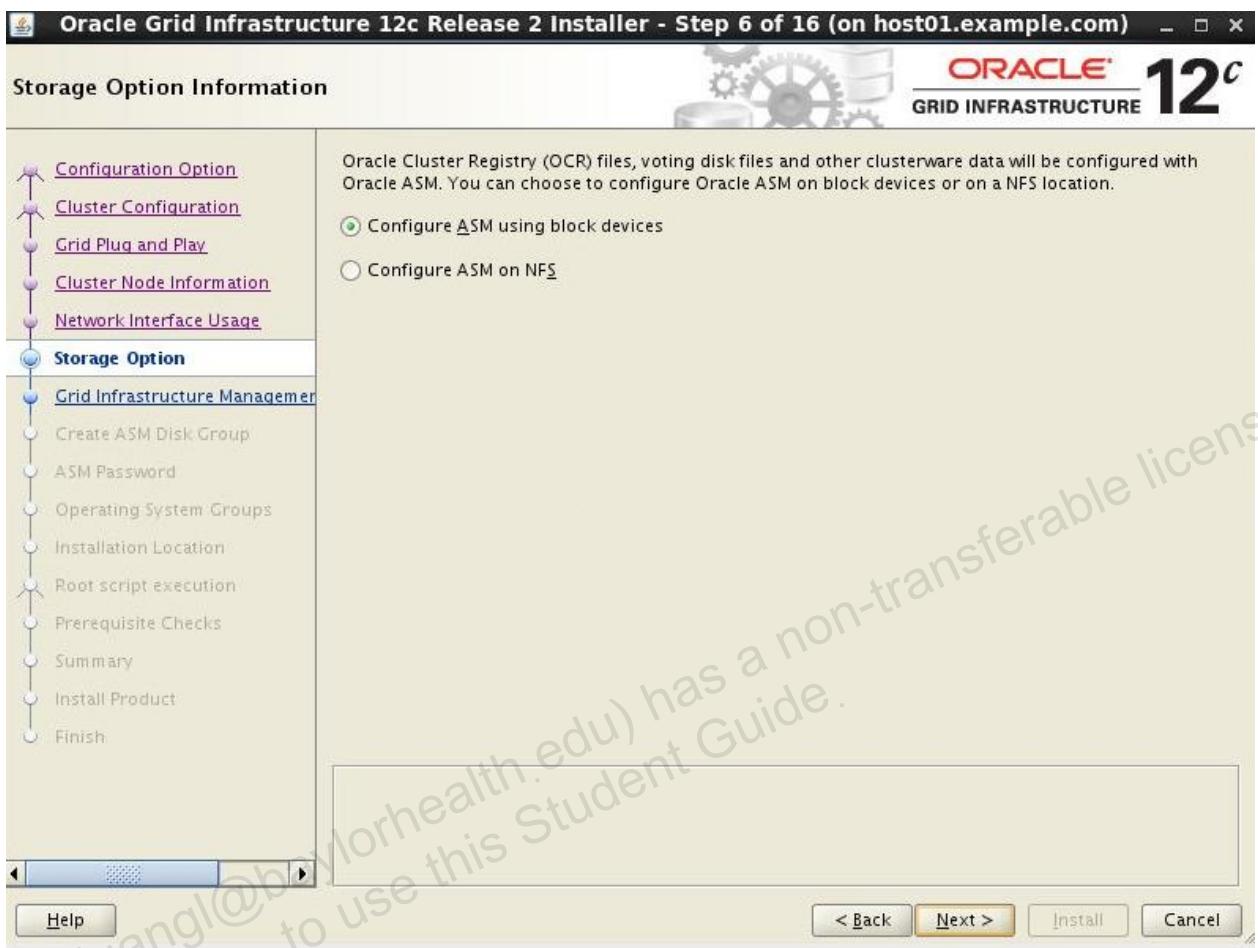
13. Click the SSH Connectivity button. Enter the grid password (*please refer to the Course Practice Environment: Security Credentials page for account passwords*) into the OS Password field and click Test to confirm that the required SSH connectivity is configured across the cluster. Your lab environment is preconfigured with the required SSH connectivity so you will next see a dialog confirming this. Click OK to continue. Review the information in the Cluster Node Information page and click Next. Your screen Cluster Node Information page should look like the one below:



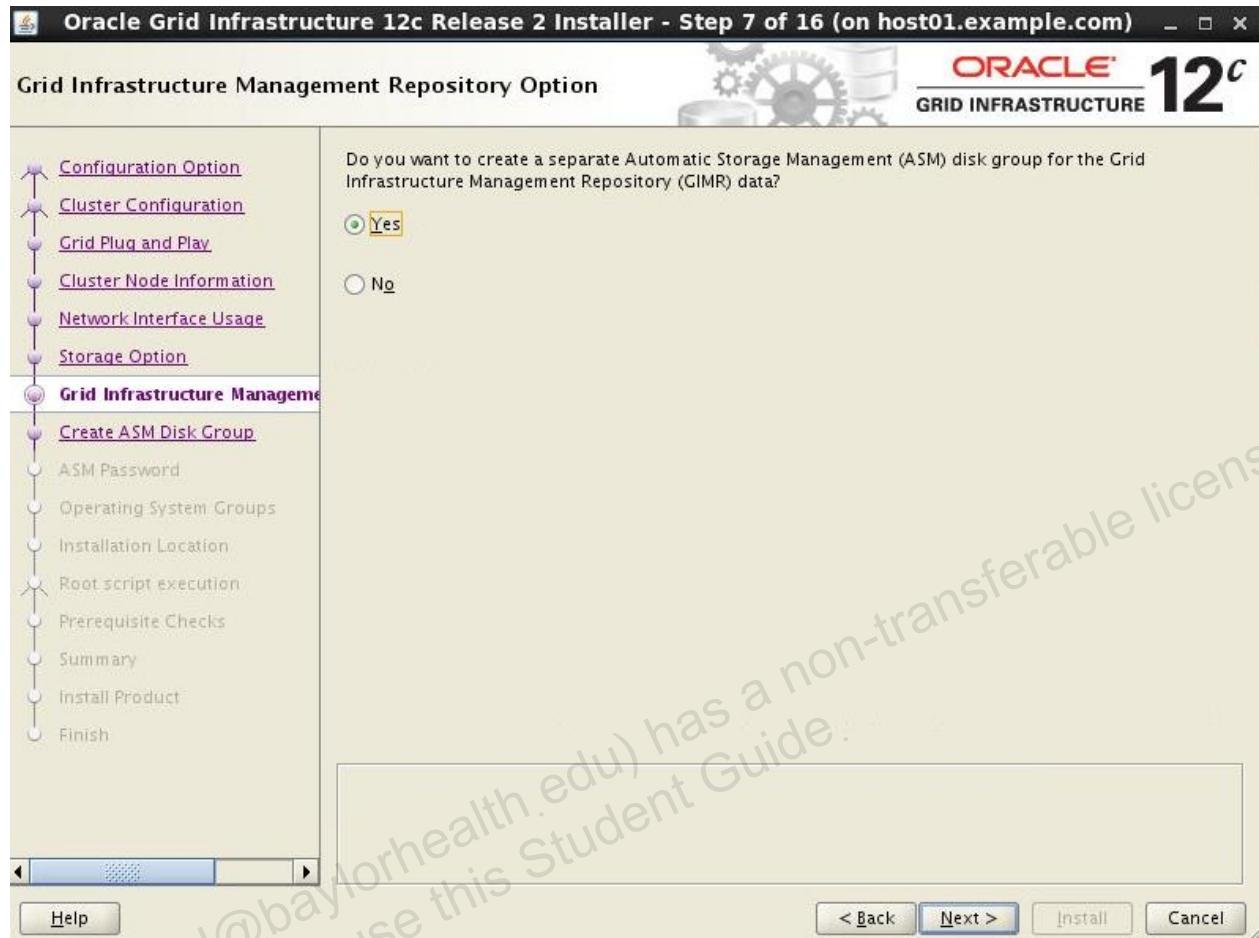
14. On the Specify Network Interface Usage page, select Public for eth0, ASM for eth1, and Private for eth2 and eth3. Click “Next” to continue.



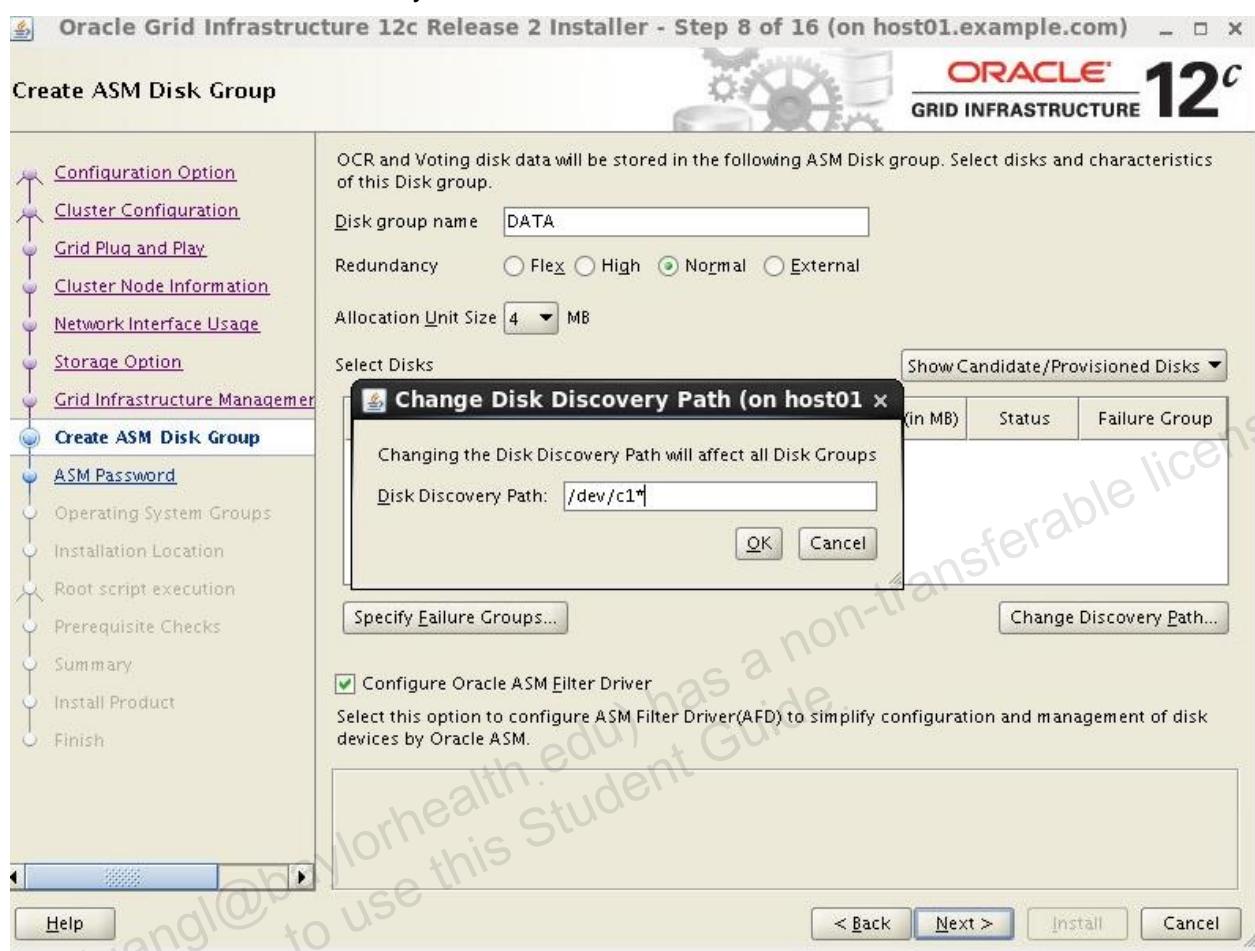
15. On the Storage Option Information page, click the “Configure ASM using block devices” button and click Next.



16. On the Grid Infrastructure Management Repository Option page, click YES then click Next.

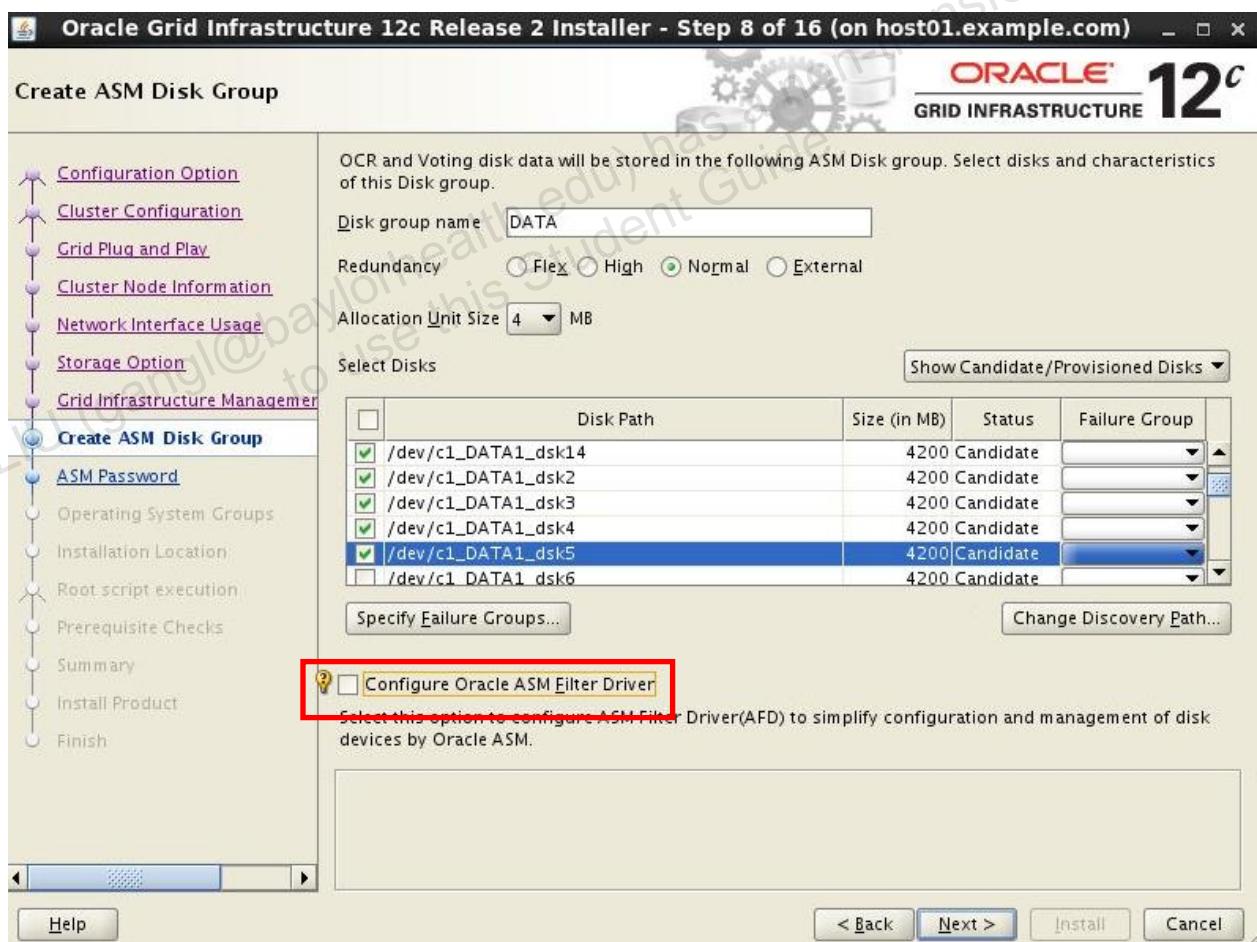


17. On the Create ASM Disk Group page, click the “Change Discovery Path” button. Enter /dev/c1* in the Disk Discovery Path field and click OK.

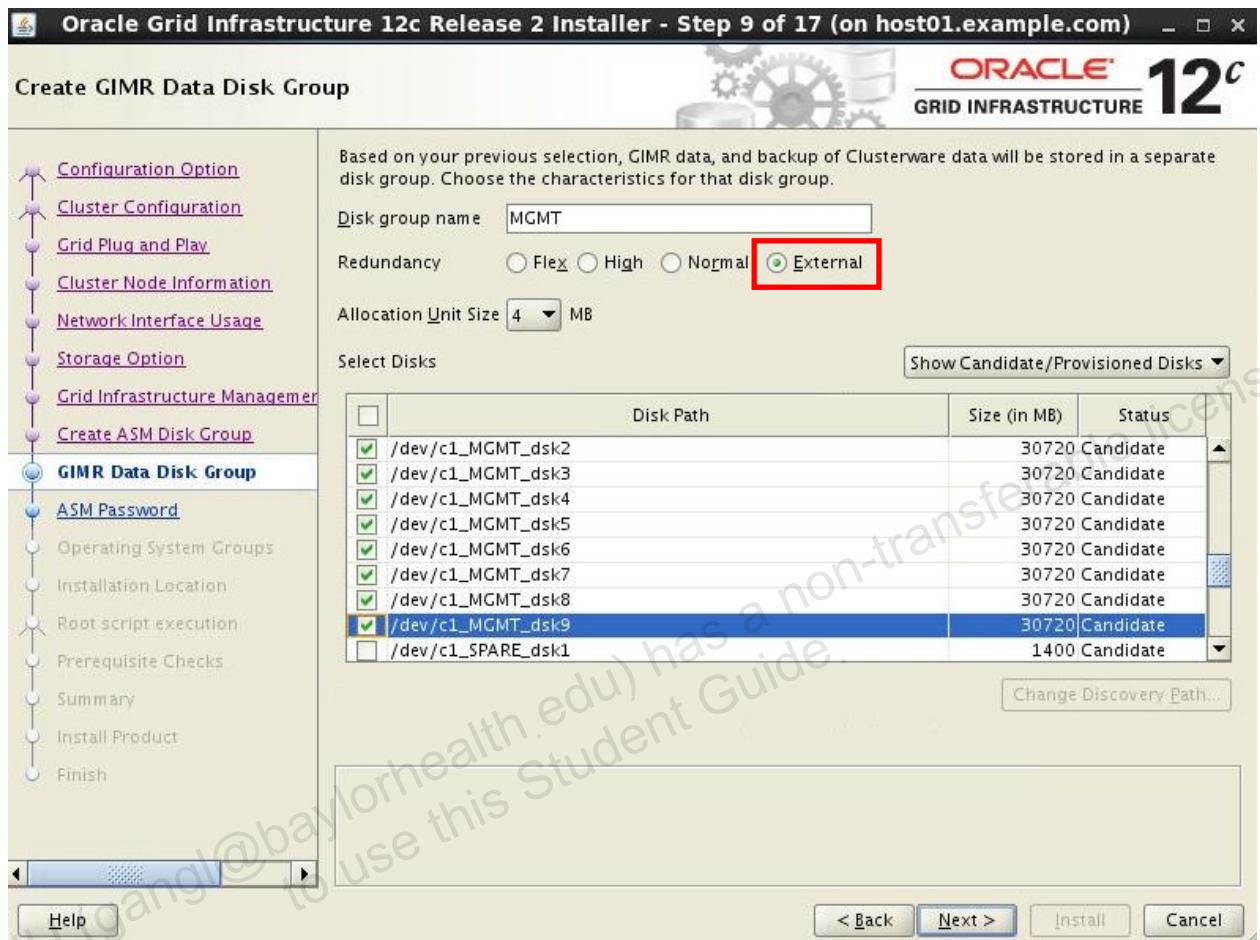


18. On the Create ASM Disk Group page, select the first 10 candidate disks (`c1_DATA1` disks) in the list for the DATA disk group. Make sure Redundancy is **Normal** and the “Configure Oracle ASM Filter Driver” check box is **NOT selected**. Click “Next”.

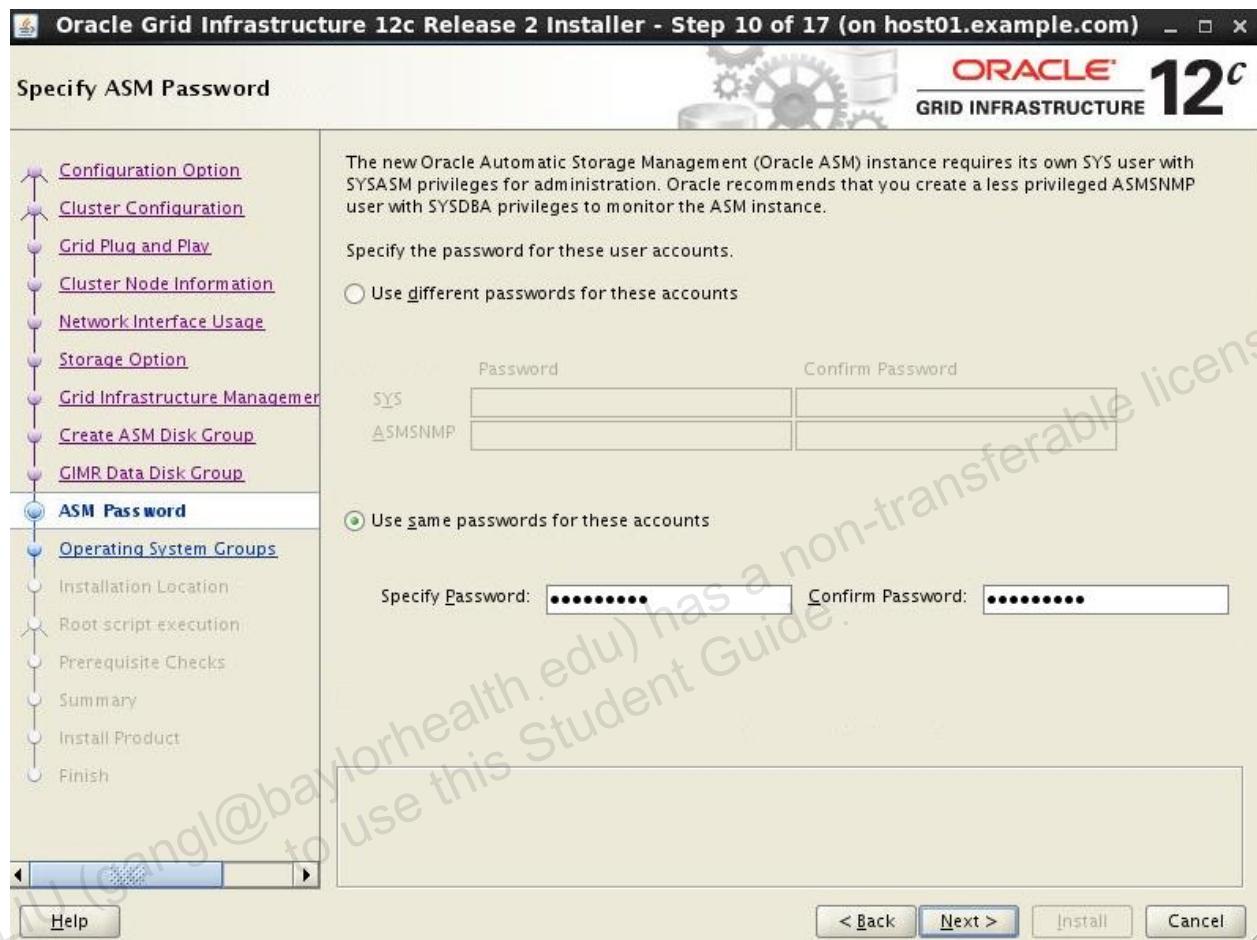
- `/dev/c1_DATA1_dsk1`
- `/dev/c1_DATA1_dsk10`
- `/dev/c1_DATA1_dsk11`
- `/dev/c1_DATA1_dsk12`
- `/dev/c1_DATA1_dsk13`
- `/dev/c1_DATA1_dsk14`
- `/dev/c1_DATA1_dsk2`
- `/dev/c1_DATA1_dsk3`
- `/dev/c1_DATA1_dsk4`
- `/dev/c1_DATA1_dsk5`



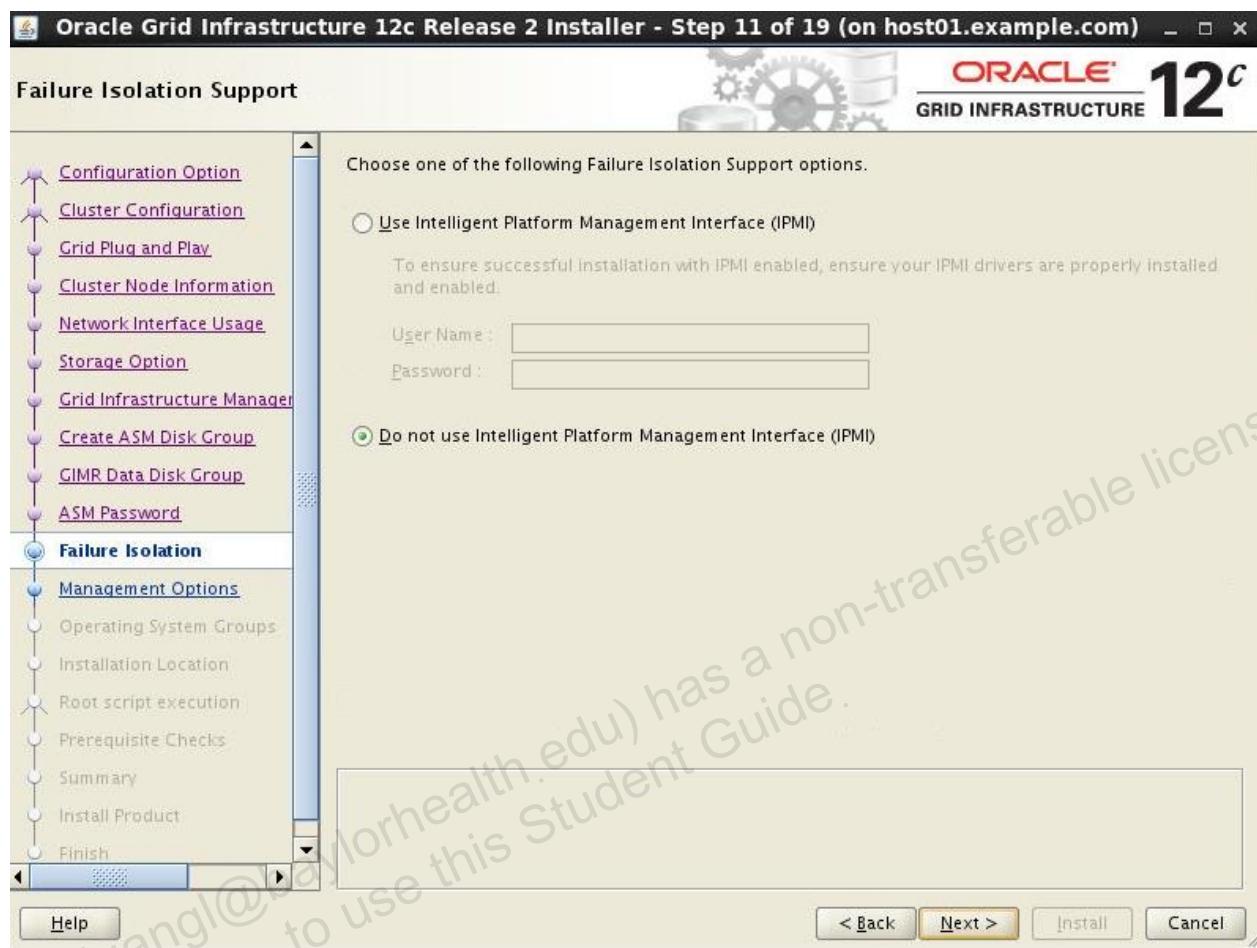
19. On the Create GIMR Data Disk Group page, select **All** candidate disks (`c1_MGMT1_dsk1~c1_MGMT_dsk14` disks) in the list for the MGMT disk group. There will be 14 disks. Set the Redundancy to **External**. Click “Next” to continue.



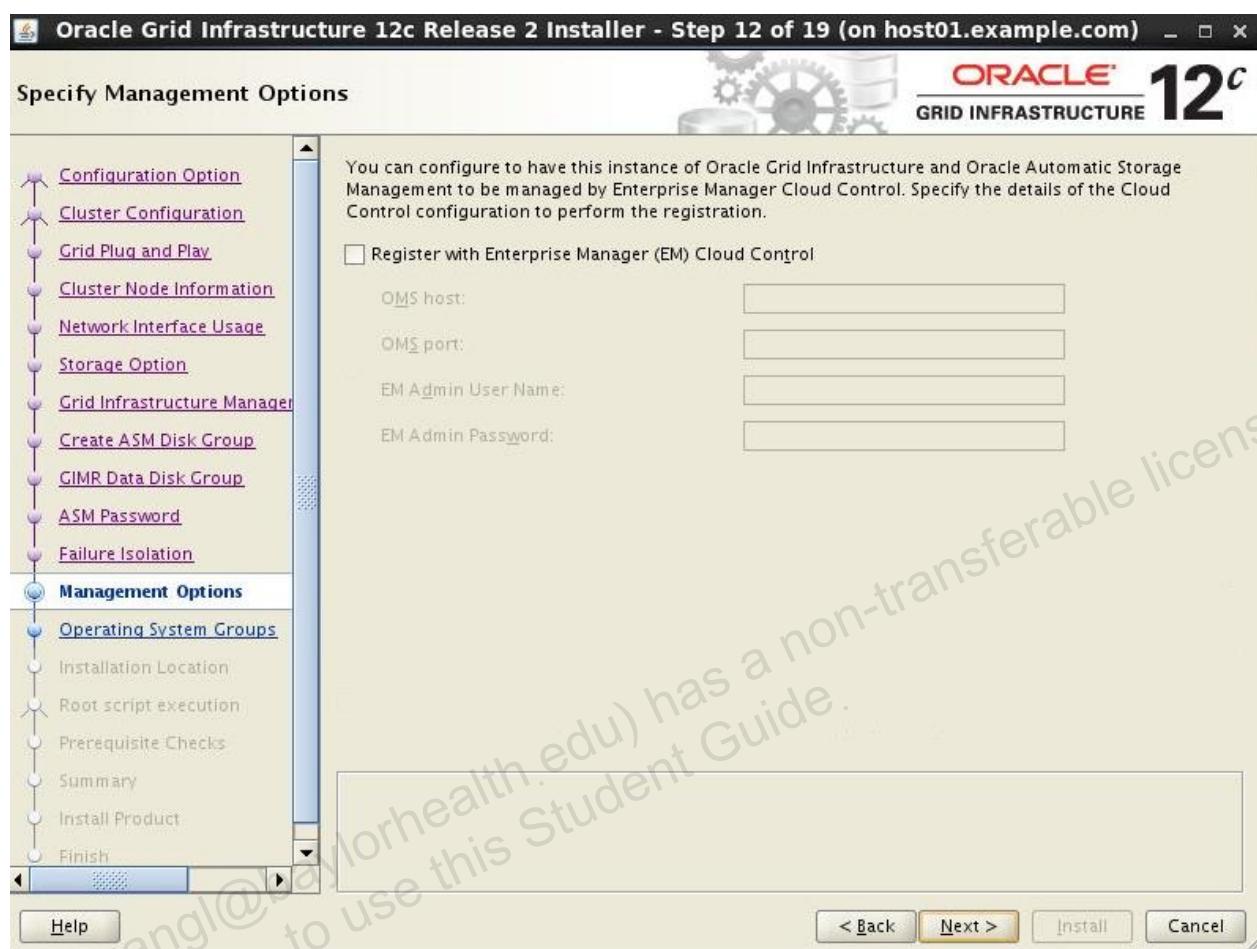
20. On the Specify ASM Password page, click the “Use same passwords for these accounts” radio button and enter the SYS password (as specified in the **Course Practice Environment: Security Credentials** page) in the Specify Password and Confirm Password fields. Click “Next” to continue.



21. On the Failure Isolation Support screen, click Next to accept the default setting (Do not use IPMI).



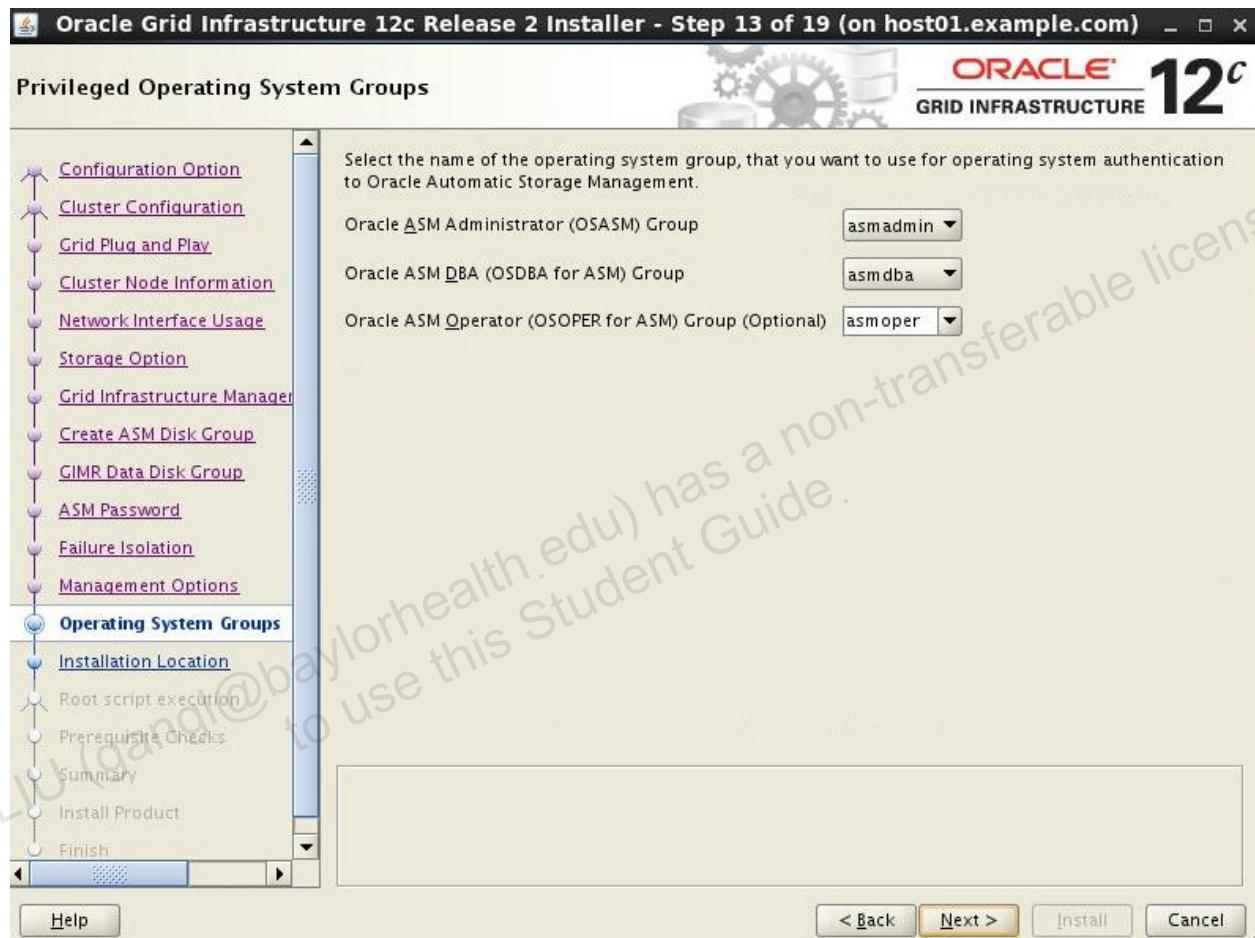
22. On the Specify Management Options screen, accept the defaults (nothing selected) and click Next to continue.



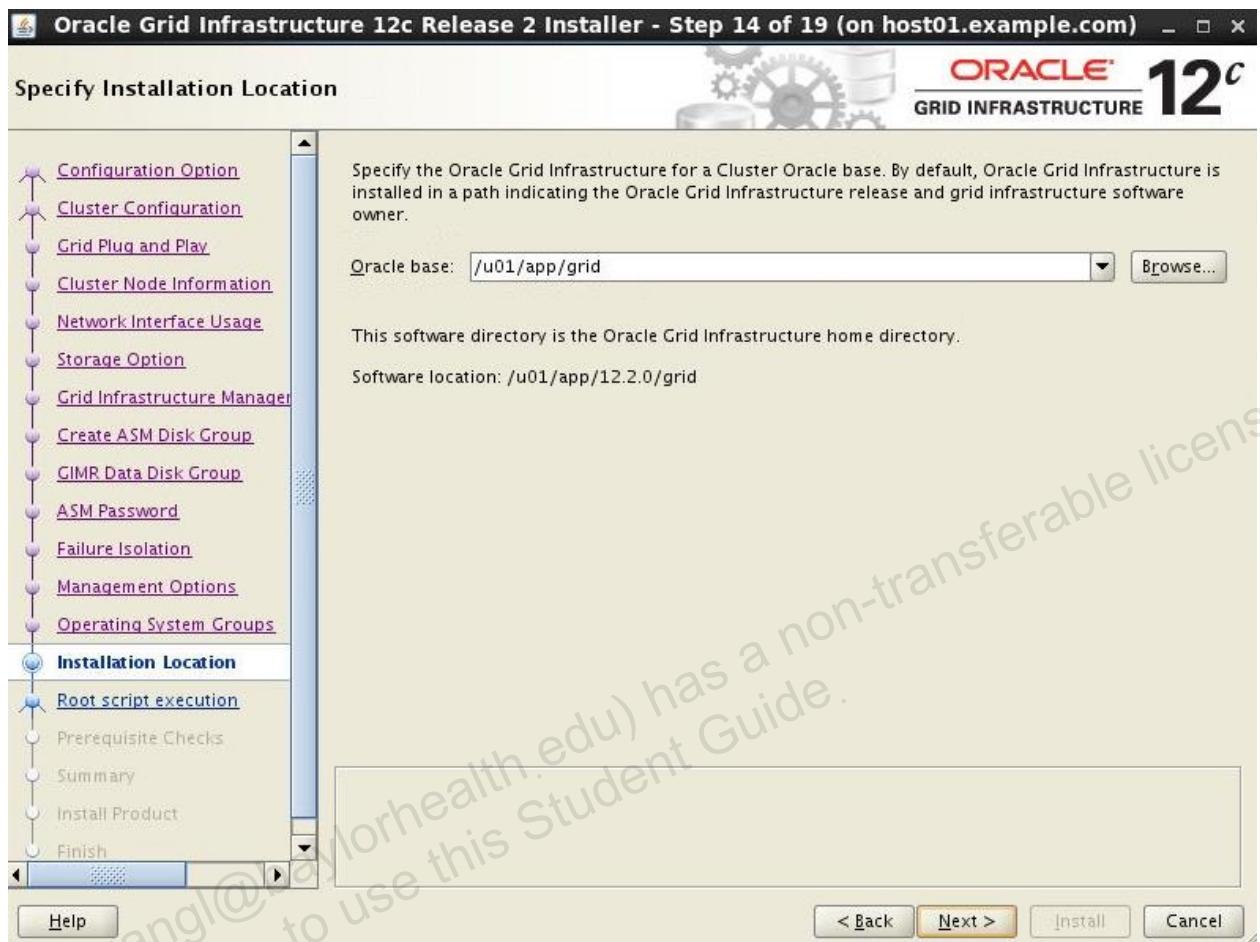
23. On the Privileged Operating System Groups screen, the values should default to the following:

- Oracle ASM Administrator Group: asmadmin
- Oracle ASM DBA Group: asmdba
- Oracle ASM Operator Group: asmoper

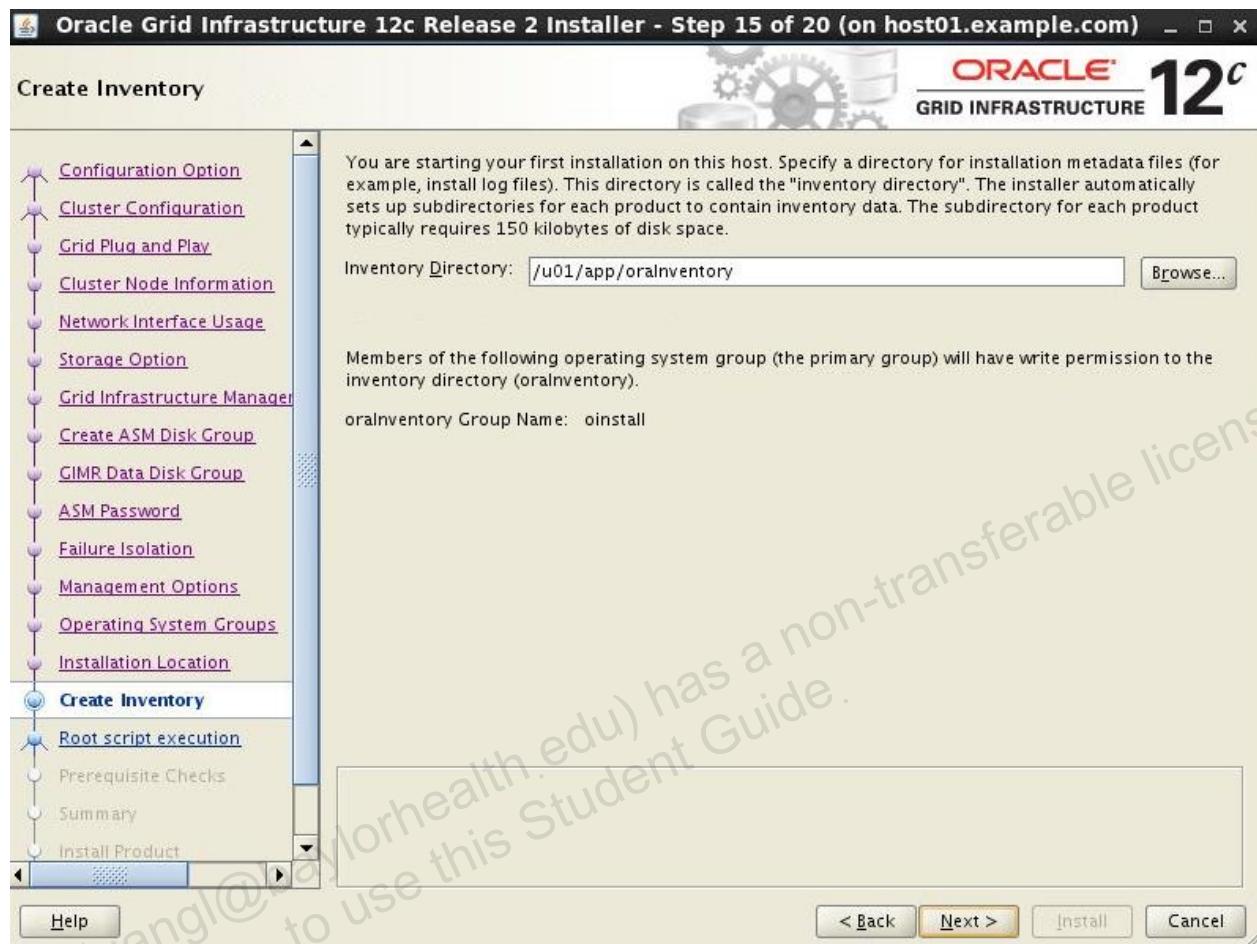
Click Next to accept the default values.



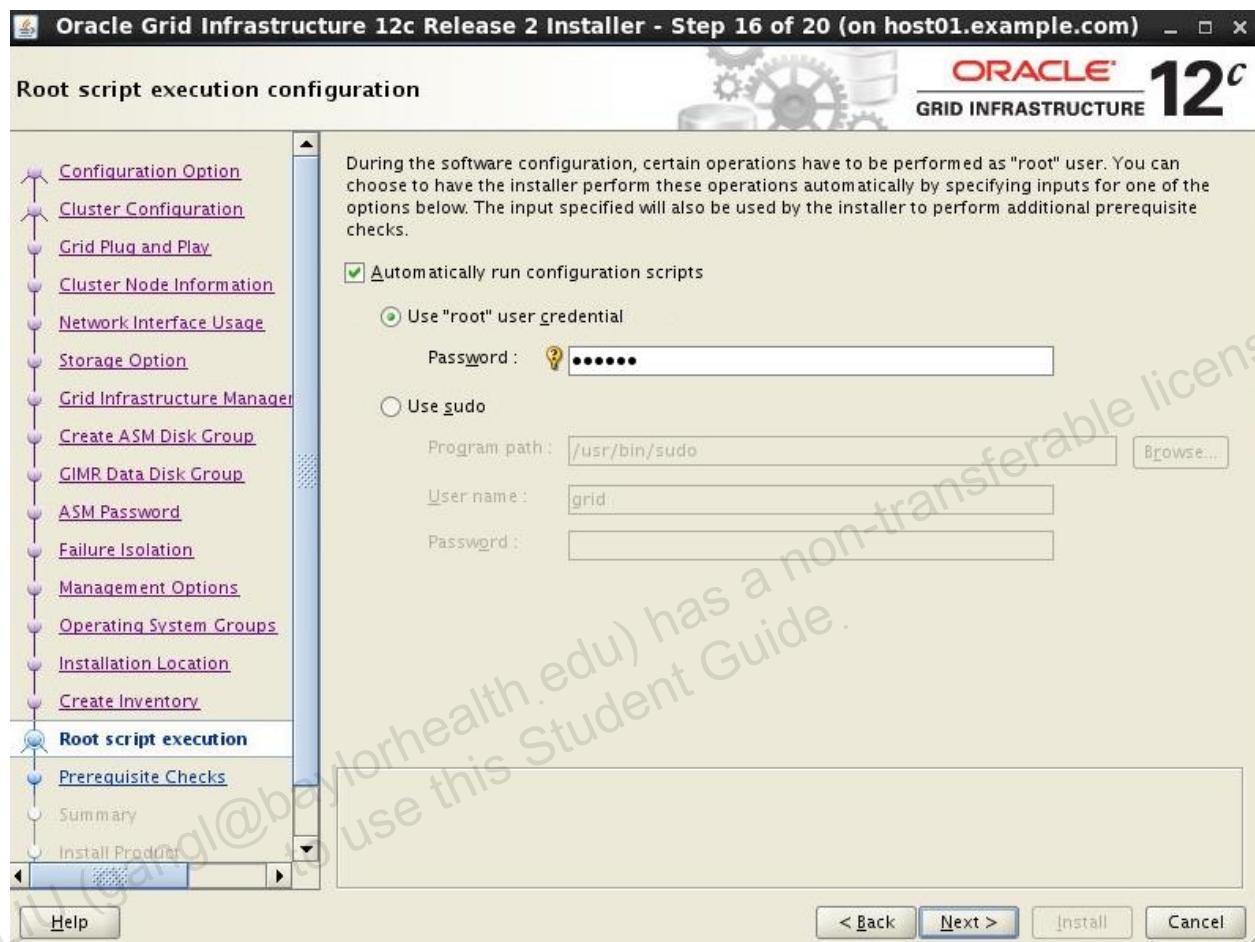
24. On the Specify Installation Location page, Oracle base should default to /u01/app/grid. Accept this value and click “Next” to continue.



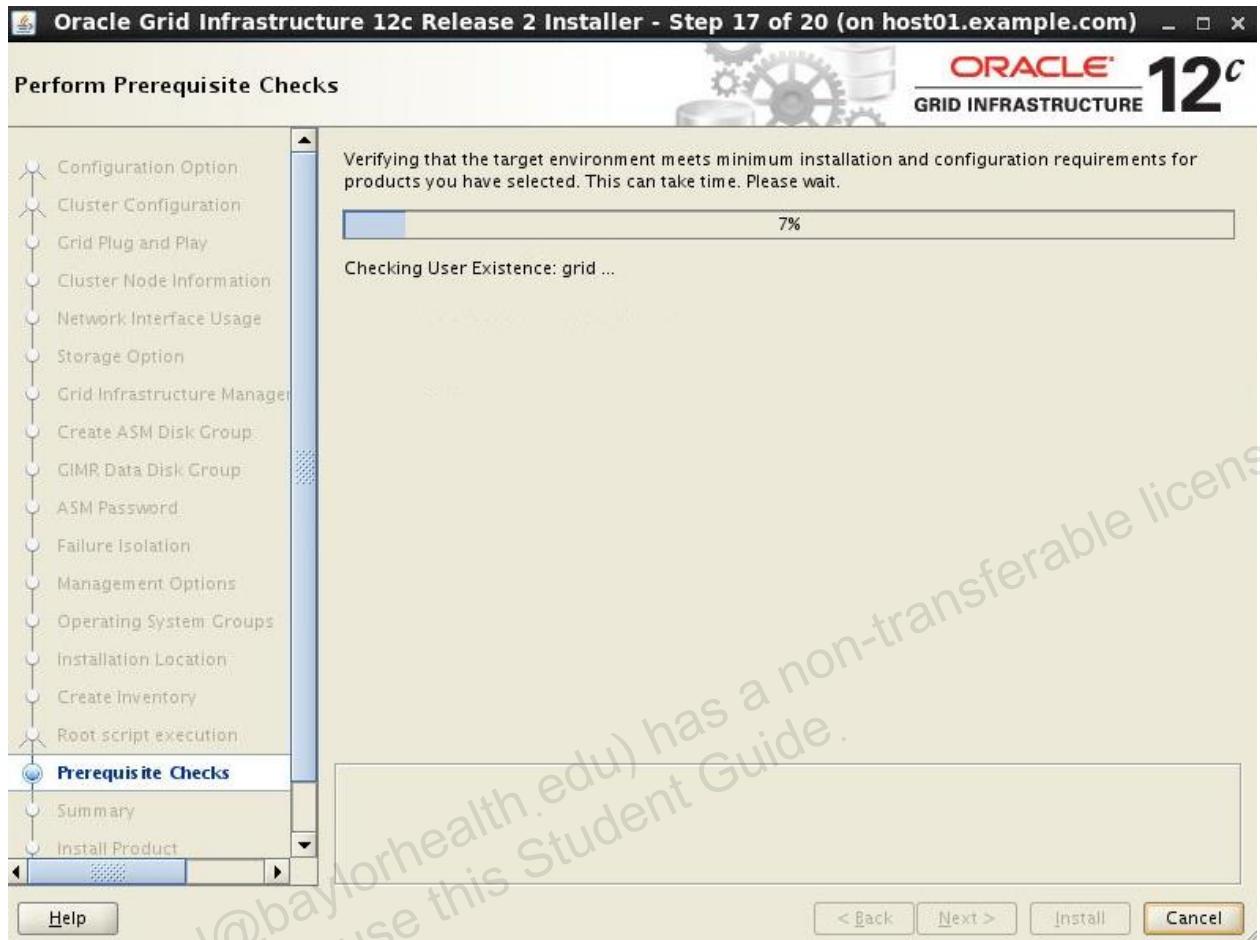
25. On the Create Inventory screen, click Next to accept the default installation inventory location of /u01/app/oralnventory.



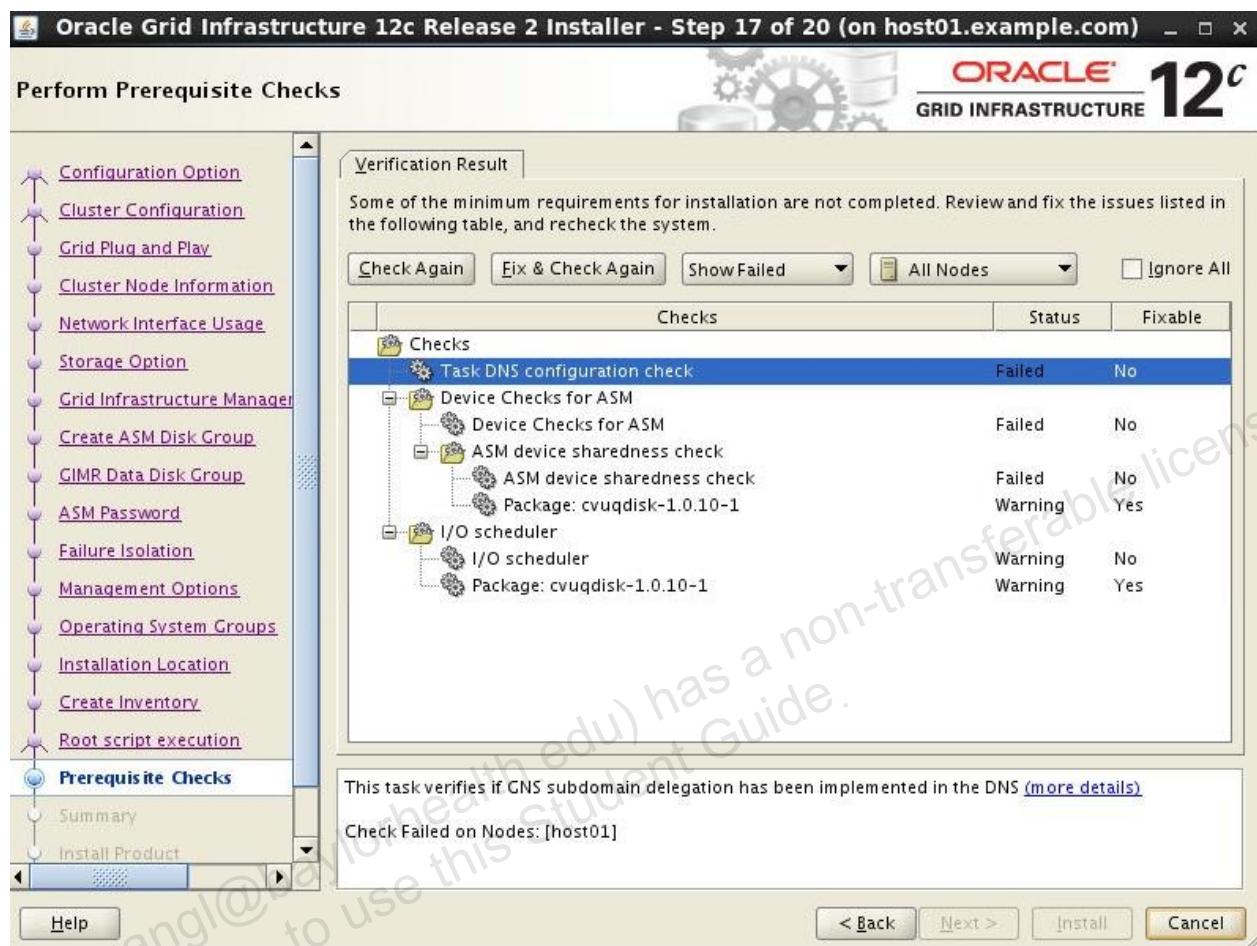
26. On the Root script execution configuration page, click the “Automatically run configuration scripts” check box. Click the “Use “root” user credential” button and enter the root password (as specified in the **Course Practice Environment: Security Credentials** page) in the Password field. Click “Next” to continue.



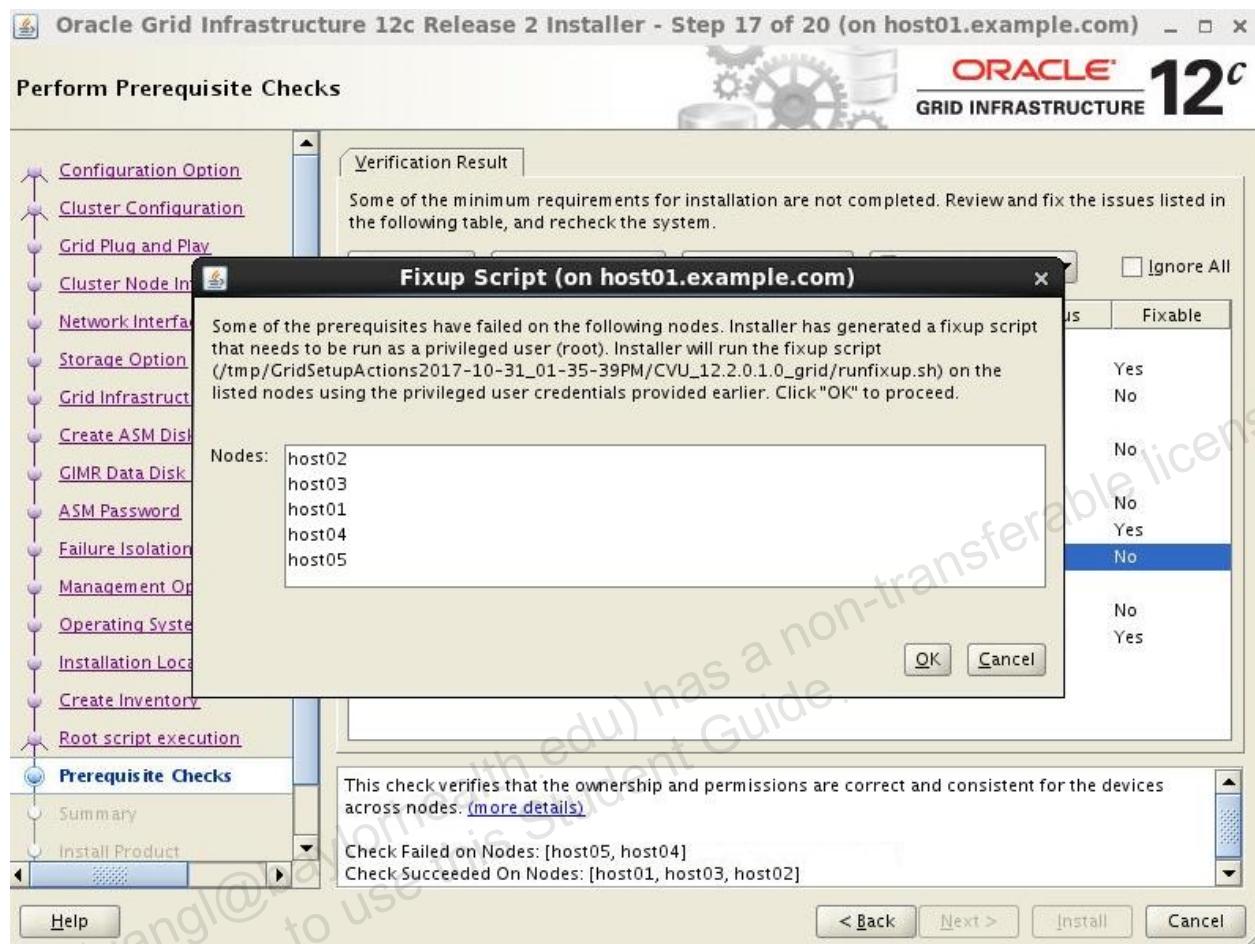
27. Wait while a series of prerequisite checks are performed.



28. On the Perform Prerequisite Checks page, a list of exceptions is displayed after the checks have completed. Click the **Fix and Check Again** button.



29. The Fixup Script dialog box will appear. The generated script will need to be run on host01, host02, host03, host04, and host05 as the root user.



30. Open a new terminal window and ssh as root to host01. Run the fixup script on host01, and then run it on host02, host03, host04, and host05. (Your script name will vary from the example as the filename contains a timestamp.)

```
[oracle@dns ~]$ ssh root@host01
Password:

[root@host01 ~]# /tmp/GridSetupActions2017-10-31_01-35-39PM/CVU_12.2.0.1.0_grid/runfixup.sh
All Fix-up operations were completed successfully.

[root@host01 ~]# ssh host02 /tmp/GridSetupActions2017-10-31_01-35-39PM/CVU_12.2.0.1.0_grid/runfixup.sh
All Fix-up operations were completed successfully.

[root@host01 ~]# ssh host03 /tmp/GridSetupActions2017-10-31_01-35-39PM/CVU_12.2.0.1.0_grid/runfixup.sh
All Fix-up operations were completed successfully.
```

```
[root@host01 ~]# ssh host04 /tmp/GridSetupActions2017-10-31_01-35-39PM/CVU_12.2.0.1.0_grid/runfixup.sh
```

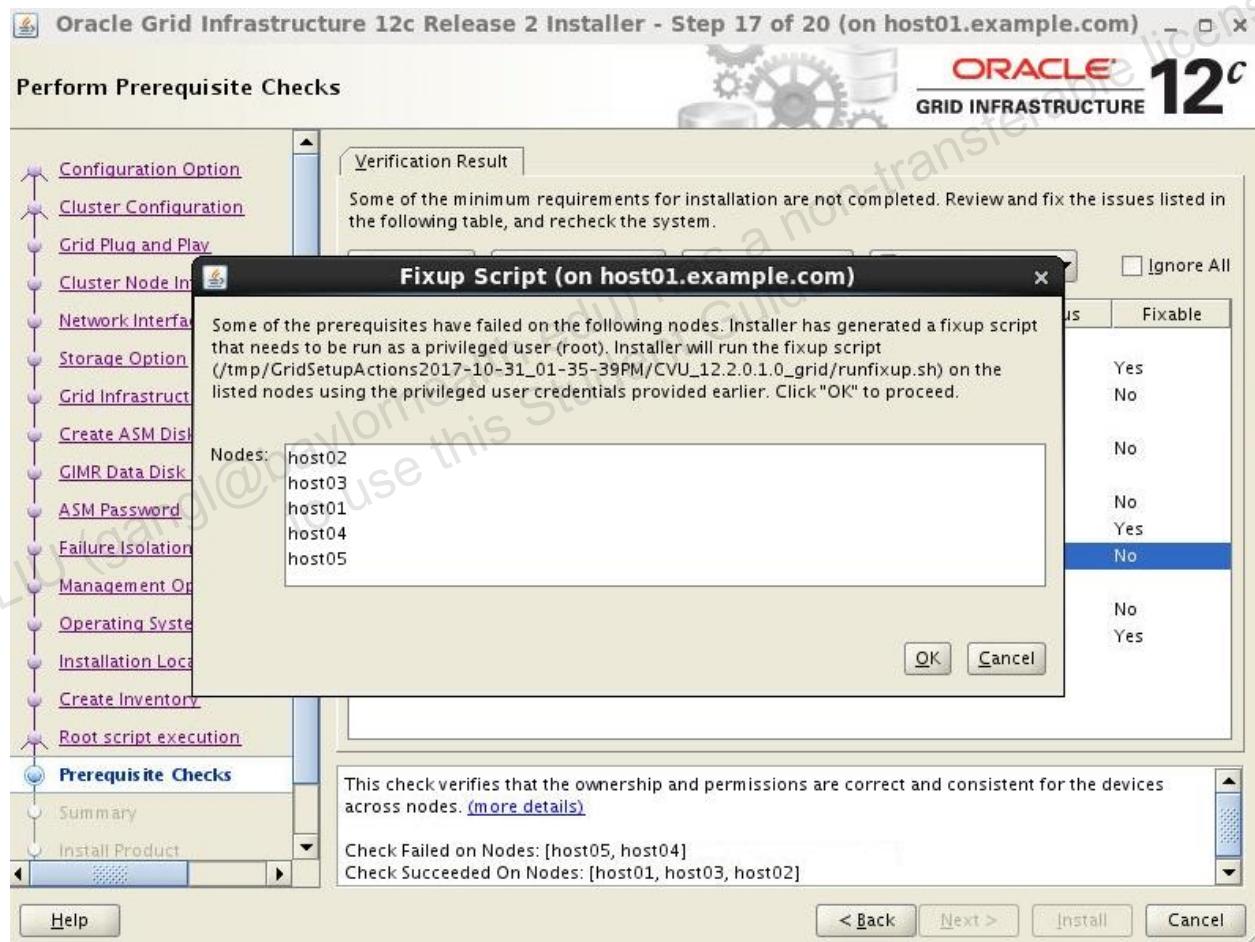
All Fix-up operations were completed successfully.

```
[root@host01 ~]# ssh host05 /tmp/GridSetupActions2017-10-31_01-35-39PM/CVU_12.2.0.1.0_grid/runfixup.sh
```

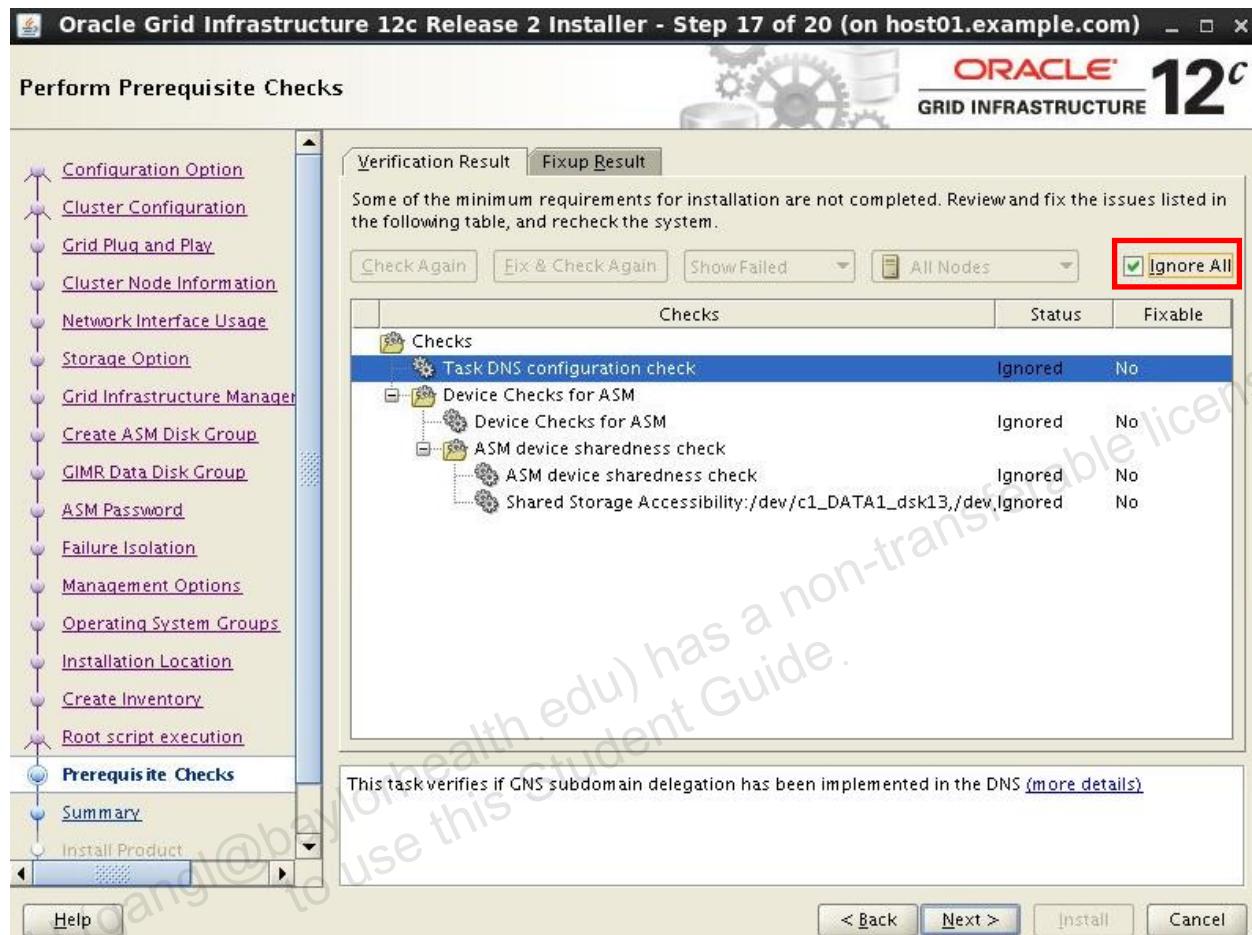
All Fix-up operations were completed successfully.

```
[root@host01 ~]#
```

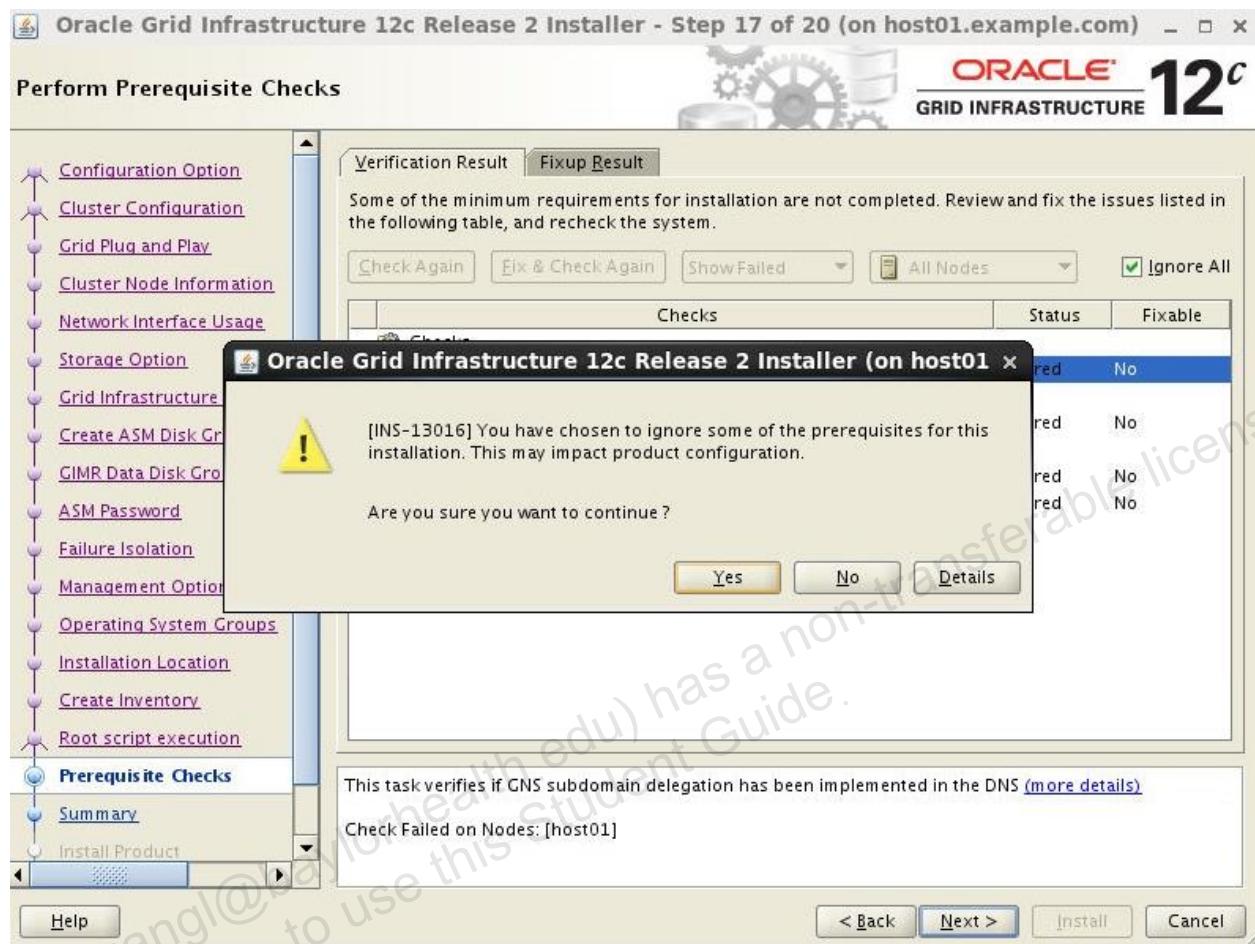
31. Return to the installer and click OK on the Fixup Script dialog box.



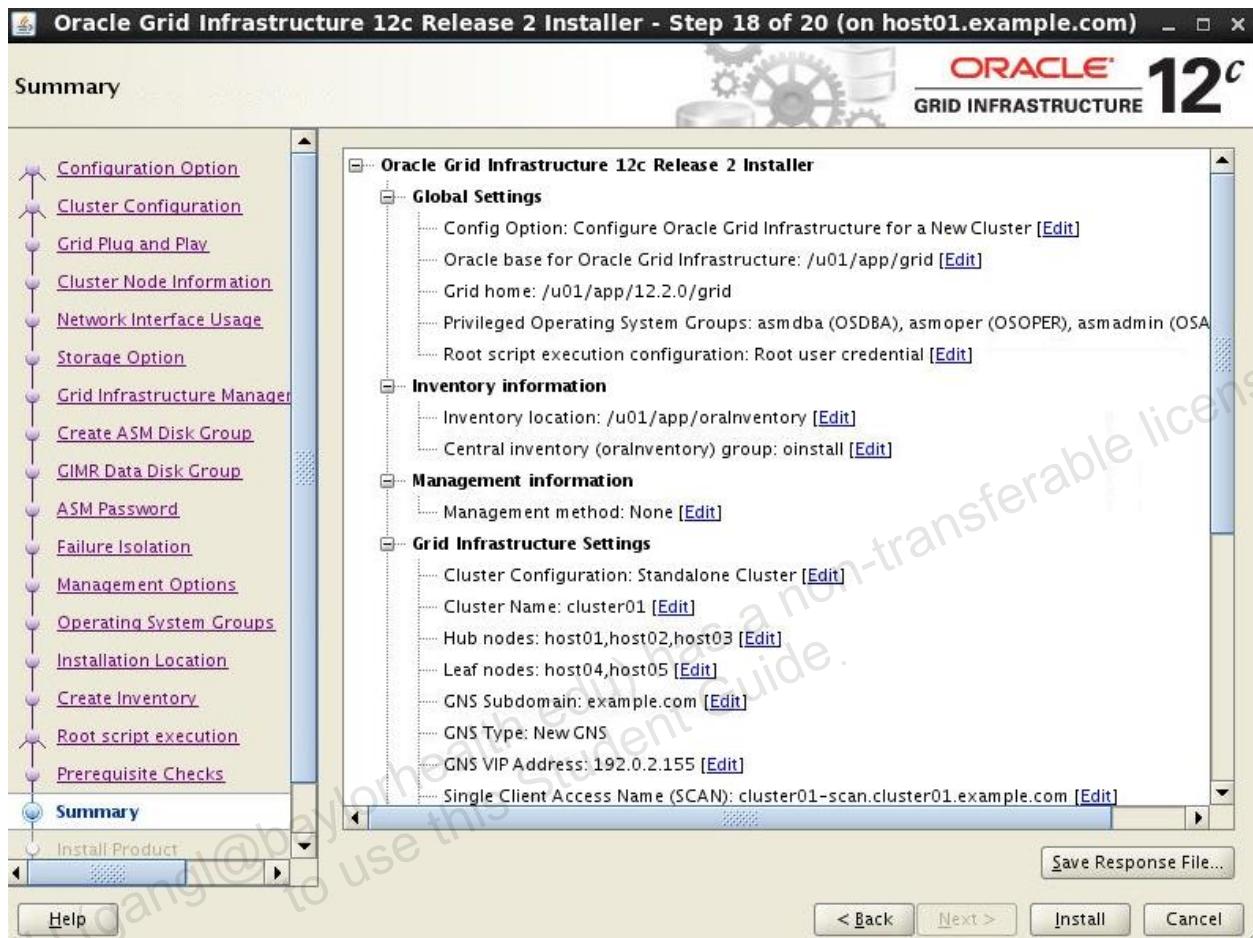
32. After dismissing the Fixup Script dialog box, prerequisites will be checked again. The DNS and ASM exceptions can be ignored. Click the “Ignore All” check box and click “Next” to continue.



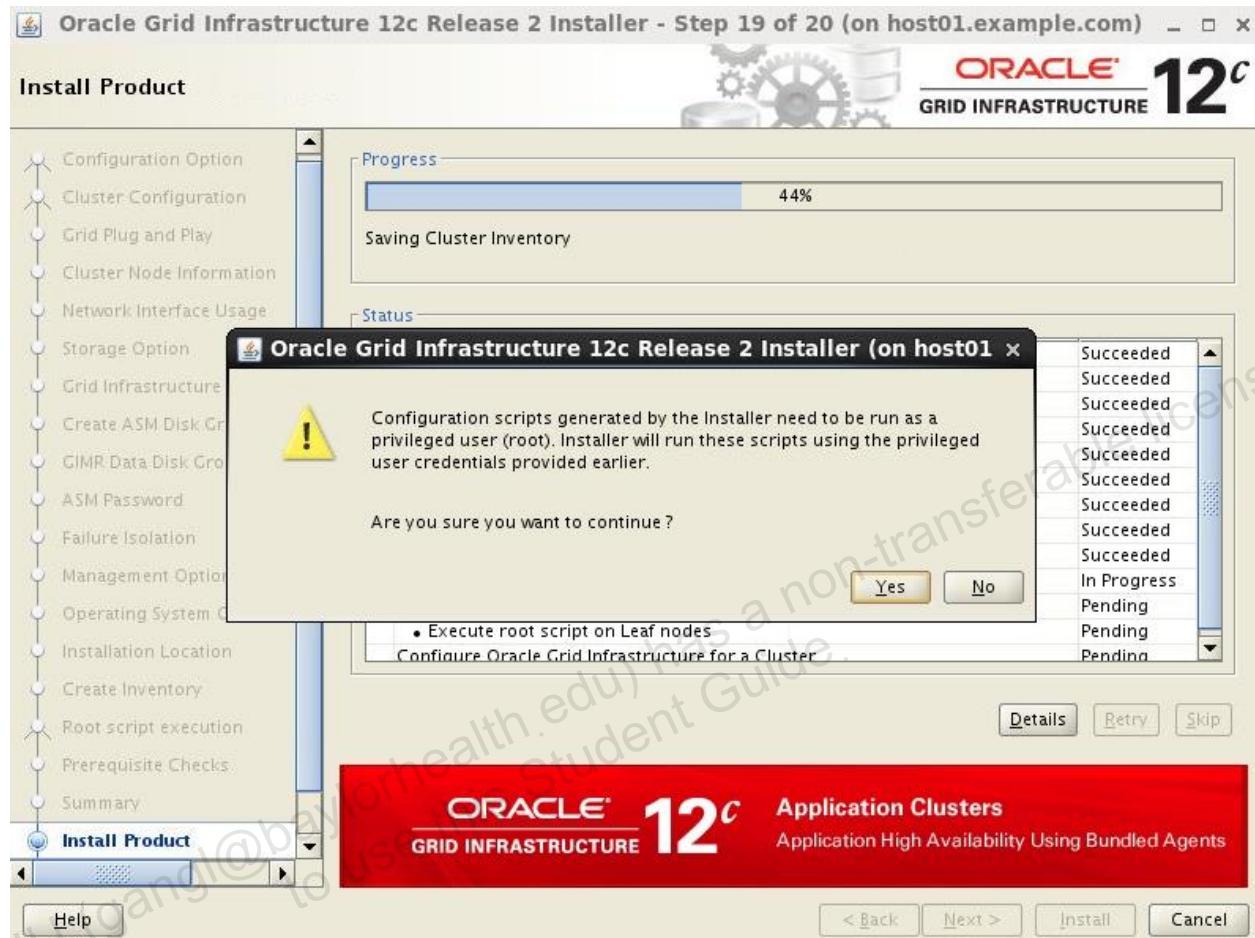
33. A dialog box will appear asking you to confirm your decision to proceed. Click Yes to continue.



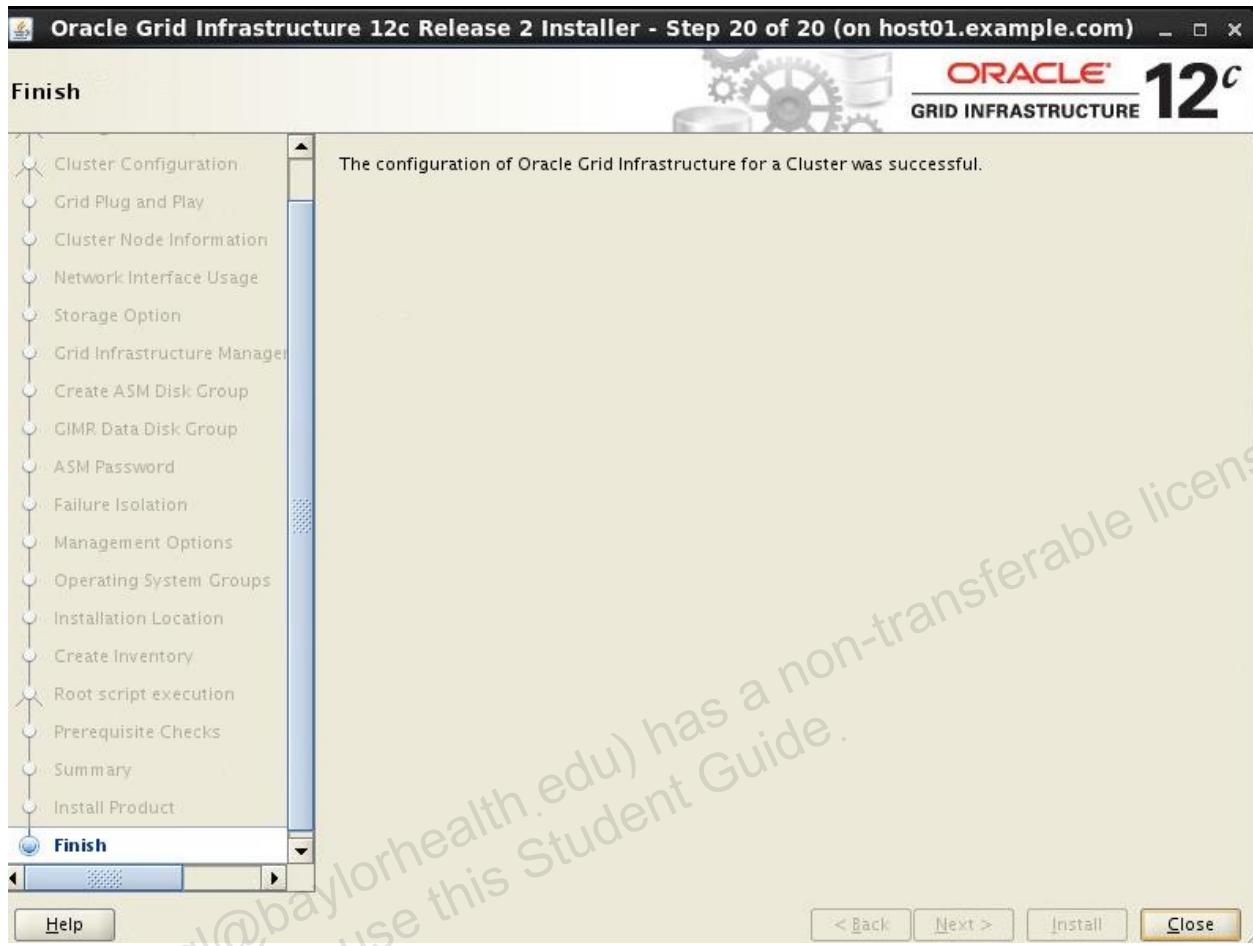
34. Examine the Summary screen. When ready, click Install to begin the installation. Oracle Grid Infrastructure release 12.2 will now install on the cluster. The Install Product screen follows the course of the installation.



35. After the installer runs for a short time, a dialog box will open and confirm you want to run the `root` scripts using the credentials provided earlier. Click Yes. *It takes about 65min to complete the installation at this stage (44%).*



36. When the installation is complete, click “Close”.



37. Back in your terminal session, configure the environment using the `oraenv` script. Enter `+ASM1` when you are prompted for an `ORACLE_SID` value.

```
[grid@host01 ~]$ . oraenv
ORACLE_SID = [grid] ? +ASM1
The Oracle base has been set to /u01/app/grid
[grid@host01 ~]$
```

38. Now check the configuration of the nodes, asm, and cluster.

```
[grid@host01 ~]$ crsctl get cluster class
CRS-41008: Cluster class is 'Standalone Cluster'

[grid@host01 ~]$ crsctl get cluster mode status
Cluster is running in "flex" mode

[grid@host01 ~]$ crsctl get cluster configuration
Name      : cluster01
```

```

Class      : Standalone Cluster
Type       : flex
The cluster is not extended.

[grid@host01 ~]$ crsctl get node role status -all
Node 'host01' active role is 'hub'
Node 'host02' active role is 'hub'
Node 'host03' active role is 'hub'
Node 'host04' active role is 'leaf'
Node 'host05' active role is 'leaf'

[grid@host01 ~]$ asmcmd showclustermode
ASM cluster : Flex mode enabled

[grid@host01 ~]$

```

39. Check the status of the cluster. Ensure that all the listed services are online on all the cluster nodes.

```

[grid@host01 ~]$ crsctl check cluster -all
*****
host01:
CRS-4537: Cluster Ready Services is online
CRS-4529: Cluster Synchronization Services is online
CRS-4533: Event Manager is online
*****
host02:
CRS-4537: Cluster Ready Services is online
CRS-4529: Cluster Synchronization Services is online
CRS-4533: Event Manager is online
*****
host03:
CRS-4537: Cluster Ready Services is online
CRS-4529: Cluster Synchronization Services is online
CRS-4533: Event Manager is online
*****
host04:
CRS-4537: Cluster Ready Services is online
CRS-4529: Cluster Synchronization Services is online
CRS-4533: Event Manager is online
*****
host05:
CRS-4537: Cluster Ready Services is online

```

```
CRS-4529: Cluster Synchronization Services is online
CRS-4533: Event Manager is online
*****
[grid@host01 ~]$
```

40. List the Clusterware resources. Ensure that all the Clusterware resources are running as shown in the following output.

[grid@host01 ~]\$ crsctl status resource -t				
Name	Target	State	Server	State details
Local Resources				
ora.ASMNET1LSNR_ASM.lsnr				
	ONLINE	ONLINE	host01	STABLE
	ONLINE	ONLINE	host02	STABLE
	ONLINE	ONLINE	host03	STABLE
ora.DATA.dg				
	ONLINE	ONLINE	host01	STABLE
	ONLINE	ONLINE	host02	STABLE
	ONLINE	ONLINE	host03	STABLE
ora.LISTENER.lsnr				
	ONLINE	ONLINE	host01	STABLE
	ONLINE	ONLINE	host02	STABLE
	ONLINE	ONLINE	host03	STABLE
ora.LISTENER_LEAF.lsnr				
	OFFLINE	OFFLINE	host04	STABLE
	OFFLINE	OFFLINE	host05	STABLE
ora.MGMT.dg				
	ONLINE	ONLINE	host01	STABLE
	ONLINE	ONLINE	host02	STABLE
	ONLINE	ONLINE	host03	STABLE
ora.chad				
	ONLINE	ONLINE	host01	STABLE
	ONLINE	ONLINE	host02	STABLE
	ONLINE	ONLINE	host03	STABLE
	ONLINE	ONLINE	host04	STABLE
	ONLINE	ONLINE	host05	STABLE
ora.net1.network				
	ONLINE	ONLINE	host01	STABLE
	ONLINE	ONLINE	host02	STABLE
	ONLINE	ONLINE	host03	STABLE
ora.ons				
	ONLINE	ONLINE	host01	STABLE

	ONLINE	ONLINE	host02	STABLE
	ONLINE	ONLINE	host03	STABLE
ora.proxy_advm				
	OFFLINE	OFFLINE	host01	STABLE
	OFFLINE	OFFLINE	host02	STABLE
	OFFLINE	OFFLINE	host03	STABLE
<hr/>				
Cluster Resources				
<hr/>				
ora.LISTENER_SCAN1.lsnr	1	ONLINE	ONLINE	host03
				STABLE
ora.LISTENER_SCAN2.lsnr	1	ONLINE	ONLINE	host02
				STABLE
ora.LISTENER_SCAN3.lsnr	1	ONLINE	ONLINE	host01
				STABLE
ora.MGMTLSNR	1	ONLINE	ONLINE	host01
				169.254.116.21
				192.168.1.101
				192.168.2.1
				01, STABLE
ora.asm	1	ONLINE	ONLINE	host01
	2	ONLINE	ONLINE	host03
	3	ONLINE	ONLINE	host02
ora.cvu	1	ONLINE	ONLINE	host01
ora.gns	1	ONLINE	ONLINE	host01
ora.gns.vip	1	ONLINE	ONLINE	host01
ora.host01.vip	1	ONLINE	ONLINE	host01
ora.host02.vip	1	ONLINE	ONLINE	host01
ora.host03.vip	1	ONLINE	ONLINE	host02
ora.mgmtdb	1	ONLINE	ONLINE	host03
ora.qosmserver	1	ONLINE	ONLINE	host01
ora.scan1.vip	1	ONLINE	ONLINE	host01
ora.scan2.vip	1	ONLINE	ONLINE	host03

```
1           ONLINE  ONLINE      host02      STABLE
ora.scan3.vip
1           ONLINE  ONLINE      host01      STABLE
-----
[grid@host01 ~]$
```

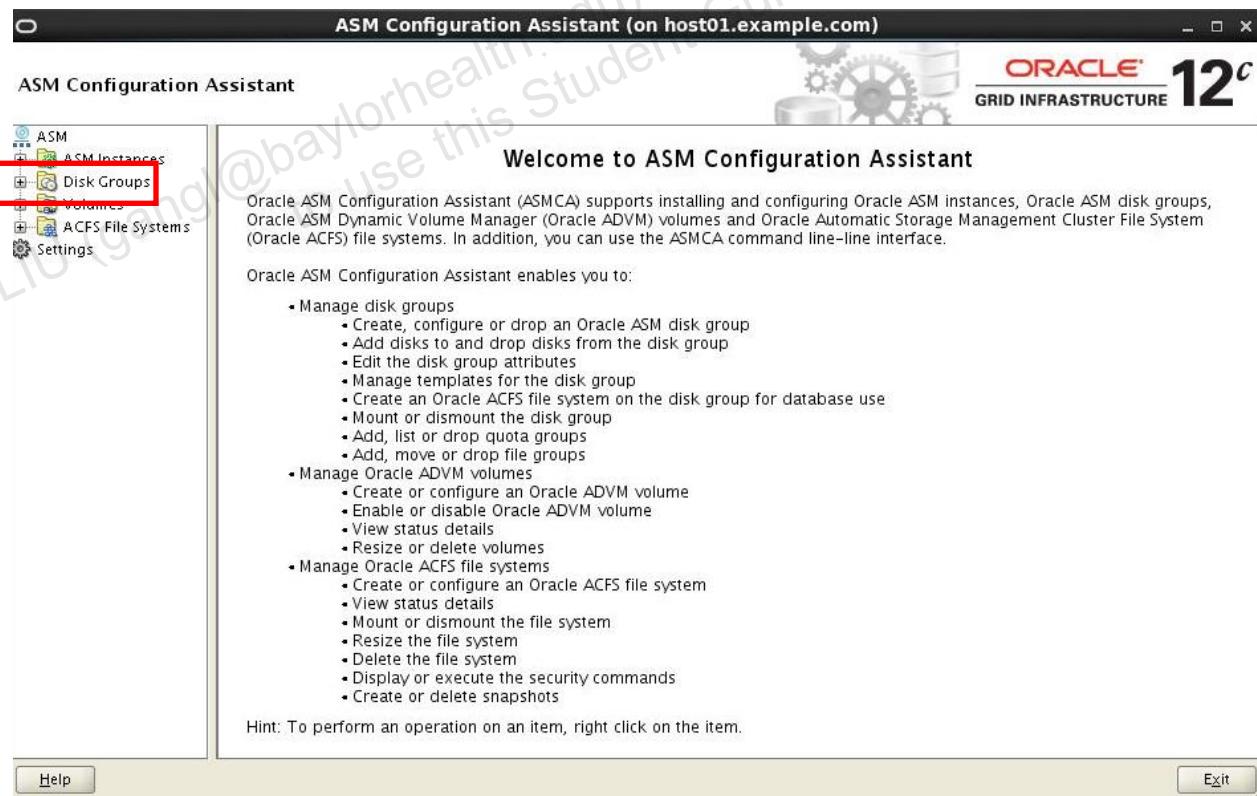
41. Stop and disable the `ora.cvu` resource using SRVCTL to conserve disk space.

```
[grid@host01 ~]$ srvctl stop cvu
[grid@host01 ~]$ srvctl disable cvu
[grid@host01 ~]$
```

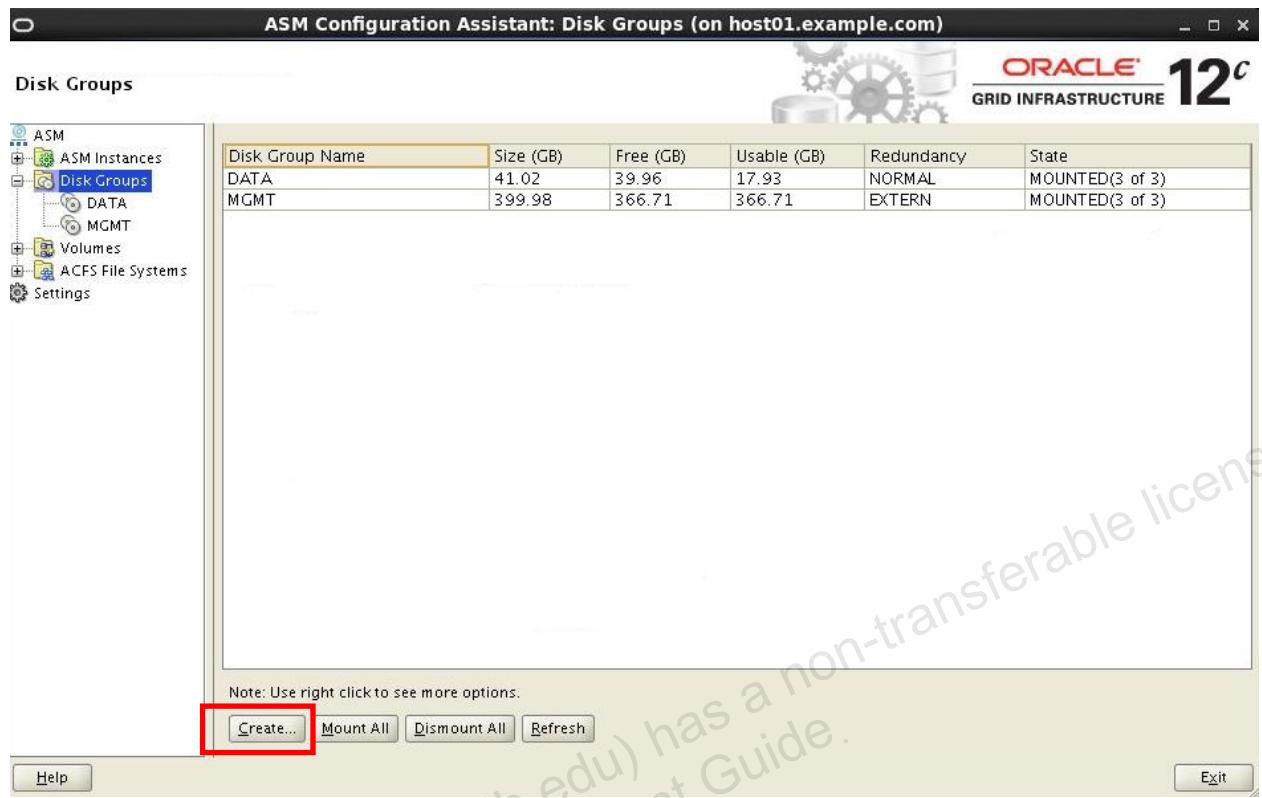
42. Next, you will create another ASM disk group to host the Fast Recovery Area (FRA). Start the ASM Configuration Assistant (`asmca`).

```
[grid@host01 ~]$ asmca
```

43. After the ASM Configuration Assistant appears, click on Disk Groups.



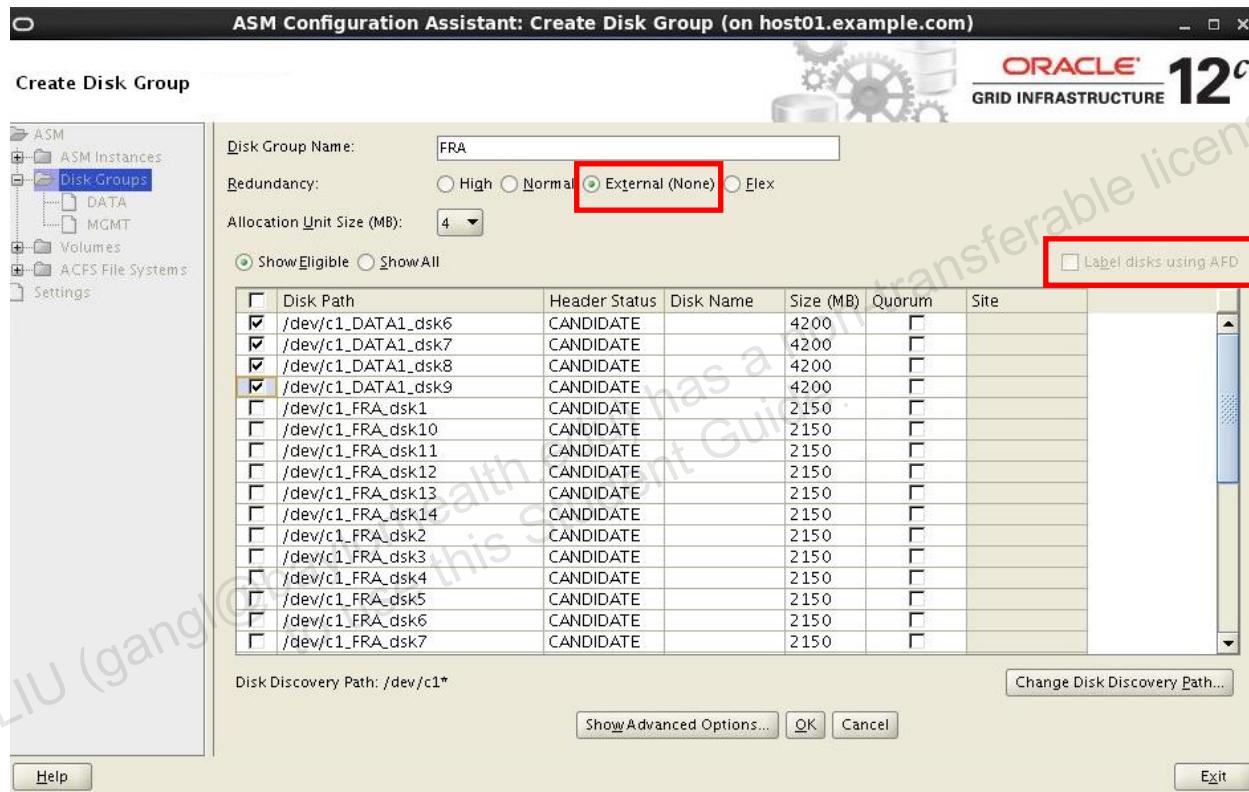
44. Click on Create.



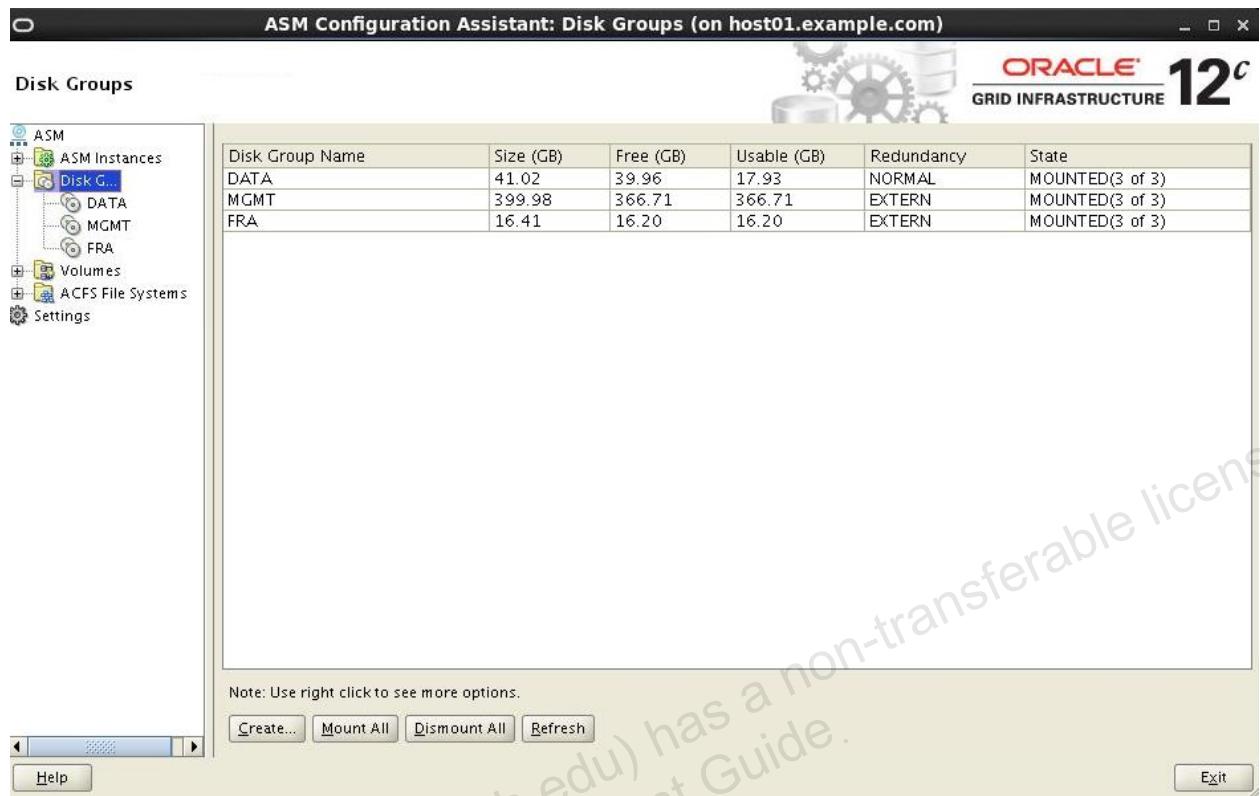
45. In the Create Disk Group window, enter FRA as the disk group name and select the following four candidate disks (c1_DATA1_dsk).

- /dev/c1_DATA1_dsk6
- /dev/c1_DATA1_dsk7
- /dev/c1_DATA1_dsk8
- /dev/c1_DATA1_dsk9

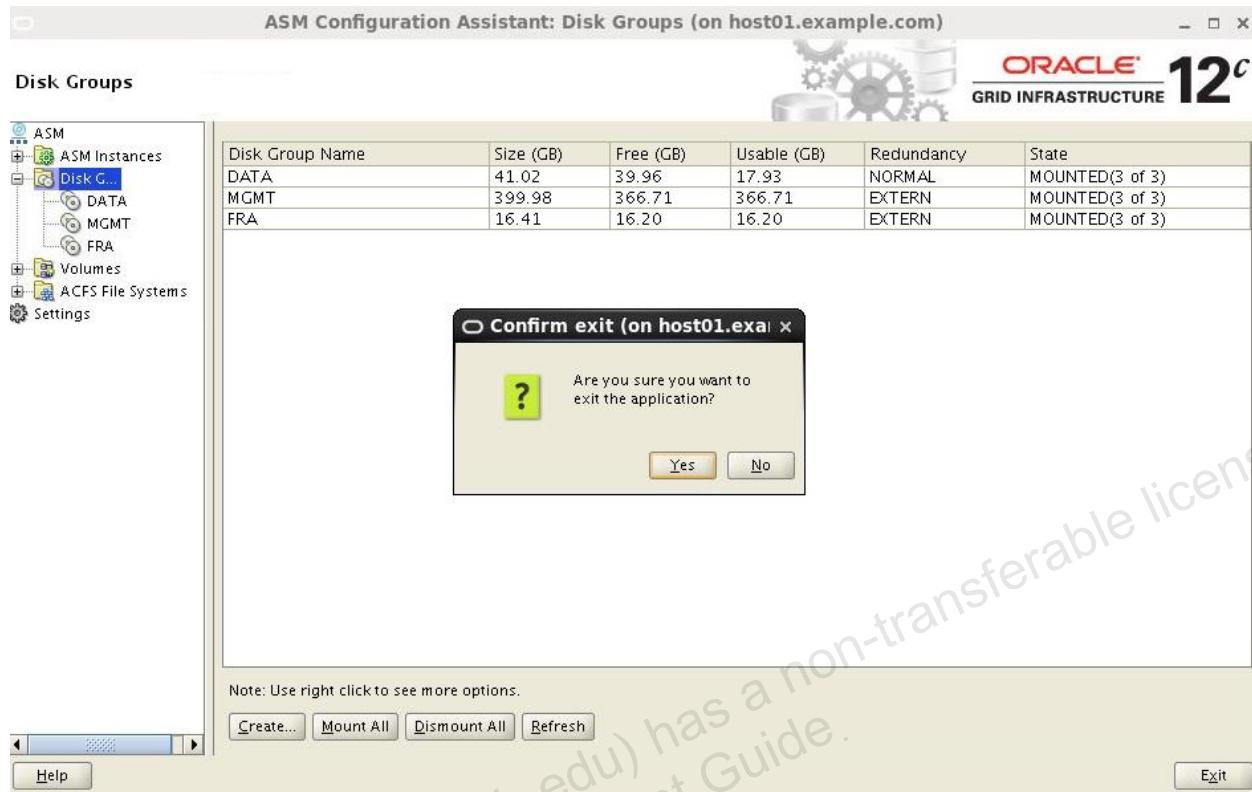
Make sure the Redundancy is **External** and The ***Label disks using AFD*** is **NOT** selected. Then click OK to create the disk group.



46. Click Exit to quit the ASM Configuration Assistant.



47. Click Yes to confirm that you want to quit the ASM Configuration Assistant.



48. Close all terminal windows opened for this practice.

Practices for Lesson 2: RAC Databases Overview & Architecture

Practices for Lesson 2

There are no practices for this lesson.

Practices for Lesson 3: Grid Infrastructure Overview

Practices for Lesson 3: Overview

Overview

In this practice, you will install the database software to both HUB and LEAF nodes in your cluster.

Practice 3-1: Installing RAC Database Software

Overview

In this practice, you will install the database software to both HUB and LEAF nodes in your cluster.

1. Open a terminal window as `root` on `host01`. Change the permissions on `/u01/app` to allow the `oracle` user write privileges. Repeat these commands on `host02`, `host03`, `host04`, and `host05`.

```
[oracle@dns ~]$ ssh root@host01
root@host01's password:

[root@host01 ~]# chmod 775 /u01/app
[root@host01 ~]# ssh host02 chmod 775 /u01/app
[root@host01 ~]# ssh host03 chmod 775 /u01/app
[root@host01 ~]# ssh host04 chmod 775 /u01/app
[root@host01 ~]# ssh host05 chmod 775 /u01/app
[root@host01 ~]#
```

2. Open a terminal window and `ssh` to `host01` as the `oracle` user. Change the working directory to `/stage/database` and execute `runInstaller`.

```
[oracle@dns ~]$ ssh -X oracle@host01
oracle@host01's password:

[oracle@host01 ~]$ cd /stage/database
[oracle@host01 database]$ ./runInstaller
```

3. On the Configure Security Updates page, uncheck “I wish to receive security updates via My Oracle Support” and click Next.



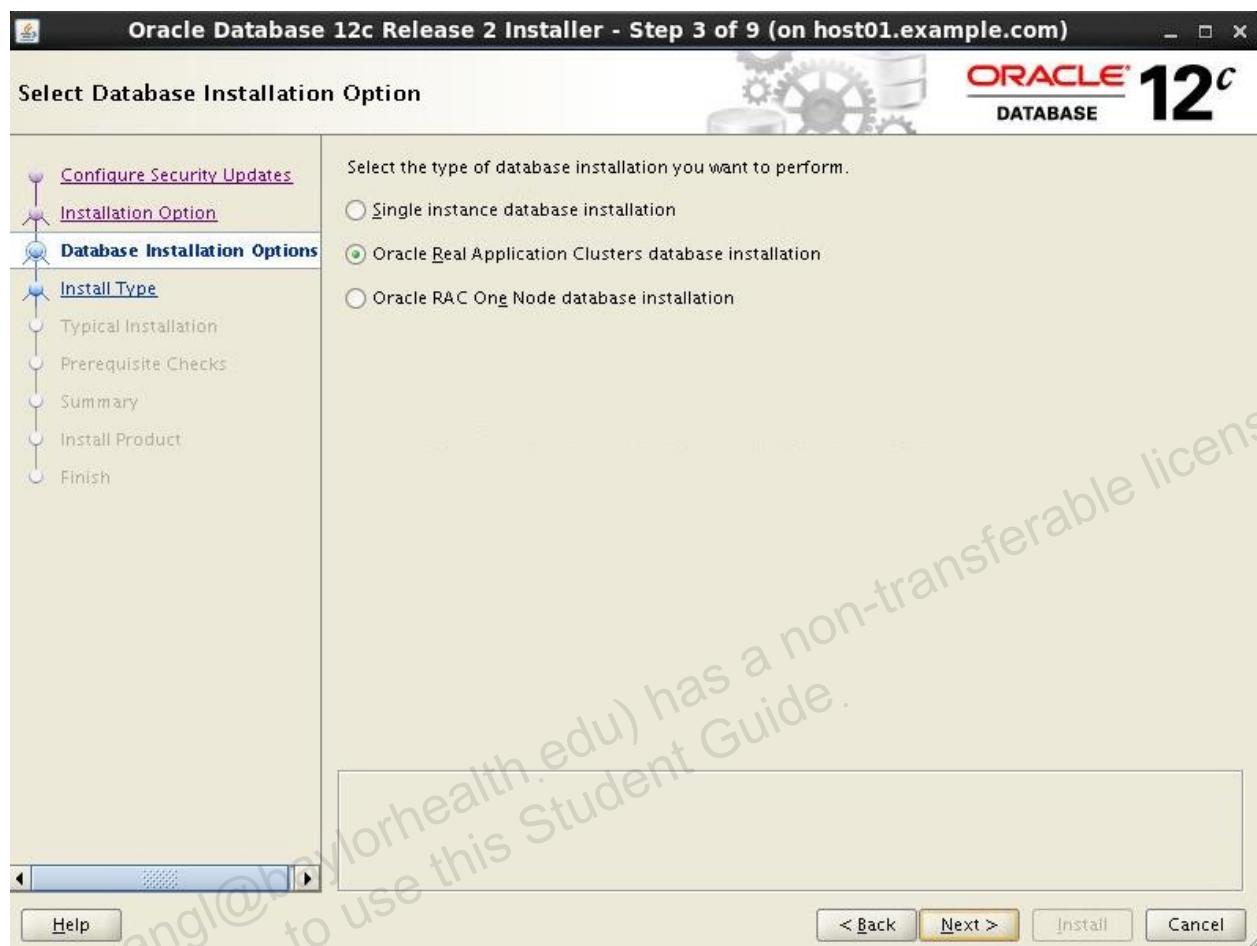
4. Click Yes on the dialog box confirming your wish regarding security updates.



5. On the Select Installation Option page, select “Install database software only” and click Next.

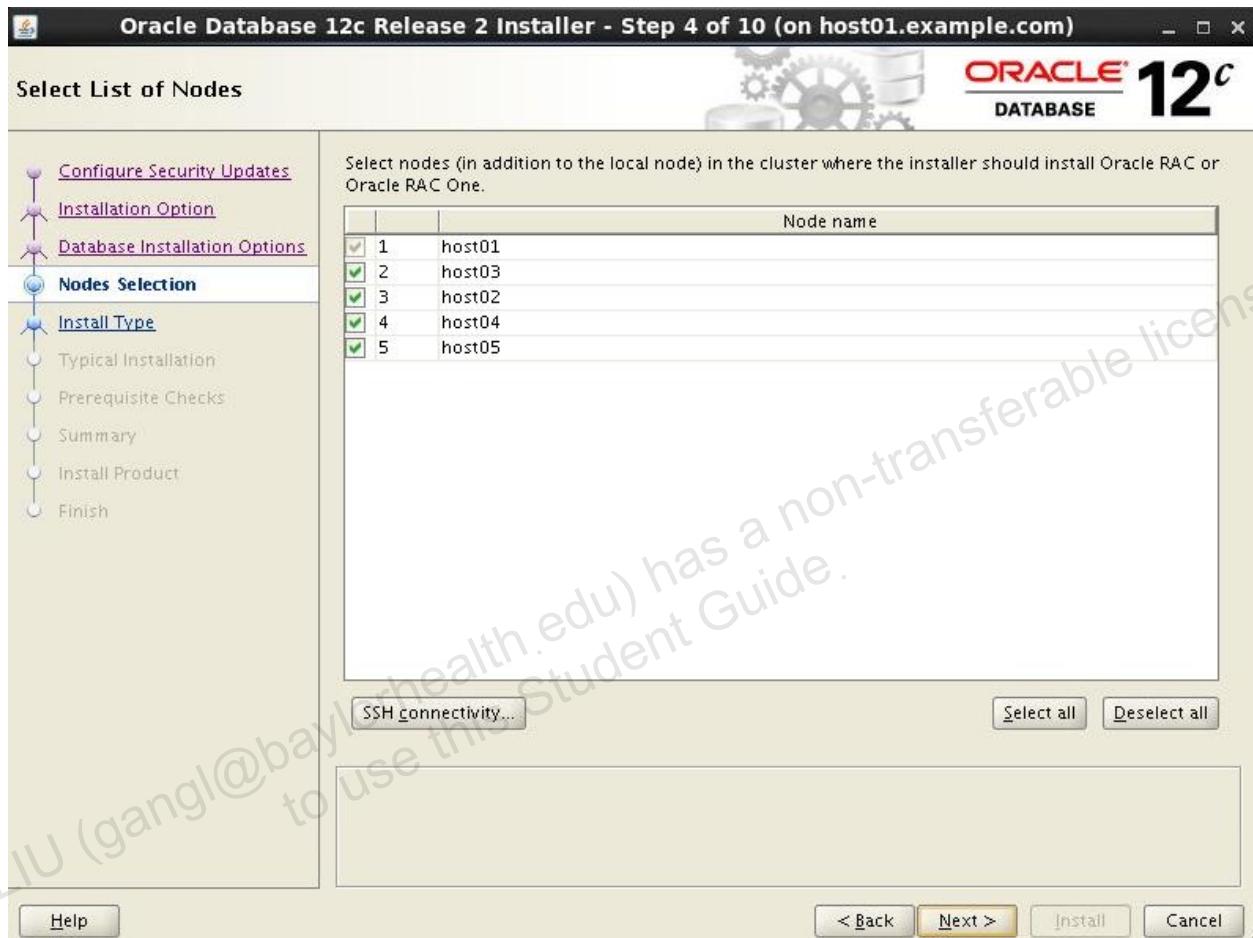


6. On the Select Database Installation Option page select “Oracle Real Application Clusters database installation” and click Next.



7. On the Select List of Nodes page, ensure all nodes are checked and click Next.

Note: Oracle recommends that you install the Oracle Database software on all the cluster nodes, even Leaf Nodes. This simplifies things if you ever want to convert a Leaf Node to a Hub Node and run database instances on it or if you plan to run database instances in the Leaf Nodes.



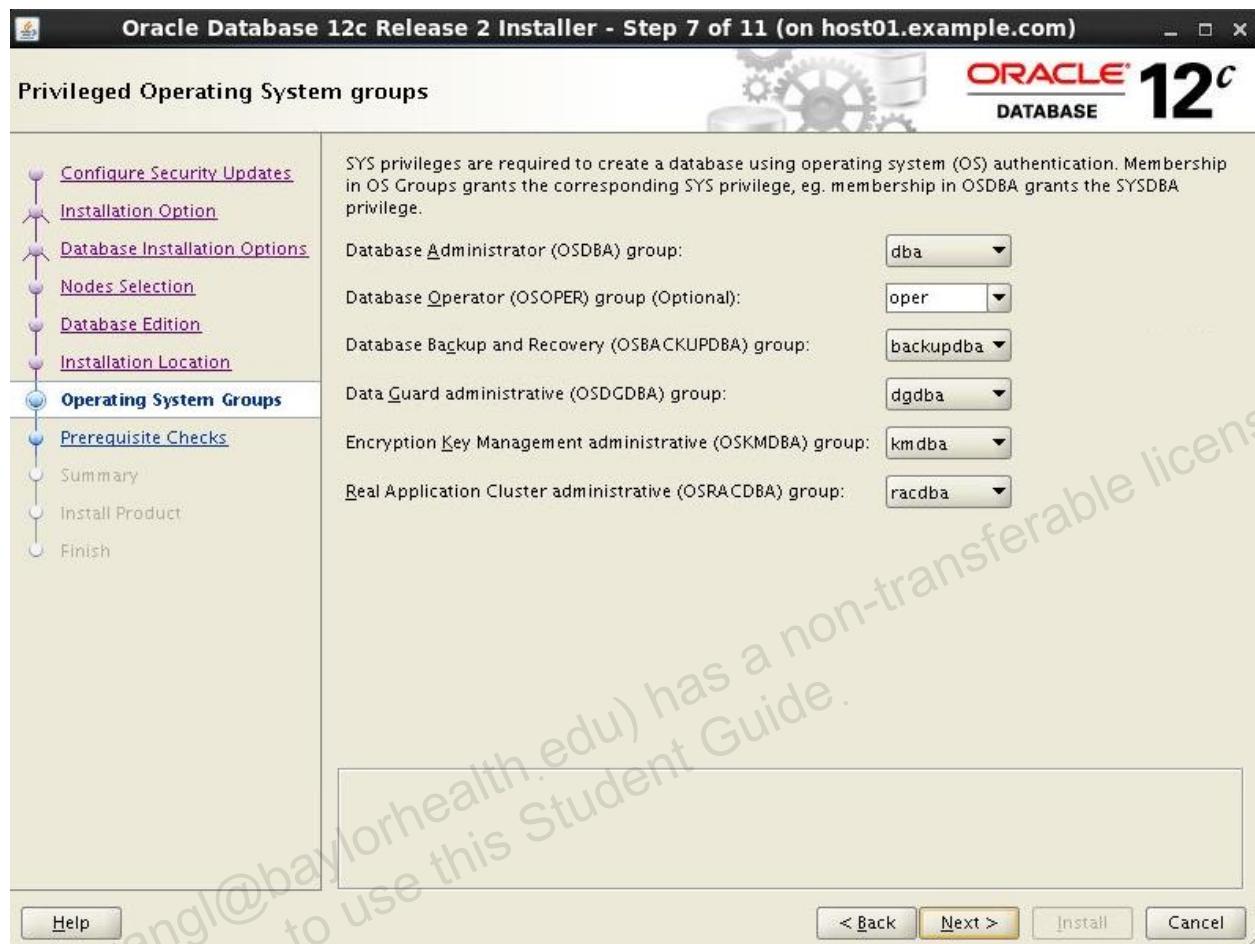
- On the Select Database Edition page, ensure “Enterprise Edition” is selected and click Next.



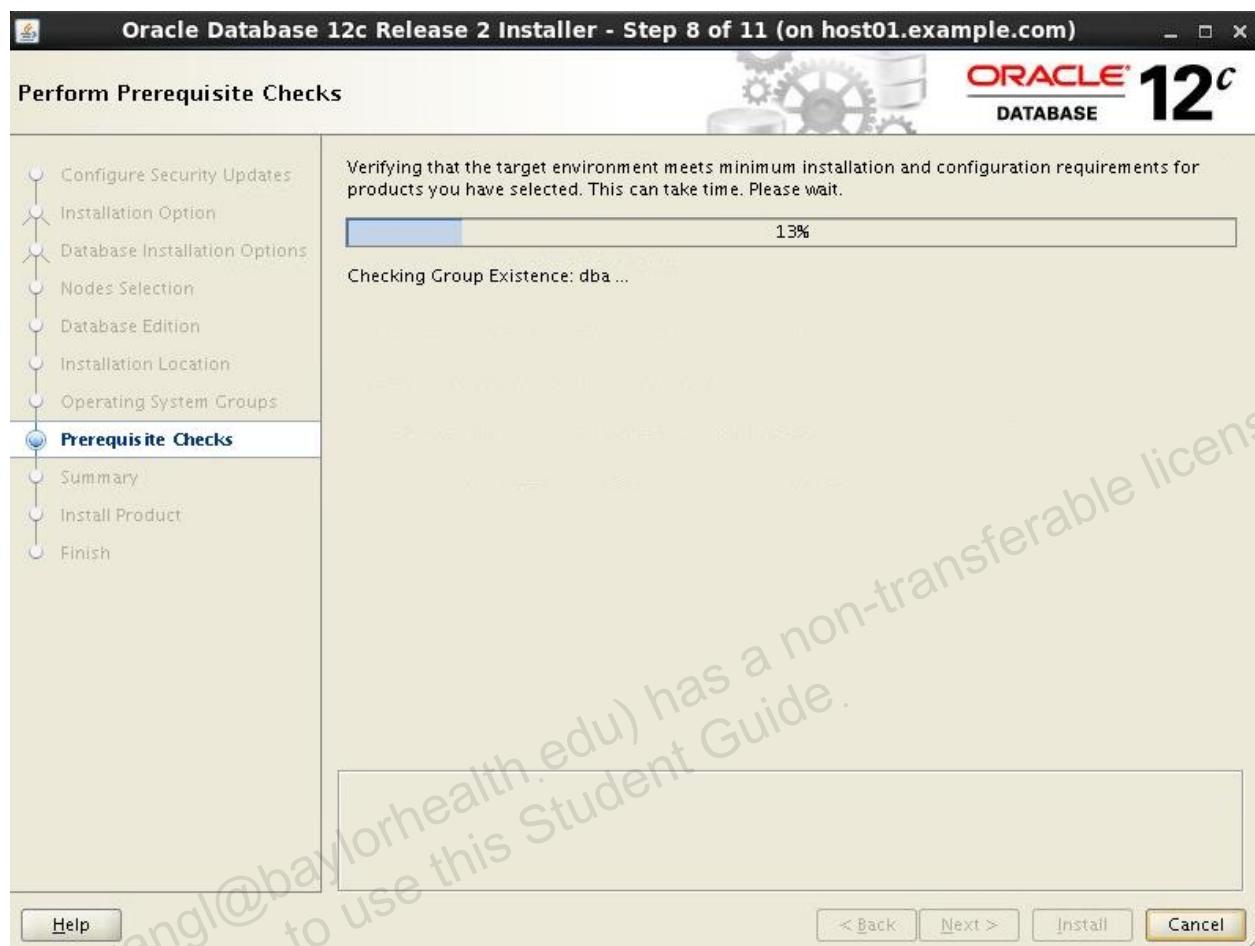
9. On the Specify Installation Location page, accept the defaults as shown here and click Next.



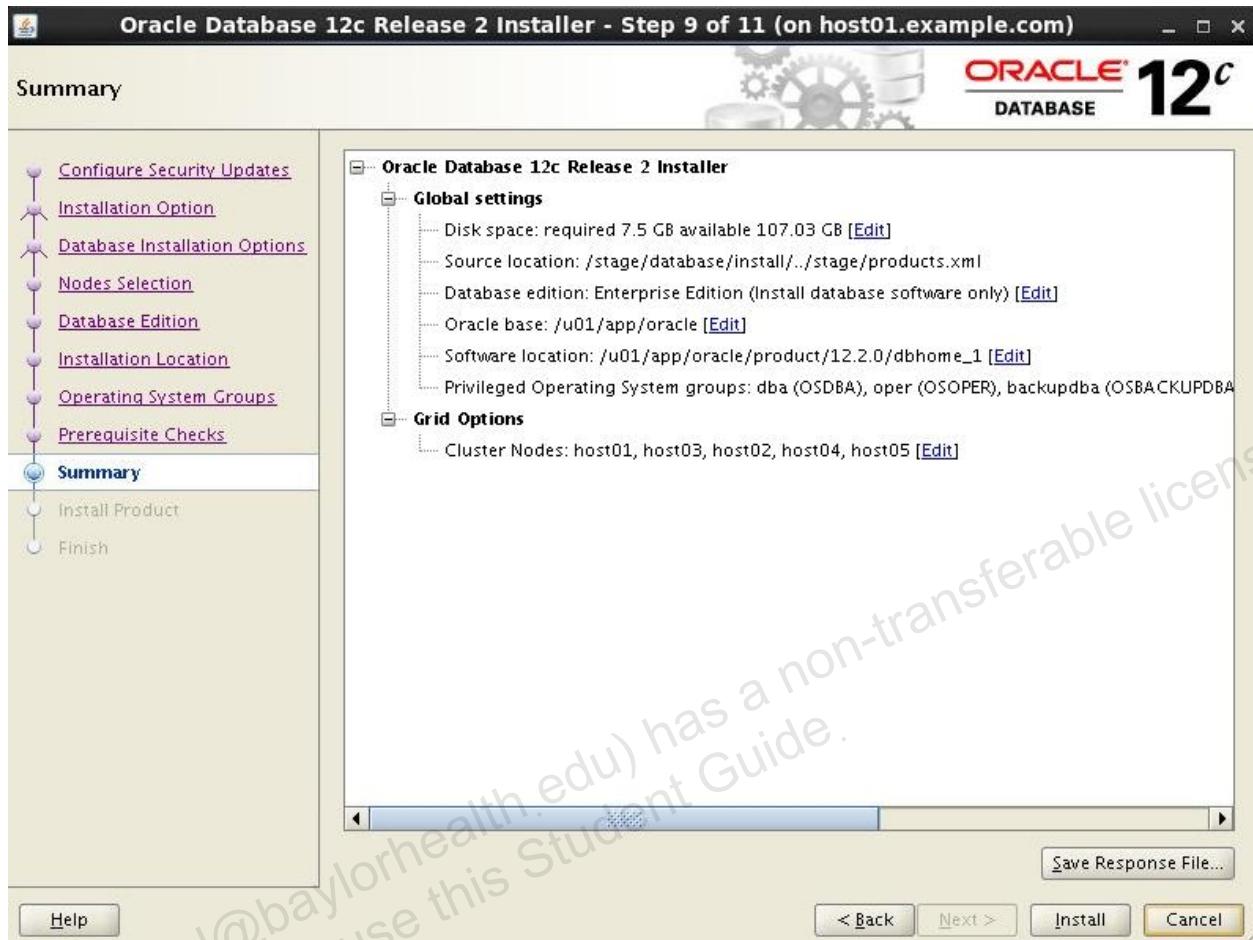
10. On the Privileged Operating System groups, accept the defaults as shown here and click Next.



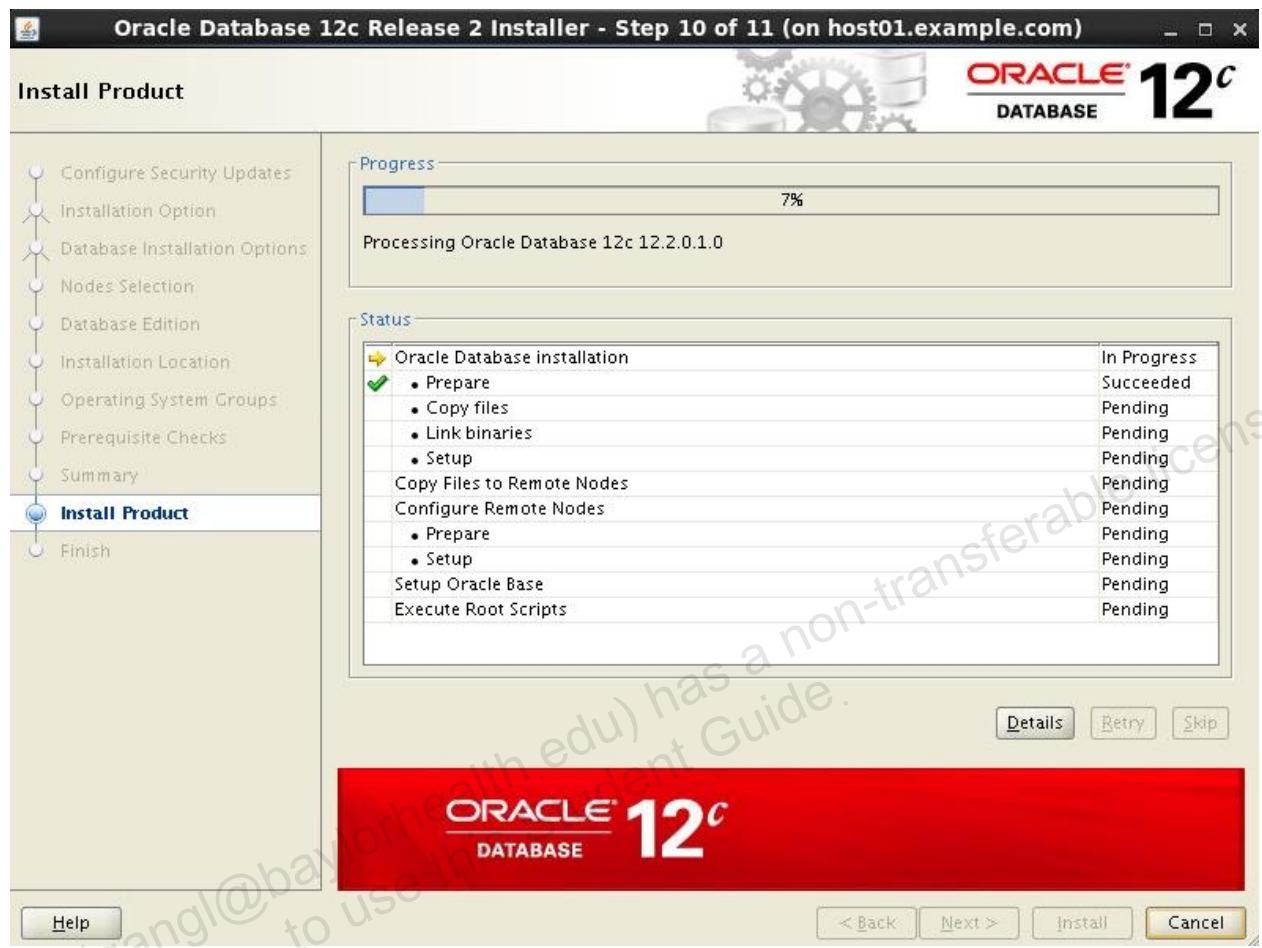
11. The Perform Prerequisite Checks page shows the progress of checks performed before the installation begins.



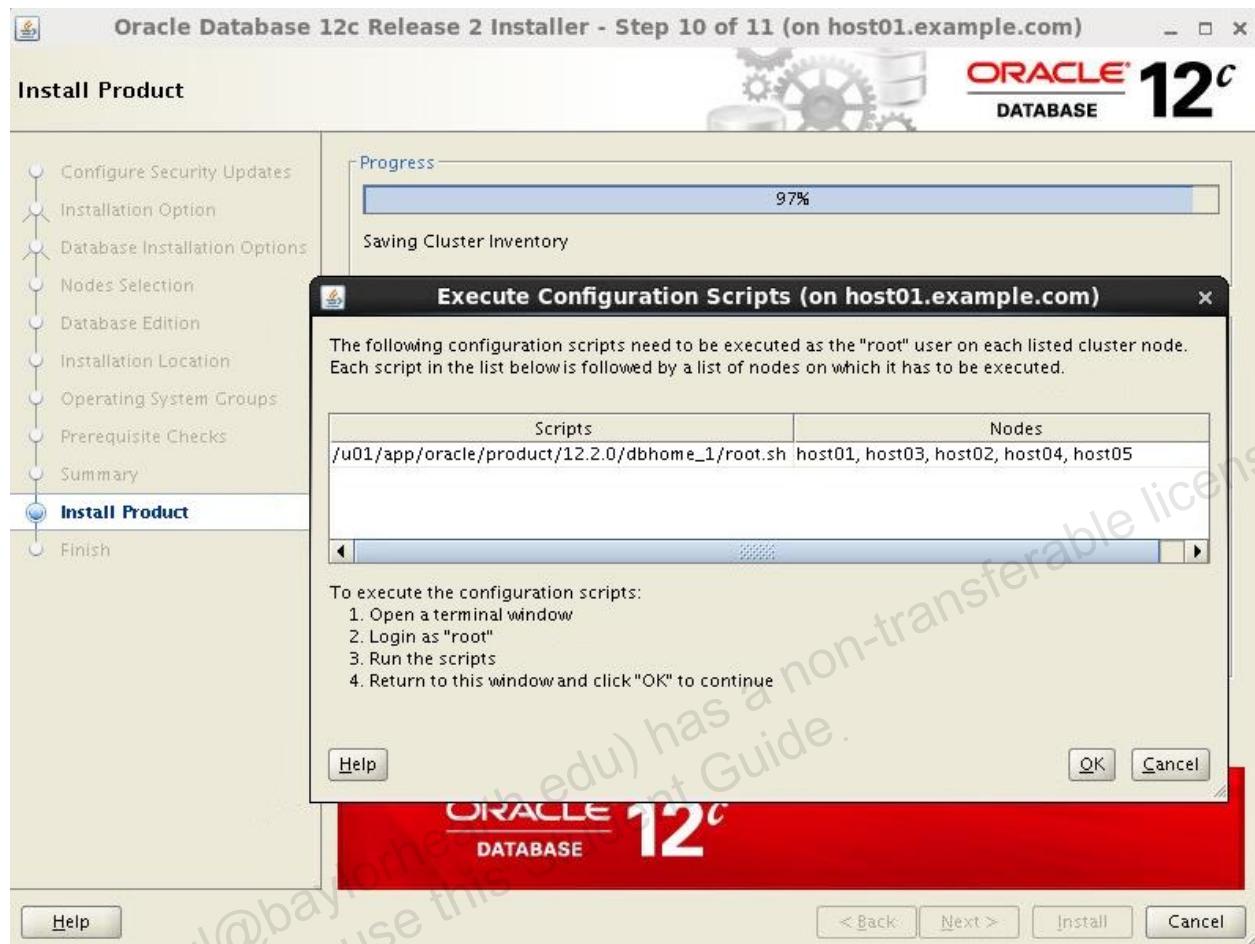
12. Review the information on the Summary page and click Next.



13. Monitor the installation on the Install Product page. *It takes about 25min to complete the installation at this stage.*



14. At the end of the installation (97%), you will be prompted to run the `root` scripts.



15. Return to the `root` terminal and run the `root` scripts on host01, host02, host03, host04, and host05.

```
[root@host01 ~]# /u01/app/oracle/product/12.2.0/dbhome_1/root.sh
Performing root user operation.
```

The following environment variables are set as:

```
ORACLE_OWNER= oracle
ORACLE_HOME= /u01/app/oracle/product/12.2.0/dbhome_1
```

Enter the full pathname of the local bin directory:
[/usr/local/bin]:

The contents of "dbhome" have not changed. No need to overwrite.
The contents of "oraenv" have not changed. No need to overwrite.
The contents of "coraenv" have not changed. No need to overwrite.

Entries will be added to the /etc/oratab file as needed by
Database Configuration Assistant when a database is created
Finished running generic part of root script.

Now product-specific root actions will be performed.

```
[root@host01 ~]# ssh host02  
/u01/app/oracle/product/12.2.0/dbhome_1/root.sh
```

Performing root user operation.

The following environment variables are set as:

```
ORACLE_OWNER= oracle
```

```
ORACLE_HOME= /u01/app/oracle/product/12.2.0/dbhome_1
```

Enter the full pathname of the local bin directory:

[/usr/local/bin]:

The contents of "dbhome" have not changed. No need to overwrite.

The contents of "oraenv" have not changed. No need to overwrite.

The contents of "coraenv" have not changed. No need to overwrite.

Entries will be added to the /etc/oratab file as needed by
Database Configuration Assistant when a database is created
Finished running generic part of root script.

Now product-specific root actions will be performed.

```
[root@host01 ~]# ssh host03  
/u01/app/oracle/product/12.2.0/dbhome_1/root.sh
```

Performing root user operation.

The following environment variables are set as:

```
ORACLE_OWNER= oracle
```

```
ORACLE_HOME= /u01/app/oracle/product/12.2.0/dbhome_1
```

Enter the full pathname of the local bin directory:

[/usr/local/bin]:

The contents of "dbhome" have not changed. No need to overwrite.

The contents of "oraenv" have not changed. No need to overwrite.

The contents of "coraenv" have not changed. No need to overwrite.

Entries will be added to the /etc/oratab file as needed by
Database Configuration Assistant when a database is created
Finished running generic part of root script.

Now product-specific root actions will be performed.

```
[root@host01 ~]# ssh host04  
/u01/app/oracle/product/12.2.0/dbhome_1/root.sh
```

Performing root user operation.

The following environment variables are set as:

```
ORACLE_OWNER= oracle
```

```
ORACLE_HOME= /u01/app/oracle/product/12.2.0/dbhome_1

Enter the full pathname of the local bin directory:
[/usr/local/bin]: 

The contents of "dbhome" have not changed. No need to overwrite.
The contents of "oraenv" have not changed. No need to overwrite.
The contents of "coraenv" have not changed. No need to overwrite.

Entries will be added to the /etc/oratab file as needed by
Database Configuration Assistant when a database is created
Finished running generic part of root script.
Now product-specific root actions will be performed.

[root@host01 ~]# ssh host05
/u01/app/oracle/product/12.2.0/dbhome_1/root.sh
Performing root user operation.

The following environment variables are set as:
ORACLE_OWNER= oracle
ORACLE_HOME= /u01/app/oracle/product/12.2.0/dbhome_1

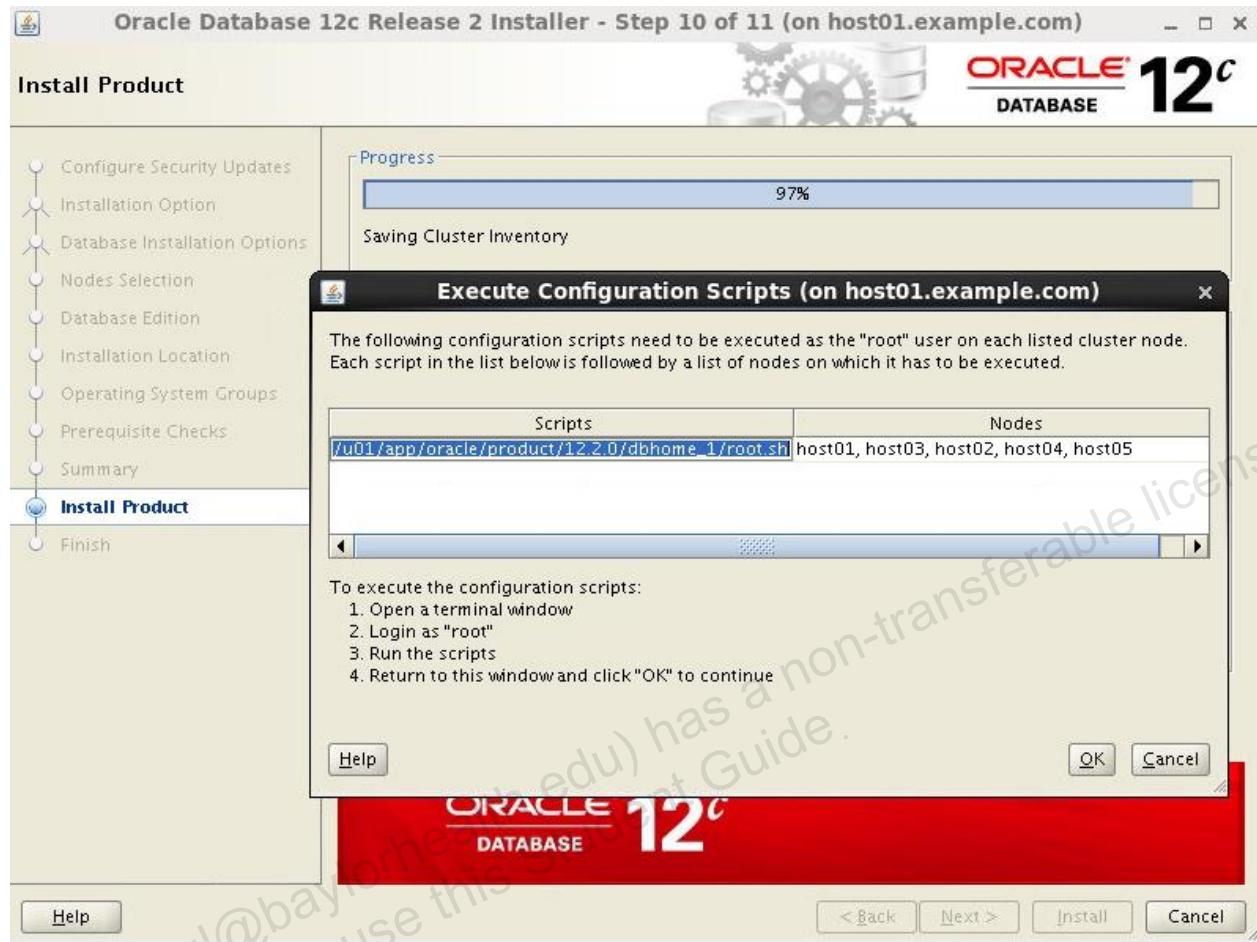
Enter the full pathname of the local bin directory:
[/usr/local/bin]: 

The contents of "dbhome" have not changed. No need to overwrite.
The contents of "oraenv" have not changed. No need to overwrite.
The contents of "coraenv" have not changed. No need to overwrite.

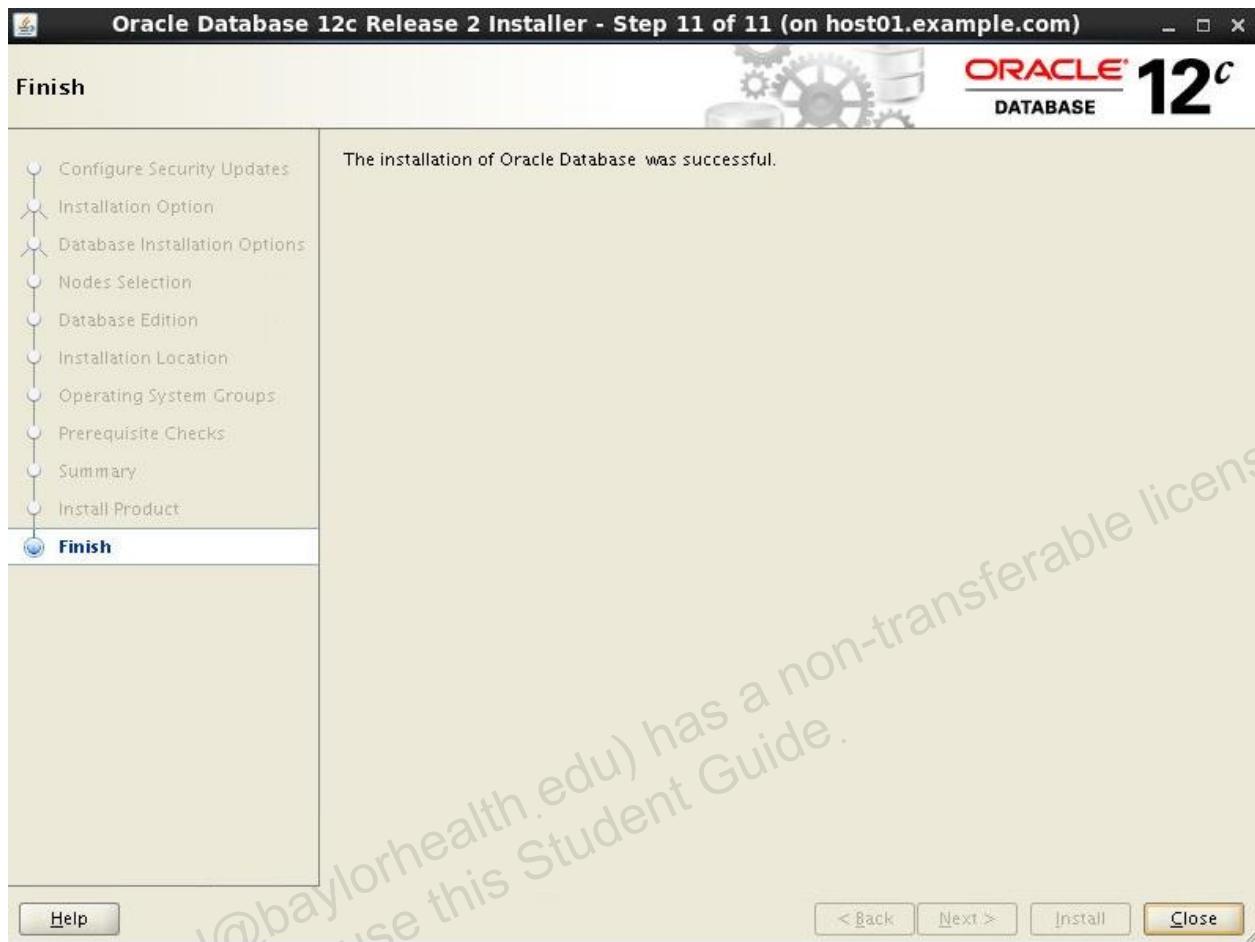
Entries will be added to the /etc/oratab file as needed by
Database Configuration Assistant when a database is created
Finished running generic part of root script.
Now product-specific root actions will be performed.

[root@host01 ~]#
```

16. After the root scripts have been run, return to the installer and click OK on the dialog box.



17. Click Close on the Finish page to dismiss the installer.



18. Close all terminals opened for this practice.

Practice 3-2: Creating a RAC Database

Overview

In this practice, you will create a RAC database in the HUB Nodes.

1. Open a terminal window on your dns node and ssh to host01 as the oracle user. Set the environment for the Oracle database software.

```
[oracle@dns ~]$ ssh -X host01  
oracle@host01's password:  
  
[oracle@host01 ~]$ export  
ORACLE_HOME=/u01/app/oracle/product/12.2.0/dbhome_1  
  
[oracle@host01 ~]$ export PATH=$PATH:$ORACLE_HOME/bin  
  
[oracle@host01 ~]$
```

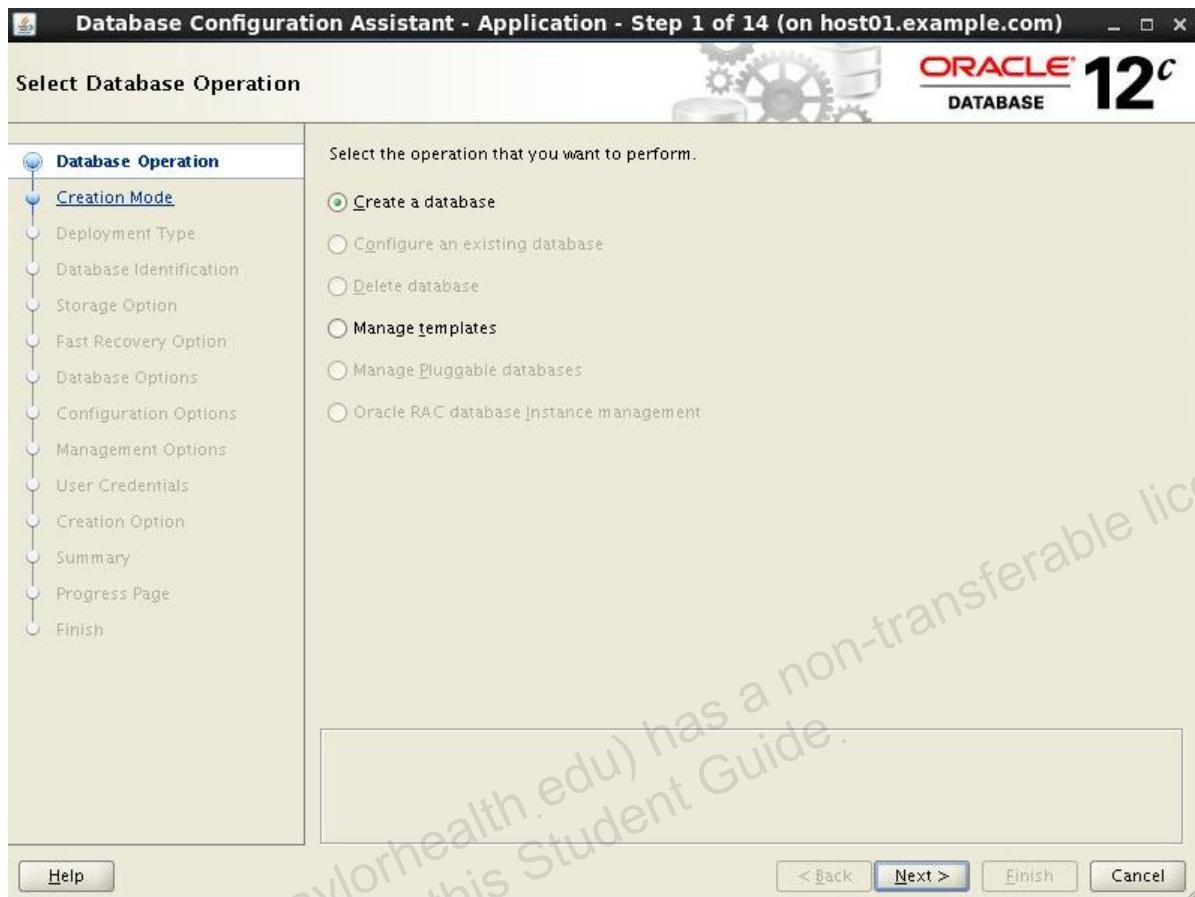
2. Create a server pool called orcldb using srvctl.

```
[oracle@host01 ~]$ srvctl add serverpool -serverpool orcldb -max 3  
-category hub  
  
[oracle@host01 ~]$
```

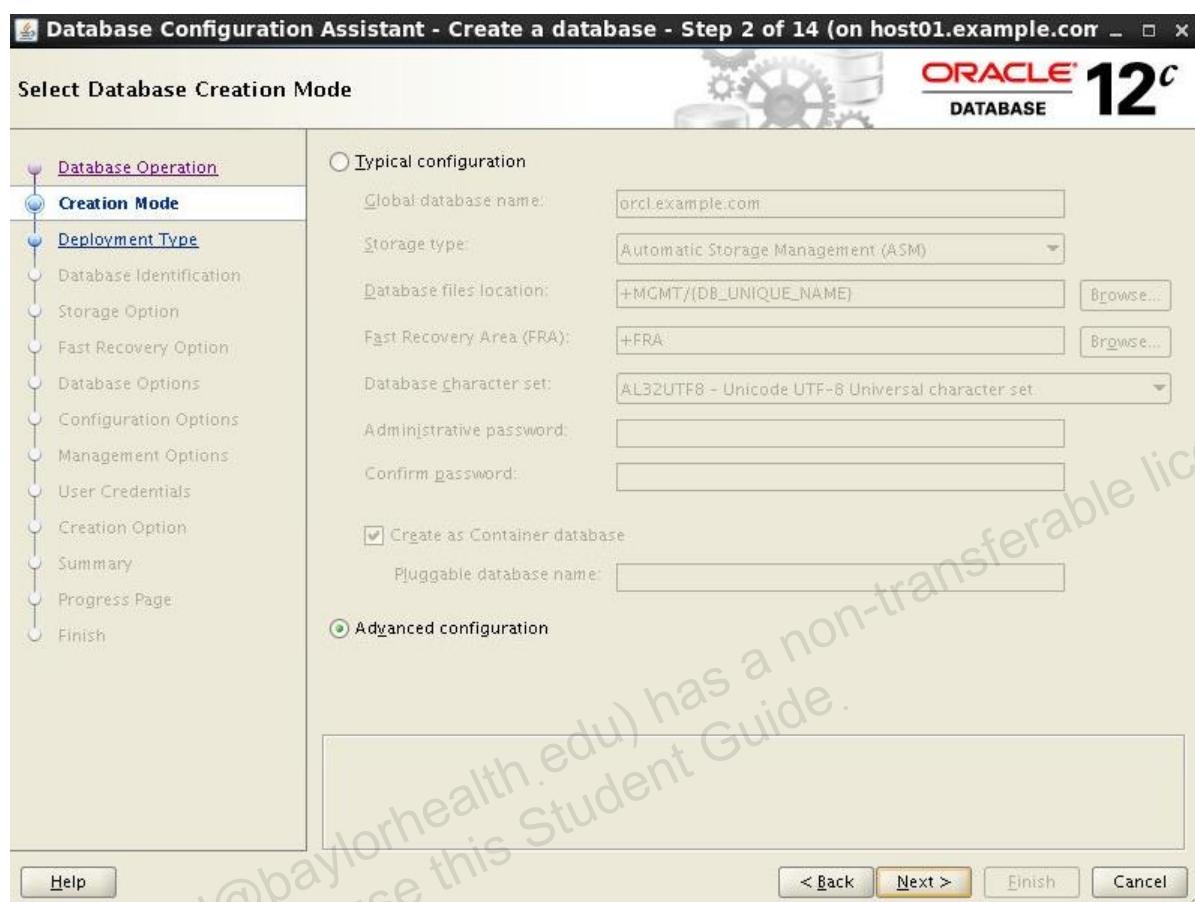
3. Change directory to \$ORACLE_HOME/bin and run DBCA.

```
[oracle@host01 ~]$ dbca
```

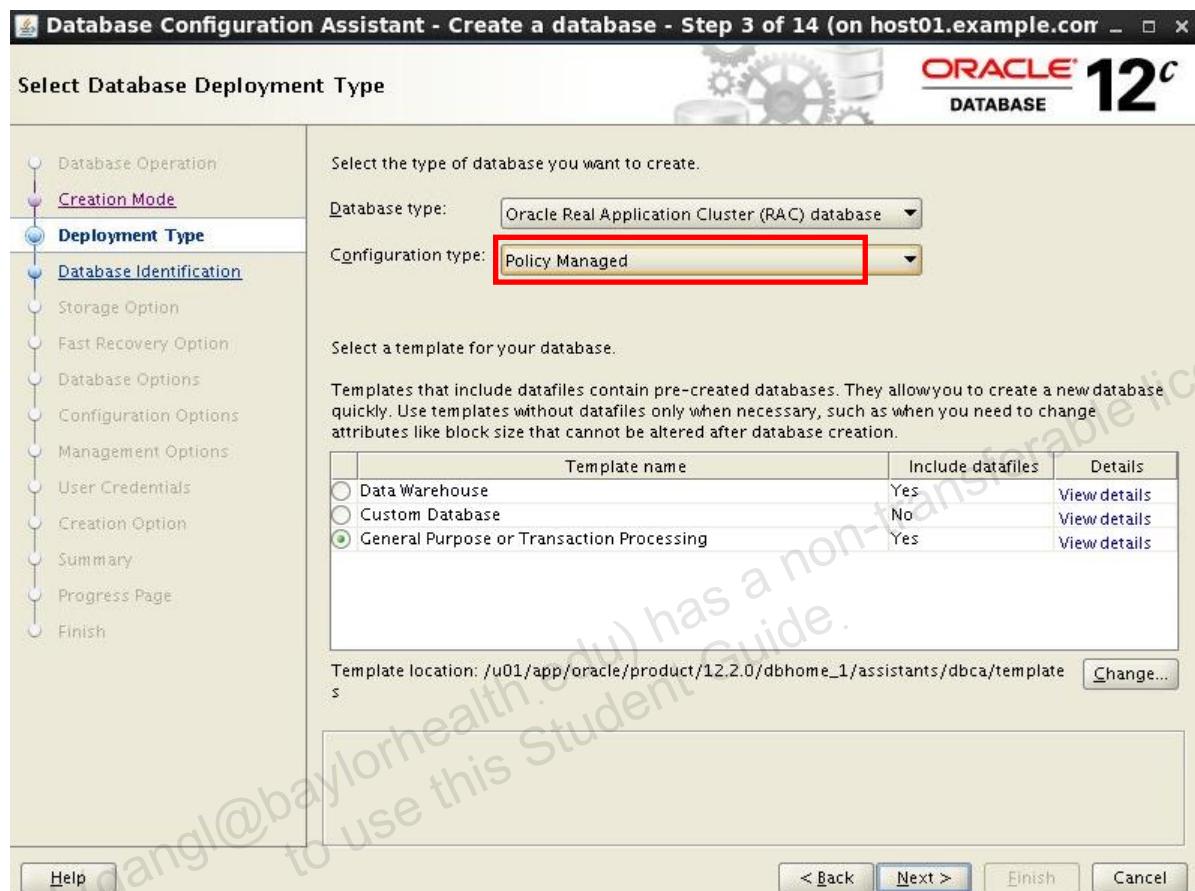
4. On the “Select Database Operation” page, select Create a database and click Next.



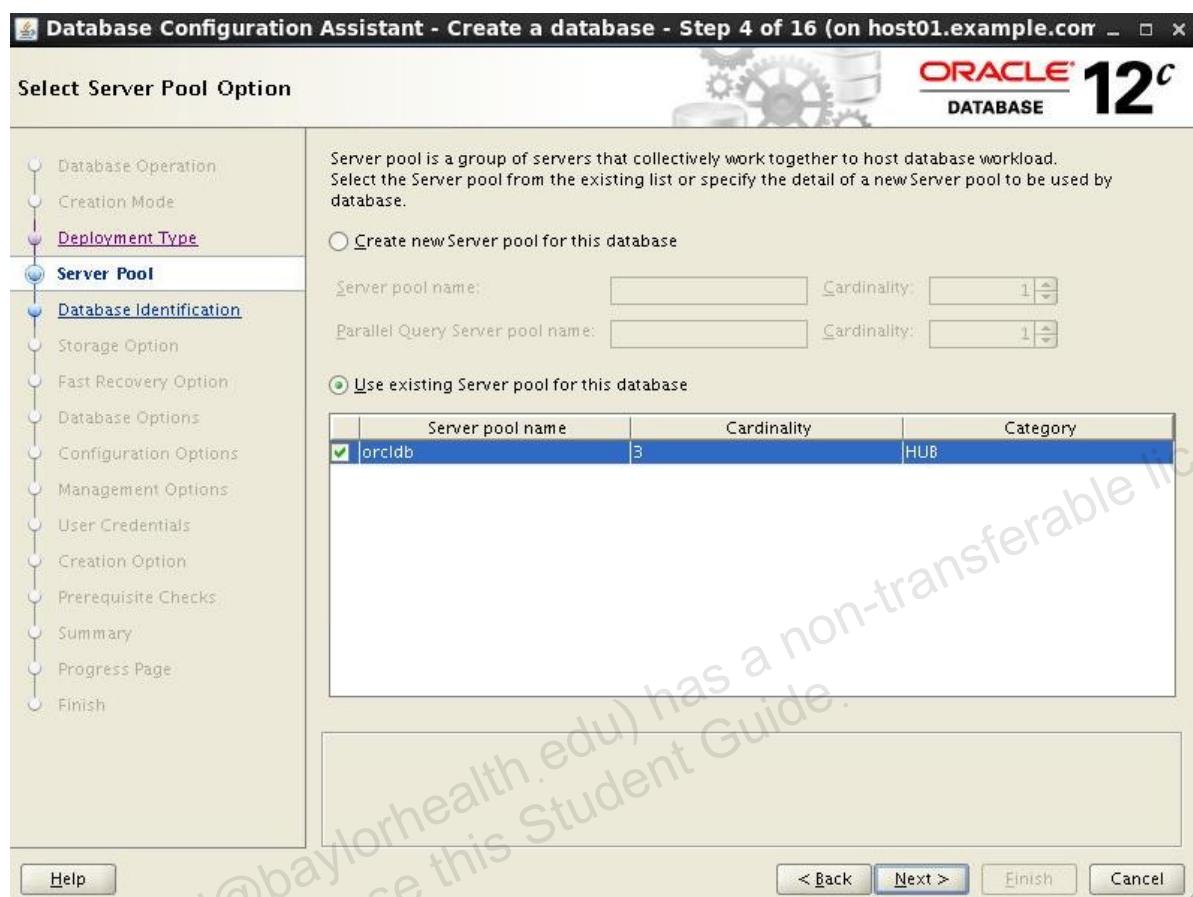
5. On The “Select Database Creation Mode” page, select “Advanced configuration” and click Next.



6. On the “Select Database Deployment Type” page, select “Oracle Real Application Cluster (RAC) database” from the Database type pull-down list and “Policy Managed” from the Configuration type pull-down list. Select the “General Purpose or Transaction Processing” template and click Next.



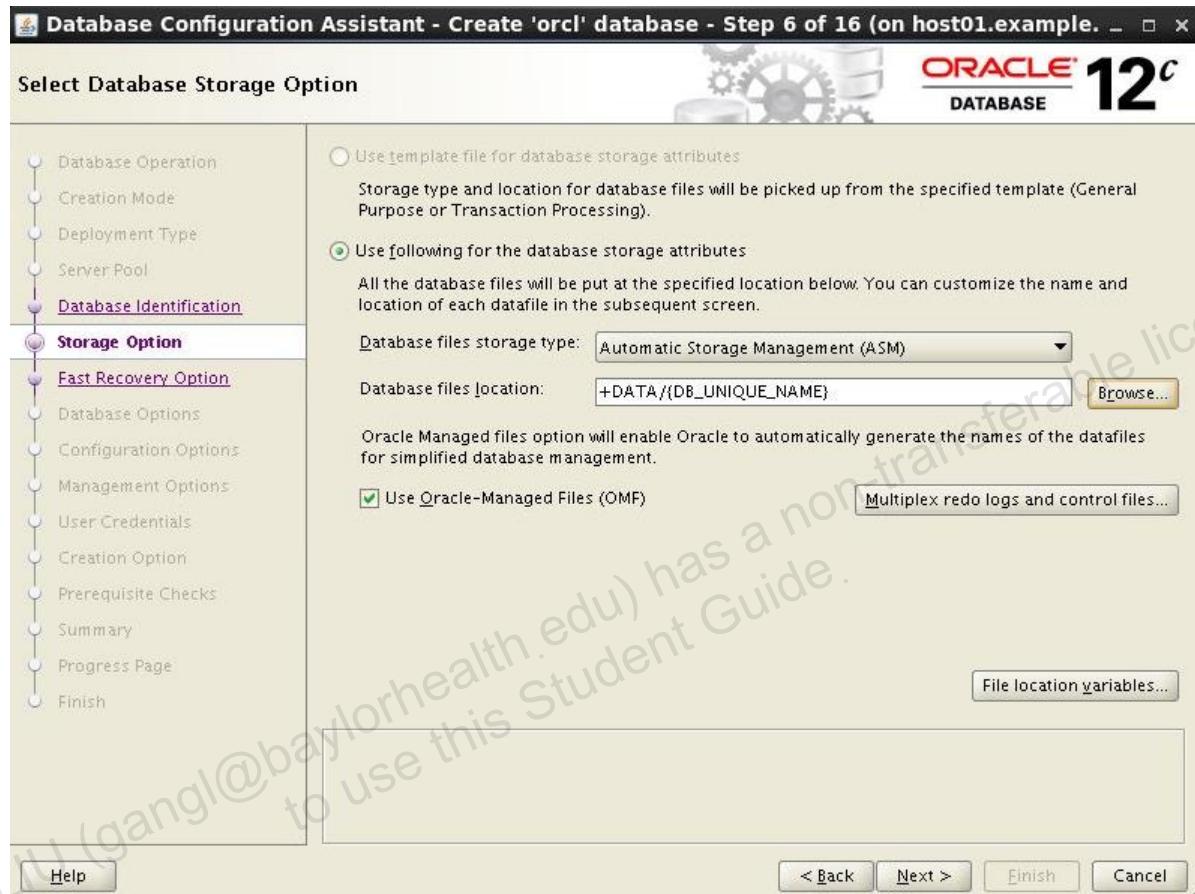
7. On the Select Server Pool Option page, click the “Use existing Server pool for this database” button. Select the `orcldb` Server pool and click Next.



8. Ensure the Global database name is `orcl`. Do **NOT** accept the default database name. Make sure the Create as Container database check box is **NOT** checked. Click Next to continue.



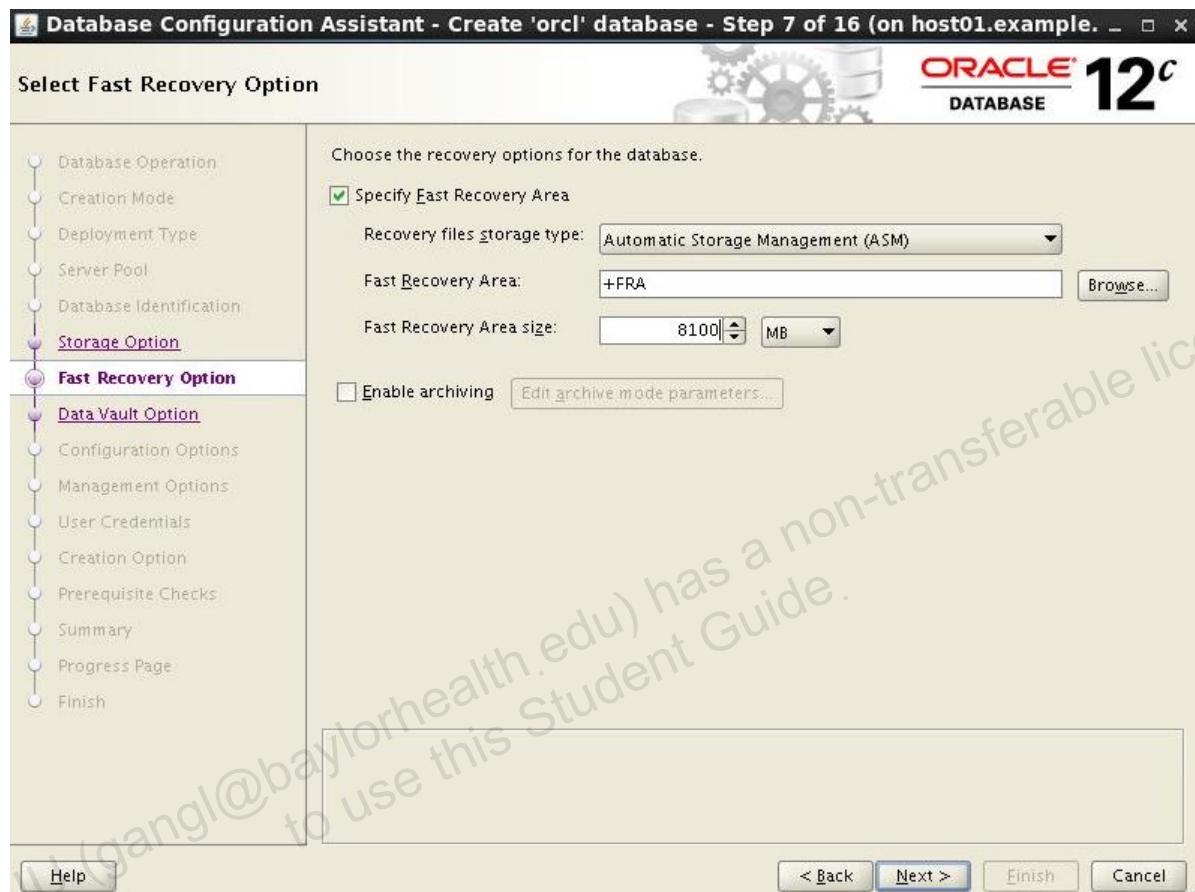
9. On the Select Database Storage Option page, click the “Use following for the database storage attributes” button. Make sure “Automatic Storage Management (ASM)” is selected from the Database files storage type pull down list and the Database files location is **+DATA/{DB_UNIQUE_NAME}**. Make sure the “Use Oracle-Managed Files (OMF)” check box is selected and click Next.



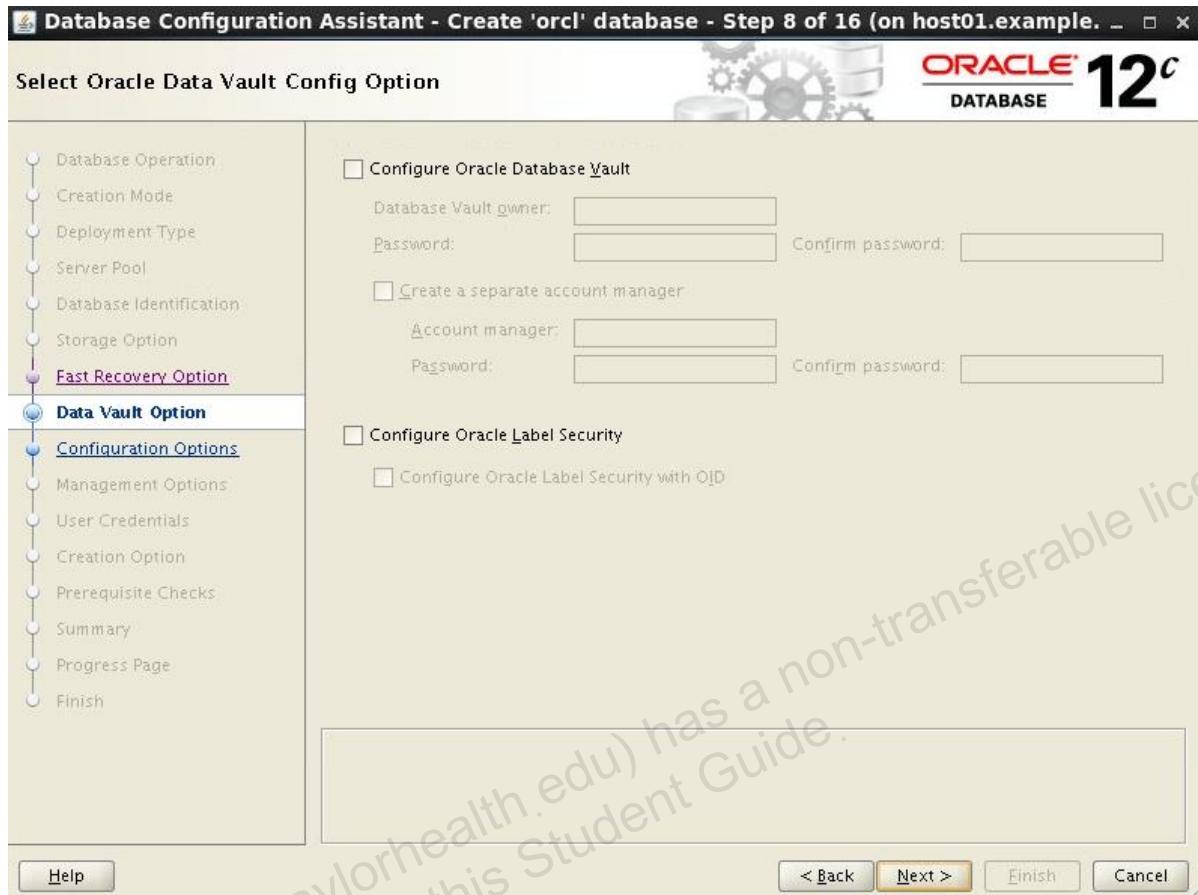
10. On the Select Fast Recovery Option, make the following adjustments:

- Fast Recovery Area: +FRA
- Fast Recovery Area Size: 8100

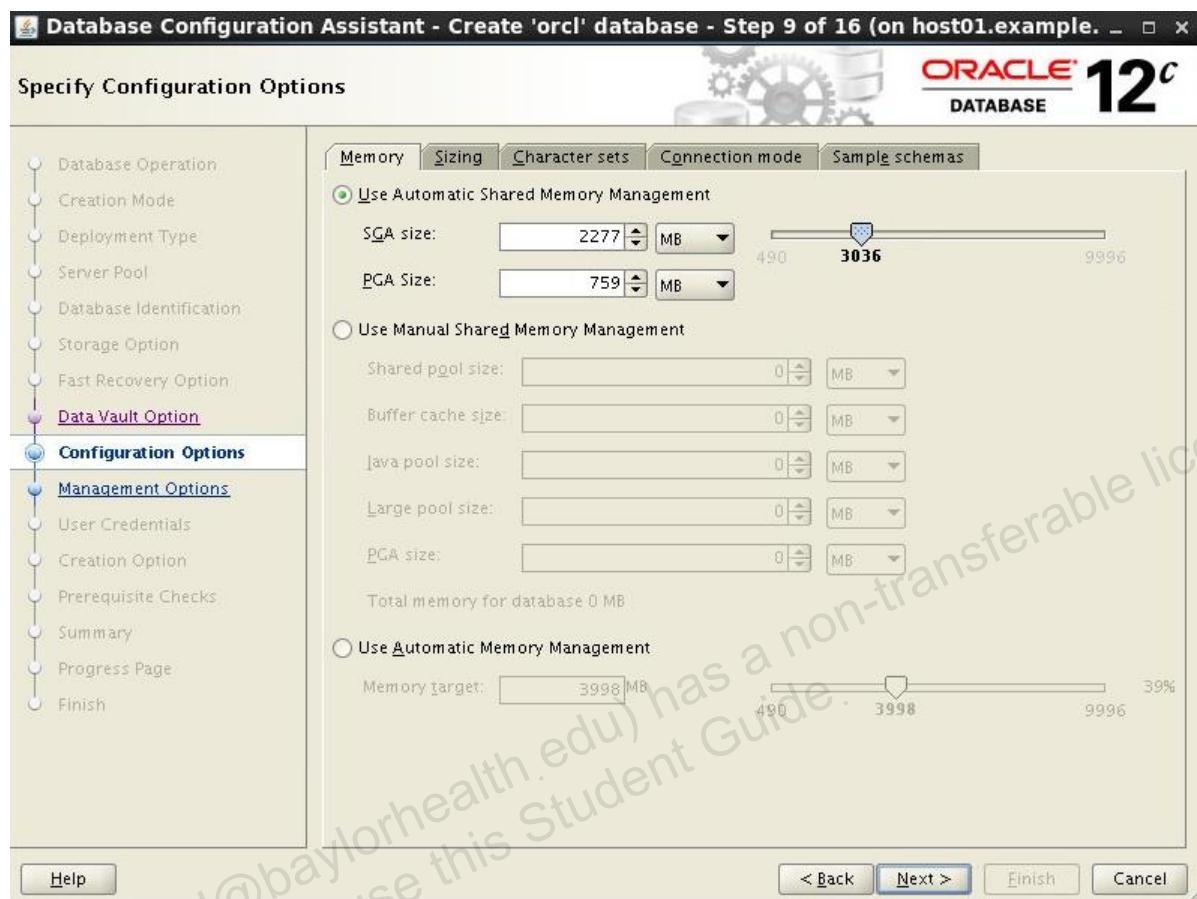
Click Next to continue.



11. On the Select Oracle Data Vault Config Option page, click Next.



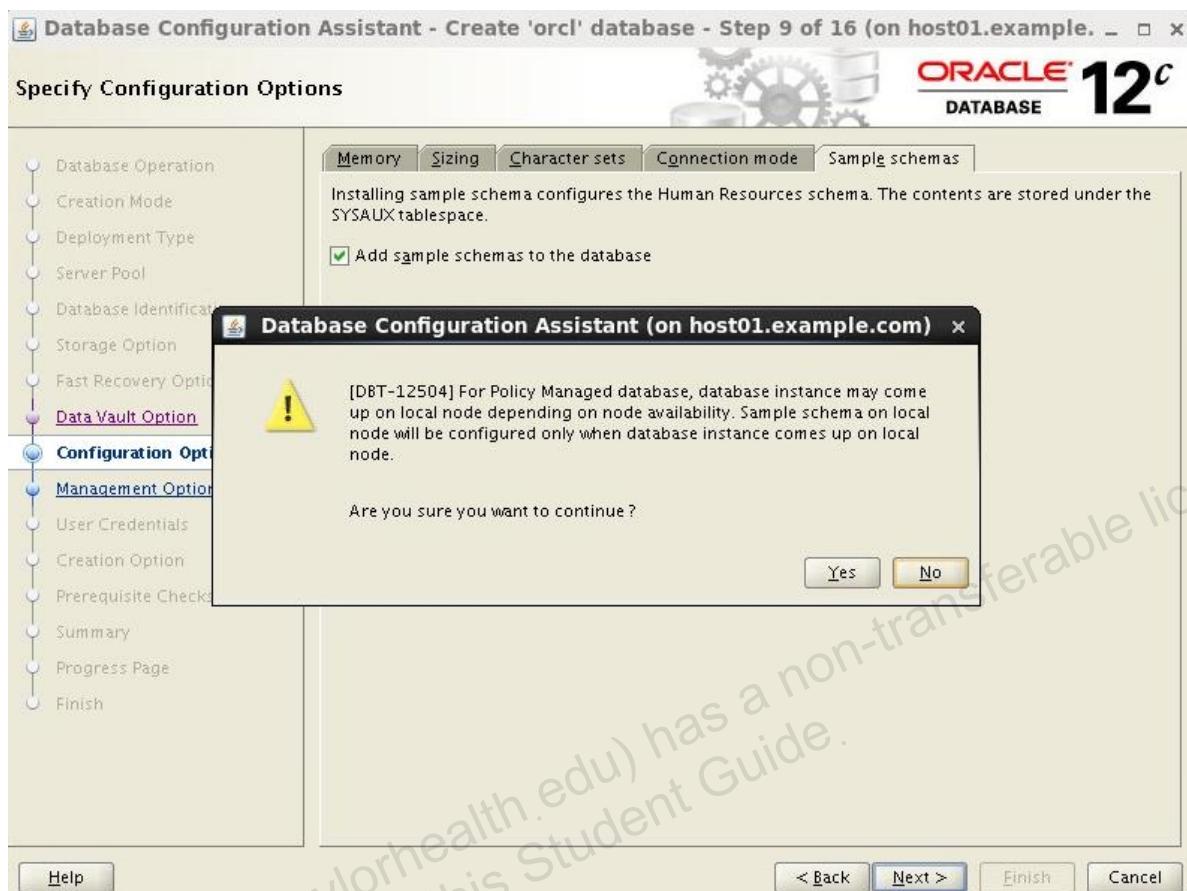
12. On the Specify Configuration Options page, set Memory Size (SGA and PGA) to around 3000. Click the Sample Schemas folder tab.



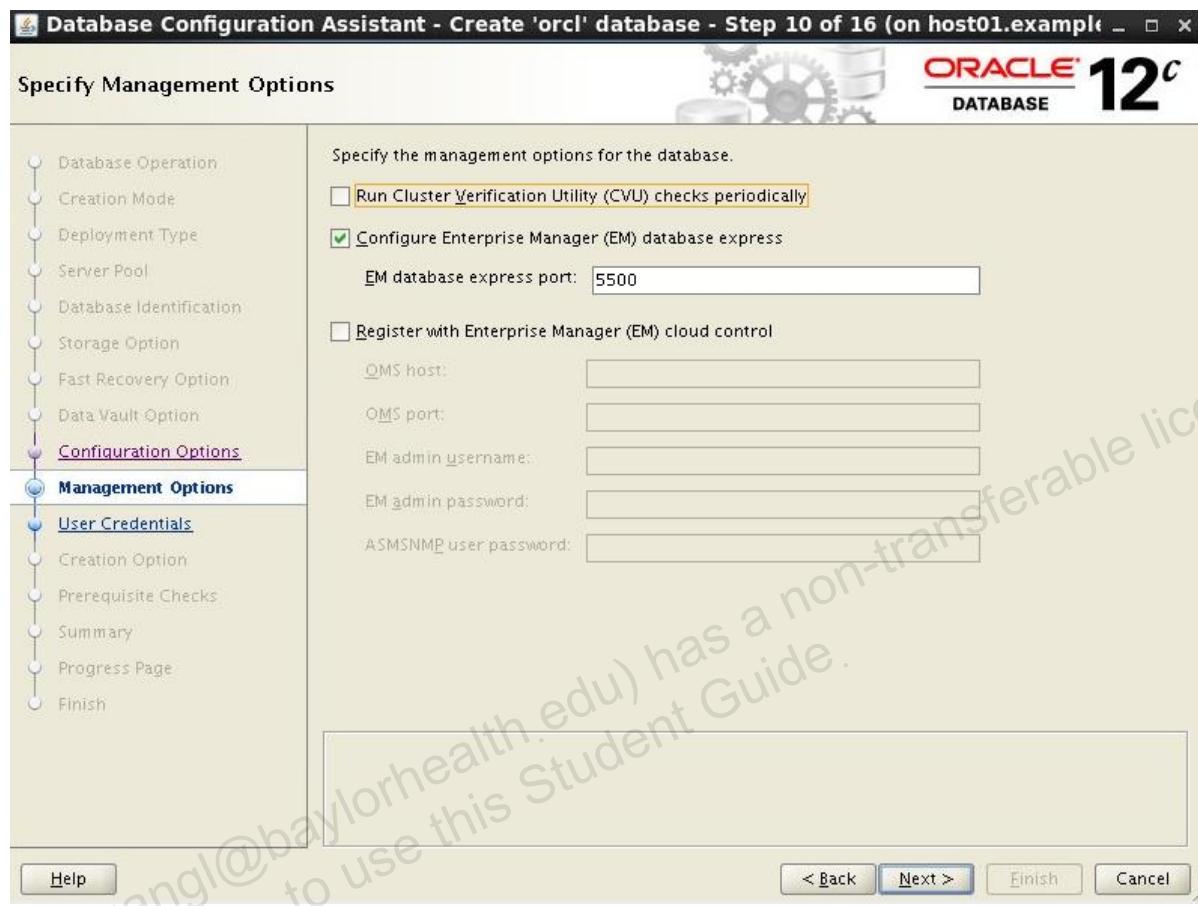
13. Click the “Add sample schemas to the database” check box and click Next.



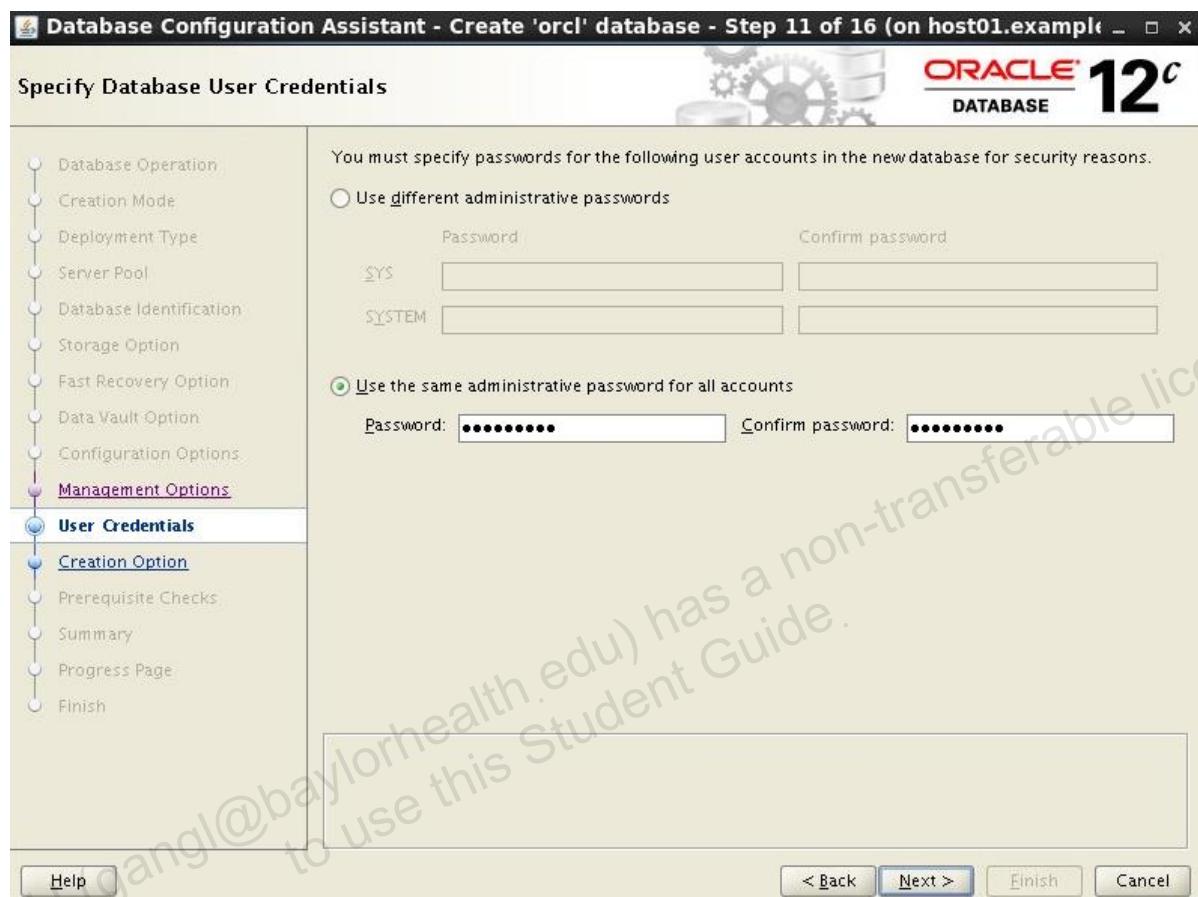
14. Click Yes to dismiss the dialog box.



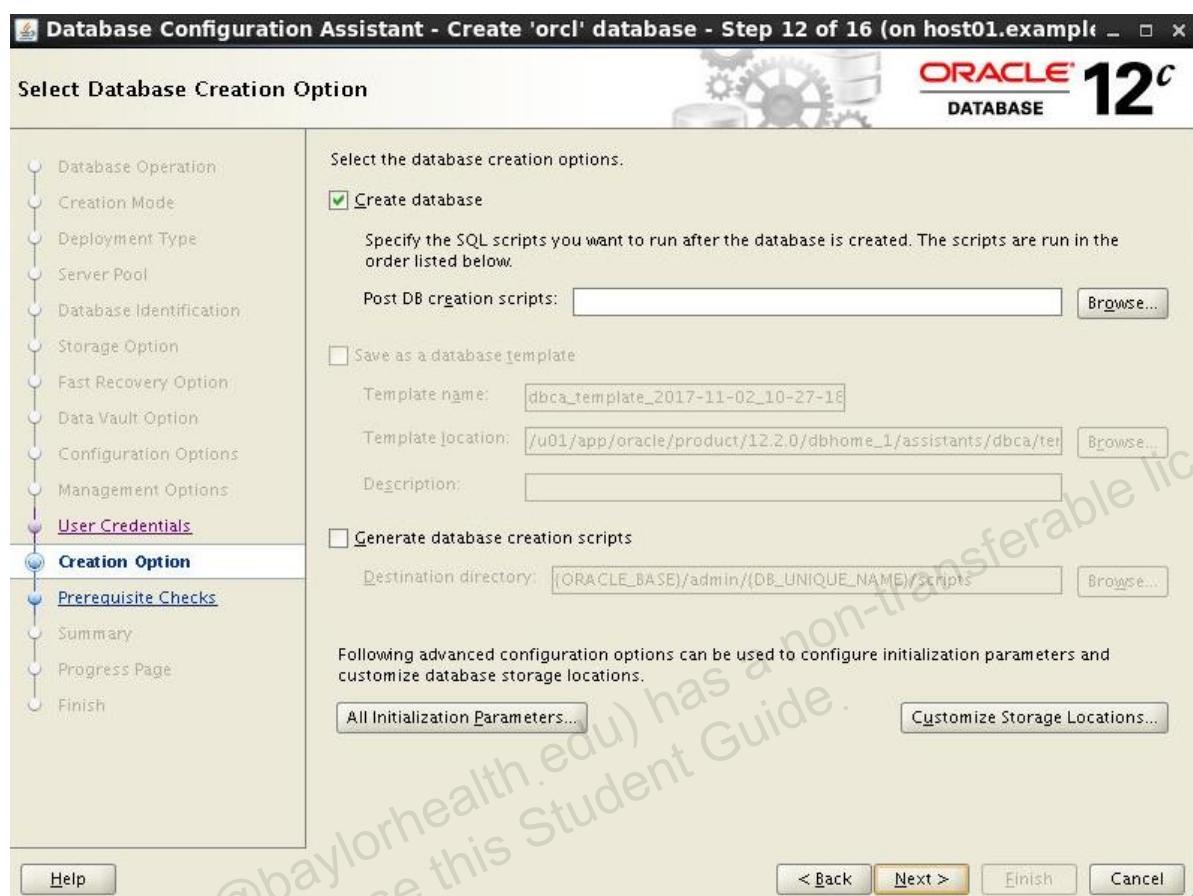
15. On the Management Options screen, uncheck Run Cluster Verification Utility (CVU) Checks Periodically. Ensure Configure EM Database Express is **checked** and accept the default port value of 5500. Click next to continue.



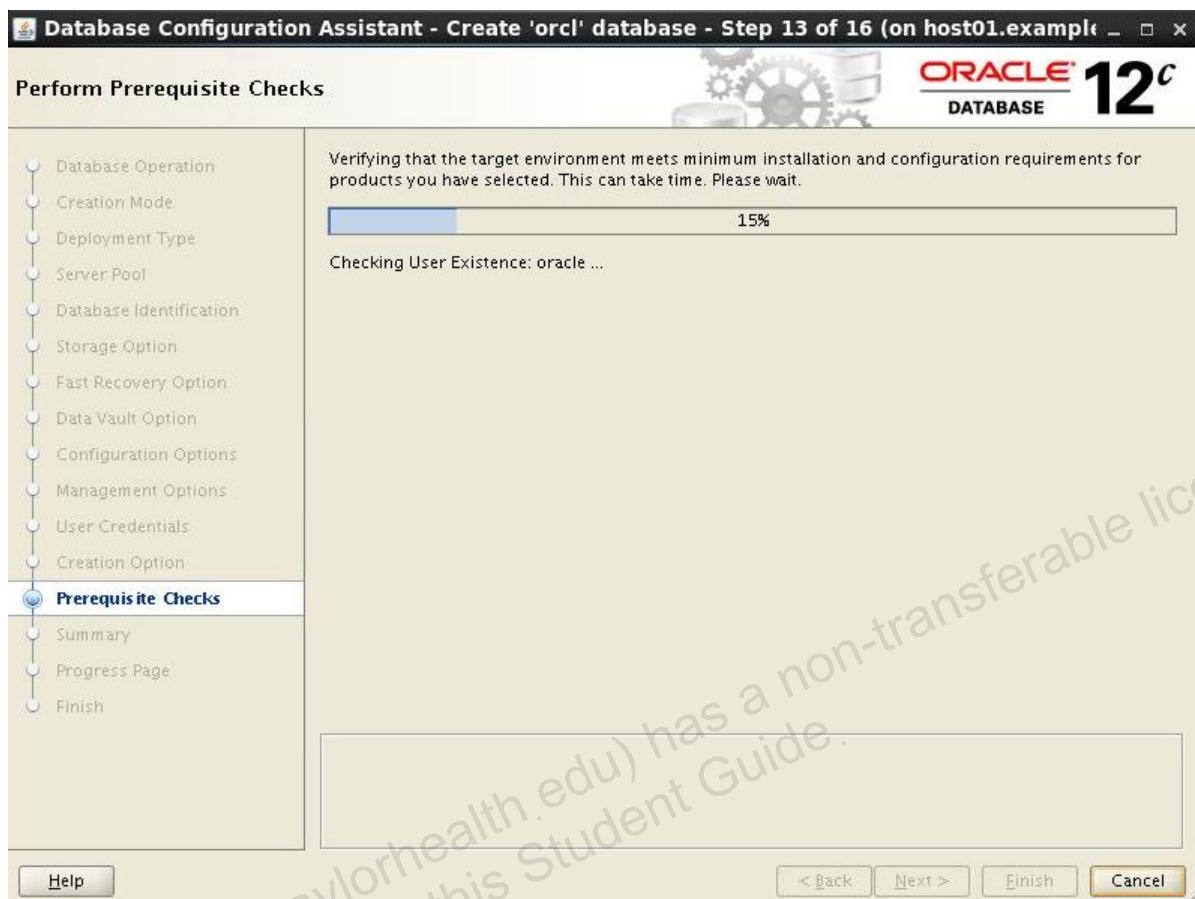
16. On the Specify database User Credentials page, click the “Use the same administrative password for all accounts” button. Enter the `sys` password (*please refer to the Course Practice Environment: Security Credentials page for account passwords*) in both the Password and Confirm Password fields and click Next.



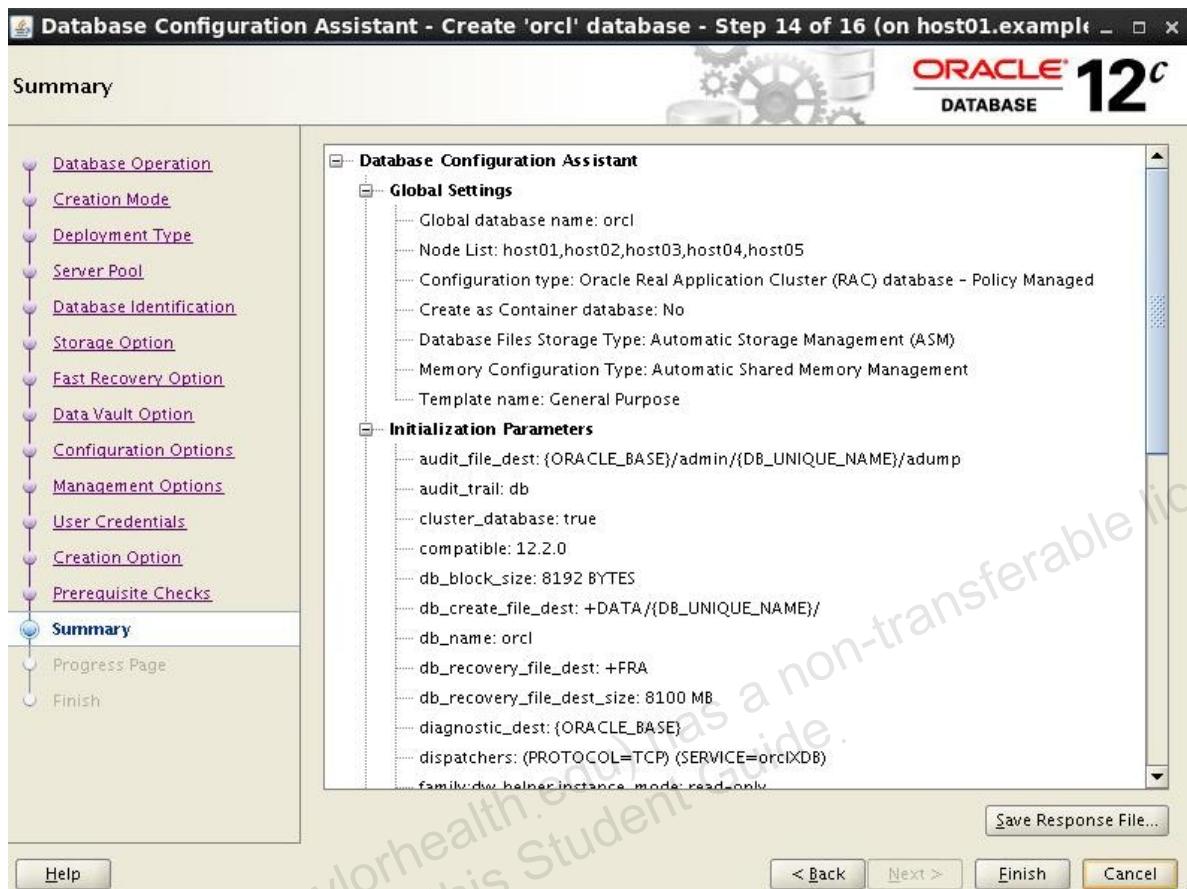
17. On the Select Database Creation Options page, make sure the “Create database” check box is selected and click Next.



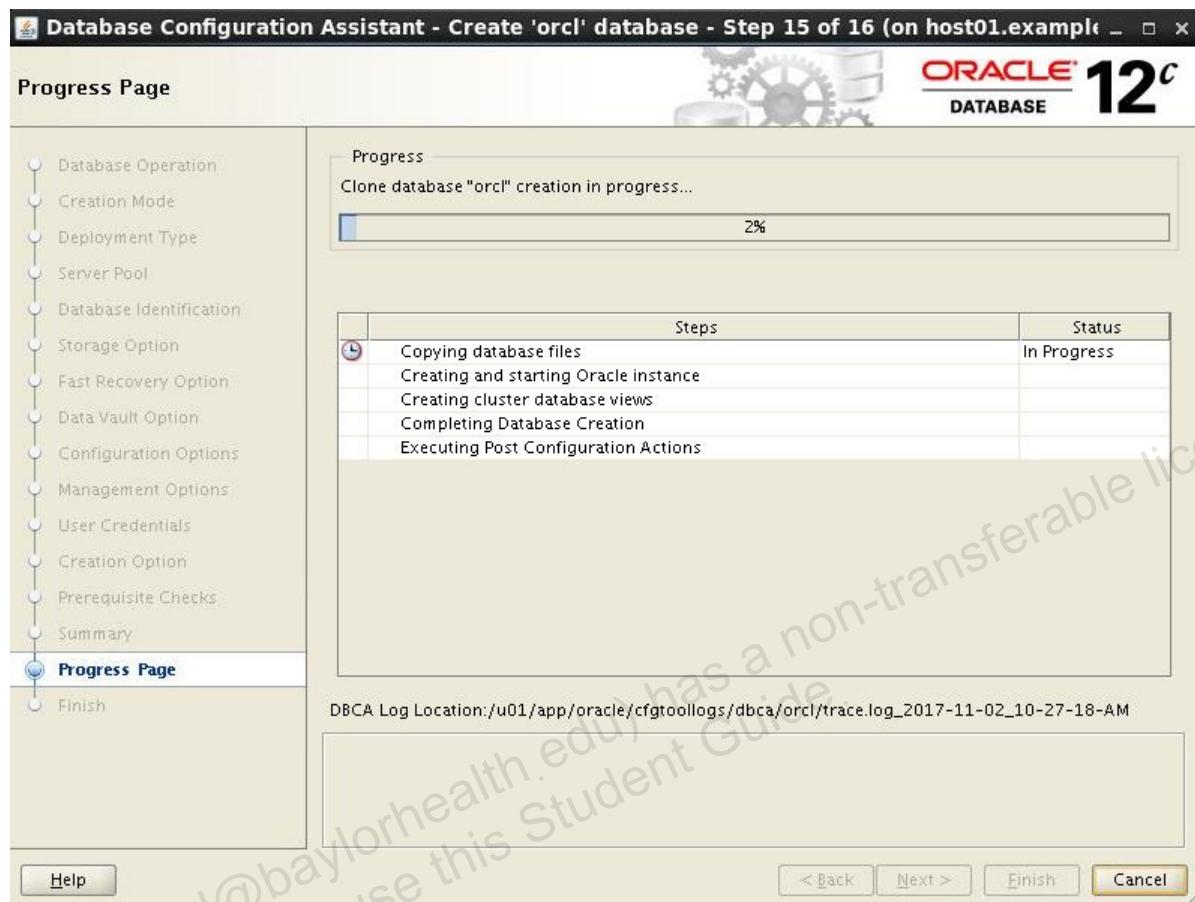
18. Wait while a series of prerequisite checks are performed.



19. On the Summary page, review the information and click Finish.



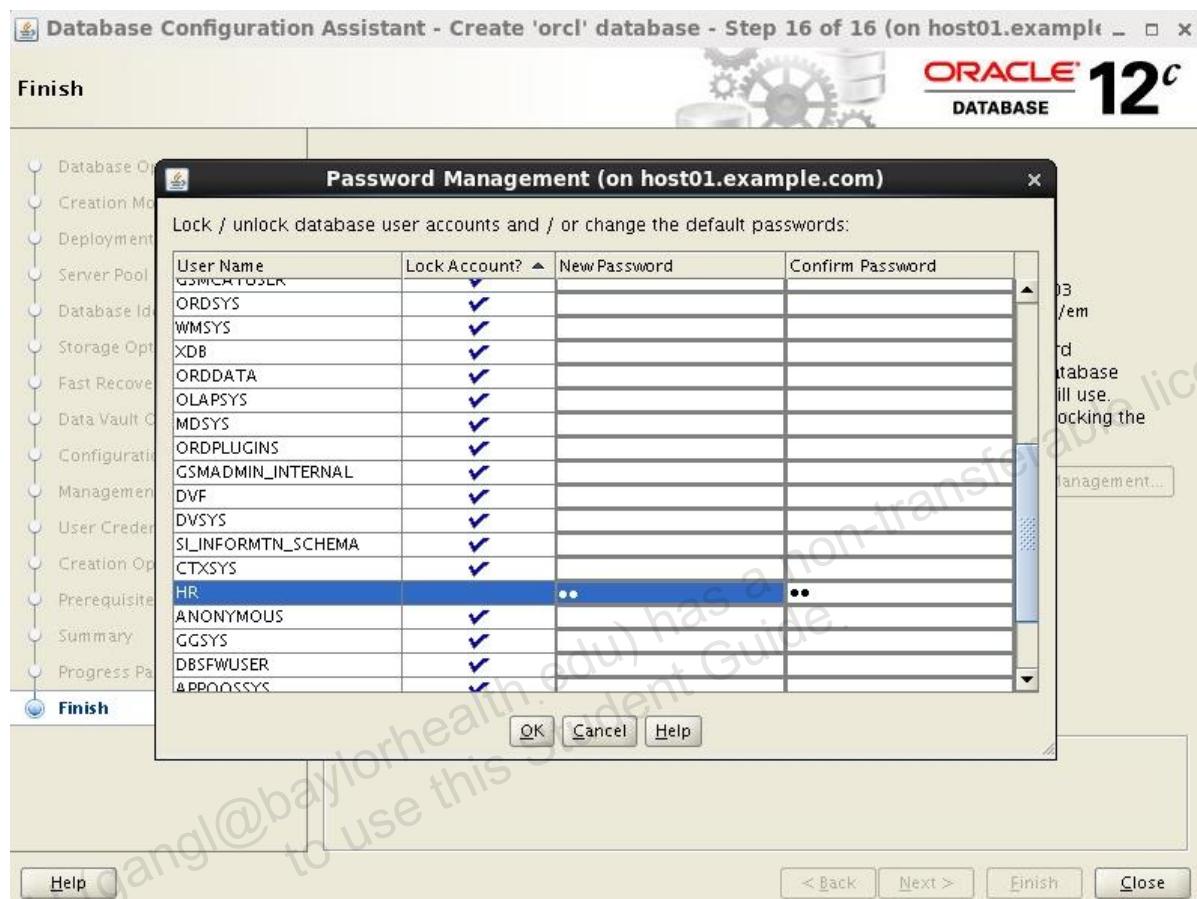
20. Wait a few moments and follow the database creation process on the Progress Page. *It takes about 20min to complete the rest of steps (step 19 ~ step 25).*



21. On the Finish page, click the “Password Management” button.



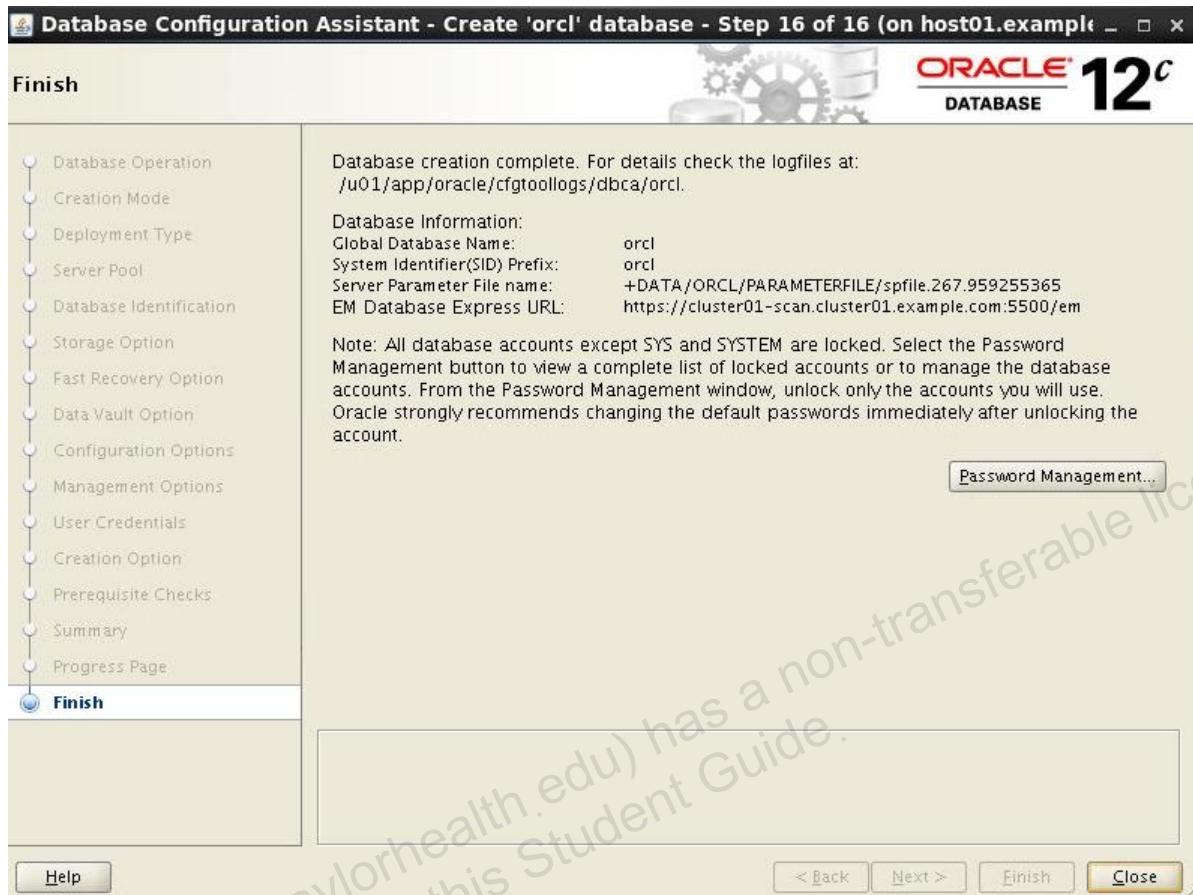
22. Select the HR account and enter the HR password (*please refer to the Course Practice Environment: Security Credentials page for account passwords*) in the New Password and Confirm Password fields, click the Lock Account field to unlock the account and click OK.



23. Click Yes to dismiss the dialog box.



24. Click Close to dismiss DBCA.



25. Return to the oracle terminal and check the database status and configuration details with SRVCTL.

```
[oracle@host01 ~]$ srvctl status database -db orcl
Instance orcl_1 is running on node host02
Instance orcl_2 is running on node host03
Instance orcl_3 is running on node host01

[oracle@host01 ~]$ srvctl config database -db orcl
Database unique name: orcl
Database name: orcl
Oracle home: /u01/app/oracle/product/12.2.0/dbhome_1
Oracle user: oracle
Spfile: +DATA/ORCL/PARAMETERFILE/spfile.297.958993403
Password file: +DATA/ORCL/PASSWORD/pwdorcl.284.958993065
Domain:
Start options: open
Stop options: immediate
Database role: PRIMARY
```

```
Management policy: AUTOMATIC
Server pools: orcldb
Disk Groups: FRA,DATA
Mount point paths:
Services:
Type: RAC
Start concurrency:
Stop concurrency:
OSDBA group: dba
OSOPER group: oper
Database instances:
Configured nodes:
CSS critical: no
CPU count: 0
Memory target: 0
Maximum memory: 0
Default network number for database services:
Database is policy managed

[oracle@host01 ~]$
```

26. Close all terminals opened for this practice.

Practices for Lesson 4: Oracle RAC Administration

Practices for Lesson 4: Overview

Overview

In this practice, you will make the local and remote connections to contrast operating system and password file authentication.

In this practice, you will make multiple Oracle database authenticated connections to a database instance and notice the effects of load-balanced connections.

In this practice, you will use the `srvctl` utility to stop all resource components executing from a single Oracle home location.

In this practice, you will configure leaf nodes to run read-only instances.

In this practice, you will work with local temporary tablespaces on read-write and read-only instances.

Practice 4-1: Operating System and Password File Authenticated Connections

Overview

In this practice, you will make the local and remote connections to contrast operating system and password file authentication.

1. Connect to your first node as the `oracle` user and update the `/etc/oratab` file.

```
[oracle@dns ~]$ ssh oracle@host01
oracle@host01's password:

[oracle@host01 ~]$ vi /etc/oratab

***** Add the orcl entry at the end *****
...
orcl:/u01/app/oracle/product/12.2.0/dbhome_1:N

:wq!

[oracle@host01 ~]$
```

2. Set up your environment variables by using the `oraenv` script.

```
[oracle@host01 ~]$ . oraenv
ORACLE_SID = [oracle] ? orcl
The Oracle base remains unchanged with value /u01/app/oracle

[oracle@host01 ~]$
```

3. Identify all the database instance names that are currently executing on your machine by using the Linux `ps` command.

Note: All database instances have a mandatory background process named pmon, and the instance name will be part of the complete process name.

```
[oracle@host01 ~]$ ps -ef | grep -i pmon
grid      13193      1  0 Nov01 ?          00:00:07 mdb_pmon_-MGMTDB
oracle    21316  17541  0 12:36 pts/5      00:00:00 grep -i pmon
oracle    25617      1  0 11:04 ?          00:00:00 ora_pmon_orcl_3
grid      31287      1  0 Nov01 ?          00:00:07 asm_pmon_+ASM1

[oracle@host01 ~]$
```

4. Attempt to make a local connection to the `orcl_n` instance by using SQL*Plus with the `sysdba` privilege. This is known as operating system authentication because a password is not needed. What happens when you are trying to connect to the instance?

```
[oracle@host01 ~]$ sqlplus / as sysdba

SQL*Plus: Release 12.2.0.1.0 Production on Thu Nov 2 12:40:20
2017

Copyright (c) 1982, 2016, Oracle. All rights reserved.

Connected to an idle instance.

SQL> exit
Disconnected
[oracle@host01 ~]$
```

5. Attempt to connect to the instance by using a network connection string `@orcl` with the `sysdba` privilege. This is known as password file authentication. Is the connection successful this time?

```
[oracle@host01 ~]$ sqlplus sys@orcl as sysdba

SQL*Plus: Release 12.2.0.1.0 Production on Thu Nov 2 12:43:09
2017

Copyright (c) 1982, 2016, Oracle. All rights reserved.

Enter password: **** << Password is not displayed

Connected to:
Oracle Database 12c Enterprise Edition Release 12.2.0.1.0 -
64bit Production

SQL> exit
Disconnected from Oracle Database 12c Enterprise Edition Release
12.2.0.1.0 - 64bit Production
[oracle@host01 ~]$
```

6. Display the values of the environment variables (`ORACLE_BASE`, `ORACLE_HOME`, `ORACLE_SID`, and so on) that were defined with the `oraenv` script in step 2.

```
[oracle@host01 ~]$ env | grep ORA
ORACLE_SID=orcl
ORACLE_BASE=/u01/app/oracle
```

```
ORACLE_HOME=/u01/app/oracle/product/12.2.0/dbhome_1  
[oracle@host01 ~]$
```

7. Modify the `ORACLE_SID` environment variable to match the actual database instance name for the `orcl` database.

```
[oracle@host01 ~]$ export ORACLE_SID=orcl_3  
[oracle@host01 ~]$
```

8. Attempt the local connection with system authentication to the local instance by using SQL*Plus with the `sysdba` privilege. This is the same command as in step 4.

```
[oracle@host01 ~]$ sqlplus / as sysdba  
  
SQL*Plus: Release 12.2.0.1.0 Production on Thu Nov 2 12:46:41  
2017  
  
Copyright (c) 1982, 2016, Oracle. All rights reserved.  
  
Connected to:  
Oracle Database 12c Enterprise Edition Release 12.2.0.1.0 -  
64bit Production  
  
SQL>
```

9. Query the `instance_name` column of the `v$instance` dynamic performance view to validate the instance that you connected with. Exit SQL*Plus when finished.

```
SQL> select instance_name from v$instance;  
  
INSTANCE_NAME  
-----  
orcl_3  
  
SQL> exit  
Disconnected from Oracle Database 12c Enterprise Edition Release  
12.2.0.1.0 - 64bit Production  
[oracle@host01 ~]$
```

Practice 4-2: Oracle Database Authenticated Connections

Overview

In this practice, you will make multiple Oracle database authenticated connections to a database instance and notice the effects of load-balanced connections.

- From your first node, connected as the `oracle` user, validate the instance names on each host.

```
[oracle@host01 ~]$ srvctl status database -db orcl
Instance orcl_1 is running on node host02
Instance orcl_2 is running on node host03
Instance orcl_3 is running on node host01

[oracle@host01 ~]$
```

- Connect to a database instance by using SQL*Plus with the `system` account. This is known as Oracle database authentication. After it is connected, query the `instance_name` column from the `v$instance` dynamic performance view.

Note: Your instance names may vary from the ones displayed below:

```
[oracle@host01 ~]$ sqlplus system@orcl

SQL*Plus: Release 12.2.0.1.0 Production on Thu Nov 2 12:53:59
2017

Copyright (c) 1982, 2016, Oracle. All rights reserved.

Enter password: ***** << Password is not displayed

Connected to:
Oracle Database 12c Enterprise Edition Release 12.2.0.1.0 -
64bit Production

SQL> select instance_name from v$instance;

INSTANCE_NAME
-----
orcl_2

SQL>
```

3. Use the SQL*Plus host command to temporarily exit SQL*Plus and return to the operating system prompt.

Note: SQL*Plus is still running when this is performed. Repeat the previous step from the operating system prompt to establish a second SQL*Plus session and database instance connection. What instance name did you connect to?

```
SQL> host
[oracle@host01 ~]$ sqlplus system@orcl

SQL*Plus: Release 12.2.0.1.0 Production on Thu Nov 2 12:53:59
2017

Copyright (c) 1982, 2016, Oracle. All rights reserved.

Enter password: ***** << Password is not displayed

Connected to:
Oracle Database 12c Enterprise Edition Release 12.2.0.1.0 -
64bit Production

SQL> select instance_name from v$instance;

INSTANCE_NAME
-----
orcl_1

SQL>
```

4. Use the SQL*Plus host command to temporarily exit SQL*Plus and return to the operating system prompt.

Note: SQL*Plus is still running when this is performed. Validate that you are still on your first node. Repeat the previous step from the operating system prompt to establish a third SQL*Plus session and database instance connection. What instance name did you connect to?

```
SQL> host
[oracle@host01 ~]$ sqlplus system@orcl

SQL*Plus: Release 12.2.0.1.0 Production on Thu Nov 2 12:53:59
2017

Copyright (c) 1982, 2016, Oracle. All rights reserved.

Enter password: ***** << Password is not displayed

Connected to:
```

```
Oracle Database 12c Enterprise Edition Release 12.2.0.1.0 -  
64bit Production  
  
SQL> select instance_name from v$instance;  
  
INSTANCE_NAME  
-----  
orcl_3  
  
SQL>
```

5. Exit the three SQL*Plus sessions that are currently executing on the first node.

```
SQL> exit  
Disconnected from Oracle Database 12c Enterprise Edition Release  
12.2.0.1.0 - 64bit Production  
  
[oracle@host01 ~]$ exit  
exit  
  
SQL> exit  
Disconnected from Oracle Database 12c Enterprise Edition Release  
12.2.0.1.0 - 64bit Production  
  
[oracle@host01 ~]$ exit  
exit  
  
SQL> exit  
Disconnected from Oracle Database 12c Enterprise Edition Release  
12.2.0.1.0 - 64bit Production  
[oracle@host01 ~]$
```

Practice 4-3: Stopping a Complete ORACLE_HOME Component Stack

Overview

In this practice, you will use the `srvctl` utility to stop all resource components executing from a single Oracle home location.

1. Validate that the instances are running on each node of the cluster using the `ps` command.

Note: The instance locations (-MGMTDB, orcl_<n>, and +ASM<n>) may be different.

```
[oracle@host01 ~]$ ps -ef|grep -i pmon
grid      13193      1  0 Nov01 ?          00:00:07 mdb_pmon_-MGMTDB
oracle    25617      1  0 11:04 ?          00:00:00 ora_pmon_orcl_3
oracle    27849 17541  0 12:59 pts/5      00:00:00 grep -i pmon
grid      31287      1  0 Nov01 ?          00:00:07 asm_pmon_+ASM1

[oracle@host01 ~]$ ssh host02 ps -ef|grep -i pmon
oracle    3335      1  0 11:04 ?          00:00:00 ora_pmon_orcl_1
grid      18628      1  0 Nov01 ?          00:00:07 asm_pmon_+ASM3

[oracle@host01 ~]$ ssh host03 ps -ef|grep -i pmon
grid      9786      1  0 Nov01 ?          00:00:07 asm_pmon_+ASM2
oracle   25000      1  0 11:04 ?          00:00:00 ora_pmon_orcl_2

[oracle@host01 ~]$
```

2. Display the syntax usage help for the `srvctl status home` command.

```
[oracle@host01 ~]$ srvctl status home -help

Displays the current state of all resources for the Oracle
home.

Usage: srvctl status home -oraclehome <oracle_home> -statefile
<state_file> -node <node_name>
      -oraclehome <path>           Oracle home path
      -statefile <state_file>       Specify a file path for the
                                     srvctl status home command to store the state of the resources
      -node <node_name>            Node name
      -help                         Print usage

[oracle@host01 ~]$
```

3. Use the `srvctl status home` command to check the state of all resources running from the `/u01/app/oracle/product/12.2.0/dbhome_1` home location. Create the required state file in the `/tmp` directory with the file name `host01_dbhome_state1.dmp` for the first node only.

```
[oracle@host01 ~]$ srvctl status home -oraclehome
/u01/app/oracle/product/12.2.0/dbhome_1 -statefile
/tmp/host01_dbhome_state1.dmp -node host01
Database orcl is running on node host01

[oracle@host01 ~]$
```

4. Display the syntax usage help for the `srvctl stop home` command.

```
[oracle@host01 ~]$ srvctl stop home -help

Stops all Oracle clusterware resources that run from the Oracle
home.

Usage: srvctl stop home -oraclehome <oracle_home> -statefile
<state_file> -node <node_name> [-stopoption <stop_options>] [-
force]
      -oraclehome <path>          Oracle home path
      -statefile <state_file>       Specify a file path for the
srvctl stop home command to store the state of the resources
      -node <node_name>           Node name
      -stopoption <stop_options>    Stop options for the
database. Examples of shutdown options are NORMAL,
TRANSACTIONAL, IMMEDIATE, or ABORT.
      -force                      Force stop
      -help                       Print usage

[oracle@host01 ~]$
```

5. Stop all resources executing from `/u01/app/oracle/product/12.2.0/dbhome_1`. Do not use the optional parameters identified by square brackets “`[]`” displayed in the syntax usage help.

```
[oracle@host01 ~]$ srvctl stop home -oraclehome
/u01/app/oracle/product/12.2.0/dbhome_1 -node host01 -statefile
/tmp/host01_dbhome_state2.dmp

[oracle@host01 ~]$
```

6. Check the status of the database instances on each node. .

```
[oracle@host01 ~]$ srvctl status database -db orcl
Instance orcl_1 is running on node host02
Instance orcl_2 is running on node host03
Instance orcl_3 is not running on node host01
[oracle@host01 ~]$
```

7. Start all resources for the /u01/app/oracle/product/12.2.0/dbhome_1 home using the state file created by the stop command.

```
[oracle@host01 ~]$ srvctl start home -oraclehome
/u01/app/oracle/product/12.2.0/dbhome_1 -node host01 -statefile
/tmp/host01_dbhome_state2.dmp

[oracle@host01 ~]$
```

8. Check the status of the database instances on each node. What do you observe?

```
[oracle@host01 ~]$ srvctl status database -db orcl
Instance orcl_1 is running on node host02
Instance orcl_2 is running on node host03
Instance orcl_3 is running on node host01

[oracle@host01 ~]$
```

The instance (orcl_3 in this example) on host01 was restored back to the original state using the information in the host01_dbhome_state2.dmp file.

9. Close all terminal windows opened for this practice.

Practice 4-4: Configuring Leaf Nodes to run Read-Only Instances

Overview

In this practice, you will configure leaf nodes to run read-only instances.

1. Open a terminal to connect to host01 as the root user. Check the network resource with crsctl.

Name	Target	State	Server	State details
<hr/>				
Local Resources				
ora.ASMNET1LSNR_ASM.lsnr				
	ONLINE	ONLINE	host01	STABLE
	ONLINE	ONLINE	host02	STABLE
	ONLINE	ONLINE	host03	STABLE
ora.DATA.dg				
	ONLINE	ONLINE	host01	STABLE
	ONLINE	ONLINE	host02	STABLE
	ONLINE	ONLINE	host03	STABLE
ora.FRA.dg				
	ONLINE	ONLINE	host01	STABLE
	ONLINE	ONLINE	host02	STABLE
	ONLINE	ONLINE	host03	STABLE
ora.LISTENER.lsnr				
	ONLINE	ONLINE	host01	STABLE
	ONLINE	ONLINE	host02	STABLE
	ONLINE	ONLINE	host03	STABLE
ora.LISTENER_LEAF.lsnr				
	OFFLINE	OFFLINE	host04	STABLE
	OFFLINE	OFFLINE	host05	STABLE

```

ora.MGMT.dg
    ONLINE  ONLINE      host01      STABLE
    ONLINE  ONLINE      host02      STABLE
    ONLINE  ONLINE      host03      STABLE

ora.chad
    ONLINE  ONLINE      host01      STABLE
    ONLINE  ONLINE      host02      STABLE
    ONLINE  ONLINE      host03      STABLE
    ONLINE  ONLINE      host04      STABLE
    ONLINE  ONLINE      host05      STABLE

ora.net1.network
    ONLINE  ONLINE      host01      STABLE
    ONLINE  ONLINE      host02      STABLE
    ONLINE  ONLINE      host03      STABLE
    ...
[root@host01 ~]#

```

2. As the `root` user, extend the network resource to `host04` and `host05` using `SRVCTL`. Recheck the network resource.

```

[root@host01 ~]# srvctl modify network -netnum 1 -extendtoleaf
yes

[root@host01 ~]# crsctl stat res ora.net1.network -t

-----
Name        Target   State       Server     State details
-----
Local Resources
-----
ora.net1.network
    ONLINE  ONLINE      host01      STABLE
    ONLINE  ONLINE      host02      STABLE
    ONLINE  ONLINE      host03      STABLE
    OFFLINE OFFLINE      host04      STABLE
    OFFLINE OFFLINE      host05      STABLE
-----
[root@host01 ~]#

```

3. Check the host VIPs on host01, host02, host03, host04, and host05 using SRVCTL.

```
[root@host01 ~]# srvctl status vip -node host01
VIP 192.0.2.254 is enabled
VIP 192.0.2.254 is running on node: host01

[root@host01 ~]# srvctl status vip -node host02
VIP 192.0.2.231 is enabled
VIP 192.0.2.231 is running on node: host02

[root@host01 ~]# srvctl status vip -node host03
VIP 192.0.2.220 is enabled
VIP 192.0.2.220 is running on node: host03

[root@host01 ~]# srvctl status vip -node host04
PRKO-2165 : VIP does not exist on node(s) : host04

[root@host01 ~]# srvctl status vip -node host05
PRKO-2165 : VIP does not exist on node(s) : host05

[root@host01 ~]#
```

4. As the root user, add VIP resources to host04 and host05 using SRVCTL and check the status of the new VIPs. **Note:** Since we chose the “Configure nodes Virtual IPs as assigned by the Dynamic Networks” option during the Grid Infrastructure installation, the –address option is not allowed.

```
[root@host01 ~]# srvctl add vip -node host04 -netnum 1 -address host04-vip/255.255.255.0
PRCR-1170 : Failed to add resource ora.host04.vip
PRCN-3028 : VIP ipv4 address modification is not allowed because
the network type is "dhcp"

[root@host01 ~]# srvctl add vip -node host04 -netnum 1

[root@host01 ~]# srvctl add vip -node host05 -netnum 1

[root@host01 ~]# srvctl status vip -node host04
VIP host04 is enabled
VIP host04 is not running
```

```
[root@host01 ~]# srvctl status vip -node host05
VIP host05 is enabled
VIP host05 is not running

[root@host01 ~]#
```

5. Start the VIPs on host04 and host05. Check the VIP and Network resource status. Both should be up.

```
[root@host01 ~]# srvctl start vip -node host04

[root@host01 ~]# srvctl start vip -node host05

[root@host01 ~]# crsctl stat res -t

-----
Name          Target   State        Server      State details
-----
Local Resources

-----
ora.ASMNET1LSNR_ASM.lsnr
    ONLINE   ONLINE    host01      STABLE
    ONLINE   ONLINE    host02      STABLE
    ONLINE   ONLINE    host03      STABLE
ora.DATA.dg
    ONLINE   ONLINE    host01      STABLE
    ONLINE   ONLINE    host02      STABLE
    ONLINE   ONLINE    host03      STABLE
ora.FRA.dg
    ONLINE   ONLINE    host01      STABLE
    ONLINE   ONLINE    host02      STABLE
    ONLINE   ONLINE    host03      STABLE
ora.LISTENER.lsnr
    ONLINE   ONLINE    host01      STABLE
    ONLINE   ONLINE    host02      STABLE
    ONLINE   ONLINE    host03      STABLE
ora.LISTENER_LEAF.lsnr
    OFFLINE  OFFLINE   host04      STABLE
    OFFLINE  OFFLINE   host05      STABLE
```

ora.MGMT.dg	ONLINE	ONLINE	host01	STABLE
	ONLINE	ONLINE	host02	STABLE
	ONLINE	ONLINE	host03	STABLE
ora.chad	ONLINE	ONLINE	host01	STABLE
	ONLINE	ONLINE	host02	STABLE
	ONLINE	ONLINE	host03	STABLE
	ONLINE	ONLINE	host04	STABLE
	ONLINE	ONLINE	host05	STABLE
ora.net1.network	ONLINE	ONLINE	host01	STABLE
	ONLINE	ONLINE	host02	STABLE
	ONLINE	ONLINE	host03	STABLE
	ONLINE	ONLINE	host04	STABLE
	ONLINE	ONLINE	host05	STABLE
...				
<hr/>				
Cluster Resources				
<hr/>				
ora.LISTENER_SCAN1.lsnr	1	ONLINE	ONLINE	host03
				STABLE
ora.LISTENER_SCAN2.lsnr	1	ONLINE	ONLINE	host02
				STABLE
ora.LISTENER_SCAN3.lsnr	1	ONLINE	ONLINE	host01
				STABLE
...				
<hr/>				
ora.host01.vip	1	ONLINE	ONLINE	host01
				STABLE
ora.host02.vip	1	ONLINE	ONLINE	host02
				STABLE
ora.host03.vip	1	ONLINE	ONLINE	host03
				STABLE
ora.host04.vip	1	ONLINE	ONLINE	host04
				STABLE

ora.host05.vip				
1	ONLINE	ONLINE	host05	STABLE
...				
[root@host01 ~]#				

6. Open a terminal to connect to host01 as the grid user and set the environment. Check listener status for the cluster. You will need to extend listeners to the leaf nodes and start them.

```
[oracle@dns ~]$ ssh grid@host01
grid@host01's password:

[grid@host01 ~]$ . oraenv
ORACLE_SID = [grid] ? +ASM1
The Oracle base has been set to /u01/app/grid

[grid@host01 ~]$ srvctl status listener
Listener LISTENER is enabled
Listener LISTENER is running on node(s): host02,host03, host01

[grid@host01 ~]$ srvctl modify listener -listener LISTENER -
extendtoleaf yes

[grid@host01 ~]$ srvctl start listener -listener LISTENER -node
host04

[grid@host01 ~]$ srvctl start listener -listener LISTENER -node
host05

[grid@host01 ~]$ srvctl status listener
Listener LISTENER is enabled
Listener LISTENER is running on node(s):
host02,host03,host01,host04,host05

[grid@host01 ~]$
```

7. Open a terminal to connect to host01 as the oracle user to edit the /etc/oratab file on host01. Add five lines for instances orcl_1, orcl_2, orcl_3, orcl_4, and orcl_5 as shown below. Repeat this on host02, host03, host04, and host05. If you prefer, you can add them using the /stage/RAC/labs/less_04/oratab_upd.sh script.

```
[oracle@dns ~]$ ssh oracle@host01
oracle@host01's password:

[oracle@host01 ~]$ vi /etc/oratab

# This file is used by ORACLE utilities. It is created by
root.sh
# and updated by either Database Configuration Assistant while
creating
# a database or ASM Configuration Assistant while creating ASM
instance.

# A colon, ':', is used as the field terminator. A new line
terminates
# the entry. Lines beginning with a pound sign, '#', are
comments.

#
# Entries are of the form:
# $ORACLE_SID:$ORACLE_HOME:<N|Y>:
#
# The first and second fields are the system identifier and home
# directory of the database respectively. The third field
indicates
# to the dbstart utility that the database should , "Y", or
should not,
# "N", be brought up at system boot time.
#
# Multiple entries with the same $ORACLE_SID are not allowed.
#
#
+ASM1:/u01/app/12.2.0/grid:N          # line added by Agent
-MGMTDB:/u01/app/12.2.0/grid:N
orcl:/u01/app/oracle/product/12.2.0/dbhome_1:N
orcl_1:/u01/app/oracle/product/12.2.0/dbhome_1:N
orcl_2:/u01/app/oracle/product/12.2.0/dbhome_1:N
```

```

orcl_3:/u01/app/oracle/product/12.2.0/dbhome_1:N
orcl_4:/u01/app/oracle/product/12.2.0/dbhome_1:N
orcl_5:/u01/app/oracle/product/12.2.0/dbhome_1:N

**** Save file and Repeat for host02, host03, host04, and host06
*****
OR
**** You can use the oratab_upd.sh script to add the entries
*****
```

[oracle@host01 ~]\$ /stage/RAC/labs/less_04/oratab_upd.sh

[oracle@host01 ~]\$

8. As the oracle user, check and see which orcl instance is running on host01. In this example, the instance is orcl_3. Set the environment for that instance.
 - Add a database service called OLTPSVC using the orcldb server pool.
 - Add a server pool for the leaf nodes called RACLEAF.
 - Add a service for the leaf nodes called LEAFSVC using the RACLEAF pool.

```

[oracle@host01 ~]$ ps -ef|grep ora_smon
oracle    9054  5263  0 08:02 pts/5    00:00:00 grep ora_smon
oracle   32060     1  0 Nov02 ?        00:00:04 ora_smon_orcl_3

[oracle@host01 ~]$ . oraenv
ORACLE_SID = [oracle] ? orcl_3
The Oracle base has been set to /u01/app/oracle

[oracle@host01 ~]$ srvctl add service -db orcl -service OLTPSVC
-serverpool orcldb
[oracle@host01 ~]$ srvctl add svrpool -serverpool RACLEAF -
category LEAF
[oracle@host01 ~]$ srvctl add service -db orcl -service LEAFSVC
-rfpool RACLEAF

[oracle@host01 ~]$
```

9. Check the status of the new services. Start them using SRVCTL. It will take a few moments for the LEAFSVC service to start as starting it will cause database instances to start on host04 and host05. Recheck the status of the new services and check the status of the orcl database.

```
[oracle@host01 ~]$ srvctl status service -db orcl
Service LEAFSVC is not running.
Service OLTPSVC is not running.

[oracle@host01 ~]$ srvctl start service -db orcl -service OLTPSVC

[oracle@host01 ~]$ srvctl start service -db orcl -service LEAFSVC

[oracle@host01 ~]$ srvctl status service -db orcl
Service LEAFSVC is running on nodes: host04,host05
Service OLTPSVC is running on nodes: host02,host03,host01

[oracle@host01 ~]$
```

10. From your oracle terminal, start SQL*Plus and run the query shown below to check this.

```
[oracle@host01 ~]$ sqlplus / as sysdba

SQL*Plus: Release 12.2.0.1.0 Production on Fri Nov 3 08:13:53
2017

Copyright (c) 1982, 2016, Oracle. All rights reserved.

Connected to:
Oracle Database 12c Enterprise Edition Release 12.2.0.1.0 -
64bit Production

SQL> col host_name for a20

SQL> select instance_name, host_name, status, database_status,
instance_mode from gv$instance order by 1;

INSTANCE_NAME HOST_NAME STATUS DATABASE_STATUS INSTANCE_MODE
----- ----- -----
orcl_1 host02.example.com OPEN ACTIVE REGULAR
```

orcl_2	host03.example.com	OPEN	ACTIVE	REGULAR
orcl_3	host01.example.com	OPEN	ACTIVE	REGULAR
orcl_4	host04.example.com	OPEN	ACTIVE	REGULAR
orcl_5	host05.example.com	OPEN	ACTIVE	REGULAR

SQL>

Note that the instances on host04 and host05 are NOT open read-only.

11. Setting up reader nodes manually as you have done does not set the `rf_node.instance_mode` parameter to `read-only`. You will have to do this using SQL*Plus. In addition, set the parameter `_read_mostly_enable_logon` to true to allow non-privileged users to logon to read-only instances. When finished, use SRVCTL to stop and then restart the database.

```
SQL> alter system set instance_mode='read-only' family='rf_node'
      scope=spfile;
```

System altered.

```
SQL> alter system set "_read_mostly_enable_logon=true"
      scope=spfile;
```

System altered.

```
SQL> exit
```

Disconnected from Oracle Database 12c Enterprise Edition Release
12.2.0.1.0 - 64bit Production

```
[oracle@host01 ~]$ srvctl stop database -db orcl
```

```
[oracle@host01 ~]$ srvctl start database -db orcl
```

```
[oracle@host01 ~]$
```

12. Check which instance has been started on host01. It will most likely be different from the one running there before the database was bounced. Set the environment for that instance, restart SQL*Plus and re-run the query against gv\$instance you ran earlier. What do you see?

```
[oracle@host01 ~]$ ps -ef|grep ora_smon
oracle    16948      1  0 08:27 ?          00:00:00 ora_smon_orcl_5
oracle    17662  5263  0 08:28 pts/5      00:00:00 grep ora_smon

[oracle@host01 ~]$ . oraenv
ORACLE_SID = [orcl_3] ? orcl_5
The Oracle base remains unchanged with value /u01/app/oracle

[oracle@host01 ~]$ sqlplus / as sysdba

SQL*Plus: Release 12.2.0.1.0 Production on Fri Nov 3 08:29:50
2017

Copyright (c) 1982, 2016, Oracle. All rights reserved.

Connected to:
Oracle Database 12c Enterprise Edition Release 12.2.0.1.0 -
64bit Production

SQL> col host_name for a20

SQL> select instance_name, host_name, status, database_status,
instance_mode from gv$instance order by 1;

INSTANCE_NAME HOST_NAME           STATUS DATABASE_STATUS INSTANCE_MO
-----
orcl_1        host04.example.com  OPEN   ACTIVE      READ ONLY
orcl_2        host05.example.com  OPEN   ACTIVE      READ ONLY
orcl_3        host02.example.com  OPEN   ACTIVE      REGULAR
orcl_4        host03.example.com  OPEN   ACTIVE      REGULAR
orcl_5        host01.example.com  OPEN   ACTIVE      REGULAR

Note that database instances on host04 and host05 are now correctly running in read-only mode.
```

```
SQL> exit
Disconnected from Oracle Database 12c Enterprise Edition Release
12.2.0.1.0 - 64bit Production

[oracle@host01 ~]$
```

13. Next, let's quickly test the services. Edit

`$ORACLE_HOME/network/admin/tnsnames.ora` and add connect strings for the OLTPSVC and LEAFSVC services at the bottom of the file as shown below. If you prefer, you can add them using the `/stage/RAC/labs/less_04/add_svcs_tns.sh` script. Test them with `tnsping`.

```
[oracle@host01 ~]$ vi $ORACLE_HOME/network/admin/tnsnames.ora
# tnsnames.ora Network Configuration File:
/u01/app/oracle/product/12.2.0/dbhome_1/network/admin/tnsnames.ora
# Generated by Oracle configuration tools.

...
ORCL =
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL = TCP) (HOST = cluster01-
scan.cluster01.example.com) (PORT = 1521))
    (CONNECT_DATA =
      (SERVER = DEDICATED)
      (SERVICE_NAME = orcl)
    )
  )

OLTPSVC = (DESCRIPTION = (ADDRESS = (PROTOCOL = TCP) (HOST =
cluster01-scan.cluster01.example.com) (PORT = 1521)) (LOAD_BALANCE =
YES) (CONNECT_DATA = (SERVER = DEDICATED) (SERVICE_NAME = oltpsvc)))

LEAFSVC = (DESCRIPTION = (ADDRESS = (PROTOCOL = TCP) (HOST =
cluster01-scan.cluster01.example.com) (PORT = 1521)) (LOAD_BALANCE =
YES) (CONNECT_DATA = (SERVER = DEDICATED) (SERVICE_NAME = leafsvc)))

**** Or add then by executing /stage/RAC/labs/less_04/add_svcs_tns.sh
****

[oracle@host01 ~]$ /stage/RAC/labs/less_04/add_svcs_tns.sh
```

```
[oracle@host01 ~]$ tnsping oltpsvc
...
Used parameter files:

Used TNSNAMES adapter to resolve the alias
Attempting to contact (DESCRIPTION = (ADDRESS = (PROTOCOL =
TCP) (HOST = cluster01-scan.cluster01.example.com) (PORT =
1521)) (LOAD_BALANCE = YES) (CONNECT_DATA = (SERVER =
DEDICATED) (SERVICE_NAME = oltpsvc)))
OK (30 msec)

[oracle@host01 ~]$ tnsping leafsvc
...
Used parameter files:

Used TNSNAMES adapter to resolve the alias
Attempting to contact (DESCRIPTION = (ADDRESS = (PROTOCOL =
TCP) (HOST = cluster01-scan.cluster01.example.com) (PORT =
1521)) (LOAD_BALANCE = YES) (CONNECT_DATA = (SERVER =
DEDICATED) (SERVICE_NAME = leafsvc)))
OK (40 msec)

[oracle@host01 ~]$
```

14. Make some connections to the services. Query V\$INSTANCE to make sure they are working as configured.

```
[oracle@host01 ~]$ sqlplus sys/<SYS_PASSWORD>@oltpsvc as sysdba

SQL*Plus: Release 12.2.0.1.0 Production on Fri Nov 3 14:03:27 2017

Copyright (c) 1982, 2016, Oracle. All rights reserved.

Connected to:
Oracle Database 12c Enterprise Edition Release 12.2.0.1.0 - 64bit
Production

SQL> col host_name for a20
```

```
SQL> select instance_name, host_name, instance_mode from v$instance;
```

INSTANCE_NAME	HOST_NAME	INSTANCE_MO
orcl_3	host02.example.com	REGULAR

```
SQL> connect sys/<SYS_PASSWORD>@oltpsvc as sysdba  
Connected.
```

```
SQL> select instance_name, host_name, instance_mode from v$instance;
```

INSTANCE_NAME	HOST_NAME	INSTANCE_MO
orcl_4	host03.example.com	REGULAR

```
SQL> connect sys/<SYS_PASSWORD>@leafsvc as sysdba  
Connected.
```

```
SQL> select instance_name, host_name, instance_mode from v$instance;
```

INSTANCE_NAME	HOST_NAME	INSTANCE_MO
orcl_1	host04.example.com	READ ONLY

```
SQL> connect sys/<SYS_PASSWORD>@leafsvc as sysdba  
Connected.
```

```
SQL> select instance_name, host_name, instance_mode from v$instance;
```

INSTANCE_NAME	HOST_NAME	INSTANCE_MO
orcl_2	host05.example.com	READ ONLY

```
SQL> exit  
[oracle@host01 ~]$
```

15. Close all terminals opened for this practice.

Practice 4-5: Working with Temporary Tablespaces

Overview

In this practice, you will work with local temporary tablespaces on read-write and read-only instances.

1. Open a new terminal window to connect to host01 as oracle user and set the environment using oraenv.

```
[oracle@dns ~]$ ssh oracle@host01
oracle@host01's password:

[oracle@host01 ~]$ ps -ef|grep ora_smon
oracle    16948      1  0 08:27 ?          00:00:00 ora_smon_orcl_5
oracle    16973 16899  0 10:14 pts/8      00:00:00 grep ora_smon

[oracle@host01 ~]$ . oraenv
ORACLE_SID = [oracle] ? orcl_5
The Oracle base has been set to /u01/app/oracle

[oracle@host01 ~]$
```

2. Start a SQL*Plus session as sysdba and create two local temporary tablespaces, one called TEMP_LEAF and another one called TEMP_ALL. Create TEMP_LEAF using the CREATE LOCAL TEMPORARY TABLESPACE for LEAF command and TEMP_ALL using the CREATE LOCAL TEMPORARY TABLESPACE for ALL command.

```
[oracle@host01 ~]$ sqlplus / as sysdba

SQL*Plus: Release 12.2.0.1.0 Production on Fri Nov 3 10:15:35
2017

Copyright (c) 1982, 2016, Oracle. All rights reserved.

Connected to:
Oracle Database 12c Enterprise Edition Release 12.2.0.1.0 -
64bit Production
SQL> CREATE LOCAL TEMPORARY TABLESPACE for LEAF TEMP_LEAF
TEMPFILE 'temp_leaf01.dbf' SIZE 5M AUTOEXTEND ON;

Tablespace created.
```

```
SQL> CREATE LOCAL TEMPORARY TABLESPACE for ALL TEMP ALL TEMPFILE
'temp_all01.dbf' SIZE 5M AUTOEXTEND ON;

Tablespace created.

SQL>
```

3. Look at the temp files created using SQL*Plus. In this example, `orcl_1` and `orcl_2` are running on the Leaf Nodes. The rest of instances are running on the Hub Nodes.

```
SQL> !srvctl status database -db orcl
Instance orcl_1 is running on node host04
Instance orcl_2 is running on node host05
Instance orcl_3 is running on node host02
Instance orcl_4 is running on node host03
Instance orcl_5 is running on node host01

SQL> set linesize 120
SQL> col file_name format a65
SQL> break on inst_id
SQL> select inst_id, file_name, shared from dba_temp_files order by
inst_id;

INST_ID FILE_NAME                               SHARED
----- -----
  1 /u01/app/oracle/product/12.2.0/dbhome_1/dbs/temp_leaf01.dbf_1 LOCAL_ON_LEAF
    /u01/app/oracle/product/12.2.0/dbhome_1/dbs/temp_all01.dbf_1 LOCAL_ON_ALL
  2 /u01/app/oracle/product/12.2.0/dbhome_1/dbs/temp_all01.dbf_2 LOCAL_ON_ALL
    /u01/app/oracle/product/12.2.0/dbhome_1/dbs/temp_leaf01.dbf_2 LOCAL_ON_LEAF
  3 /u01/app/oracle/product/12.2.0/dbhome_1/dbs/temp_all01.dbf_3 LOCAL_ON_ALL
  4 /u01/app/oracle/product/12.2.0/dbhome_1/dbs/temp_all01.dbf_4 LOCAL_ON_ALL
  5 /u01/app/oracle/product/12.2.0/dbhome_1/dbs/temp_all01.dbf_5 LOCAL_ON_ALL
+DATA/ORCL/TEMPFILE/temp.296.958993285          SHARED

8 rows selected.

SQL>
```

4. Grant dba privileges to hr.

```
SQL> grant dba to hr;  
Grant succeeded.  
SQL>
```

5. Alter the hr user and assign TEMP_LEAVE as the local temporary tablespace. Confirm the assignment.

```
SQL> alter user hr local temporary tablespace TEMP_LEAVE;  
  
User altered.  
  
SQL> select username, TEMPORARY_TABLESPACE,  
LOCAL_TEMP_TABLESPACE from dba_users where username = 'HR';  
  
USERNAME    TEMPORARY_TABLESPACE      LOCAL_TEMP_TABLESPACE  
-----  
HR          TEMP                  TEMP_LEAVE  
SQL>
```

6. Grant SELECT on col\$ for hr. Connect as hr and count the records in col\$.

```
SQL> grant select on col$ to hr;  
  
Grant succeeded.  
  
SQL> connect hr/<hr_password>  
Connected.  
  
SQL> select count(*) from sys.col$;  
  
COUNT(*)  
-----  
114532  
  
1 row selected.  
  
SQL>
```

7. As hr, create a table called test as SELECT from sys.col\$. Perform an INSERT AS SELECT into test to add more records. Re-execute the command (/) 5 times as shown below then COMMIT.

```
SQL> create table test as select obj#, col#, name from sys.col$;

Table created.

SQL> insert into test select * from test;

114532 rows created.

SQL> /

229064 rows created.

SQL> /

458128 rows created.

SQL> /

916256 rows created.

SQL> /

1832512 rows created.

SQL> /

3665024 rows created.

SQL> commit;

Commit complete.

SQL> select count(*) from test;

COUNT(*)
-----
7330048

SQL>
```

8. Run the query below to check tempfile activity for the instance.

```
SQL> select f.name, s.phyrd, s.phywrt from v$tempfile f, v$tempstat
  s where f.file# = s.file# order by 1;
```

NAME	PHYRDS	PHYWRTS
+DATA/ORCL/TEMPFILE/temp.296.958993285	1	4
/u01/app/oracle/product/12.2.0/dbhome_1/dbs/temp_all01.dbf	0	0
/u01/app/oracle/product/12.2.0/dbhome_1/dbs/temp_leaf01.dbf	0	0

SQL>

9. Now, as hr logged into orcl_5 (a read-write instance), set autotrace to traceonly (to suppress query output) and select all records in the table.

```
SQL> show user
USER is "HR"

SQL> set autotrace traceonly
SQL> select * from test order by 1, 2;

7333376 rows selected.

Execution Plan
-----
Plan hash value: 2007178810

-----
| Id  | Operation          | Name | Rows  | Bytes | TempSpc| Cost (%CPU) | Time  |
| 0  | SELECT STATEMENT   |      | 114K | 2014K|        | 763  (1) | 00:00:01 |
| 1  |  SORT ORDER BY     |      | 114K | 2014K| 3160K | 763  (1) | 00:00:01 |
| 2  | TABLE ACCESS FULL| TEST | 114K | 2014K|        | 109  (1) | 00:00:01 |
-----

Statistics
-----
      195 recursive calls
      964 db block gets
    24139 consistent gets
```

```

24715 physical reads
    0 redo size
134929753 bytes sent via SQL*Net to client
    5378365 bytes received via SQL*Net from client
    488889 SQL*Net roundtrips to/from client
    0 sorts (memory)
    1 sorts (disk)
7333312 rows processed
SQL>

```

10. Set AUTOTRACE OFF and re-run the query from step 8.

```

SQL> set autotrace off

SQL> select f.name, s.phyrd, s.phywrt from v$tempfile f, v$tempstat
  s where f.file# = s.file# order by 1;

NAME          PHYRDS   PHYWRTS
-----        -----
+DATA/ORCL/TEMPFILE/ temp.296.958993285           983     1580
/u01/app/oracle/product/12.2.0/dbhome_1/dbs/temp_all01.dbf      0       0
/u01/app/oracle/product/12.2.0/dbhome_1/dbs/temp_leaf01.dbf      0       0

SQL>

```

Note that the shared tempfile is being used.

11. Open a new terminal window to connect to the leaf node host04 as oracle user and set the environment using oraenv. Login as hr to the read-only instance. In this example, that is orcl_1. Set AUTOTRACE to TRACEONLY and rerun the query against the test table.

```

[oracle@dns ~]$ ssh oracle@host04
oracle@host04's password:

[oracle@host04 ~]$ ps -ef|grep ora_smon
oracle    16469      1  0 16:48 ?          00:00:00 ora_smon_orcl_1
oracle    25899 25605  0 17:30 pts/0      00:00:00 grep ora_smon

[oracle@host04 ~]$ . oraenv

```

```
ORACLE_SID = [oracle] ? orcl_1
The Oracle base has been set to /u01/app/oracle

[oracle@host04 ~]$ sqlplus hr/<hr_password>

SQL> set autotrace traceonly

SQL> select * from hr.test order by 1, 2;
...
ERROR:
ORA-16000: database or pluggable database open for read-only access
SP2-0612: Error generating AUTOTRACE EXPLAIN report

Statistics
-----
      1593 recursive calls
       385 db block gets
     24250 consistent gets
    48847 physical reads
        0 redo size
 34929753 bytes sent via SQL*Net to client
 5378365 bytes received via SQL*Net from client
 488889 SQL*Net roundtrips to/from client
     195 sorts (memory)
       1 sorts (disk)
 7333312 rows processed

SQL>
```

You can disregard the ORA-16000 and SP2-0612 messages. This is a read-only instance and PLAN_TABLE cannot be created or updated. This is unimportant as we are only using AUTOTRACE TRACEONLY to suppress excessive output from the query.

12. Set AUTOTRACE OFF and re-run the query from step 10.

```
SQL> set autotrace off

SQL> col name format a60
SQL> set lines 150
```

SQL> **select f.name, s.phyrd, s.phywrt from v\$tempfile f, v\$tempstat s where f.file# = s.file# order by 1;**

NAME	PHYRDS	PHYWRTS
+DATA/ORCL/TEMPFILE/ temp.296.958993285	1	0
/u01/app/oracle/product/12.2.0/dbhome_1/dbs/temp_all01.dbf	1	0
/u01/app/oracle/product/12.2.0/dbhome_1/dbs/temp_leaf01.dbf	968	992

SQL>

Note that the TEMP_leaf local temporary tablespace was used.

13. Switch to the terminal session on host01 and drop two local temporary tablespaces.

SQL> **drop tablespace temp_leaf including contents and datafiles;**
Tablespace dropped.

SQL> **drop tablespace temp_all including contents and datafiles;**
Tablespace dropped.

SQL>

14. Close all terminals opened for this practice.

Practices for Lesson 5: Upgrading and Patching Oracle RAC

Practices for Lesson 5

There are no practices for this lesson.

Practices for Lesson 6: Managing Backup and Recovery for RAC

Practices for Lesson 6: Overview

Overview

In this practice, you will configure ARCHIVELOG mode for your RAC database.

In this practice, you will designate two read-only instances running on the Leaf Nodes of your policy-managed database responsible for performing parallel backups of the database. The database will be backed up to the +FRA ASM disk group by default.

Practice 6-1: Configuring Archive Log Mode

Overview

In this practice, you will configure ARCHIVELOG mode for your RAC database.

1. Open a terminal session to host01 as the oracle user and set up the environment variables using the oraenv script for the database instance. Determine the instance running on host01. Change the value of the ORACLE_SID variable to allow local system authenticated connections.

```
[oracle@dns ~] ssh oracle@host01
Oracle@host01's Password:

[oracle@host01 ~]$ . oraenv
ORACLE_SID = [oracle] ? orcl
The Oracle base has been set to /u01/app/oracle

[oracle@host01 ~]$ srvctl status database -db orcl
Instance orcl_1 is running on node host04
Instance orcl_2 is running on node host05
Instance orcl_3 is running on node host02
Instance orcl_4 is running on node host03
Instance orcl_5 is running on node host01

[oracle@host01 ~]$ export ORACLE_SID=orcl_5
[oracle@host01 ~]$
```

2. Make a local connection using operating system authentication to the database instance, and then use the archive log list SQL command to determine whether the database is in archive log mode. Exit SQL*Plus when done.

```
[oracle@host01 ~]$ sqlplus / as sysdba

SQL*Plus: Release 12.2.0.1.0 Production on Mon Nov 6 08:22:05
2017

Copyright (c) 1982, 2016, Oracle. All rights reserved.

Connected to:
Oracle Database 12c Enterprise Edition Release 12.2.0.1.0 -
64bit Production

SQL> archive log list
Database log mode          No Archive Mode
```

```

Automatic archival           Disabled
Archive destination          USE_DB_RECOVERY_FILE_DEST
Oldest online log sequence   1
Current log sequence         1

```

SQL> **exit**

Disconnected from Oracle Database 12c Enterprise Edition Release
12.2.0.1.0 - 64bit Production
[oracle@host01 ~]\$

- Stop the `orcl` database on each node of the cluster by using the `srvctl stop database` command.

```
[oracle@host01 ~]$ srvctl stop database -db orcl
[oracle@host01 ~]$
```

- Verify that the `orcl` database is not running on any node of the cluster by using the `srvctl status database` command.

```
[oracle@host01 ~]$ srvctl status database -db orcl
Instance orcl_1 is not running on node host04
Instance orcl_2 is not running on node host05
Instance orcl_3 is not running on node host02
Instance orcl_4 is not running on node host03
Instance orcl_5 is not running on node host01

[oracle@host01 ~]$
```

- Make a local connection using operating system authentication to the local database instance, and then start up the database on only the first node with the `mount` option.

```
[oracle@host01 ~]$ sqlplus / as sysdba

SQL*Plus: Release 12.2.0.1.0 Production on Mon Nov 6 08:26:10
2017

Copyright (c) 1982, 2016, Oracle. All rights reserved.

Connected to an idle instance.

SQL> startup mount
ORACLE instance started.

Total System Global Area 2399141888 bytes
```

```

Fixed Size           8623496 bytes
Variable Size       721422968 bytes
Database Buffers    1660944384 bytes
Redo Buffers        8151040 bytes
Database mounted.
SQL>

```

6. Issue the `alter database archivelog` SQL command to change the archive mode of the database, and then verify the results by using the `archive log list` SQL command.

```

SQL> alter database archivelog;

Database altered.

SQL> archive log list
Database log mode          Archive Mode
Automatic archival         Enabled
Archive destination        USE_DB_RECOVERY_FILE_DEST
Oldest online log sequence 80
Next log sequence to archive 81
Current log sequence       81
SQL>

```

7. Shut down the database instance with the `immediate` option and exit SQL*Plus. Use the `srvctl` utility to restart the database instances on all nodes of the cluster.

```

SQL> shutdown immediate
ORA-01109: database not open

Database dismounted.
ORACLE instance shut down.

SQL> exit
Disconnected from Oracle Database 12c Enterprise Edition Release
12.1.0.2.0 - 64bit Production
With the Partitioning, Real Application Clusters, Automatic
Storage Management, OLAP,
Advanced Analytics and Real Application Testing options

[oracle@host01 ~]$ srvctl start database -db orcl
[oracle@host01 ~]$

```

8. Verify that the `orcl` database is running on all the five nodes of your cluster by using the `srvctl status database` command.

```
[oracle@host01 ~]$ srvctl status database -db orcl
Instance orcl_1 is running on node host04
Instance orcl_2 is running on node host05
Instance orcl_3 is running on node host02
Instance orcl_4 is running on node host03
Instance orcl_5 is running on node host01

[oracle@host01 ~]$
```

Practice 6-2: Configuring RMAN and Performing Parallel Backups

Overview

In this practice, you will designate two read-only instances running on the Leaf Nodes of your policy-managed database responsible for performing parallel backups of the database. The database will be backed up to the +FRA ASM disk group by default.

1. Using the recovery manager utility (RMAN), connect to the `orcl` database as the target database.

```
[oracle@host01 ~]$ rman target /  
  
Recovery Manager: Release 12.2.0.1.0 - Production on Mon Nov 6  
08:38:16 2017  
  
Copyright (c) 1982, 2017, Oracle and/or its affiliates. All  
rights reserved.  
  
connected to target database: ORCL (DBID=1487037122)  
  
RMAN>
```

2. Display all of the current RMAN settings.

```
RMAN> show all;  
  
using target database control file instead of recovery catalog  
RMAN configuration parameters for database with db_unique_name  
ORCL are:  
CONFIGURE RETENTION POLICY TO REDUNDANCY 1; # default  
CONFIGURE BACKUP OPTIMIZATION OFF; # default  
CONFIGURE DEFAULT DEVICE TYPE TO DISK; # default  
CONFIGURE CONTROLFILE AUTOBACKUP ON; # default  
CONFIGURE CONTROLFILE AUTOBACKUP FORMAT FOR DEVICE TYPE DISK TO  
'%F'; # default  
CONFIGURE DEVICE TYPE DISK PARALLELISM 1 BACKUP TYPE TO  
BACKUPSET; # default  
CONFIGURE DATAFILE BACKUP COPIES FOR DEVICE TYPE DISK TO 1; #  
default  
CONFIGURE ARCHIVELOG BACKUP COPIES FOR DEVICE TYPE DISK TO 1; #  
default  
CONFIGURE MAXSETSIZE TO UNLIMITED; # default  
CONFIGURE ENCRYPTION FOR DATABASE OFF; # default  
CONFIGURE ENCRYPTION ALGORITHM 'AES128'; # default  
CONFIGURE COMPRESSION ALGORITHM 'BASIC' AS OF RELEASE 'DEFAULT'  
OPTIMIZE FOR LOAD TRUE ; # default
```

```

CONFIGURE RMAN OUTPUT TO KEEP FOR 7 DAYS; # default
CONFIGURE ARCHIVELOG DELETION POLICY TO NONE; # default
CONFIGURE SNAPSHOT CONTROLFILE NAME TO
'/u01/app/oracle/product/12.2.0/dbhome_1/dbs/snapcf_orcl_5.f'; # default

RMAN>

```

3. Configure RMAN to automatically back up the control file and server parameter file each time any backup operation is performed.

```

RMAN> configure controlfile autobackup on;

new RMAN configuration parameters:
CONFIGURE CONTROLFILE AUTOBACKUP ON;
new RMAN configuration parameters are successfully stored

RMAN>

```

4. Configure channels to use automatic load balancing. Set parallelism to 4, then configure the connect string. **Note:** You already created the leafsvc service in practice 4-4.

```

RMAN> CONFIGURE DEVICE TYPE disk PARALLELISM 4;

new RMAN configuration parameters:
CONFIGURE DEVICE TYPE DISK PARALLELISM 4 BACKUP TYPE TO
BACKUPSET;
new RMAN configuration parameters are successfully stored

RMAN> configure channel device type disk connect =
'sys/<sys_password>@leafsvc';

new RMAN configuration parameters:
CONFIGURE CHANNEL DEVICE TYPE DISK CONNECT '*';
new RMAN configuration parameters are successfully stored

RMAN>

```

5. Open a second terminal session as the oracle user and set up the environment variables for the orcl database. Invoke SQL*plus as the system user, and run the /stage/RAC/labs/less_06/monitor_rman.sql script. Do not exit the first session with the RMAN prompt or this second session with the SQL prompt.

```

[oracle@host01 ~]$ . oraenv
ORACLE_SID = [oracle] ? orcl

```

```
The Oracle base has been set to /u01/app/oracle
[oracle@host01 ~]$ srvctl status database -db orcl
Instance orcl_1 is running on node host04
Instance orcl_2 is running on node host05
Instance orcl_3 is running on node host02
Instance orcl_4 is running on node host03
Instance orcl_5 is running on node host01

[oracle@host01 ~]$ export ORACLE_SID=orcl_5
[oracle@host01 ~]$ sqlplus / as sysdba

SQL*Plus: Release 12.2.0.1.0 Production on Mon Nov 6 08:42:31
2017

Copyright (c) 1982, 2016, Oracle. All rights reserved.

Connected to:
Oracle Database 12c Enterprise Edition Release 12.2.0.1.0 -
64bit Production

SQL> @/stage/RAC/labs/less_06/monitor_rman.sql

no rows selected

SQL>
```

6. In the first session with the RMAN prompt, perform a full database backup with archive logs. The backup should happen only on the designated leaf nodes (host04 and host05 in our case) as the backup nodes. **Immediately move to the next step.**

```
RMAN> backup database plus archivelog;

Starting backup at 06-NOV-17
current log archived
using target database control file instead of recovery catalog
allocated channel: ORA_DISK_1
channel ORA_DISK_1: SID=29 instance=orcl_2 device type=DISK
allocated channel: ORA_DISK_2
channel ORA_DISK_2: SID=36 instance=orcl_1 device type=DISK
allocated channel: ORA_DISK_3
channel ORA_DISK_3: SID=38 instance=orcl_2 device type=DISK
allocated channel: ORA_DISK_4
```

```
channel ORA_DISK_4: SID=37 instance=orcl_1 device type=DISK
channel ORA_DISK_1: starting archived log backup set
channel ORA_DISK_1: specifying archived log(s) in backup set
input archived log thread=3 sequence=1 RECID=3 STAMP=959330600
channel ORA_DISK_1: starting piece 1 at 06-NOV-17
channel ORA_DISK_2: starting archived log backup set
channel ORA_DISK_2: specifying archived log(s) in backup set
input archived log thread=2 sequence=1 RECID=2 STAMP=959330600
channel ORA_DISK_2: starting piece 1 at 06-NOV-17
channel ORA_DISK_3: starting archived log backup set
channel ORA_DISK_3: specifying archived log(s) in backup set
input archived log thread=1 sequence=2 RECID=1 STAMP=959330598
channel ORA_DISK_3: starting piece 1 at 06-NOV-17
channel ORA_DISK_4: starting archived log backup set
channel ORA_DISK_4: specifying archived log(s) in backup set
input archived log thread=1 sequence=6 RECID=16 STAMP=959331231
input archived log thread=2 sequence=5 RECID=15 STAMP=959331229
input archived log thread=3 sequence=5 RECID=13 STAMP=959330814
input archived log thread=3 sequence=6 RECID=14 STAMP=959331229
input archived log thread=3 sequence=7 RECID=19 STAMP=959331265
input archived log thread=2 sequence=6 RECID=18 STAMP=959331264
input archived log thread=1 sequence=7 RECID=17 STAMP=959331264
input archived log thread=1 sequence=8 RECID=20 STAMP=959331642
input archived log thread=2 sequence=7 RECID=21 STAMP=959331643
channel ORA_DISK_4: starting piece 1 at 06-NOV-17
channel ORA_DISK_1: finished piece 1 at 06-NOV-17
piece
handle=+FRA/ORCL/BACKUPSET/2017_11_06/annnf0_tag20171106t091006_
0.258.959332207 tag=TAG20171106T091006 comment=NONE
channel ORA_DISK_1: backup set complete, elapsed time: 00:00:02
channel ORA_DISK_1: starting archived log backup set
channel ORA_DISK_1: specifying archived log(s) in backup set
input archived log thread=3 sequence=8 RECID=22 STAMP=959331643
input archived log thread=1 sequence=9 RECID=23 STAMP=959331663
input archived log thread=2 sequence=8 RECID=24 STAMP=959331664
input archived log thread=3 sequence=9 RECID=25 STAMP=959331664
input archived log thread=1 sequence=10 RECID=26 STAMP=959332050
input archived log thread=2 sequence=9 RECID=27 STAMP=959332051
input archived log thread=3 sequence=10 RECID=28 STAMP=959332051
input archived log thread=1 sequence=11 RECID=29 STAMP=959332069
input archived log thread=2 sequence=10 RECID=30 STAMP=959332069
channel ORA_DISK_1: starting piece 1 at 06-NOV-17
channel ORA_DISK_2: finished piece 1 at 06-NOV-17
```

```
piece
handle=+FRA/ORCL/BACKUPSET/2017_11_06/annnf0_tag20171106t091006_
0.278.959332209 tag=TAG20171106T091006 comment=NONE
channel ORA_DISK_2: backup set complete, elapsed time: 00:00:03
channel ORA_DISK_2: starting archived log backup set
channel ORA_DISK_2: specifying archived log(s) in backup set
input archived log thread=1 sequence=3 RECID=4 STAMP=959330633
input archived log thread=2 sequence=2 RECID=6 STAMP=959330635
input archived log thread=3 sequence=2 RECID=5 STAMP=959330635
input archived log thread=1 sequence=4 RECID=9 STAMP=959330777
input archived log thread=2 sequence=3 RECID=7 STAMP=959330775
input archived log thread=3 sequence=3 RECID=8 STAMP=959330776
input archived log thread=2 sequence=4 RECID=11 STAMP=959330811
input archived log thread=3 sequence=4 RECID=12 STAMP=959330811
input archived log thread=1 sequence=5 RECID=10 STAMP=959330810
channel ORA_DISK_2: starting piece 1 at 06-NOV-17
channel ORA_DISK_3: finished piece 1 at 06-NOV-17
piece
handle=+FRA/ORCL/BACKUPSET/2017_11_06/annnf0_tag20171106t091006_
0.257.959332209 tag=TAG20171106T091006 comment=NONE
channel ORA_DISK_3: backup set complete, elapsed time: 00:00:02
channel ORA_DISK_3: starting archived log backup set
channel ORA_DISK_3: specifying archived log(s) in backup set
input archived log thread=3 sequence=11 RECID=31 STAMP=959332069
input archived log thread=1 sequence=12 RECID=32 STAMP=959332186
input archived log thread=2 sequence=11 RECID=33 STAMP=959332186
input archived log thread=3 sequence=12 RECID=34 STAMP=959332186
channel ORA_DISK_3: starting piece 1 at 06-NOV-17
channel ORA_DISK_1: finished piece 1 at 06-NOV-17
piece
handle=+FRA/ORCL/BACKUPSET/2017_11_06/annnf0_tag20171106t091006_
0.280.959332211 tag=TAG20171106T091006 comment=NONE
channel ORA_DISK_1: backup set complete, elapsed time: 00:00:00
channel ORA_DISK_2: finished piece 1 at 06-NOV-17
piece
handle=+FRA/ORCL/BACKUPSET/2017_11_06/annnf0_tag20171106t091006_
0.283.959332211 tag=TAG20171106T091006 comment=NONE
channel ORA_DISK_2: backup set complete, elapsed time: 00:00:00
channel ORA_DISK_4: finished piece 1 at 06-NOV-17
piece
handle=+FRA/ORCL/BACKUPSET/2017_11_06/annnf0_tag20171106t091006_
0.279.959332209 tag=TAG20171106T091006 comment=NONE
channel ORA_DISK_4: backup set complete, elapsed time: 00:00:01
channel ORA_DISK_3: finished piece 1 at 06-NOV-17
```

```
piece
handle=+FRA/ORCL/BACKUPSET/2017_11_06/annnf0_tag20171106t091006_
0.282.959332211 tag=TAG20171106T091006 comment=NONE
channel ORA_DISK_3: backup set complete, elapsed time: 00:00:01
Finished backup at 06-NOV-17

Starting backup at 06-NOV-17
using channel ORA_DISK_3
using channel ORA_DISK_1
using channel ORA_DISK_4
using channel ORA_DISK_2
channel ORA_DISK_1: starting full datafile backup set
channel ORA_DISK_1: specifying datafile(s) in backup set
input datafile file number=00001
name=+DATA/ORCL/DATAFILE/system.258.959256231
input datafile file number=00007
name=+DATA/ORCL/DATAFILE/users.261.959256333
channel ORA_DISK_1: starting piece 1 at 06-NOV-17
channel ORA_DISK_2: starting full datafile backup set
channel ORA_DISK_2: specifying datafile(s) in backup set
input datafile file number=00003
name=+DATA/ORCL/DATAFILE/sysaux.259.959256297
input datafile file number=00005
name=+DATA/ORCL/DATAFILE/undotbs2.274.959256735
channel ORA_DISK_2: starting piece 1 at 06-NOV-17
channel ORA_DISK_3: starting full datafile backup set
channel ORA_DISK_3: specifying datafile(s) in backup set
input datafile file number=00002
name=+DATA/ORCL/DATAFILE/undotbs3.275.959256737
input datafile file number=00004
name=+DATA/ORCL/DATAFILE/undotbs1.260.959256331
channel ORA_DISK_3: starting piece 1 at 06-NOV-17
channel ORA_DISK_3: finished piece 1 at 06-NOV-17
piece
handle=+FRA/ORCL/BACKUPSET/2017_11_06/nnndf0_tag20171106t091011_
0.286.959332213 tag=TAG20171106T091011 comment=NONE
channel ORA_DISK_3: backup set complete, elapsed time: 00:00:04
channel ORA_DISK_1: finished piece 1 at 06-NOV-17
piece
handle=+FRA/ORCL/BACKUPSET/2017_11_06/nnndf0_tag20171106t091011_
0.259.959332213 tag=TAG20171106T091011 comment=NONE
channel ORA_DISK_1: backup set complete, elapsed time: 00:00:09
channel ORA_DISK_2: finished piece 1 at 06-NOV-17
```

```
piece
handle=+FRA/ORCL/BACKUPSET/2017_11_06/nnndf0_tag20171106t091011_
0.285.959332213 tag=TAG20171106T091011 comment=NONE
channel ORA_DISK_2: backup set complete, elapsed time: 00:00:08
Finished backup at 06-NOV-17

Starting backup at 06-NOV-17
current log archived
using channel ORA_DISK_3
using channel ORA_DISK_1
using channel ORA_DISK_4
using channel ORA_DISK_2
channel ORA_DISK_1: starting archived log backup set
channel ORA_DISK_1: specifying archived log(s) in backup set
input archived log thread=1 sequence=13 RECID=37 STAMP=959332228
channel ORA_DISK_1: starting piece 1 at 06-NOV-17
channel ORA_DISK_2: starting archived log backup set
channel ORA_DISK_2: specifying archived log(s) in backup set
input archived log thread=3 sequence=13 RECID=36 STAMP=959332226
channel ORA_DISK_2: starting piece 1 at 06-NOV-17
channel ORA_DISK_3: starting archived log backup set
channel ORA_DISK_3: specifying archived log(s) in backup set
input archived log thread=2 sequence=12 RECID=35 STAMP=959332222
channel ORA_DISK_3: starting piece 1 at 06-NOV-17
channel ORA_DISK_1: finished piece 1 at 06-NOV-17
piece
handle=+FRA/ORCL/BACKUPSET/2017_11_06/annnf0_tag20171106t091028_
0.318.959332229 tag=TAG20171106T091028 comment=NONE
channel ORA_DISK_1: backup set complete, elapsed time: 00:00:01
channel ORA_DISK_2: finished piece 1 at 06-NOV-17
piece
handle=+FRA/ORCL/BACKUPSET/2017_11_06/annnf0_tag20171106t091028_
0.319.959332229 tag=TAG20171106T091028 comment=NONE
channel ORA_DISK_2: backup set complete, elapsed time: 00:00:00
channel ORA_DISK_3: finished piece 1 at 06-NOV-17
piece
handle=+FRA/ORCL/BACKUPSET/2017_11_06/annnf0_tag20171106t091028_
0.320.959332229 tag=TAG20171106T091028 comment=NONE
channel ORA_DISK_3: backup set complete, elapsed time: 00:00:00
Finished backup at 06-NOV-17

Starting Control File and SPFILE Autobackup at 06-NOV-17
```

```

piece
handle=+FRA/ORCL/AUTOBACKUP/2017_11_06/s_959332229.321.959332231
comment=NONE
Finished Control File and SPFILE Autobackup at 06-NOV-17

RMAN>

```

- While the backup is in progress, rerun the query on the second terminal window to monitor the RMAN backup session progress within the cluster. The backup should be done in parallel, with work distributed to two leaf nodes of the cluster. Enter the slash (/) symbol and press the Enter key to rerun the query. It may be necessary to do this multiple times until the output appears. When the backup finishes, exit SQL*Plus.

Note: You may not be able to see the output because it takes about **a minute** to complete the full database backup.

```

SQL> /
no rows selected

SQL> /
INST_ID      SID      SERIAL#      CONTEXT      SOFAR      TOTALWORK  %_COMPLETE
-----  -----
1          275      58586           1      79614      88320    90.14
2          38       32484           1      76926     103040    74.66

SQL> /
no rows selected

SQL> exit

[oracle@host01 ~]$

```

- Shut down the database using `srvctl` so the ARCHIVELOG mode can be disabled for your RAC database.

```
[oracle@host01 ~]$ srvctl stop database -db orcl
[oracle@host01 ~]$
```

- Make a local connection using operating system authentication to the local database instance, and then start up the database on only the first node with the `mount` option. Disable archivelog mode with the `alter database noarchivelog` statement. Confirm this operation with the `archive log list` statement. Shut down the database and exit SQL*Plus when finished.

```
[oracle@host01 ~]$ sqlplus / as sysdba

SQL*Plus: Release 12.2.0.1.0 Production on Mon Nov 6 09:21:02
2017

Copyright (c) 1982, 2016, Oracle. All rights reserved.

Connected to an idle instance.

SQL> startup mount
ORACLE instance started.

Total System Global Area 2399141888 bytes
Fixed Size                  8623496 bytes
Variable Size                721422968 bytes
Database Buffers            1660944384 bytes
Redo Buffers                 8151040 bytes
Database mounted.

SQL> alter database noarchivelog;

Database altered.

SQL> archive log list
Database log mode           No Archive Mode
Automatic archival          Disabled
Archive destination          USE_DB_RECOVERY_FILE_DEST
Oldest online log sequence   11
Current log sequence         13

SQL> shutdown immediate;
ORA-01109: database not open

Database dismounted.
ORACLE instance shut down.

SQL> exit
Disconnected from Oracle Database 12c Enterprise Edition Release
12.2.0.1.0 - 64bit Production
[oracle@host01 ~]$
```

10. Use `srvctl` to re-start your database. Ensure that all instances are up.

```
[oracle@host01 ~]$ srvctl start database -db orcl

[oracle@host01 ~]$ srvctl status database -db orcl
Instance orcl_1 is running on node host04
Instance orcl_2 is running on node host05
Instance orcl_3 is running on node host02
Instance orcl_4 is running on node host03
Instance orcl_5 is running on node host01

[oracle@host01 ~]$
```

11. Close all terminal windows opened for this practice.

Practices for Lesson 7: Global Resource Management Concepts

Practices for Lesson 7

There are no practices for this lesson.

Practices for Lesson 8: RAC Database Monitoring and Tuning

Practices for Lesson 8: Overview

Overview

The goal of this practice is to show you how to manually discover performance issues by using the EM Express performance pages as well as ADDM. **This first part generates a workload that uses a bad RAC application design.**

The goal of this practice is to show you how to manually discover performance issues by using the EM Express performance pages as well as ADDM. **In this second part of the practice, you are going to correct the previously found issue by creating a sequence number instead of by using a table.**

The goal of this practice is to show you how to manually discover performance issues by using the EM Express performance pages as well as ADDM. **This last part generates the same workload as in the previous practice, but uses more cache entries for sequence number S.**

Practice 8-1: ADDM and RAC Part I

Overview

The goal of this practice is to show you how to manually discover performance issues by using the EM Express performance pages as well as ADDM. This first part generates a workload that uses a bad RAC application design.

Note that all the necessary scripts for this practice are located in the `/stage/RAC/labs/less_08` directory.

1. Connect to your first node as the `oracle` user. Be sure to use the `-X` option. Set up your environment variables by using the `oraenv` script. Determine that the Oracle instance is running on `host01`. You will need this information throughout these practices.

```
[oracle@dns ~]$ ssh -X oracle@host01
oracle@host01's password:

[oracle@host01 ~]$ . oraenv
ORACLE_SID = [oracle] ? orcl
The Oracle base has been set to /u01/app/oracle

[oracle@host01 ~]$ ps -ef|grep ora_smon
oracle    18660  3459  0 17:25 pts/3      00:00:00 grep ora_smon
oracle    31630      1  0 Jan13 ?          00:01:06 ora_smon_orcl_5

[oracle@host01 ~]$
```

2. Execute the `setupseq1.sh` script from the `/stage/RAC/labs/less_08` directory to set up the necessary configuration for this practice.

```
[oracle@host01 ~]$ cd /stage/RAC/labs/less_08

[oracle@host01 less_08]$ ./setupseq1.sh

PL/SQL procedure successfully completed.

PL/SQL procedure successfully completed.

drop user jmw cascade
*
ERROR at line 1:
ORA-01918: user 'JMW' does not exist
```

```
drop tablespace seq including contents and datafiles
*
ERROR at line 1:
ORA-00959: tablespace 'SEQ' does not exist
```

Tablespace created.

User created.

Grant succeeded.

```
drop sequence s
*
ERROR at line 1:
ORA-02289: sequence does not exist
```

```
drop table s purge
*
ERROR at line 1:
ORA-00942: table or view does not exist
```

```
drop table t purge
*
ERROR at line 1:
ORA-00942: table or view does not exist
```

Table created.

Table created.

Index created.

```
1 row created.

Commit complete.

PL/SQL procedure successfully completed.

[oracle@host01 less_08]$
```

3. Login as SYSDBA to set the EM Database Express port to 5501.

```
[oracle@host01 less_08]$ srvctl status database -db orcl
Instance orcl_1 is running on node host04
Instance orcl_2 is running on node host05
Instance orcl_3 is running on node host03
Instance orcl_4 is running on node host02
Instance orcl_5 is running on node host01

[oracle@host01 less_08]$ export ORACLE_SID=orcl_5

[oracle@host01 less_08]$ sqlplus / as sysdba

SQL*Plus: Release 12.2.0.1.0 Production on Tue Nov 7 09:16:03
2017

Copyright (c) 1982, 2016, Oracle. All rights reserved.

Connected to:
Oracle Database 12c Enterprise Edition Release 12.2.0.1.0 -
64bit Production

SQL> exec dbms_xdb_config.sethttpport(5501);

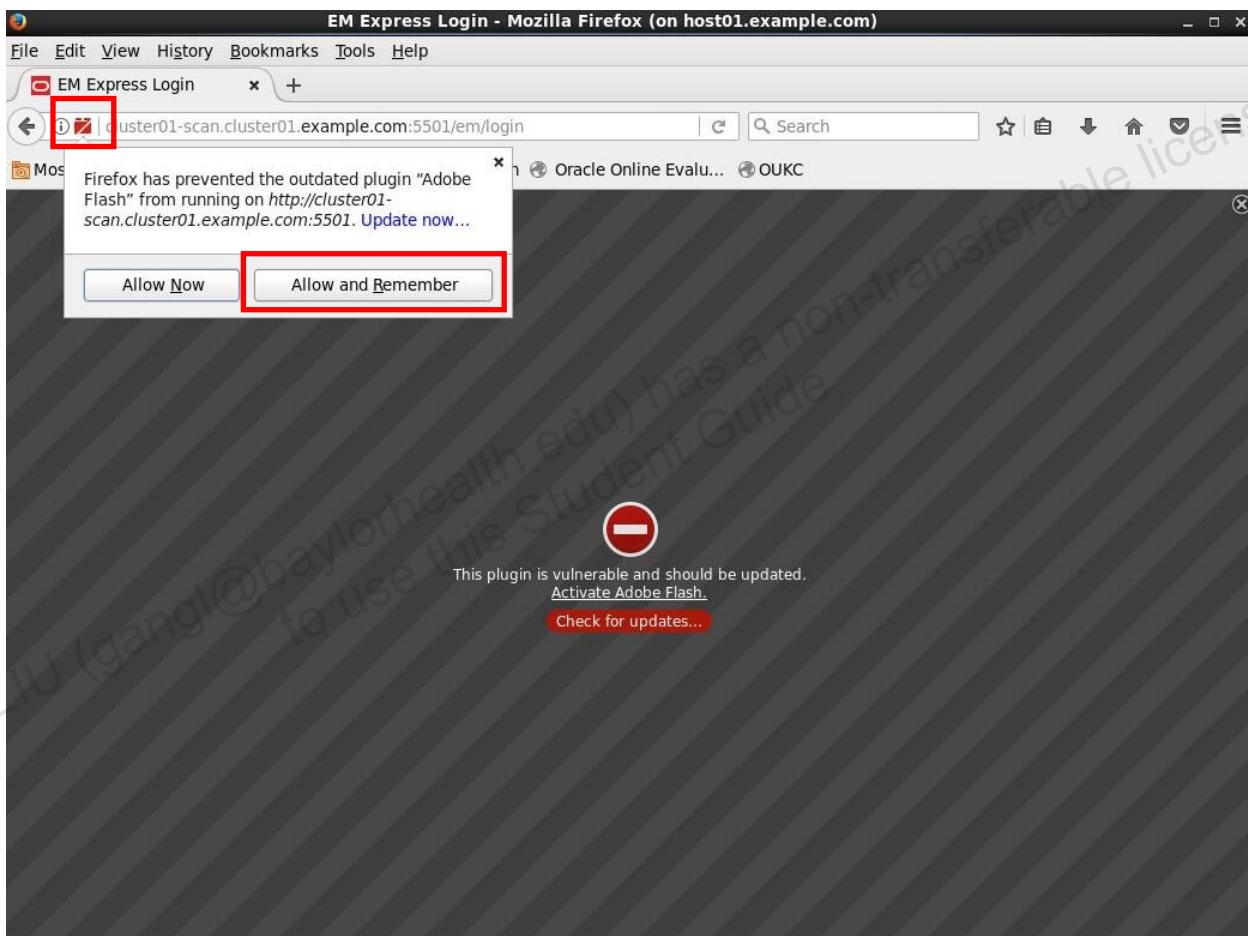
PL/SQL procedure successfully completed.

SQL> exit
Disconnected from Oracle Database 12c Enterprise Edition Release
12.2.0.1.0 - 64bit Production

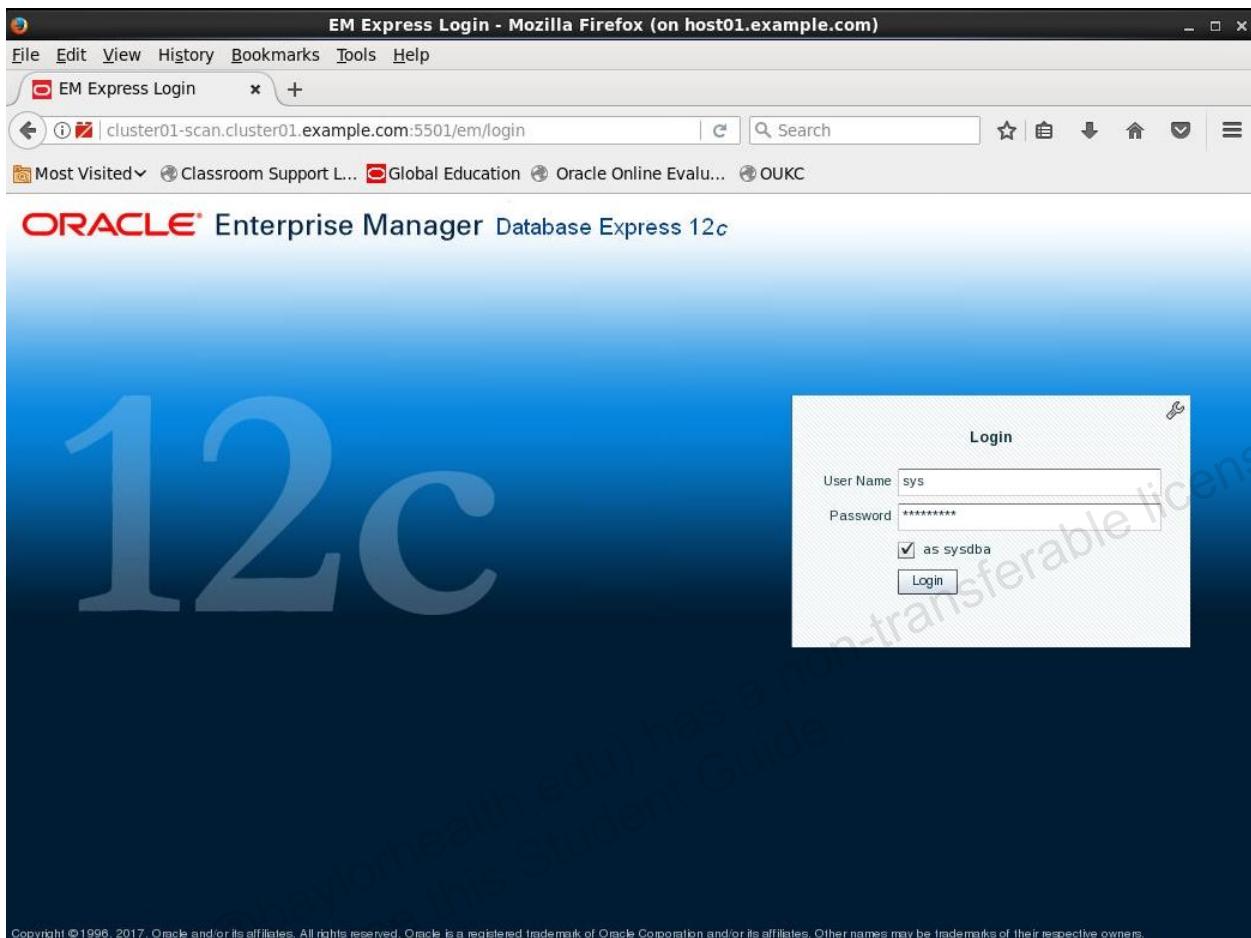
[oracle@host01 less_08]$
```

4. Start a browser and enter the following address: **http://cluster01-scan.cluster01.example.com:5501/em**. If you don't see the EM login page due to the plugin issue,
 - Click a small red icon in the location bar.
 - Click the Allow and Remember button

```
[oracle@host01 less_08]$ firefox&
[1] 32419
[oracle@host01 less_08]$
```



5. Log in to EM Express as `sys/<sys_password>`. Make sure the connection is as SYSDBA.



6. Use PL/SQL to create a new AWR snapshot.

```
[oracle@host01 less_08]$ ./create_snapshot.sh
PL/SQL procedure successfully completed.

[oracle@host01 less_08]$
```

7. Open a second terminal to `host01` as the `oracle` user. Change the directory to `/stage/RAC/labs/less_08`. Execute the `lockinfo.sh` script. This script allows you to view global lock contention issues. Your output should show no transactions for JMW.

```
[oracle@dns ~]$ ssh oracle@host01
oracle@host01's password:

[oracle@host01 ~]$ cd /stage/RAC/labs/less_08
[oracle@host01 less_08]$ ./lockinfo.sh
```

```

SQL*Plus: Release 12.2.0.1.0 Production on Wed Nov 8 08:46:47
2017

Copyright (c) 1982, 2016, Oracle. All rights reserved.

Connected to:
Oracle Database 12c Enterprise Edition Release 12.2.0.1.0 -
64bit Production

SQL>
Sess          Op Sys          OBJ NAME or
ID  USERNAME User ID TERMINAL TRANS_ID      TY Lock Mode   Req Mode
-----
 33  SYS        oracle       Trans-4      PS Share
 42  SYS        oracle       ORA$BASE    AE Share
 42  SYS        oracle       Trans-4      PS Share
 42  SYS        oracle       Trans-1      PS Share
 42  SYS        oracle       Trans-2      PS Share
 42  SYS        oracle       Trans-5      PS Share
 42  SYS        oracle       Trans-3      PS Share
 255 SYS        oracle       Trans-5      PS Share
 257 SYS        oracle       Trans-1      PS Share
 271 SYS        oracle       ORA$BASE    AE Share
 274 SYSRAC    oracle       Trans-2      PS Share
 274 SYS        oracle       Trans-2      PS Share
 282 SYS        oracle       Trans-3      PS Share

13 rows selected.

SQL> Disconnected from Oracle Database 12c Enterprise Edition
Release 12.2.0.1.0 - 64bit Production
[oracle@host01 less_08]$

```

- From the first oracle terminal, execute the startseq1.sh script to generate a workload on host01 and host02. **Do not wait; proceed with the next step.**

```

[oracle@host01 less_08]$ ./startseq1.sh

INSTANCE_NAME
-----
orcl_5

INSTANCE_NAME
-----
orcl_4

[oracle@host01 less_08]$

```

9. From the second terminal, execute the `lockinfo.sh` script again to view information regarding possible lock contention. You can also select Blocking Sessions from the Cluster Database Performance pull-down menu. You may have to refresh several times to see a lock contention for transactions belonging to JMW.

```
[oracle@host01 less_08]$ ./lockinfo.sh

SQL*Plus: Release 12.2.0.1.0 Production on Wed Nov 8 08:46:47
2017

Copyright (c) 1982, 2016, Oracle. All rights reserved.

Connected to:
Oracle Database 12c Enterprise Edition Release 12.2.0.1.0 -
64bit Production

SQL>
Sess          Op Sys          OBJ NAME or
ID  USERNAME User ID TERMINAL TRANS_ID      TY Lock Mode  Req Mode
----  -----
42  SYS        oracle        ORA$BASE      AE Share
42  SYS        oracle        Trans-3       PS Share
42  SYS        oracle        Trans-1       PS Share
42  SYS        oracle        Trans-5       PS Share
42  SYS        oracle        Trans-2       PS Share
42  SYS        oracle        Trans-4       PS Share
49  SYS        oracle        Trans-4       PS Share
51  JMW        oracle        S             TM --Waiting-- Exclusive
51  JMW        oracle        ORA$BASE      AE Share
51  JMW        oracle        Trans-196608 TX Exclusive
255  SYS       oracle        Trans-5       PS Share
256  SYS       oracle        Trans-1       PS Share
264  SYSRAC    oracle        T             TM Row Excl
264  JMW        oracle        T             TM Row Excl
264  SYSRAC    oracle        ORA$BASE      AE Share
264  SYSRAC    oracle        S             TM Exclusive
264  JMW        oracle        S             TM Exclusive
264  JMW        oracle        ORA$BASE      AE Share
271  SYS        ORA$BASE      AE Share
274  SYS        oracle        Trans-2       PS Share
274  SYSRAC    oracle        Trans-2       PS Share
282  SYS        oracle        Trans-3       PS Share

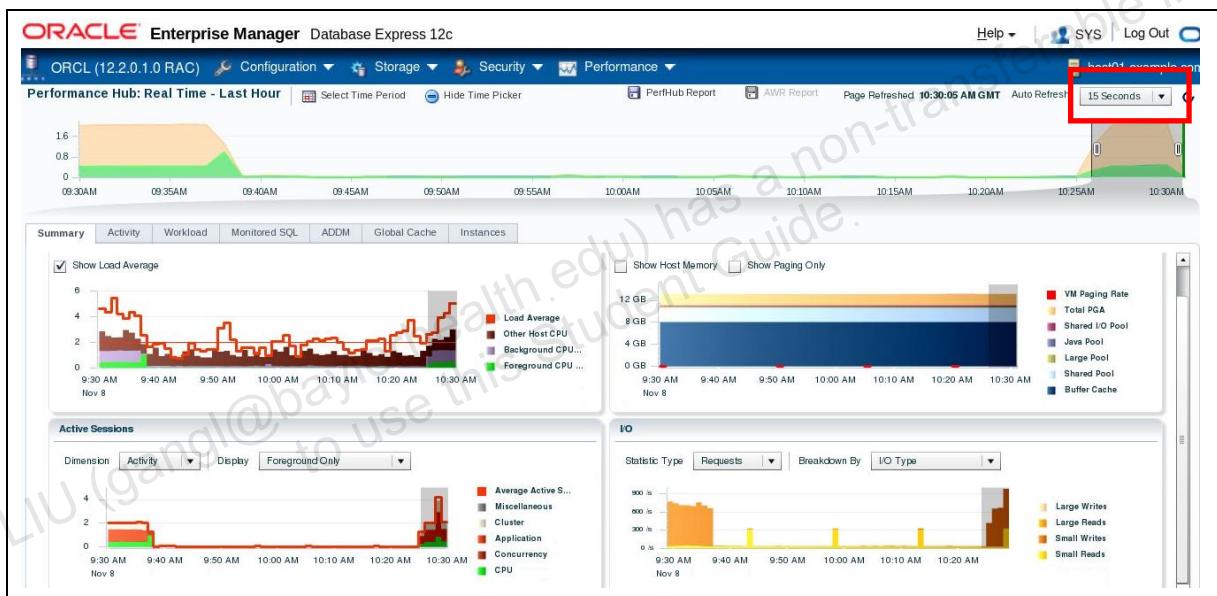
22 rows selected.

SQL> Disconnected from Oracle Database 12c Enterprise Edition
Release 12.2.0.1.0 - 64bit Production
[oracle@host01 less_08]$
```

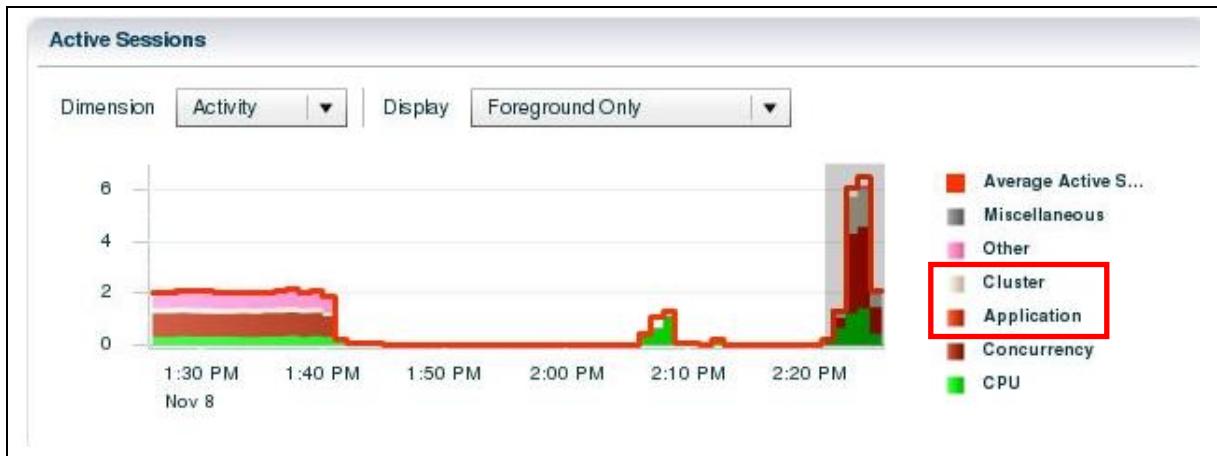
10. While the scripts are still executing, look at the Average Active Sessions graphic. Then, drill down to the Cluster wait class for the first node. What are your conclusions?
- Click Performance, and then Performance Hub.



- From there you can now see the Active Sessions graph in the bottom left corner. Make sure that Auto Refresh is set to 15 Seconds.



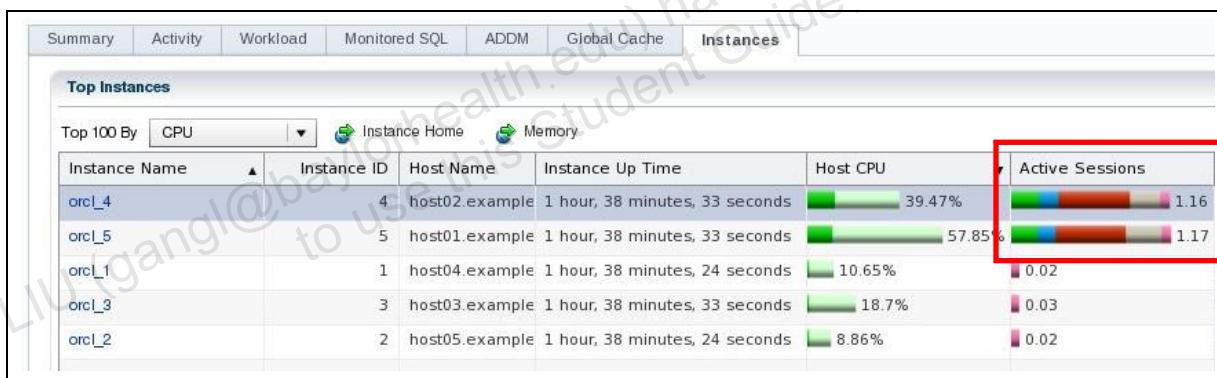
- After a few seconds, the graphic should clearly show that the **Cluster and Application wait classes** are causing most waits.



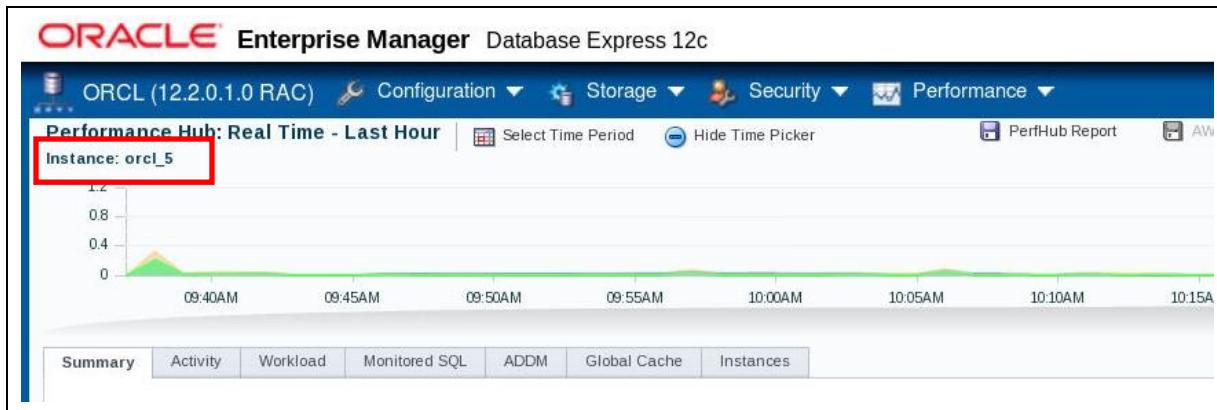
- d. Click the Workload tab. In the Workload Profile graph, place the cursor over the Transactions line at the current time on the timeline. Transactions should be around 120-180 per second.



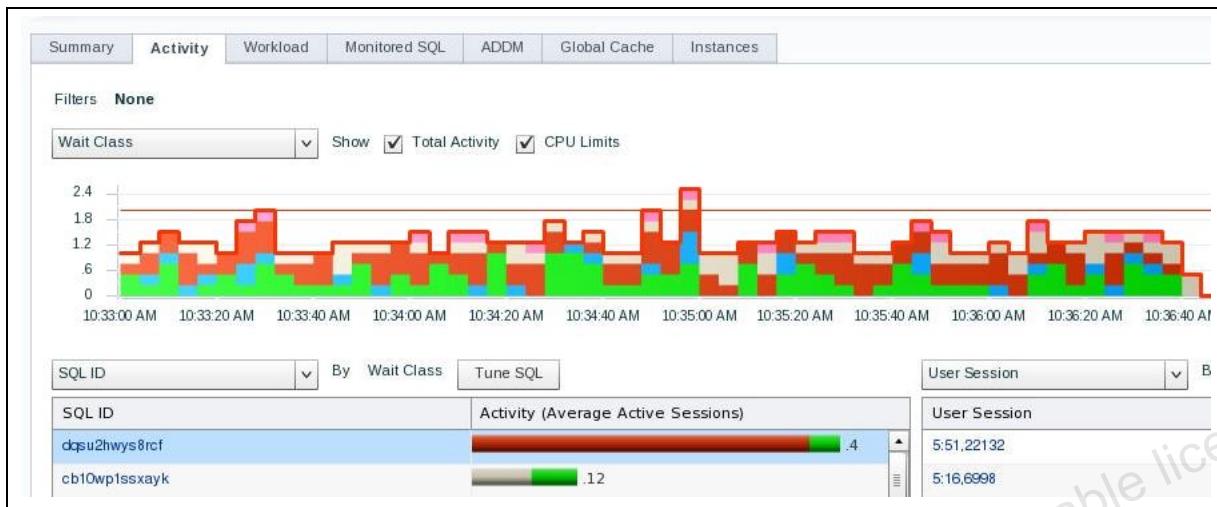
- e. Click the Instances tab and scroll down to the Top Instances section. Looking at the Active Sessions column, you should see that the number of active sessions on host01 and host02 are similar.



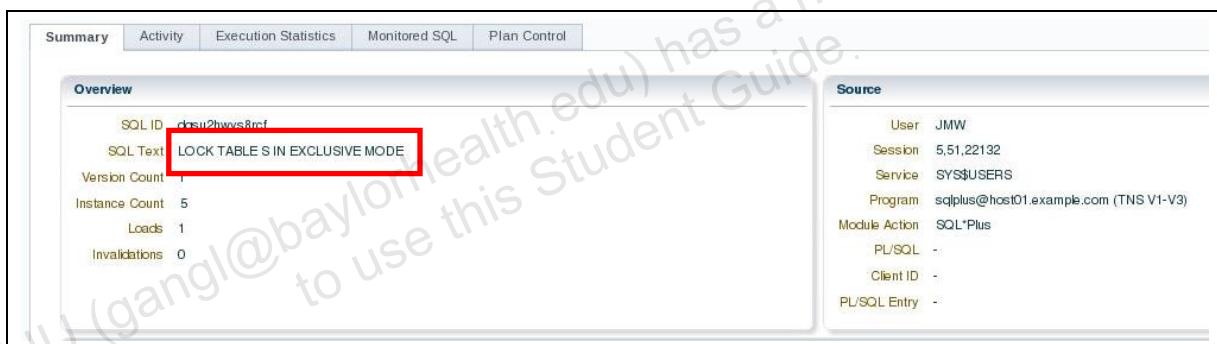
- f. Click the instance running on host01 (orcl_5 in this example). This takes you to the Performance Hub for the orcl_5 page.



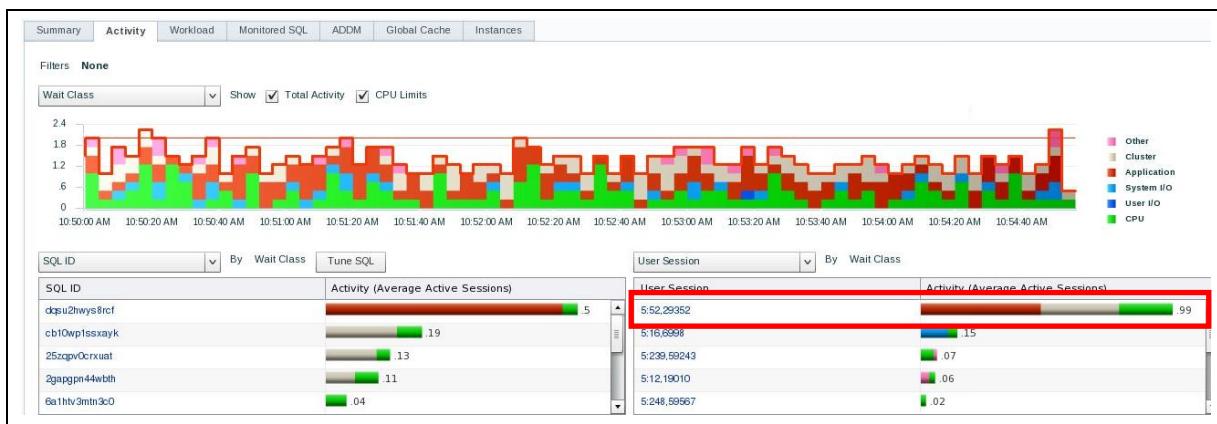
- g. Click the Activity tab, look under the SQL ID column, and then click the SQL identifier that uses most of the resource



- h. This takes you to the SQL Details page for the corresponding statement. The SQL text should be LOCK TABLE S IN EXCLUSIVE MODE, and the user should be JMW.



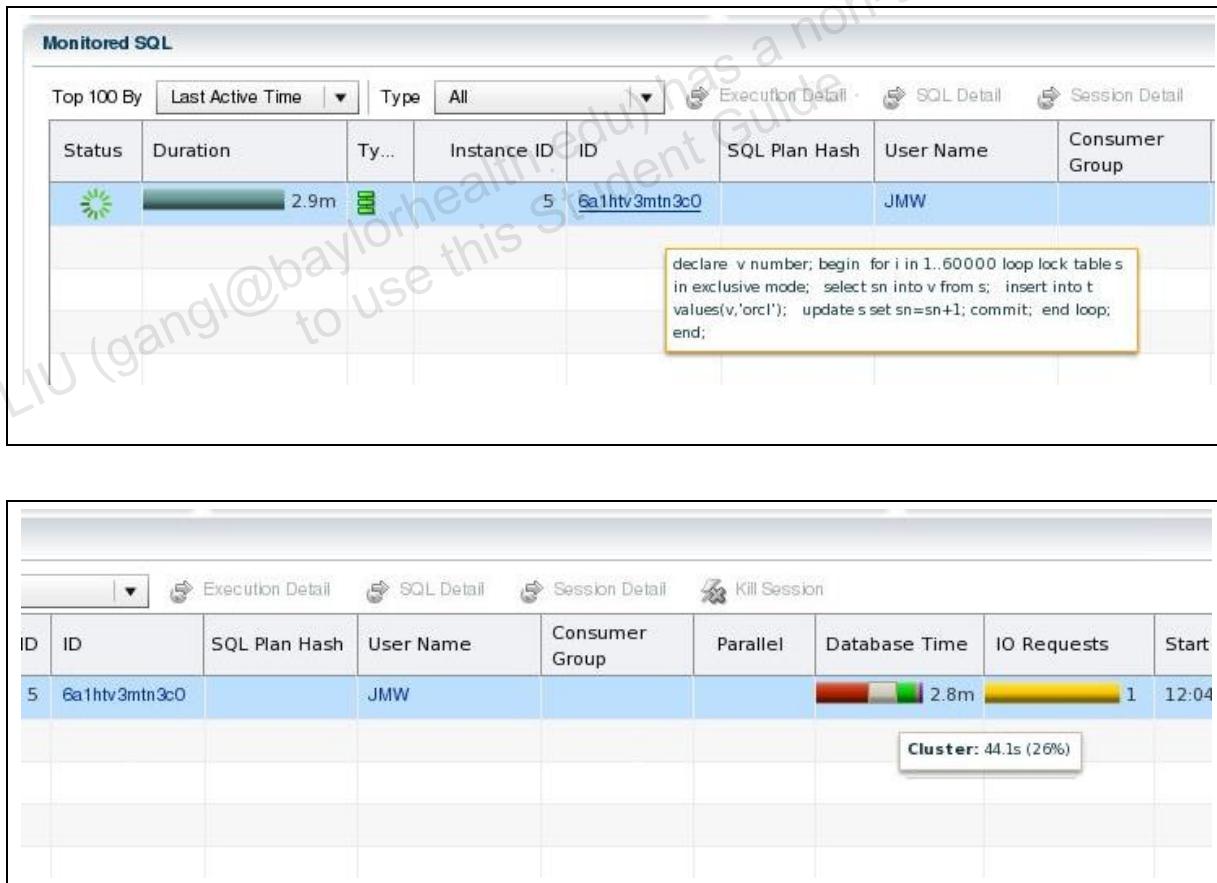
- i. Click the Back button in the web browser to return to the Performance Hub for the orcl_5 page. Under the User Session column, click the User Session ID with the highest activity.



- j. The User Name should be JMW. Beneath the Monitored SQL section, drill down on the SQL ID under the ID column.

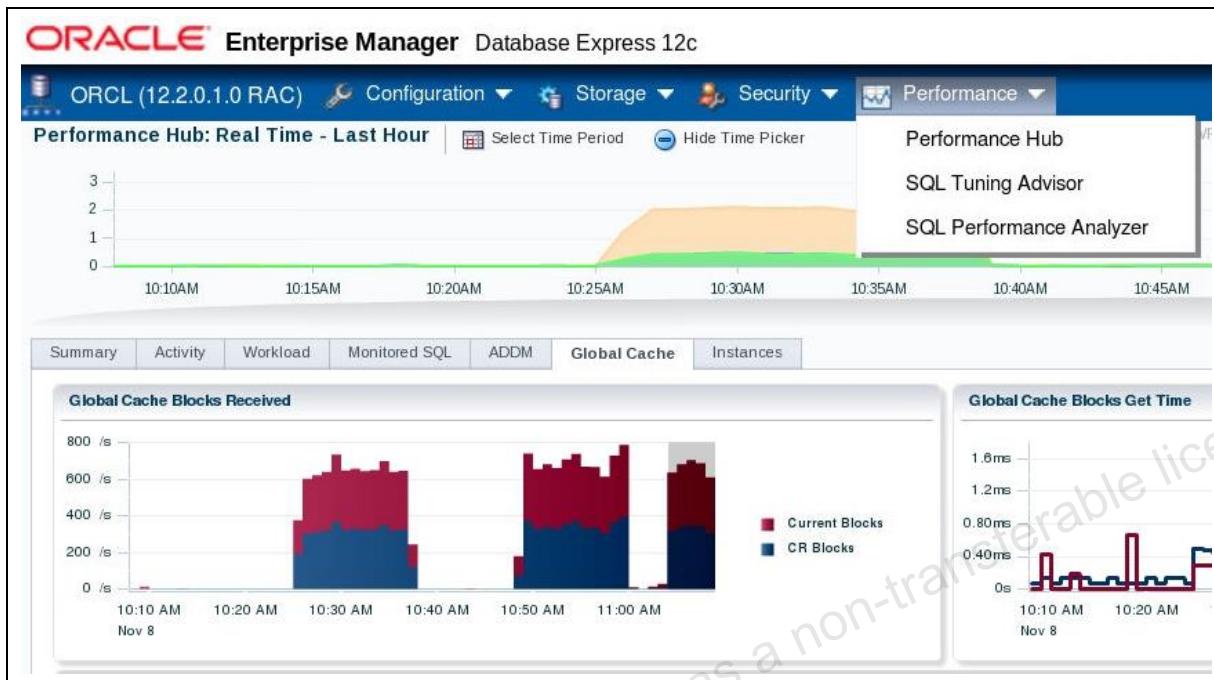


- k. Place the cursor over the SQL Text to see the entire SQL statement. You will see that the script running on the first instance is executing a SELECT/UPDATE statement on table S that causes most of the Cluster waits.



By using EM Express, you can quickly identify the top waiting SQL statements and the top waiting sessions on both instances. Here it appears that a SELECT or UPDATE statement on table S is causing most of the waits for the Cluster wait class.

- Using EM Express, Click Performance, then Performance Hub, and then click the Global Cache tab. Look at the Global Cache Blocks Received graph. What are your conclusions?



- The Global Cache Blocks Received graph clearly shows that there are a lot of blocks transferred per second on the system. You should see cumulative transfer rates (both CR and current blocks) in excess of 500 blocks per second.



- After the workload finishes, use PL/SQL to create a new AWR snapshot. If the workload is still running, press CTRL + C and hit Enter.

```
INSTANCE_NAME
```

```
-----
```

```
orcl_5
```

```

INSTANCE_NAME
-----
orcl_4

***** If the workload is still running, press CTRL + C *****
^C

PL/SQL procedure successfully completed.

PL/SQL procedure successfully completed. → Hit Enter

[oracle@host01 less_08]$ ./create_snapshot.sh

PL/SQL procedure successfully completed.

[oracle@host01 less_08]$

```

14. Using EM Express, review the latest ADDM run. What are your conclusions?
- Click Performance, then Performance Hub, and then click the ADDM tab.
 - Find the link for the latest ADDM task under ADDM Tasks and click it. This takes you to the Automatic Database Diagnostic Monitor (ADDM) page.

Task Name	Task ID	Time	Task Analysis Type
ADDM:1487037122_31	181	Wed Nov 8, 2017 11:15:48 AM	DATABASE
ADDM:1487037122_30	179	Wed Nov 8, 2017 11:02:23 AM	DATABASE
ADDM:1487037122_29	177	Wed Nov 8, 2017 10:47:36 AM	DATABASE

- c. On the Automatic Database Diagnostic Monitor (ADDM) page, the ADDM Tasks section shows you the consolidation of ADDM reports from all instances running in your cluster. This is your first entry point before drilling down to specific instances. From there, investigate the Top SQL Statements, Table Locks, and Global Cache Messaging findings.

The screenshot shows the ADDM Findings page with the following data:

Finding	Impact	Number of Recommendations
Top SQL Statements	87%	5
Table Locks	47.5%	1
Global Cache Messaging	22.4%	1

- d. Click the Top SQL Statements finding, which affects all instances, revealing LOCK TABLE, SELECT, and UPDATE commands on table JMW.S as possible problems to investigate. Click the Back button to return to the ADDM report.

The screenshot shows the Performance Finding Details for the Top SQL Statements finding. It includes the following information:

- Finding:** SQL statements consuming significant database time were found. These statements offer a good opportunity for performance improvement.
- Impact (Active Sessions):** 1.68
- Percentage of Finding's Impact (%):** 87%
- Period Start Time:** Wed Nov 8, 2017 11:02:23 AM
- End Time:** Wed Nov 8, 2017 11:15:48 AM
- Actions:** SQL Tuning
- Rationales:**
 - Investigate the LOCK TABLE statement with SQL_ID "dqsu2hwys8rcf" for possible performance improvements. You can supplement the information given here with an ASH report for this SQL_ID.
 - The SQL Tuning Advisor cannot operate on LOCK TABLE statements.
 - Database time for this SQL was divided as follows: 100% for SQL execution, 0% for PL/SQL execution and 0% for Java execution.
 - SQL statement with SQL_ID "dqsu2hwys8rcf" was executed 120000 times and had an elapsed time of 0.0066 seconds.
 - Waiting for event "enq: TM - contention" in wait class "Application" accounted for database time spent in processing the SQL statement with SQL_ID "dqsu2hwys8rcf".

- e. Click the Table Locks finding, which affects all instances, revealing that you should investigate your application logic regarding the JMW.S object.

The screenshot shows the Performance Finding Details for the Table Locks finding. It includes the following information:

- Finding:** Contention on table lock waits was consuming significant database time.
- Impact (Active Sessions):** 0.92
- Percentage of Finding's Impact (%):** 47.5%
- Period Start Time:** Wed Nov 8, 2017 11:02:23 AM
- End Time:** Wed Nov 8, 2017 11:15:48 AM
- Actions:** Application Analysis
- Rationales:**
 - Investigate application logic involving DDL and DML on provided blocked objects.
 - The TABLE "JMW.S" with object ID "74062" was locked for a significant time.

- f. Click the Back button to return to the ADDM report. Click the Global Cache Messaging finding revealing either the UPDATE S or SELECT command as responsible for significant amount of Cluster waits during the analysis period. Click the Back button.

Category	Actions	Rationales
Application Analysis	<ul style="list-style-type: none"> Look at the "Top SQL Statements" finding for SQL statements consuming significant time on Cluster waits. For example, the UPDATE statement with SQL_ID "cb10wp1ssxayk" is responsible for 35% of Cluster wait during the analysis period. 	

- g. You now have the possibility to drill down to each instance using the links located in the DB Time Breakdown By Instance table. Click the link corresponding to the most affected instance.

Task Name	Instance Name	Impact
ADDM:1487037122_5_31	orcl_5	50.7%
ADDM:1487037122_4_31	orcl_4	47.9%
ADDM:1487037122_3_31	orcl_3	1.4%

- h. On the corresponding Automatic Database Diagnostic Monitor page, you should retrieve similar top findings seen previously at the cluster level.

15. Keep all terminal windows for the next practices.

Practice 8-2: ADDM and RAC Part II

Overview

The goal of this practice is to show you how to manually discover performance issues by using the EM Express performance pages as well as ADDM. In this second part of the practice, you are going to correct the previously found issue by creating a sequence number instead of by using a table.

Note that all the necessary scripts for this practice are located in the `/stage/RAC/labs/less_08` directory.

1. Execute the `setupseq2.sh` script to create the necessary objects used for the rest of this practice.

```
[oracle@host01 less_08]$ ./setupseq2.sh

PL/SQL procedure successfully completed.

PL/SQL procedure successfully completed.

User dropped.

Tablespace dropped.

Tablespace created.

User created.

Grant succeeded.

drop table s purge
*
ERROR at line 1:
ORA-00942: table or view does not exist

drop sequence s
*
```

```
ERROR at line 1:  
ORA-02289: sequence does not exist  
  
drop table t purge  
*  
ERROR at line 1:  
ORA-00942: table or view does not exist  
  
Table created.  
  
Index created.  
  
Sequence created.  
  
PL/SQL procedure successfully completed.  
[oracle@host01 less_08]$
```

2. Use PL/SQL to create a new AWR snapshot.

```
[oracle@host01 less_08]$ ./create_snapshot.sh  
  
PL/SQL procedure successfully completed.  
[oracle@host01 less_08]$
```

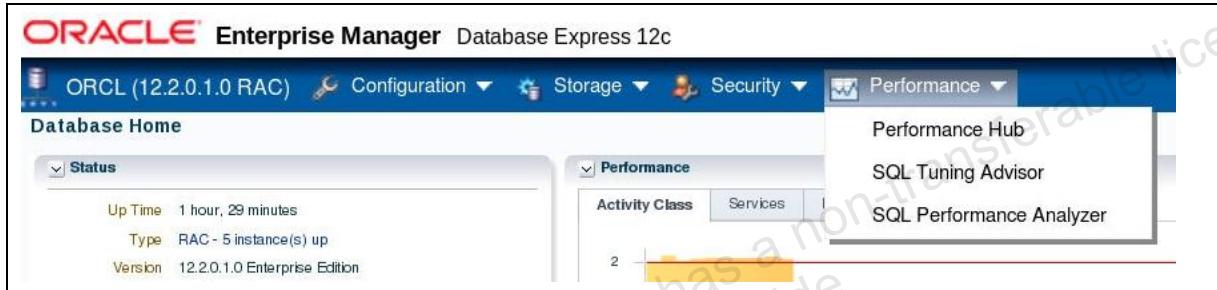
3. Execute the `startseq2.sh` script to generate a workload on the instances running on host01 and host02. **Do not wait; proceed with the next step.**

```
[oracle@host01 less_08]$ ./startseq2.sh  
[oracle@host01 less_08]$
```

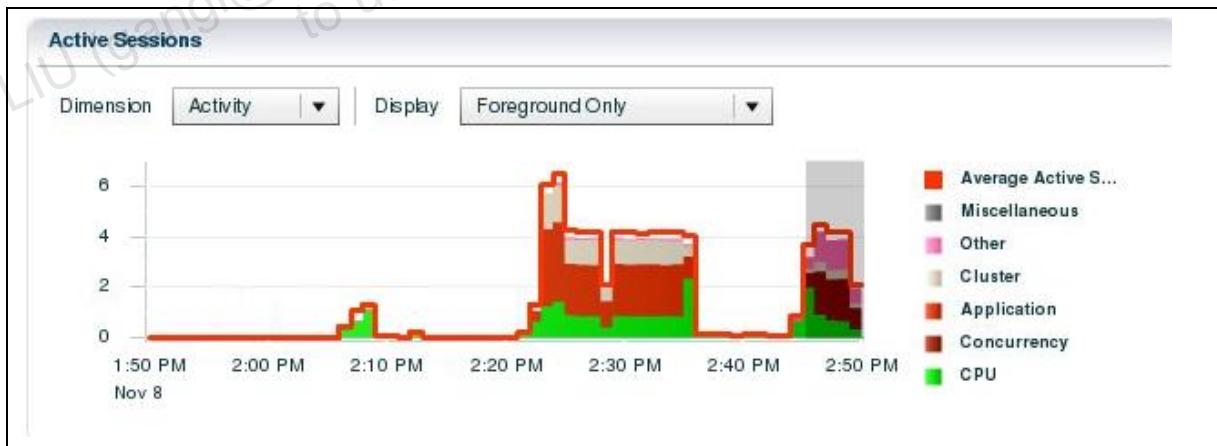
... Do not wait after this point and go to the next step.

4. While the scripts are still executing, look at the Active Sessions graphic. View the workload. What are your conclusions?

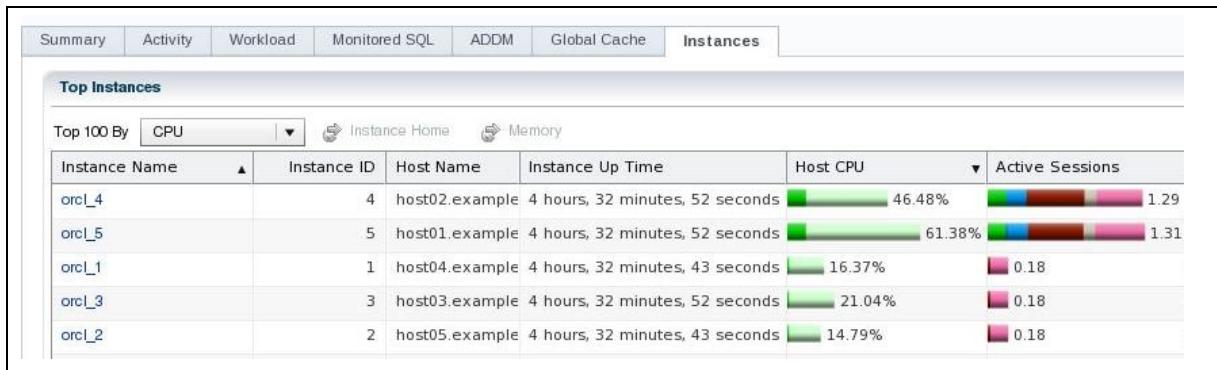
- a. Click Performance, and then Performance Hub.



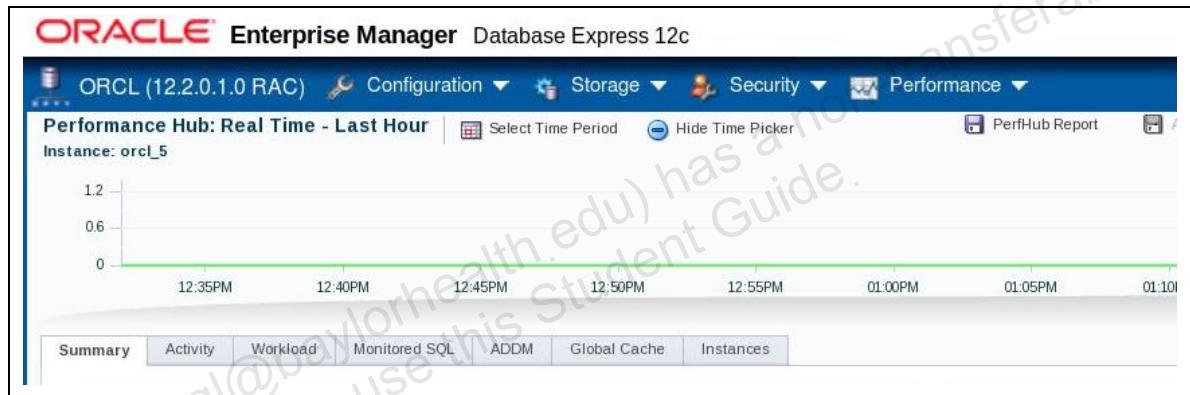
- b. From there you can now see the Average Active Sessions graph. Make sure 15 Seconds Refresh is selected from the Auto Refresh drop-down list. After a few seconds, the graphic will show that the **Cluster and Concurrency wait classes** are causing most waits.



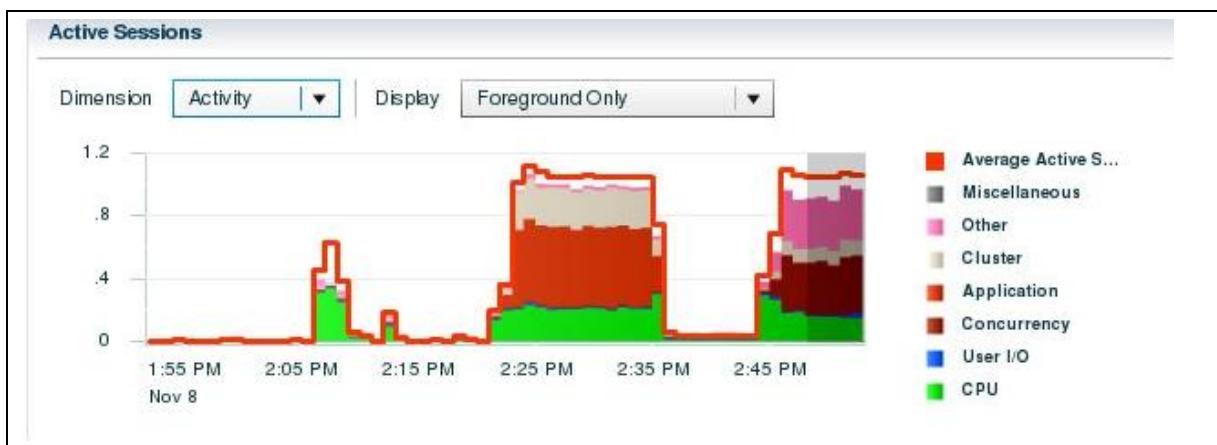
- c. Click the Instances tab and scroll down to the Top Instances section. You should see that the number of active sessions is similar on host01 and host02.



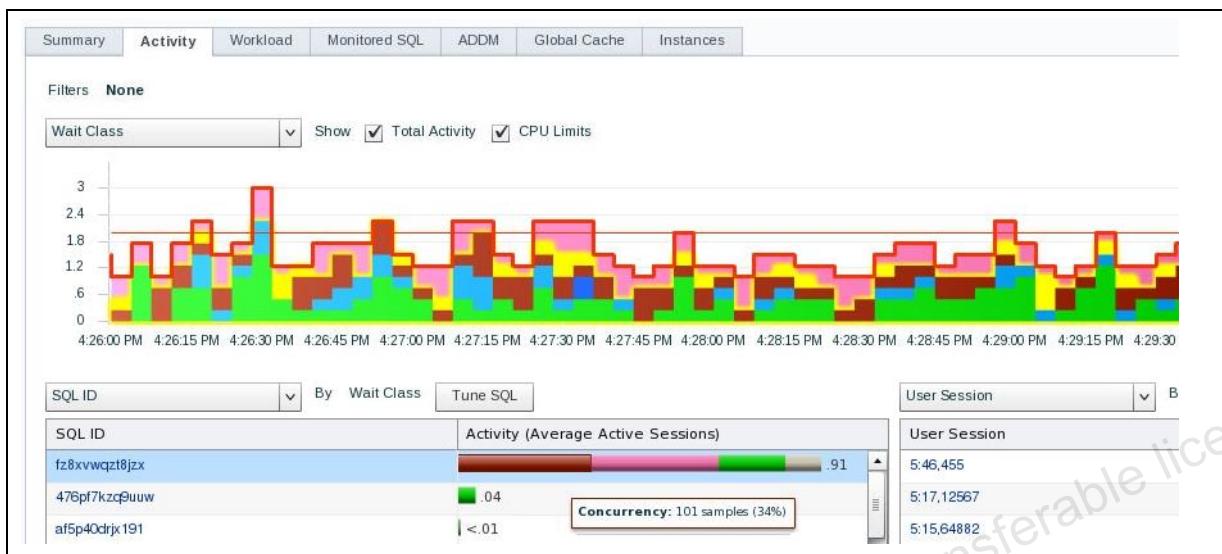
- d. Click the first instance's link, orcl_5 in this case (on host01). This takes you to the Performance Hub for the orcl_5 page.



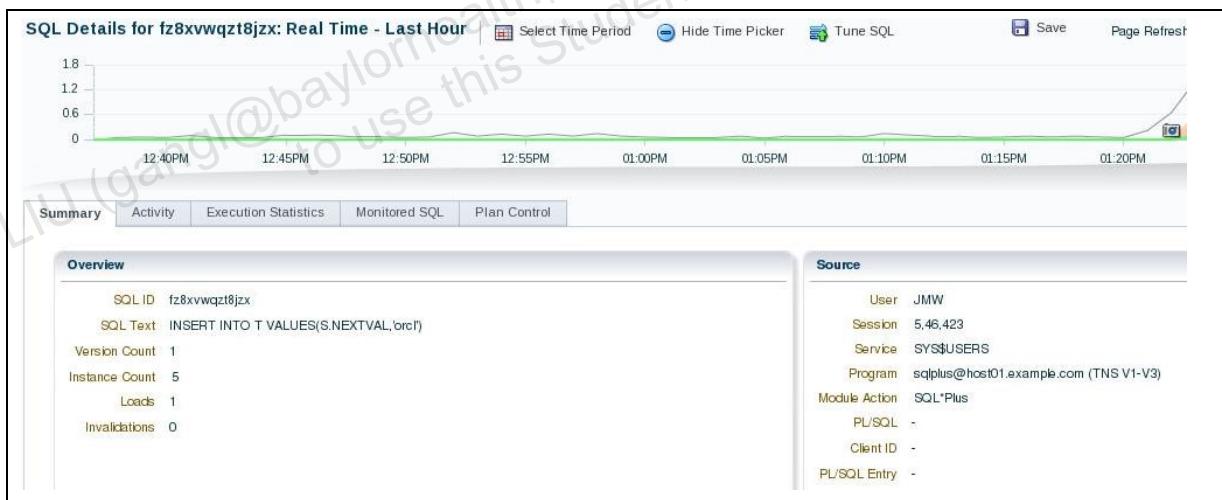
- e. On the Active Sessions graph, you can see the most significant wait events causing most of the waits in the **Cluster and Concurrency wait class** on the first instance (orcl_5 in this example).



f. Click the Activity tab.



- g. Click the SQL identifier under the SQL ID column that uses most of the resources. This takes you to the SQL Details page for the corresponding statement. You will see that the script running on the first instance is executing an INSERT statement on table T that causes most of the Cluster waits.



- h. Click Performance, then Performance Hub, and then the Workload tab. Make sure 15 Seconds Refresh is selected from the Auto Refresh drop-down list. In the Workload Profile graph, place the cursor over the Transactions line at the current time on the timeline. Transactions should be around 90~100 per second.



5. After the workload finishes, use PL/SQL to create a new AWR snapshot.

```
PL/SQL procedure successfully completed.  
PL/SQL procedure successfully completed. → Hit Enter  
[oracle@host01 less_08]$ ./create_snapshot.sh  
PL/SQL procedure successfully completed.  
[oracle@host01 less_08]$
```

6. Using EM Express, review the latest ADDM run. What are your conclusions?
 - a. Click Performance, then Performance Hub, and then the ADDM tab.
 - b. Find the link for the latest ADDM task under ADDM Tasks and click it. This takes you to the Automatic Database Diagnostic Monitor (ADDM) page.

Task Name	Task ID	Time
ADDM:1487037122_33	185	Wed Nov 8, 2017 1:42:42 PM
ADDM:1487037122_32	183	Wed Nov 8, 2017 1:22:16 PM

- c. On the Automatic Database Diagnostic Monitor (ADDM) page, the Findings section shows you the consolidation of ADDM reports from all instances running in your cluster. This is your first entry point before drilling down to specific instances. From there, investigate the Top SQL Statements, Sequence Usage, and Unusual “Concurrency” Wait Event findings.

Finding	Impact	Number of Recommendations
Top SQL Statements	93.3%	1
Sequence Usage	85%	1
Unusual "Concurrency" Wait Event	42.7%	4
Unusual "Other" Wait Event	28.3%	5
Global Cache Messaging	9.4%	1

- d. Click the Top SQL Statements finding. The Top SQL Statements should reveal an INSERT INTO T command using sequence S as a possible problem to investigate. Click the Back button to return to the ADDM report.

Performance Finding Details

Finding: SQL statements consuming significant database time were found. These statements offer a good opportunity for performance improvement.

Impact (Active Sessions): 1.71

Percentage of Finding's Impact (%): 93.3%

Period Start Time: Wed Nov 8, 2017 1:22:15 PM

End Time: Wed Nov 8, 2017 1:42:42 PM

Implement:

Category	Actions	Rationales
SQL Tuning	<ul style="list-style-type: none"> Investigate the INSERT statement with SQL_ID "fz8xvwqzt8jzx" for possible performance improvements. You can supplement the information given here with an ASH report for this SQL_ID. 	<ul style="list-style-type: none"> The SQL spent only 27% of its database time on CPU, I/O and CI Tuning Advisor is not applicable in this case. Look at performance improvements. Database time for this SQL was divided as follows: 100% for SQL PL/SQL execution and 0% for Java execution. SQL statement with SQL_ID "fz8xvwqzt8jzx" was executed 5001

- e. Click Sequence Usage. The Sequence Usage finding reveals that you should use a larger cache size for your hot sequences. Click the Back button to return to the ADDM report.

Performance Finding Details

Finding	Sequence cache misses were consuming significant database time.
Impact (Active Sessions)	156
Percentage of Finding's Impact (%)	 85%
Period Start Time	Wed Nov 8, 2017 1:22:15 PM
End Time	Wed Nov 8, 2017 1:42:42 PM

Implement

Category	Actions	Rationales
Application Analysis	<ul style="list-style-type: none"> Investigate application or look at top SQL to find hot sequences. Use a larger cache size for those sequences. Try avoiding the use of the ORDER setting if running RAC. 	

- f. Click Unusual “Concurrency” Wait Event. The Unusual “Concurrency” Wait Event finding asks you to investigate the cause for high “row cache lock” waits. Refer to the Oracle Database Reference for the description of this wait event. Click the Back button.

Performance Finding Details

Finding	Wait event "row cache lock" in wait class "Concurrency" was consuming significant database time.
Impact (Active Sessions)	0.78
Percentage of Finding's Impact (%)	 42.7%
Period Start Time	Wed Nov 8, 2017 1:22:15 PM
End Time	Wed Nov 8, 2017 1:42:42 PM

Implement

Category	Actions	Rationales
Application Analysis	<ul style="list-style-type: none"> Investigate the cause for high "row cache lock" waits. Refer to Oracle's "Database Reference" for the description of this wait event. Look at the "Top SQL Statements" finding for SQL statements consuming significant time on the "row cache lock" wait event. For example, the INSERT statement with SQL_ID "fz8xvwqzt8jzx" is responsible for 88% of these waits. 	
Application Analysis	<ul style="list-style-type: none"> Investigate the cause for high "row cache lock" waits in Module "SQL*Plus". 	

7. Keep all terminal windows for the next practice.

Practice 8-3: ADDM and RAC Part III

Overview

The goal of this practice is to show you how to manually discover performance issues by using the EM Express performance pages as well as ADDM. This last part generates the same workload as in the previous practice, but uses more cache entries for sequence number S.

Note that all the necessary scripts for this practice are located in the /stage/RAC/labs/less_08 directory.

1. Execute the `setupseq3.sh` script to create the necessary objects used for the rest of this practice.

```
[oracle@host01 less_08]$ ./setupseq3.sh

PL/SQL procedure successfully completed.

PL/SQL procedure successfully completed.

User dropped.

Tablespace dropped.

Tablespace created.

User created.

Grant succeeded.

drop table s purge
*
ERROR at line 1:
ORA-00942: table or view does not exist

drop sequence s
*
```

```
ERROR at line 1:  
ORA-02289: sequence does not exist  
  
drop table t purge  
*  
ERROR at line 1:  
ORA-00942: table or view does not exist  
  
Table created.  
  
Index created.  
  
Sequence created.  
  
PL/SQL procedure successfully completed.  
[oracle@host01 less_08]$
```

2. Use PL/SQL to create a new AWR snapshot.

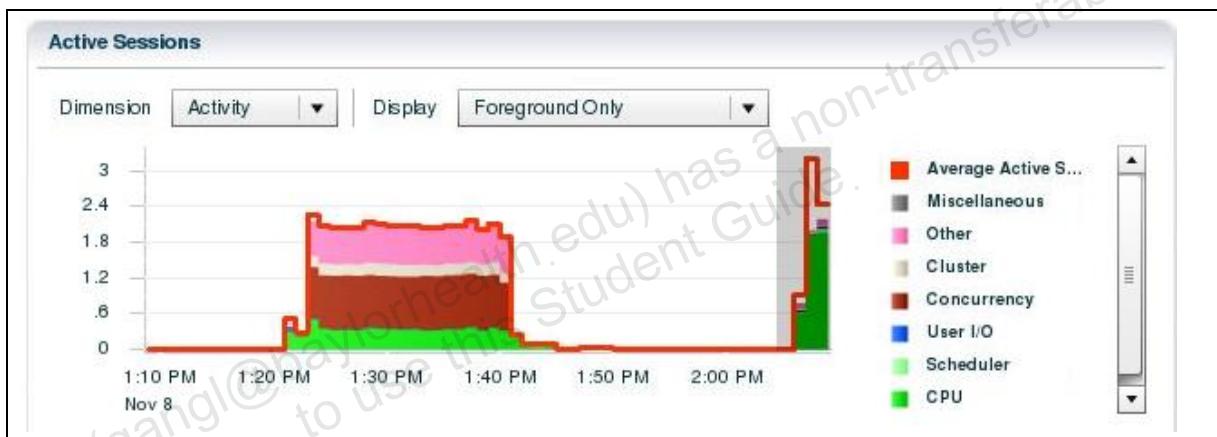
```
[oracle@host01 less_08]$ ./create_snapshot.sh  
  
PL/SQL procedure successfully completed.  
[oracle@host01 less_08]$
```

3. Execute the `startseq2.sh` script to generate the same workload on both instances of your cluster as for the previous practice. **Do not wait, and proceed with the next step.**

```
[oracle@host01 less_08]$ ./startseq2.sh
[oracle@host01 less_08]$
```

... Do not wait after this point and go to the next step.

4. Until the scripts are executed, look at the Active Sessions graphic. What are your conclusions?
- Click Performance, and then Performance Hub. Make sure 15 Seconds Refresh is selected from the Auto Refresh drop-down list. After all the scripts have finished their execution, the Active Sessions graph will clearly show that there are no significant waits on your cluster.



- Next, click the Workload tab. In the Workload Profile graph, place the cursor over the Transactions line at the current time on the timeline. After a few moments, transactions should be in excess of 1000 per second.



5. After the workload finishes, use PL/SQL to create a new AWR snapshot.

```
[oracle@host01 less_08]$ 
PL/SQL procedure successfully completed.

PL/SQL procedure successfully completed. → Hit Enter

[oracle@host01 less_08]$ ./create_snapshot.sh

PL/SQL procedure successfully completed.

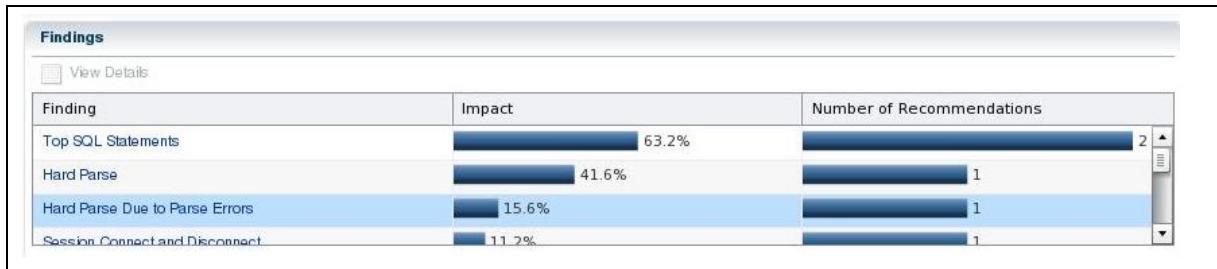
[oracle@host01 less_08]$
```

6. Using EM Express, review the latest ADDM run. What are your conclusions?

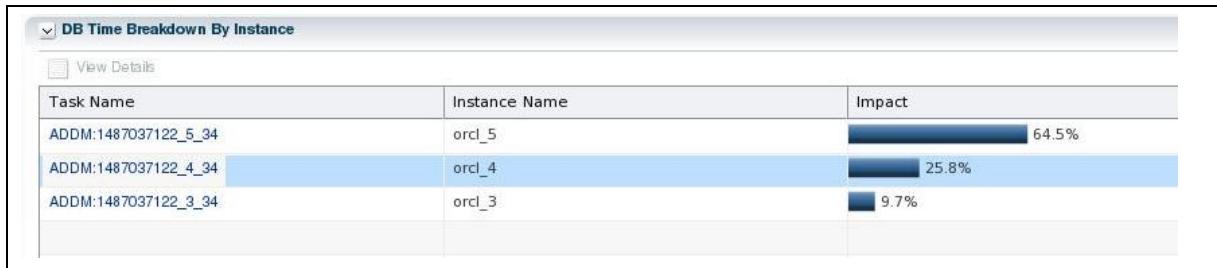
- a. Click Performance, then Performance Hub, and then the ADDM tab.

Task Name	Task ID	Time
ADDM:1487037122_34	187	Wed Nov 8, 2017 2:11:23 PM
ADDM:1487037122_33	185	Wed Nov 8, 2017 1:42:42 PM
ADDM:1487037122_32	183	Wed Nov 8, 2017 1:22:16 PM

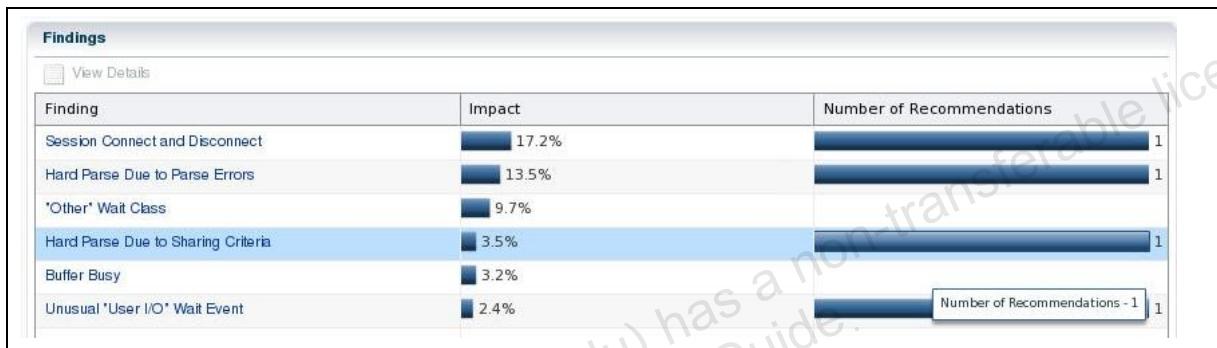
- b. Find the link for the latest ADDM task under ADDM Tasks and click it. This takes you to the Automatic Database Diagnostic Monitor (ADDM) page. On the Automatic Database Diagnostic Monitor (ADDM) page, the Findings section shows you the consolidation of ADDM reports from all instances running in your cluster.



- c. In the DB Time Breakdown By Instance section, drill down on the task link for `orcl_5`.



- d. From there, investigate the Buffer Busy findings. You should no longer see the Sequence Usage, nor specific instances impacted.



7. Close all terminal windows opened for this practice.

Practices for Lesson 9: Managing High Availability of Services

Practices for Lesson 9: Overview

Overview

In this practice, you will create a service called `prod1`. You then observe what happens to your service when you terminate the instances on which it is running.

In this practice, you will use EM Express to determine the amount of resources used by sessions executing under a particular service.

Practice 9-1: Working with Services

Overview

In this practice, you will create a service called prod1. You then observe what happens to your service when you terminate the instances on which it is running.

1. Use SRVCTL to create a singleton service called PROD1.
 - a. Connect to your first node as the `oracle` user. Set up your environment variables using the `oraenv` script. Change directory to `/stage/RAC/labs/less_09`.

```
[oracle@dns ~]$ ssh oracle@host01
oracle@host01's password:

[oracle@host01 ~]$ . oraenv
ORACLE_SID = [oracle] ? orcl
The Oracle base has been set to /u01/app/oracle

[oracle@host01 ~]$ cd /stage/RAC/labs/less_09

[oracle@host01 less_09]$
```

- b. Use SRVCTL to create a SINGLETON service called PROD1 using the `orcldb` server pool.

```
[oracle@host01 less_09]$ srvctl add service -db orcl -service
PROD1 -serverpool orcldb -cardinality singleton -policy manual

[oracle@host01 less_09]$ srvctl status service -db orcl
Service LEAFSVC is running on nodes: host04,host05
Service OLTPSVC is running on nodes: host02,host03,host01
Service PROD1 is not running.

[oracle@host01 less_09]$
```

- c. Use SRVCTL to start the PROD1 service

```
[oracle@host01 less_09]$ srvctl start service -db orcl -service
prod1
[oracle@host01 less_09]$ srvctl status service -db orcl -service
prod1
Service PROD1 is running on nodes: host01

[oracle@host01 less_09]$
```

- d. Add the following entry for the PROD1 service in the tnsnames.ora on all three hosts.

```
prod1 = (DESCRIPTION = (ADDRESS = (PROTOCOL = TCP)(HOST =
cluster01-scan.cluster01.example.com)(PORT = 1521))(LOAD_BALANCE
= YES)(CONNECT_DATA = (SERVER = DEDICATED)(SERVICE_NAME =
prod1)))
```

Use the add_tns.sh script to add the entries on all three hosts if you like.

```
[oracle@host01 less_09]$ cat add_tns.sh
cat /stage/RAC/labs/less_09/prod1_tns.txt>>
/u01/app/oracle/product/12.2.0/dbhome_1/network/admin/tnsnames.ora

ssh host02 "cat /stage/RAC/labs/less_09/prod1_tns.txt>>
/u01/app/oracle/product/12.2.0/dbhome_1/network/admin/tnsnames.ora"

ssh host03 "cat /stage/RAC/labs/less_09/prod1_tns.txt>>
/u01/app/oracle/product/12.2.0/dbhome_1/network/admin/tnsnames.ora"

echo "PROD1 tnsnames.ora entry completed for HOST01"
echo "PROD1 tnsnames.ora entry completed for HOST02"
echo "PROD1 tnsnames.ora entry completed for HOST03"

[oracle@host01 less_09]$ cat prod1_tns.txt
prod1 = (DESCRIPTION =
(ADDRESS = (PROTOCOL = TCP)(HOST = cluster01-
scan.cluster01.example.com)(PORT = 1521))
(LOAD_BALANCE = YES) (CONNECT_DATA = (SERVER =
DEDICATED)(SERVICE_NAME = prod1)))

[oracle@host01 less_09]$ ./add_tns.sh
PROD1 tnsnames.ora entry completed for HOST01
PROD1 tnsnames.ora entry completed for HOST02
PROD1 tnsnames.ora entry completed for HOST03

[oracle@host01 less_09]$
```

2. Use the `srvctl` command to check the status of the new service. On which host is your service running? (It may be different from the example below)

```
[oracle@host01 less_09]$ srvctl status service -db orcl -service
prod1
Service prod1 is running on nodes: host01
[oracle@host01 ~]$
```

3. Change user to grid, set the environment and check the status of the PROD1 service using CRSCTL.

```
[oracle@host01 less_09]$ su - grid
Password:

[grid@host01 ~]$ . oraenv
ORACLE_SID = [grid] ? +ASM1
The Oracle base has been set to /u01/app/grid

[grid@host01 ~]$ crsctl stat res -t
-----
Name          Target  State       Server           State details
-----
Local Resources
-----
ora.ASMNET1LSNR_ASM.lsnr
    ONLINE  ONLINE   host01          STABLE
    ONLINE  ONLINE   host02          STABLE
    ONLINE  ONLINE   host03          STABLE
ora.DATA.dg
    ONLINE  ONLINE   host01          STABLE
    ONLINE  ONLINE   host02          STABLE
    ONLINE  ONLINE   host03          STABLE
ora.FRA.dg
    ONLINE  ONLINE   host01          STABLE
    ONLINE  ONLINE   host02          STABLE
    ONLINE  ONLINE   host03          STABLE
ora.LISTENER.lsnr
    ONLINE  ONLINE   host01          STABLE
    ONLINE  ONLINE   host02          STABLE
    ONLINE  ONLINE   host03          STABLE
    ONLINE  ONLINE   host04          STABLE
    ONLINE  ONLINE   host05          STABLE
ora.LISTENER_LEAF.lsnr
    OFFLINE OFFLINE  host04          STABLE
    OFFLINE OFFLINE  host05          STABLE
ora.MGMT.dg
    ONLINE  ONLINE   host01          STABLE
    ONLINE  ONLINE   host02          STABLE
    ONLINE  ONLINE   host03          STABLE
ora.chad
    ONLINE  ONLINE   host01          STABLE
    ONLINE  ONLINE   host02          STABLE
    ONLINE  ONLINE   host03          STABLE
    ONLINE  ONLINE   host04          STABLE
```

	ONLINE	ONLINE	host05	STABLE
ora.net1.network	ONLINE	ONLINE	host01	STABLE
	ONLINE	ONLINE	host02	STABLE
	ONLINE	ONLINE	host03	STABLE
	ONLINE	ONLINE	host04	STABLE
	ONLINE	ONLINE	host05	STABLE
ora.ons	ONLINE	ONLINE	host01	STABLE
	ONLINE	ONLINE	host02	STABLE
	ONLINE	ONLINE	host03	STABLE
ora.proxy_advm	OFFLINE	OFFLINE	host01	STABLE
	OFFLINE	OFFLINE	host02	STABLE
	OFFLINE	OFFLINE	host03	STABLE
<hr/>				
Cluster Resources				
<hr/>				
ora.LISTENER_SCAN1.lsnr	1	ONLINE	ONLINE	host03
				STABLE
ora.LISTENER_SCAN2.lsnr	1	ONLINE	ONLINE	host02
				STABLE
ora.LISTENER_SCAN3.lsnr	1	ONLINE	ONLINE	host01
				STABLE
ora.MGMTLSNR	1	ONLINE	ONLINE	host01
				169.254.116.21
				192.168.1.101
				192.168.2.101,
				STABLE
ora.asm	1	ONLINE	ONLINE	host01
	2	ONLINE	ONLINE	host03
	3	ONLINE	ONLINE	host02
ora.cvu	1	OFFLINE	OFFLINE	
				STABLE
ora.gns	1	ONLINE	ONLINE	host01
				STABLE
ora.gns.vip	1	ONLINE	ONLINE	host01
				STABLE
ora.host01.vip	1	ONLINE	ONLINE	host01
				STABLE
ora.host02.vip	1	ONLINE	ONLINE	host01
				STABLE
ora.host03.vip	1	ONLINE	ONLINE	host02
				STABLE
ora.host04.vip	1	ONLINE	ONLINE	host03
				STABLE

	1	ONLINE	ONLINE	host04	STABLE
ora.host05.vip	1	ONLINE	ONLINE	host05	STABLE
ora.mgmtdb	1	ONLINE	ONLINE	host01	Open, STABLE
ora.orcl.db	1	ONLINE	ONLINE	host04	Open, Readonly, HOME=/u01/app/oracle/product/12.2.0/dbhome_1, STABLE
	2	ONLINE	ONLINE	host05	Open, Readonly, HOME=/u01/app/oracle/product/12.2.0/dbhome_1, STABLE
	3	ONLINE	ONLINE	host03	Open, HOME=/u01/app/oracle/product/12.2.0/dbhome_1, STABLE
	4	ONLINE	ONLINE	host02	Open, HOME=/u01/app/oracle/product/12.2.0/dbhome_1, STABLE
	5	ONLINE	ONLINE	host01	Open, HOME=/u01/app/oracle/product/12.2.0/dbhome_1, STABLE
ora.orcl.leafsvc.svc					
	1	ONLINE	ONLINE	host04	STABLE
	2	ONLINE	ONLINE	host05	STABLE
ora.orcl.oltpsvc.svc					
	1	ONLINE	ONLINE	host03	STABLE
	2	ONLINE	ONLINE	host01	STABLE
	3	ONLINE	ONLINE	host02	STABLE
ora.orcl.prod1.svc					
	1	ONLINE	ONLINE	host01	STABLE
ora.qosmserver					
	1	ONLINE	ONLINE	host01	STABLE
ora.scan1.vip					
	1	ONLINE	ONLINE	host03	STABLE
ora.scan2.vip					
	1	ONLINE	ONLINE	host02	STABLE
ora.scan3.vip					
	1	ONLINE	ONLINE	host01	STABLE
<hr/>					
[grid@host01 ~]\$ exit					
logout					
[oracle@host01 less_09]\$					

4. Connect to the service and query V\$INSTANCE and determine what instance you are connected to. Check the database status to determine the host on which the instance is running.

```
[oracle@host01 less_09]$ sqlplus sys/<sys_password>@prod1 as sysdba

SQL*Plus: Release 12.2.0.1.0 Production on Tue Nov 7 09:16:03
2017

Copyright (c) 1982, 2016, Oracle. All rights reserved.

Connected to:
Oracle Database 12c Enterprise Edition Release 12.2.0.1.0 -
64bit Production

SQL> select instance_name from v$instance;

INSTANCE_NAME
-----
orcl_5

SQL> exit
Disconnected from Oracle Database 12c Enterprise Edition Release
12.2.0.1.0 - 64bit Production

[oracle@host01 less_09]$ srvctl status database -db orcl
Instance orcl_1 is running on node host04
Instance orcl_2 is running on node host05
Instance orcl_3 is running on node host03
Instance orcl_4 is running on node host02
Instance orcl_5 is running on node host01
[oracle@host01 less_09]$
```

5. From a terminal session on the node hosting the PROD1 service, as the oracle user, crash the instance on that node. In this example, the service is running on host01. Use ssh to log into the host, find the database pmon process and kill the ora_pmon_orcl_n process. Use the pkill -9 -f ora_pmon_orcl_n command to crash the database instance. The orcl_1 instance will crash and the Clusterware services will restart it very quickly. Exit back to host01.

```
[oracle@host01 less_09]$ ssh host01

[oracle@host01 ~]$ ps -ef|grep ora_pmon
```

```

oracle      4305      1  0 Sep13 ?          00:04:25 ora_pmon_orcl_5
oracle    26772 26746  0 13:47 pts/1        00:00:00 grep ora_pmon

[oracle@host01 ~]$ pkill -9 -f ora_pmon_orcl_5

[oracle@host01 ~]$ exit
logout
Connection to host01 closed.
[oracle@host01 less_09]$

```

6. Use `SRVCTL` to check the status of the `PROD1` service. Where is the service running now? In the example below, the service has been failed over to `host02`. What instance is running on the system hosting the service?

```

[oracle@host01 less_09]$ srvctl status service -db orcl -service
prod1
Service PROD1 is running on nodes: host02

[oracle@host01 less_09]$ srvctl status database -db orcl
Instance orcl_1 is running on node host04
Instance orcl_2 is running on node host05
Instance orcl_3 is running on node host03
Instance orcl_4 is running on node host02
Instance orcl_5 is running on node host01

[oracle@host01 less_09]$

```

7. Make a connection to the database using the `PROD1` service. What instance are you connected to?

```

[oracle@host01 less_09]$ sqlplus sys/<sys_password>@prod1 as
sysdba

SQL*Plus: Release 12.2.0.1.0 Production on Tue Nov 7 09:39:58
2017

Copyright (c) 1982, 2016, Oracle. All rights reserved.

Connected to:
Oracle Database 12c Enterprise Edition Release 12.2.0.1.0 -
64bit Production

SQL> select instance_name from v$instance;

```

```
INSTANCE_NAME
-----
orcl_4

SQL> exit
Disconnected from Oracle Database 12c Enterprise Edition Release
12.2.0.1.0 - 64bit Production

[oracle@host01 less_09]$ srvctl status database -db orcl
Instance orcl_1 is running on node host04
Instance orcl_2 is running on node host05
Instance orcl_3 is running on node host03
Instance orcl_4 is running on node host02
Instance orcl_5 is running on node host01

[oracle@host01 less_09]$
```

8. Use SRVCTL to relocate the PROD1 service back to the original host.

```
[oracle@host01 less_09]$ srvctl relocate service -db orcl -
service PROD1 -c host02 -n host01

[oracle@host01 less_09]$
```

9. Verify that the PROD1 service has been relocated to the host specified in the previous step.

```
[oracle@host01 less_09]$ srvctl status service -db orcl -service
prod1
Service PROD1 is running on nodes: host01

[oracle@host01 less_09]$
```

10. Close all terminal sessions opened for this practice.

Practice 9-2: Monitoring Services

Overview

In this practice, you will use EM Express to determine the amount of resources used by sessions executing under a particular service.

1. Connect to your first node as the `oracle` user. Set up your environment variables using the `oraenv` script. Change directory to `/stage/RAC/labs/less_09`. Execute the `createuser.sh` script. This script creates a new user called `jmw` identified by the password `jmw`. The default tablespace of this user is `USERS`, and its temporary tablespace is `TEMP`. This new user has the `CONNECT`, `RESOURCE`, and `DBA` roles.

```
[oracle@dns ~]$ ssh oracle@host01
oracle@host01's password:

[oracle@host01 ~]$ . oraenv
ORACLE_SID = [oracle] ? orcl
The Oracle base has been set to /u01/app/oracle

[oracle@host01 ~]cd /stage/RAC/labs/less_09

[oracle@host01 less_09]$ cat createuser.sh
export HOST=`hostname -s`
export ORACLE_HOME=/u01/app/oracle/product/12.2.0/dbhome_1
export ORACLE_SID=`$ORACLE_HOME/bin/srvctl status database -db
orcl|grep $HOST|cut -f2 -d" "
export PATH=$PATH:$ORACLE_HOME/bin

/u01/app/oracle/product/12.2.0/dbhome_1/bin/sqlplus -s /NOLOG
<<EOF

connect / as sysdba
drop user jmw cascade;
create user jmw identified by jmw default tablespace users
temporary tablespace temp;
grant connect, resource, dba to jmw;

EOF

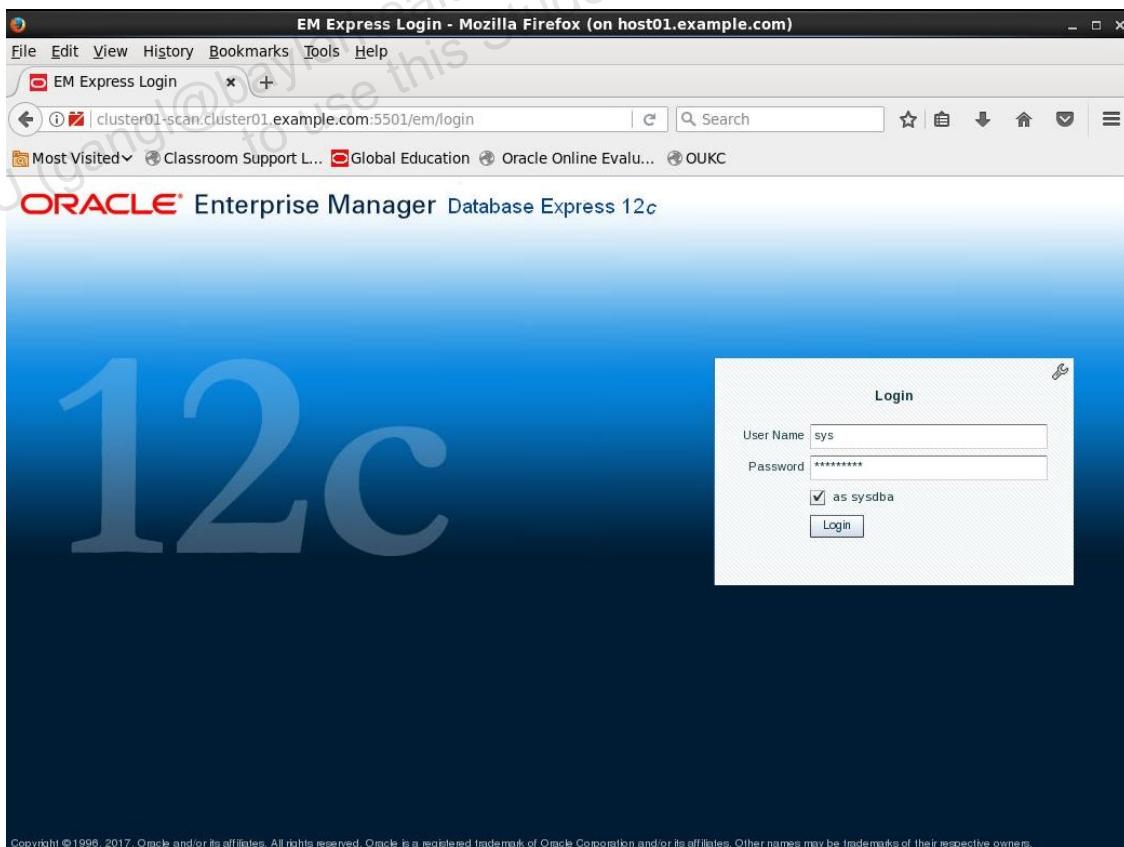
[oracle@host01 less_09]$ ./createuser.sh

User dropped.
```

```
User created.  
Grant succeeded.  
[oracle@host01 less_09]$
```

2. Open a second terminal to your first node as the `oracle` user. Be sure to use the `-x` option. Set up your environment variables using the `oraenv` script. Next, start Firefox and enter the following address:
 - <http://cluster01-scan.cluster01.example.com:5501/em>
 - Login to EM Express as `sys/<sys_password>` as SYSDBA.

```
[oracle@dns ~]$ ssh -X oracle@host01  
oracle@host01's password:  
  
[oracle@host01 ~]$ . oraenv  
ORACLE_SID = [oracle] ? orcl  
The Oracle base has been set to /u01/app/oracle  
  
[oracle@host01 ~]$ firefox&  
[oracle@host01 ~]$
```



3. From the first terminal session, connect to prod1 as jmw using SQL*Plus. When connected, determine the instance on which your session is currently running.

Then execute the following query:

```
select count(*) from dba_objects,dba_objects,dba_objects
```

Do not wait; instead, proceed with the next step.

```
[oracle@host01 less_09]$ sqlplus jmw/<jmw_password>@PROD1

SQL> select instance_name from v$instance;

INSTANCE_NAME
-----
orcl_5

SQL> select count(*) from dba_objects,dba_objects,dba_objects;
```

4. Open a terminal window to connect to host01 as the oracle user. Set up your environment variables using the oraenv script.

```
[oracle@dns ~]$ ssh oracle@host01
oracle@host01's password:

[oracle@host01 ~]$ . oraenv
ORACLE_SID = [oracle] ? orcl
The Oracle base has been set to /u01/app/oracle

[oracle@host01 ~]
```

5. Check statistics on your service with gv\$service_stats from a SQL*Plus session connected as SYSDBA as shown below.

```
[oracle@host01 ~]$ sqlplus sys/<sys_password>@oltpsvc as sysdba

SQL*Plus: Release 12.2.0.1.0 Production on Tue Nov 7 09:16:03
2017

Copyright (c) 1982, 2016, Oracle. All rights reserved.

Connected to:
Oracle Database 12c Enterprise Edition Release 12.2.0.1.0 -
64bit Production
```

```
SQL> select stat_name, sum(value) from gv$service_stats where
  service_name = 'PROD1' group by stat_name;
```

STAT_NAME	SUM (VALUE)
user calls	25
DB CPU	8105608
redo size	3224
db block changes	19
Main function call count	18
DB time	6934
user rollbacks	0
gc cr blocks received	1
gc cr block receive time	0
gc current blocks received	193
opened cursors cumulative	544

STAT_NAME	SUM (VALUE)
workarea executions - multipass	0
session cursor cache hits	433
user I/O wait time	237033
parse count (total)	204
physical reads	16
gc current block receive time	27
workarea executions - optimal	90
concurrency wait time	35897
parse time elapsed	866581
physical writes	0
workarea executions - onepass	0

STAT_NAME	SUM (VALUE)
execute count	568
session logical reads	35993
cluster wait time	75464
application wait time	1327
logons cumulative	2
sql execute elapsed time	11821774
user commits	0

29 rows selected.

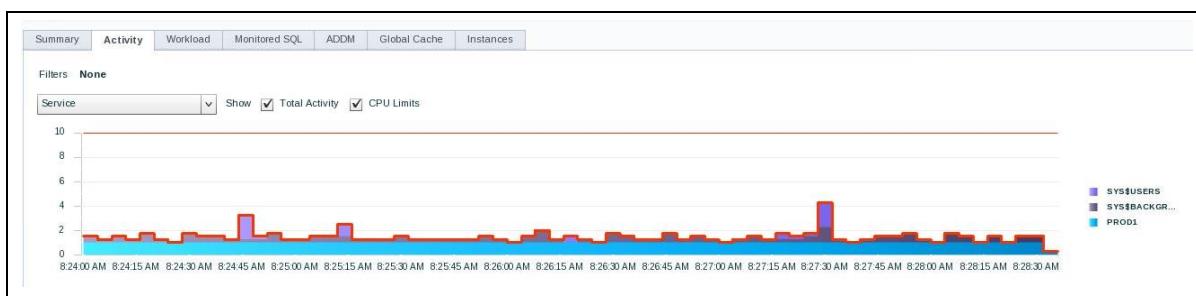
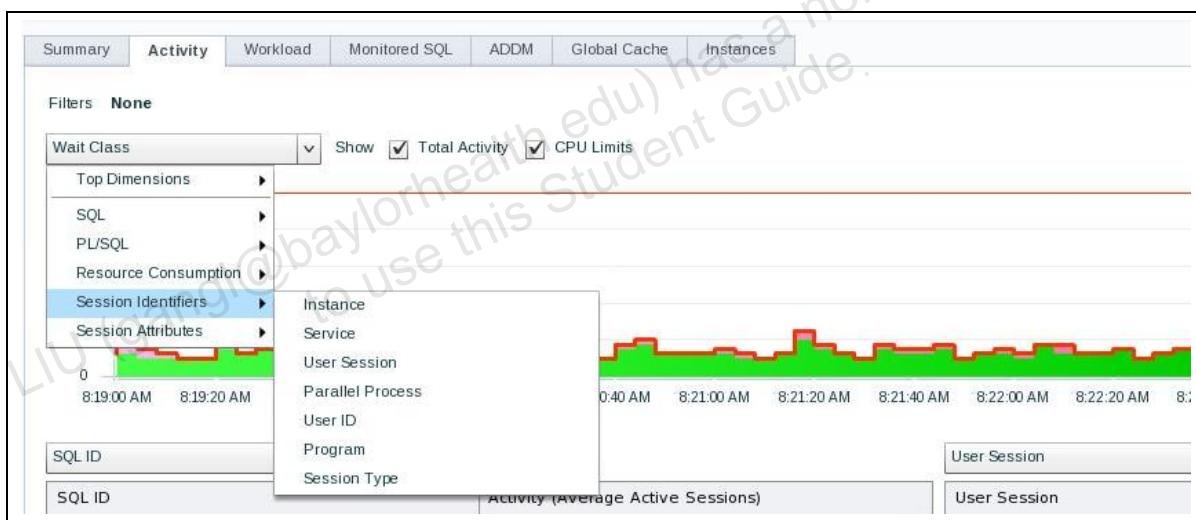
```
SQL> exit
```

```
[oracle@host01 ~] $
```

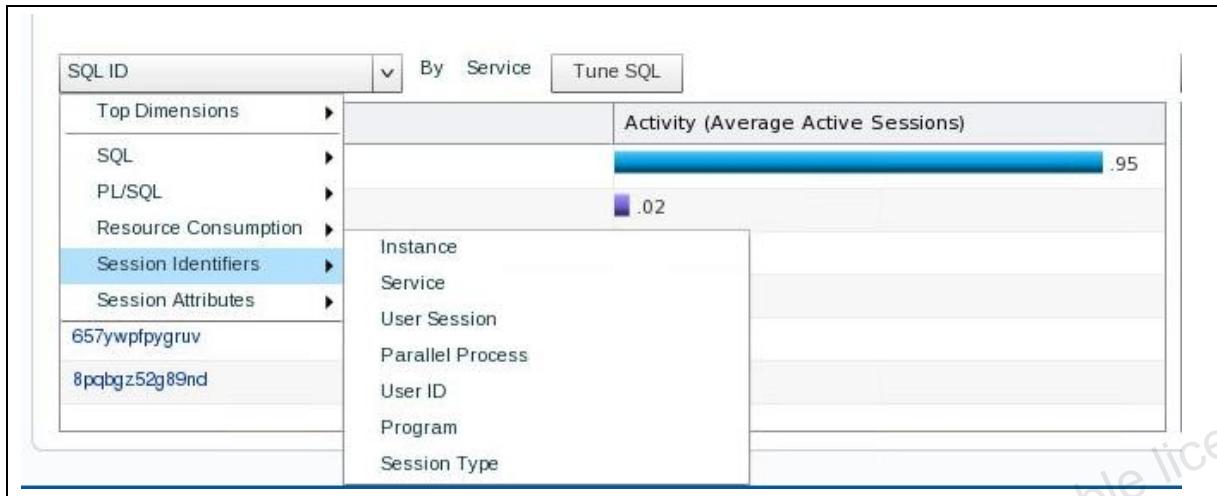
6. You can also use EM Express to view service activity.
 - a. Click Performance, then Performance Hub.



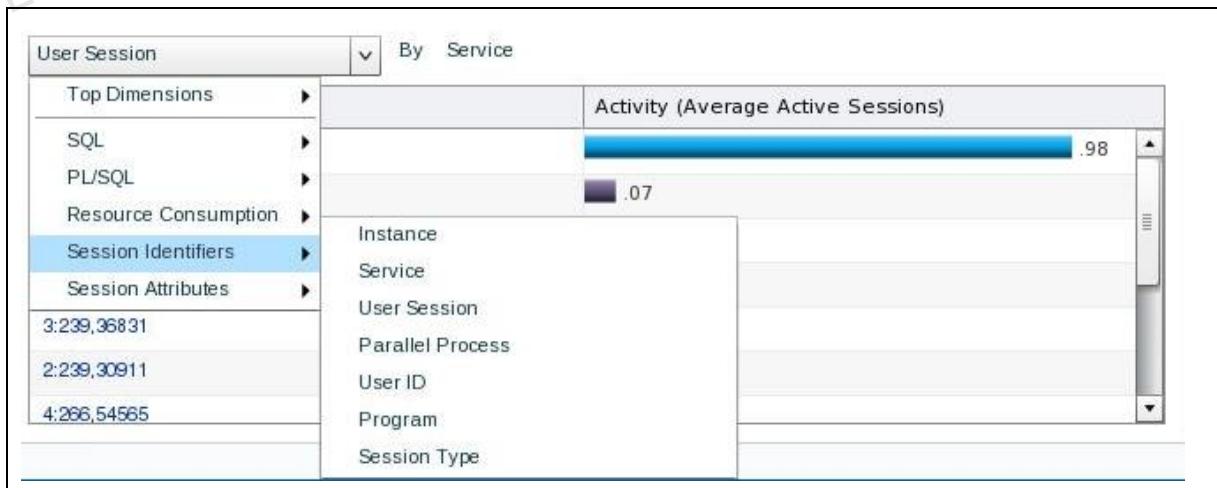
- b. Click the Activity tab. On the activity timeline graph, select Session Identifiers > Service from the pull-down list.

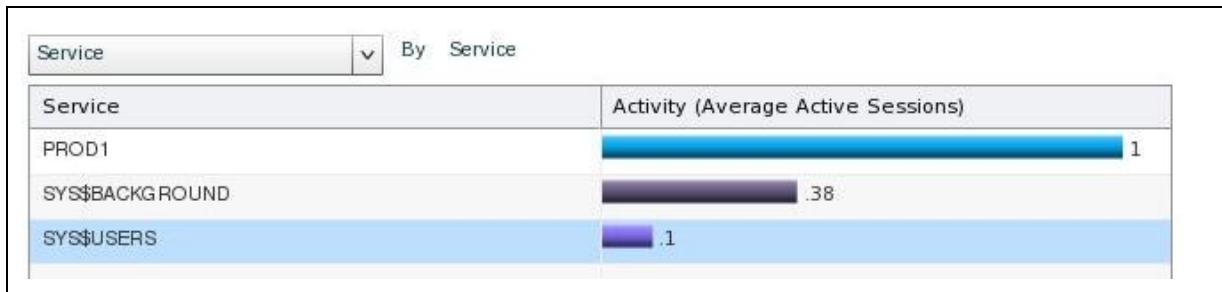


- c. On the bottom-left summary graphic, again choose Session Identifiers > Service from the pull-down list.



- d. On the bottom-right summary graphic, choose Session Identifiers > User ID from the pull-down list.





Spend a few moments and monitor the service activity for the PROD1 service. You should see the PROD1 service activity steadily increase.

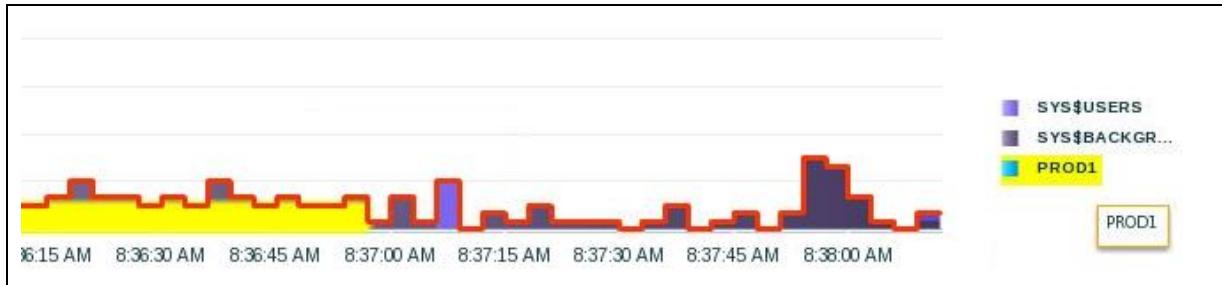
7. Go to the first terminal window. If the query is still running, stop it by issuing a <Ctrl + C>.

```
SQL> select count(*) from dba_objects,dba_objects,dba_objects;
<Ctrl + C>

^Cselect count(*) from dba_objects,dba_objects,dba_objects
          *
ERROR at line 1:
ORA-01013: user requested cancel of current operation

SQL> exit
Disconnected from Oracle Database 12c Enterprise Edition Release
12.2.0.1.0 - 64bit Production
[oracle@host01 less_09]$
```

8. Returning to EM Express, you should see the service activity for PROD1 steadily decrease until it disappears from the monitored list due to inactivity.



9. In the first terminal window, stop the PROD1 service and remove it.

```
[oracle@host01 less_09]$ srvctl stop service -db orcl -service PROD1
[oracle@host01 less_09]$ srvctl remove service -db orcl -service PROD1
[oracle@host01 less_09]$
```

10. Dismiss Firefox and EM Express and close all terminal windows opened for this practice.
11. Close all terminal windows opened for this practice.

Practices for Lesson 10: High Availability for Connections and Applications

Practices for Lesson 10: Overview

Overview

In this practice, you will use Application Continuity in a RAC environment to demonstrate how Application Continuity helps an application to seamlessly recover after the failure of a RAC instance.

Practice 10-1: Using Application Continuity

Overview

In this practice, you will use Application Continuity in a RAC environment to demonstrate how Application Continuity helps an application to seamlessly recover after the failure of a RAC instance.

1. Open a terminal window to connect to host01 as the root user and set the environment variables. Stop Oracle Clusterware in host03, host04, and host05.

```
[oracle@dns ~]$ ssh root@host01
root@host01's Password:

[root@host01 ~]# . oraenv
ORACLE_SID = [root] ? +ASM1
The Oracle base has been set to /u01/app/grid

[root@host01 ~]# crsctl stop cluster -n host03 host04 host05
CRS-2673: Attempting to stop 'ora.crsd' on 'host05'
CRS-2673: Attempting to stop 'ora.crsd' on 'host04'
CRS-2673: Attempting to stop 'ora.crsd' on 'host03'
CRS-2790: Starting shutdown of Cluster Ready Services-managed
resources on server 'host05'
CRS-2673: Attempting to stop 'ora.chad' on 'host05'
CRS-2673: Attempting to stop 'ora.orcl.leafsvc.svc' on 'host05'
CRS-2790: Starting shutdown of Cluster Ready Services-managed
resources on server 'host04'
CRS-2673: Attempting to stop 'ora.chad' on 'host04'
CRS-2673: Attempting to stop 'ora.orcl.leafsvc.svc' on 'host04'
CRS-2790: Starting shutdown of Cluster Ready Services-managed
resources on server 'host03'
CRS-2673: Attempting to stop 'ora.orcl.oltpsvc.svc' on 'host03'
CRS-2673: Attempting to stop 'ora.chad' on 'host03'
CRS-2677: Stop of 'ora.orcl.leafsvc.svc' on 'host05' succeeded
CRS-2673: Attempting to stop 'ora.orcl.db' on 'host05'
CRS-2677: Stop of 'ora.orcl.leafsvc.svc' on 'host04' succeeded
CRS-2673: Attempting to stop 'ora.orcl.db' on 'host04'
CRS-2677: Stop of 'ora.orcl.db' on 'host04' succeeded
CRS-2673: Attempting to stop 'ora.LISTENER.lsnr' on 'host04'
CRS-2677: Stop of 'ora.orcl.db' on 'host05' succeeded
CRS-2673: Attempting to stop 'ora.LISTENER.lsnr' on 'host05'
CRS-2673: Attempting to stop 'ora.LISTENER_LEAF.lsnr' on
'host05'
CRS-2677: Stop of 'ora.orcl.oltpsvc.svc' on 'host03' succeeded
```

```
CRS-2673: Attempting to stop 'ora.orcl.db' on 'host03'
CRS-2673: Attempting to stop 'ora.LISTENER_LEAF.lsnr' on
  'host04'
CRS-2677: Stop of 'ora.LISTENER_LEAF.lsnr' on 'host04' succeeded
CRS-2677: Stop of 'ora.LISTENER.lsnr' on 'host04' succeeded
CRS-2677: Stop of 'ora.LISTENER_LEAF.lsnr' on 'host05' succeeded
CRS-2677: Stop of 'ora.LISTENER.lsnr' on 'host05' succeeded
CRS-2673: Attempting to stop 'ora.host05.vip' on 'host05'
CRS-2673: Attempting to stop 'ora.host04.vip' on 'host04'
CRS-2677: Stop of 'ora.host04.vip' on 'host04' succeeded
CRS-2677: Stop of 'ora.host05.vip' on 'host05' succeeded
CRS-2677: Stop of 'ora.chad' on 'host04' succeeded
CRS-2672: Attempting to start 'ora.host04.vip' on 'host02'
CRS-2677: Stop of 'ora.orcl.db' on 'host03' succeeded
CRS-2673: Attempting to stop 'ora.DATA.dg' on 'host03'
CRS-2673: Attempting to stop 'ora.FRA.dg' on 'host03'
CRS-2673: Attempting to stop 'ora.MGMT.dg' on 'host03'
CRS-2673: Attempting to stop 'ora.LISTENER.lsnr' on 'host03'
CRS-2673: Attempting to stop 'ora.LISTENER_SCAN1.lsnr' on
  'host03'
CRS-2677: Stop of 'ora.chad' on 'host03' succeeded
CRS-2677: Stop of 'ora.FRA.dg' on 'host03' succeeded
CRS-2677: Stop of 'ora.DATA.dg' on 'host03' succeeded
CRS-2677: Stop of 'ora.MGMT.dg' on 'host03' succeeded
CRS-2673: Attempting to stop 'ora.asm' on 'host03'
CRS-2677: Stop of 'ora.LISTENER.lsnr' on 'host03' succeeded
CRS-2673: Attempting to stop 'ora.host03.vip' on 'host03'
CRS-2677: Stop of 'ora.LISTENER_SCAN1.lsnr' on 'host03'
  succeeded
CRS-2673: Attempting to stop 'ora.scan1.vip' on 'host03'
CRS-2677: Stop of 'ora.asm' on 'host03' succeeded
CRS-2673: Attempting to stop 'ora.ASMNET1LSNR_ASM.lsnr' on
  'host03'
CRS-2677: Stop of 'ora.host03.vip' on 'host03' succeeded
CRS-2677: Stop of 'ora.scan1.vip' on 'host03' succeeded
CRS-2676: Start of 'ora.host04.vip' on 'host02' succeeded
CRS-2673: Attempting to stop 'ora.net1.network' on 'host04'
CRS-2677: Stop of 'ora.net1.network' on 'host04' succeeded
CRS-2792: Shutdown of Cluster Ready Services-managed resources
  on 'host04' has completed
CRS-2677: Stop of 'ora.chad' on 'host05' succeeded
CRS-2672: Attempting to start 'ora.host05.vip' on 'host02'
CRS-2677: Stop of 'ora.ASMNET1LSNR_ASM.lsnr' on 'host03'
  succeeded
```

```
CRS-2672: Attempting to start 'ora.scan1.vip' on 'host01'
CRS-2672: Attempting to start 'ora.host03.vip' on 'host02'
CRS-2677: Stop of 'ora.crsd' on 'host04' succeeded
CRS-2673: Attempting to stop 'ora.cluster_interconnect.haip' on
'host04'

CRS-2673: Attempting to stop 'ora.ctssd' on 'host04'
CRS-2673: Attempting to stop 'ora.evmd' on 'host04'
CRS-2673: Attempting to stop 'ora.storage' on 'host04'
CRS-2677: Stop of 'ora.storage' on 'host04' succeeded
CRS-2677: Stop of 'ora.cluster_interconnect.haip' on 'host04'
succeeded

CRS-2677: Stop of 'ora.ctssd' on 'host04' succeeded
CRS-2677: Stop of 'ora.evmd' on 'host04' succeeded
CRS-2673: Attempting to stop 'ora.cssd' on 'host04'
CRS-2677: Stop of 'ora.cssd' on 'host04' succeeded
CRS-2676: Start of 'ora.host05.vip' on 'host02' succeeded
CRS-2676: Start of 'ora.scan1.vip' on 'host01' succeeded
CRS-2672: Attempting to start 'ora.LISTENER_SCAN1.lsnr' on
'host01'

CRS-2673: Attempting to stop 'ora.net1.network' on 'host05'
CRS-2677: Stop of 'ora.net1.network' on 'host05' succeeded
CRS-2792: Shutdown of Cluster Ready Services-managed resources
on 'host05' has completed

CRS-2676: Start of 'ora.host03.vip' on 'host02' succeeded
CRS-2677: Stop of 'ora.crsd' on 'host05' succeeded
CRS-2673: Attempting to stop 'ora.cluster_interconnect.haip' on
'host05'

CRS-2673: Attempting to stop 'ora.ctssd' on 'host05'
CRS-2673: Attempting to stop 'ora.evmd' on 'host05'
CRS-2673: Attempting to stop 'ora.storage' on 'host05'
CRS-2677: Stop of 'ora.storage' on 'host05' succeeded
CRS-2677: Stop of 'ora.cluster_interconnect.haip' on 'host05'
succeeded

CRS-2677: Stop of 'ora.ctssd' on 'host05' succeeded
CRS-2677: Stop of 'ora.evmd' on 'host05' succeeded
CRS-2673: Attempting to stop 'ora.cssd' on 'host05'
CRS-2677: Stop of 'ora.cssd' on 'host05' succeeded
CRS-2676: Start of 'ora.LISTENER_SCAN1.lsnr' on 'host01'
succeeded

CRS-2673: Attempting to stop 'ora.ons' on 'host03'
CRS-2677: Stop of 'ora.ons' on 'host03' succeeded
CRS-2673: Attempting to stop 'ora.net1.network' on 'host03'
CRS-2677: Stop of 'ora.net1.network' on 'host03' succeeded
```

```

CRS-2792: Shutdown of Cluster Ready Services-managed resources
on 'host03' has completed
CRS-2677: Stop of 'ora.crsd' on 'host03' succeeded
CRS-2673: Attempting to stop 'ora.ctssd' on 'host03'
CRS-2673: Attempting to stop 'ora.evmd' on 'host03'
CRS-2673: Attempting to stop 'ora.storage' on 'host03'
CRS-2677: Stop of 'ora.storage' on 'host03' succeeded
CRS-2673: Attempting to stop 'ora.asm' on 'host03'
CRS-2677: Stop of 'ora.ctssd' on 'host03' succeeded
CRS-2677: Stop of 'ora.evmd' on 'host03' succeeded
CRS-2677: Stop of 'ora.asm' on 'host03' succeeded
CRS-2673: Attempting to stop 'ora.cluster_interconnect.haip' on
'host03'
CRS-2677: Stop of 'ora.cluster_interconnect.haip' on 'host03'
succeeded
CRS-2673: Attempting to stop 'ora.cssd' on 'host03'
CRS-2677: Stop of 'ora.cssd' on 'host03' succeeded

[root@host01 ~]#

```

2. Open a terminal window connected to host01 using the oracle OS user.

```

[oracle@dns ~]$ ssh oracle@host01
oracle@host01's password:
[oracle@host01 ~]$

```

3. Configure the environment by using the oraenv script.

```

[oracle@host01 ~]$ . oraenv
ORACLE_SID = [oracle] ? orcl
The Oracle base has been set to /u01/app/oracle
[oracle@host01 ~]$

```

4. Confirm that two instances of the RAC database are up and running.

```

[oracle@host01 ~]$ srvctl status database -db orcl
Instance orcl_4 is running on node host02
Instance orcl_5 is running on node host01

[oracle@host01 ~]$

```

5. Navigate to the directory that contains the files for this practice.

```
[oracle@host01 ~]$ cd /stage/RAC/labs/less_10  
[oracle@host01 less_10]$
```

6. Execute the `setup10.sh` script from the `/stage/RAC/labs/less_10` directory to set up the necessary configuration for this practice.

```
[oracle@host01 less_10]$ ./setup10.sh  
  
User dropped.  
  
Tablespace dropped.  
  
Tablespace created.  
  
User created.  
  
Grant succeeded.  
  
drop table emp  
      *  
ERROR at line 1:  
ORA-00942: table or view does not exist  
  
Table created.  
  
[oracle@host01 less_10]$
```

7. Create a database service on the `orcl` database. Configure the service for use in conjunction with Application Continuity.

```
[oracle@host01 less_10]$ srvctl add service -db orcl -service  
actest -serverpool ora.orcldb -cardinality singleton -  
failovertype TRANSACTION -commit_outcome TRUE -failoverretry 50  
-failoverdelay 5 -retention 86400 -replay_init_time 1800 -  
notification TRUE  
  
[oracle@host01 less_10]$
```

8. Start the service.

```
[oracle@host01 less_10]$ srvctl start service -db orcl -service  
actest  
[oracle@host01 less_10]$
```

9. Examine the status of the newly created service. Take note of the node it is running on (host01 in this example) because it may be different in your environment.

```
[oracle@host01 less_10]$ srvctl status service -db orcl -service actest
Service actest is running on nodes: host01
[oracle@host01 less_10]$
```

10. Establish another terminal session connected to host01 using the oracle OS user. To differentiate this session from your primary session, it will be referred to as the ADMIN session for the rest of the practice.

```
[oracle@dns ~]$ ssh oracle@host01
oracle@host01's password:

[oracle@host01 ~]$ cd /stage/RAC/labs/less_10
[oracle@host01 less_10]$
```

11. Configure the prompt in your ADMIN session as shown below. This will help you to differentiate between your terminal sessions as you progress through this practice.

```
[oracle@host01 less_10]$ export PS1='[HOST01 ADMIN] $ '
[HOST01 ADMIN] $
```

12. Configure the environment in your ADMIN session by using the oraenv script.

```
[HOST01 ADMIN] $ ps -ef |grep ora_smon

oracle    21046      1  0 15:04 ?          00:00:00 ora_smon_orcl_5
oracle    23595 18572  0 15:09 pts/44      00:00:00 grep ora_smon

[HOST01 ADMIN] $. oraenv
ORACLE_SID = [oracle] ? orcl_5
The Oracle base has been set to /u01/app/oracle
[HOST01 ADMIN] $
```

13. Back in your second terminal window, examine the scripts that you will soon use to execute the practice application. Notice that both scripts execute the same application code (in actest.jar). The only difference is that each script references a different properties file.

```
[oracle@host01 less_10]$ cat runnoreplay
java -
classpath ./actest.jar:$ORACLE_HOME/ucp/lib/ucp.jar:$ORACLE_HOME
/jdbc/lib/ojdbc8.jar actest.ACTest actest_noreplay.properties
[oracle@host01 less_10]$
```

```
[oracle@host01 less_10]$ cat runreplay
java -
classpath ./actest.jar:$ORACLE_HOME/ucp/lib/ucp.jar:$ORACLE_HOME
/jdbc/lib/ojdbc8.jar actest.actest_replay.properties

[oracle@host01 less_10]$
```

14. Examine the properties files. Notice that the only difference is the datasource specification.

```
[oracle@host01 less_10]$ cat actest_noreplay.properties
username=scott
password=tiger
autoCommit=false

# Use standard 12.2 datasource no replay
datasource=oracle.jdbc.pool.OracleDataSource

url=jdbc:oracle:thin:@(DESCRIPTION=(ADDRESS=(PROTOCOL=tcp) (HOST=
cluster01-
scan.cluster01.example.com) (PORT=1521)) (CONNECT_DATA=(SERVICE_NA
ME=actest)))

# UCP setting:
ucp_pool_size=2
ucp_validate_connection_on_borrow=true
ucp_connection_wait_timeout=60

# Think Time taken to process the results from the database.
# Time in milliseconds.
# -1 means no sleep.
thread_think_time=20

# Number of concurrent threads running in the application
# UCP is tuned to have MAX and MIN limit set to this
number_of_threads=6

verbose=true

[oracle@host01 less_10]$
```

```
[oracle@host01 less_10]$ cat actest_replay.properties
username=scott
password=tiger
autoCommit=false

# Use new 12.2 replay datasource
datasource=oracle.jdbc.replay.OracleDataSourceImpl

url=jdbc:oracle:thin:@(DESCRIPTION=(ADDRESS=(PROTOCOL=tcp) (HOST=
cluster01-
scan.cluster01.example.com) (PORT=1521)) (CONNECT_DATA=(SERVICE_NA
ME=actest)))

# UCP setting:
ucp_pool_size=2
ucp_validate_connection_on_borrow=true
ucp_connection_wait_timeout=60

# Think Time taken to process the results from the database.
Time in milliseconds.
# -1 means no sleep.
thread_think_time=20

# Number of concurrent threads running in the application
# UCP is tuned to have MAX and MIN limit set to this
number_of_threads=6

verbose=true

[oracle@host01 less_10]$
```

```
[oracle@host01 less_10]$ diff actest_replay.properties
actest_noreplay.properties
5,6c5,6
< # Use new 12.2 replay datasource
< datasource=oracle.jdbc.replay.OracleDataSourceImpl
---
> # Use standard 12.2 datasource no replay
> datasource=oracle.jdbc.pool.OracleDataSource

[oracle@host01 less_10]$
```

Next, you will execute the practice Java application twice. Once without the benefit of Application Continuity, and once with Application Continuity enabled. Notice that you will execute the same application and the only difference is the JDBC data source that is used on each occasion. The source files containing the application code are contained in the `src` directory. Feel free to examine the application code if you like.

15. Execute the practice application without the benefit of Application Continuity. Notice that while the application runs, a periodic status message is displayed.

```
[oracle@host01 less_10]$ ./runnoreplay
#####
Connecting to
jdbc:oracle:thin:@(DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=cluster01-
scan.cluster01.example.com)(PORT=1521))(CONNECT_DATA=(SERVICE_NAME=actest)))
# of Threads : 6
UCP pool size : 2
Thread think time : 20 ms
#####
2 active connections, avg response time from db 38 ms
2 active connections, avg response time from db 23 ms
```

16. While the application continues to execute in the primary window, return to your ADMIN session and remind yourself about which node is running the `actest` service. Then, abort the database instance on the host running the `actest` service (`host01` in the example shown below). Ensure that you abort the instance on the node running the service and not the other database node.

```
[HOST01 ADMIN] $ srvctl status service -db orcl -service actest
Service actest is running on nodes: host01

[HOST01 ADMIN] $ srvctl status database -db orcl
Instance orcl_4 is running on node host02
Instance orcl_5 is running on node host01

[HOST01 ADMIN] $ ssh host01 pkill -9 -f pmon_orcl_5
[HOST01 ADMIN] $
```

17. Return to your primary window and you should see a series of errors caused by aborting the database instance. This is typical of applications that do not use Application Continuity. Press Ctrl + C to abort the application.

```

...
      at
oracle.ucp.jdbc.oracle.OracleJDBCConnectionPool.borrowConnection
(OracleJDBCConnectionPool.java:1441)
      at
oracle.ucp.jdbc.oracle.OracleConnectionConnectionPool.borrowConn
ection(OracleConnectionConnectionPool.java:81)
      at
oracle.ucp.jdbc.PoolDataSourceImpl.getConnection(PoolDataSourceI
mpl.java:1027)
      ...
      ... 4 more
.Exception occurred while getting connection:
oracle.ucp.UniversalConnectionPoolException: Cannot get
Connection from Datasource: java.sql.SQLRecoverableException:
Listener refused the connection with the following error:
ORA-12514, TNS:listener does not currently know of service
requested in connect descriptor
.
0 active connections, avg response time from db 150418377 ms
^C
[oracle@host01 less_10]$
```

18. Confirm both RAC database instances are up and running again.

```
[oracle@host01 less_10]$ srvctl status database -db orcl
Instance orcl_4 is running on node host02
Instance orcl_5 is running on node host01

[oracle@host01 less_10]$
```

19. Reexamine the status of the **actest** service. You should observe that the service is running on a different node compared to what you observed earlier. This is because the service was migrated when you aborted the database instance earlier in the practice.

```
[oracle@host01 less_10]$ srvctl status service -db orcl -service
actest
Service actest is running on nodes: host02

[oracle@host01 less_10]$
```

20. Execute the practice application with Application Continuity enabled. You should see the same period status messages as before while the application is running.

```
[oracle@host01 less_10]$ ./runreplay
#####
Connecting to
jdbc:oracle:thin:@(DESCRIPTION=(ADDRESS=(PROTOCOL=tcp) (HOST=cluster01-
scan.cluster01.example.com) (PORT=1521)) (CONNECT_DATA=(SERVICE_NAME=actest)))
# of Threads : 6
UCP pool size : 2
Thread think time : 20 ms
#####
2 active connections, avg response time from db 44 ms
2 active connections, avg response time from db 23 ms
```

21. While the application continues to execute in the primary window, return to your ADMIN session and remind yourself about which node is now running the `actest` service. Then, abort the database instance running the `actest` service (which is now `host02` in the example shown below). Ensure that you abort the instance on the node running the service and not the other database node.

```
[HOST01 ADMIN] $ srvctl status service -db orcl -service actest
Service actest is running on nodes: host02

[HOST1 ADMIN] $ srvctl status database -db orcl
Instance orcl_4 is running on node host02
Instance orcl_5 is running on node host01

[HOST01 ADMIN] $ ssh host02 pkill -9 -f pmon_orcl_4
[HOST01 ADMIN] $
```

22. Return to your primary window and you should see that the application continued in spite of the aborted database instance. You should see a pause and then a brief spike in the response time, which coincides with the time when the database instance was aborted. Now you have seen how Application Continuity masks the effect of database instance loss in a RAC database environment. Press Ctrl + C to abort the application.

```
2 active connections, avg response time from db 44 ms
2 active connections, avg response time from db 23 ms
2 active connections, avg response time from db 20 ms
2 active connections, avg response time from db 16 ms
2 active connections, avg response time from db 14 ms
2 active connections, avg response time from db 12 ms
```

```
2 active connections, avg response time from db 12 ms
2 active connections, avg response time from db 15 ms
2 active connections, avg response time from db 125 ms
2 active connections, avg response time from db 16 ms
2 active connections, avg response time from db 13 ms
2 active connections, avg response time from db 12 ms
2 active connections, avg response time from db 12 ms
2 active connections, avg response time from db 13 ms
^C
[oracle@host01 less_10]$
```

23. Confirm that both instances are running and the service migrated to the other node as expected.

```
[oracle@host01 less_10]$ srvctl status database -db orcl
Instance orcl_4 is running on node host02
Instance orcl_5 is running on node host01

[oracle@host01 less_10]$ srvctl status service -db orcl -service actest
Service actest is running on nodes: host01

[oracle@host01 less_10]$
```

24. Stop the actest service and remove it.

```
[oracle@host01 less_10]$ srvctl stop service -db orcl -service actest

[oracle@host01 less_10]$ srvctl remove service -db orcl -service actest
[oracle@host01 less_10]$
```

25. Close all terminal windows opened for this practice.

Practices for Lesson 11: Oracle RAC One Node

Practices for Lesson 11: Overview

Overview

In this practice, you will create a RAC One Node database by converting the exiting RAC database. You will perform an online database relocation. Finally, you will convert the RAC One Database to an Oracle RAC database.

Practice 11-1: RAC One Node

Overview

In this practice, you will create a RAC One Node database by converting the existing RAC database. You will perform an online database relocation. Finally, you will convert the RAC One Database to an Oracle RAC database.

1. Open a terminal window to connect to host01 as the oracle user. Set the environment and check the status of the database. **Note:** Your instance names may be different.

```
[oracle@dns ~]$ ssh oracle@host01
oracle@host01's password:

[oracle@host01 ~]$ . oraenv
ORACLE_SID = [oracle] ? orcl
The Oracle base has been set to /u01/app/oracle

[oracle@host01 ~]$ srvctl status database -db orcl
Instance orcl_4 is running on node host02
Instance orcl_5 is running on node host01
[oracle@host01 ~]$
```

2. Stop the instance on host02. Verify the database status.

```
[oracle@host01 ~]$ srvctl stop instance -f -db orcl -node host02

[oracle@host01 ~]$ srvctl status database -db orcl
Instance orcl_5 is running on node host01
Instance orcl_4 is not running on node host02
[oracle@host01 ~]$
```

3. An active service is needed to convert the RAC database to RACONENODE. Create a service called SERV1.

```
[oracle@host01 ~]$ srvctl add service -db orcl -service SERV1 -
serverpool orcldb
[oracle@host01 ~]$
```

4. Attempt to convert the RAC database to RACONENODE. What do you observe?

```
[oracle@host01 ~]$ srvctl convert database -db orcl -dbtype
RACONENODE
PRCD-1155 : Failed to convert the configuration of cluster
database orcl into its equivalent RAC One Node database
```

```
[oracle@host01 ~]$ configuration because cluster database has more than one server
pool [ora.orcldb, ora.RACLEAF] to control its placement
[oracle@host01 ~]$
```

5. Remove the components that were added for the Reader Nodes and convert the RAC database to RACONENODE.

```
[oracle@host01 ~]$ srvctl remove service -db orcl -service leafsvc
[oracle@host01 ~]$ srvctl remove srvpool -serverpool RACLEAF

[oracle@host01 ~]$ srvctl modify database -db orcl -serverpool orcldb
[oracle@host01 ~]$ srvctl convert database -db orcl -dbtype RACONENODE
[oracle@host01 ~]$
```

6. From the oracle terminal session, check your database configuration using the `srvctl config` utility.

```
[oracle@host01 ~]$ srvctl config database -db orcl
Database unique name: orcl
Database name: orcl
Oracle home: /u01/app/oracle/product/12.2.0/dbhome_1
Oracle user: oracle
Spfile: +DATA/ORCL/PARAMETERFILE/spfile.267.959256577
Password file: +DATA/ORCL/PASSWORD/pwdorcl.257.959256207
Domain:
Start options: open
Stop options: immediate
Database role: PRIMARY
Management policy: AUTOMATIC
Server pools: orcldb
Disk Groups: FRA,DATA
Mount point paths:
Services: actest,OLTPSVC,SERV1
Type: RACOneNode
Online relocation timeout: 30
Instance name prefix: orcl
Candidate servers:
OSDBA group: dba
OSOPER group: oper
Database instances: orcl_1
CSS critical: no
```

```
CPU count: 0
Memory target: 0
Maximum memory: 0
Default network number for database services:
Database is policy managed

[oracle@host01 ~]$
```

7. Use the `srvctl` utility to check the status of the `orcl` database.

```
[oracle@host01 ~]$ srvctl status database -db orcl
Instance orcl_5 is running on node host01
Online relocation: INACTIVE
[oracle@host01 ~]$
```

8. Execute `srvctl relocate database -help` to view command usage.

```
[oracle@host01 ~]$ srvctl relocate database -help

Initiate online relocation of the RAC One Node database.

Usage: srvctl relocate database -db <db_unique_name> { [-node
<target>] [-timeout <timeout>] [-stopoption <stop_option>] | -
abort [-revert] } [-verbose]

      -db <db_unique_name>           Unique name of database to
relocate

      -node <target>                 Target node to which to
relocate database

      -timeout <timeout>             Online relocation timeout in
minutes

      -abort                         Abort failed online
relocation

      -revert                         Remove target node of failed
online relocation request from the candidate server list of
administrator-managed RAC One Node database

      -stopoption <stop_option>       Override default shutdown
option for running instance (only NORMAL allowed)

      -verbose                        Verbose output

      -help                           Print usage

[oracle@host01 ~]$
```

9. Open another terminal window as `oracle` to prepare for monitoring the migration process.

```
[oracle@dns ~]$ ssh oracle@host01
oracle@host01's password:
```

```
[oracle@host01 ~]$ . oraenv
ORACLE_SID = [oracle] ? orcl
The Oracle base has been set to /u01/app/oracle
[oracle@host01 ~]$
```

10. Switch to the first terminal window. In this example, the `orcl_5` instance was running initially on `host01`. Use `srvctl` to perform an online database relocation from `host01` to `host02`. **Immediately after issuing the command, proceed to the next step!**

```
[oracle@host01 ~]$ srvctl relocate database -db orcl -node host02 -timeout 15 -verbose

<<< Immediately go to the next step>>>

Configuration updated to two instances
Instance orcl_1 started
Services relocated
Waiting for up to 15 minutes for instance orcl_5 to stop ...
Instance orcl_5 stopped
Configuration updated to one instance

[oracle@host01 ~]$
```

11. Switch to the second terminal window and run the `srvctl status database -db orcl` command several times to monitor the migration process until the relocation command finishes in the first terminal. Did you wait for 15 minutes to complete the migration process?

```
[oracle@host01 ~]$ srvctl status database -db orcl
Instance orcl_5 is running on node host01
Online relocation: ACTIVE
Source instance: orcl_5 on host01
Destination instance: orcl_1 on host02

[oracle@host01 ~]$ srvctl status database -db orcl
Instance orcl_5 is running on node host01
Online relocation: ACTIVE
Source instance: orcl_5 on host01
Destination instance: orcl_1 on host02

[oracle@host01 ~]$ srvctl status database -db orcl
Instance orcl_1 is running on node host02
Online relocation: INACTIVE
[oracle@host01 ~]$
```

Note: In our environment, the migration process would take about 1 minute even though the timeout was set to 15 minutes. The timeout option in step 10 is the amount of time after the relocated database starts and services are migrated, before the former instance of the database stops. This configured amount of time is not an upper bound on the amount of time taken by the entire operation, but only controls how long the relocated database waits for connections to migrate from the former instance to the new instance, before stopping the former instance.

- Let's convert the RAC One Node database to a RAC database. First, shut down the RAC One Node database.

```
[oracle@host01 ~]$ srvctl stop database -db orcl
[oracle@host01 ~]$
```

- Use `srvctl` to convert the database to RAC and restart the database. Check the database status to make sure that both the instances are up.

```
[oracle@host01 ~]$ srvctl convert database -db orcl -dbtype RAC

[oracle@host01]$ srvctl start database -db orcl

[oracle@host01]$ srvctl status database -db orcl
Instance orcl_1 is running on node host02
Instance orcl_2 is running on node host01

[oracle@host01]$
```

- Execute the `srvctl status service` command to view the services configuration. Note that the `serv1` service is running on all nodes. Stop the service and remove it.

```
[oracle@host01 ~]$ srvctl status service -db orcl -service serv1
Service SERV1 is running on nodes: host01,host02

[oracle@host01 ~]$ srvctl stop service -db orcl -service serv1

[oracle@host01 ~]$ srvctl remove service -db orcl -service serv1
[oracle@host01 ~]$
```

- Execute the `srvctl config database` command to view the database configuration.

```
[oracle@host01 ~]$ srvctl config database -db orcl
Database unique name: orcl
Database name: orcl
Oracle home: /u01/app/oracle/product/12.2.0/dbhome_1
Oracle user: oracle
```

```
Spfile: +DATA/ORCL/PARAMETERFILE/spfile.267.959256577
Password file: +DATA/ORCL/PASSWORD/pwdorcl.257.959256207
Domain:
Start options: open
Stop options: immediate
Database role: PRIMARY
Management policy: AUTOMATIC
Server pools: orcldb
Disk Groups: FRA,DATA
Mount point paths:
Services: OLTPSVC
Type: RAC
Start concurrency:
Stop concurrency:
OSDBA group: dba
OSOPER group: oper
Database instances:
Configured nodes:
CSS critical: no
CPU count: 0
Memory target: 0
Maximum memory: 0
Default network number for database services:
Database is policy managed

[oracle@host01 ~]$
```

16. Exit all terminal windows opened for this practice.

Practices for Lesson 12: Oracle Database In-Memory in RAC

Practices for Lesson 12: Overview

Overview

In this practice, you will reconfigure your practice environment in preparation for the CDB practices.

Practice 12-1: Reconfiguring the Environment

Overview

In this practice, you will reconfigure your practice environment in preparation for the CDB practices.

1. Open a terminal window to connect to host01 as root user to start Oracle Clusterware on host03.

```
[oracle@dns ~]$ ssh root@host01
root@host01's Password:
[root@host01 ~]# . oraenv
ORACLE_SID = [root] ? +ASM1
The Oracle base has been set to /u01/app/grid
[root@host01 ~]# crsctl start cluster -n host03
CRS-2672: Attempting to start 'ora.evmd' on 'host03'
CRS-2672: Attempting to start 'ora.cssdmonitor' on 'host03'
CRS-2676: Start of 'ora.cssdmonitor' on 'host03' succeeded
CRS-2672: Attempting to start 'ora.cssd' on 'host03'
CRS-2672: Attempting to start 'ora.diskmon' on 'host03'
CRS-2676: Start of 'ora.diskmon' on 'host03' succeeded
CRS-2676: Start of 'ora.evmd' on 'host03' succeeded
CRS-2676: Start of 'ora.cssd' on 'host03' succeeded
CRS-2672: Attempting to start 'ora.ctssd' on 'host03'
CRS-2672: Attempting to start 'ora.cluster_interconnect.haip' on 'host03'
CRS-2676: Start of 'ora.ctssd' on 'host03' succeeded
CRS-2676: Start of 'ora.cluster_interconnect.haip' on 'host03' succeeded
CRS-2672: Attempting to start 'ora.asm' on 'host03'
CRS-2676: Start of 'ora.asm' on 'host03' succeeded
CRS-2672: Attempting to start 'ora.storage' on 'host03'
CRS-2676: Start of 'ora.storage' on 'host03' succeeded
CRS-2672: Attempting to start 'ora.crsd' on 'host03'
CRS-2676: Start of 'ora.crsd' on 'host03' succeeded

[root@host01 ~]#
```

2. Open a new terminal window to connect to host01 as the oracle user to remove the existing RAC database. Navigate to /u01/app/oracle/product/12.2.0/dbhome_1/bin and execute DBCA.

```
[oracle@dns ~]# ssh -X oracle@host01
oracle@host01's password:

[oraclec@host01 ~]# cd /u01/app/oracle/product/12.2.0/dbhome_1/bin

[oracle@host01 bin]$ ./dbca
```

Step	Screen/Page Description	Choices or Values
2-1	Database Operation	Select Delete Database. Click Next.
2-2	Source Database	Select orcl and Enter the SYS user's Password and Click Next
2-3	De-register Management Options	Click Next.
2-4	Summary	Click Finish.
2-5	Database Configuration Assistant dialog box	You are informed that the instances and datafiles will be deleted. Click Yes to proceed.
2-6	Finish	You are informed that database deletion is complete. Click Close

3. Remove the orcldb server pool.

```
[oracle@host01 bin]$ /u01/app/12.2.0/grid/bin/srvctl remove svrpool
-serverpool orcldb

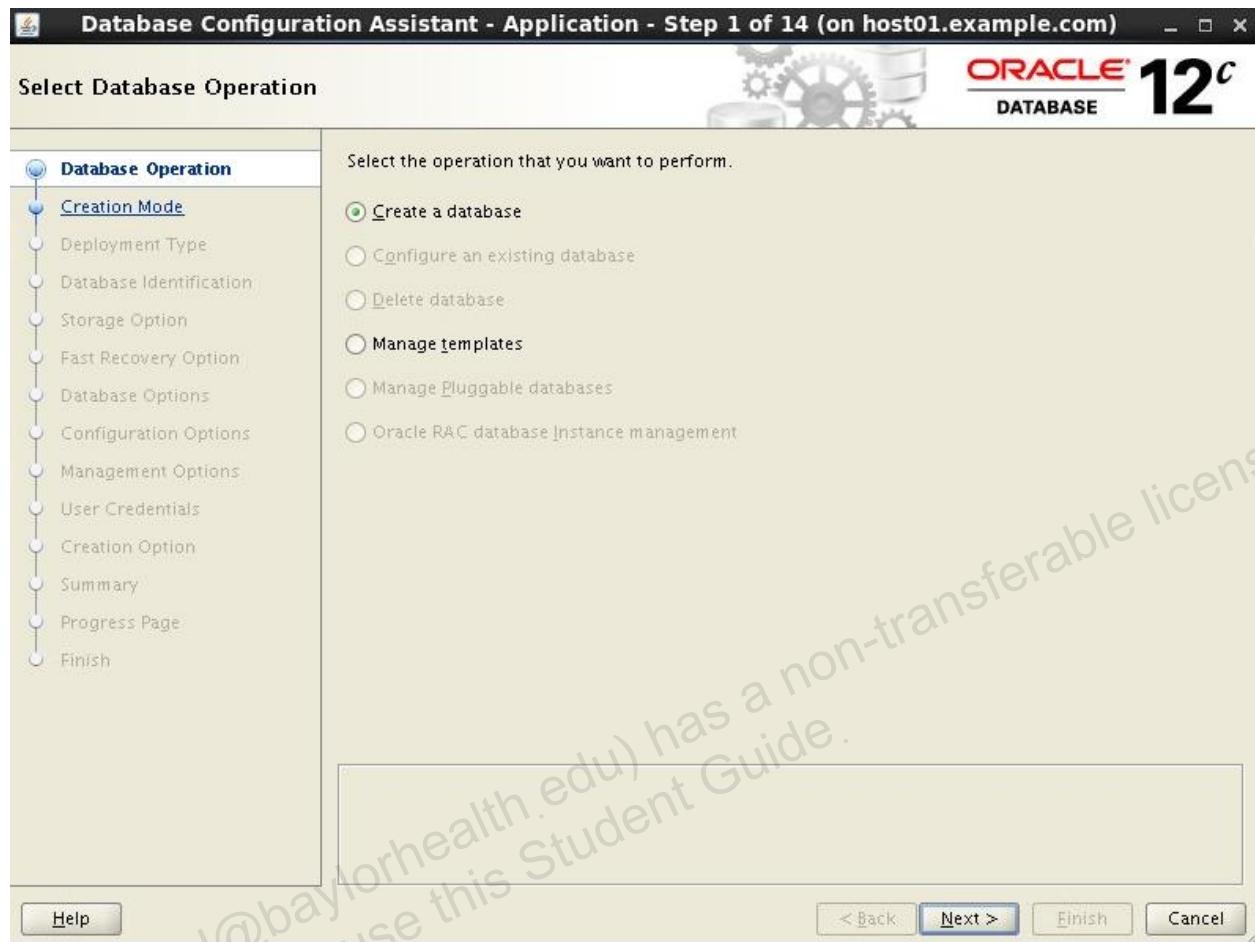
[oracle@host01 bin]$ /u01/app/12.2.0/grid/bin/srvctl status svrpool
Server pool name: Free
Active servers count: 3
Server pool name: Generic
Active servers count: 0

[oracle@host01 bin]$
```

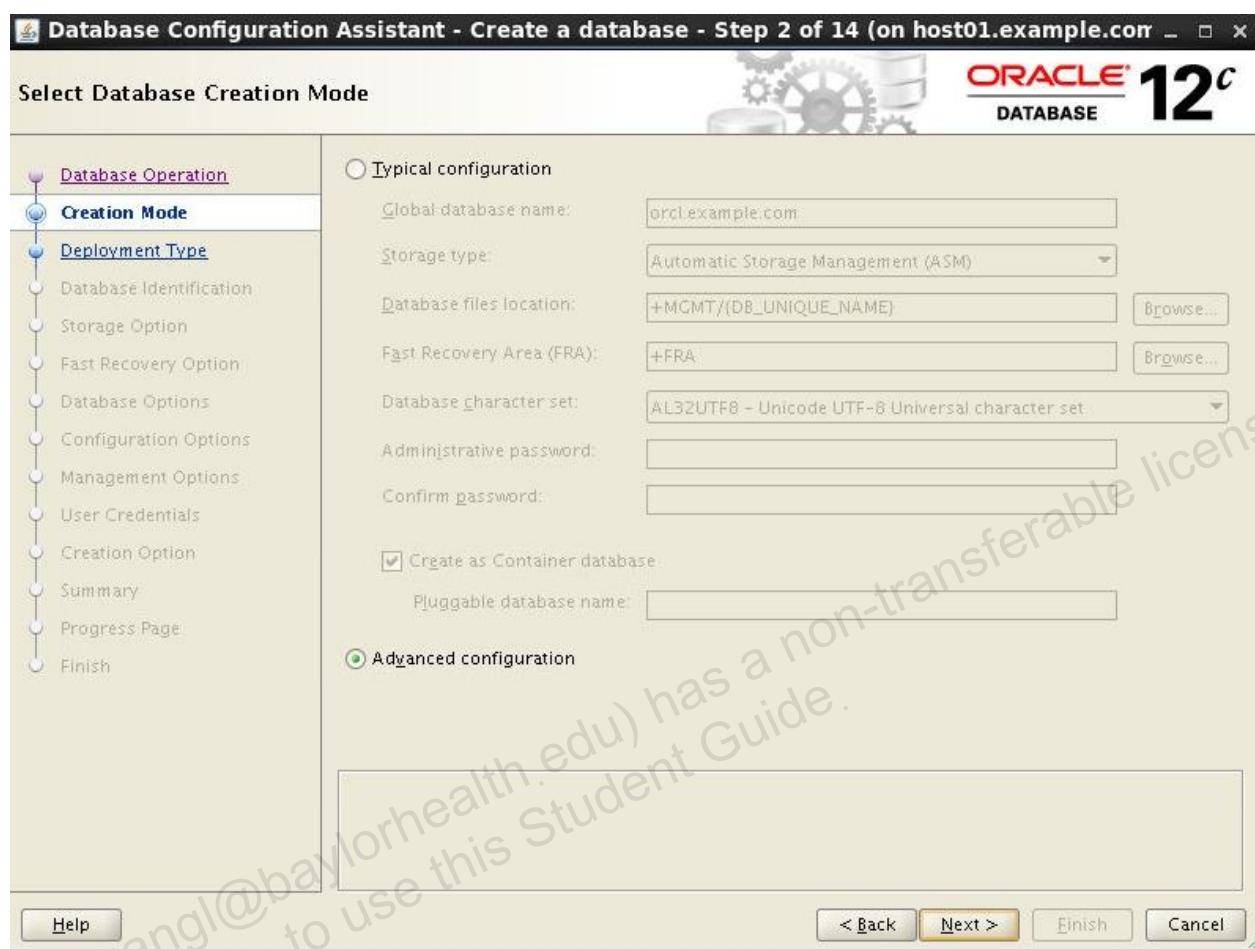
4. Start DBCA and perform the following steps.

```
[oracle@host01 bin]$ ./dbca
```

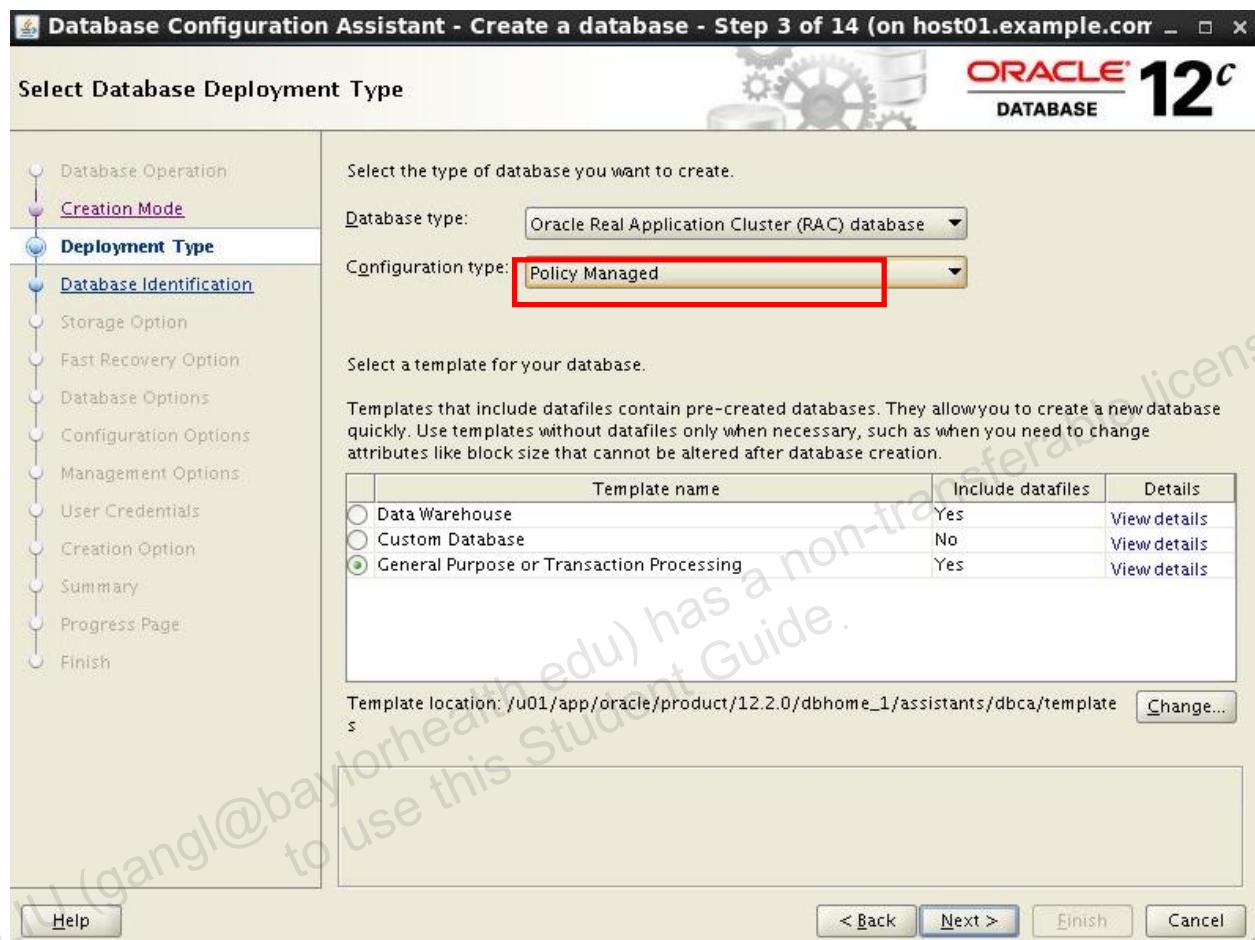
5. On the “Select Database Operation” page, select Create a database and click Next.



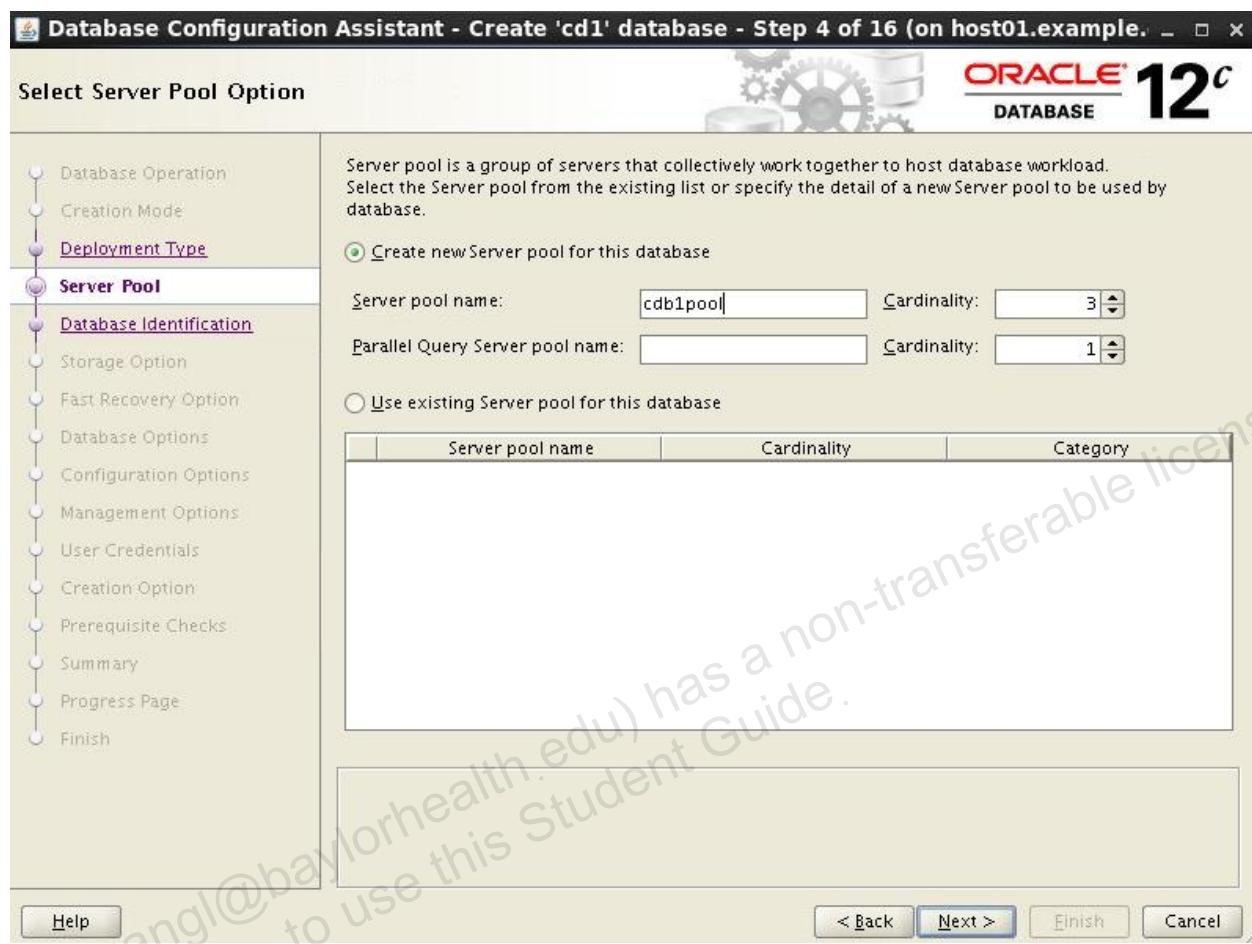
6. On The “Select Database Creation Mode” page, select “Advanced configuration” and click Next.



7. On the “Select Database Deployment Type” page, select “Oracle Real Application Cluster (RAC) database” from the Database type pull-down list and “Policy Managed” from the Configuration type pull-down list. Select the “General Purpose or Transaction Processing” template and click Next.

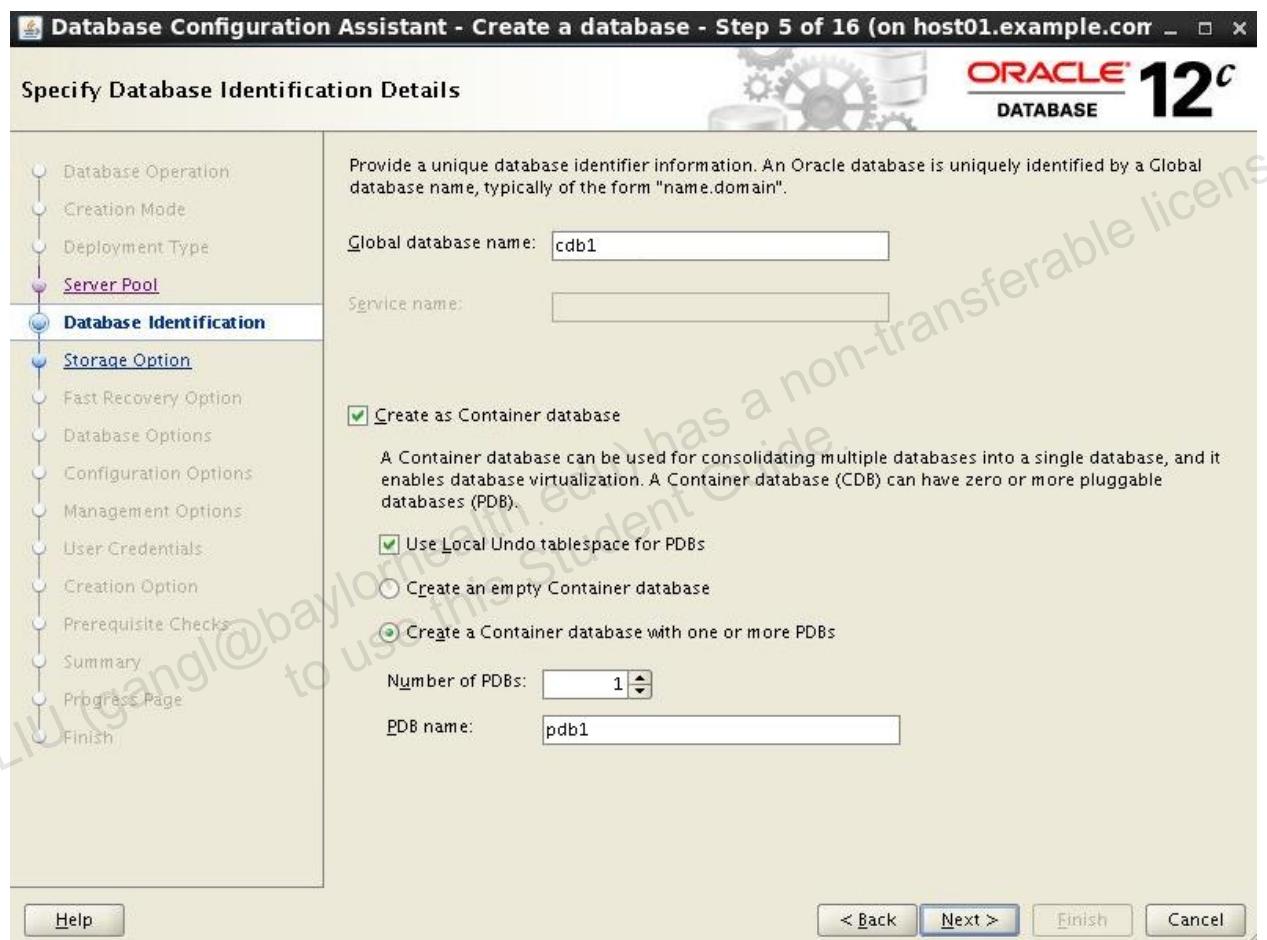


8. On the Select Server Pool Option page, specify **cdb1pool** for the Server pool Name and set its cardinality to **3**. Click Next to proceed

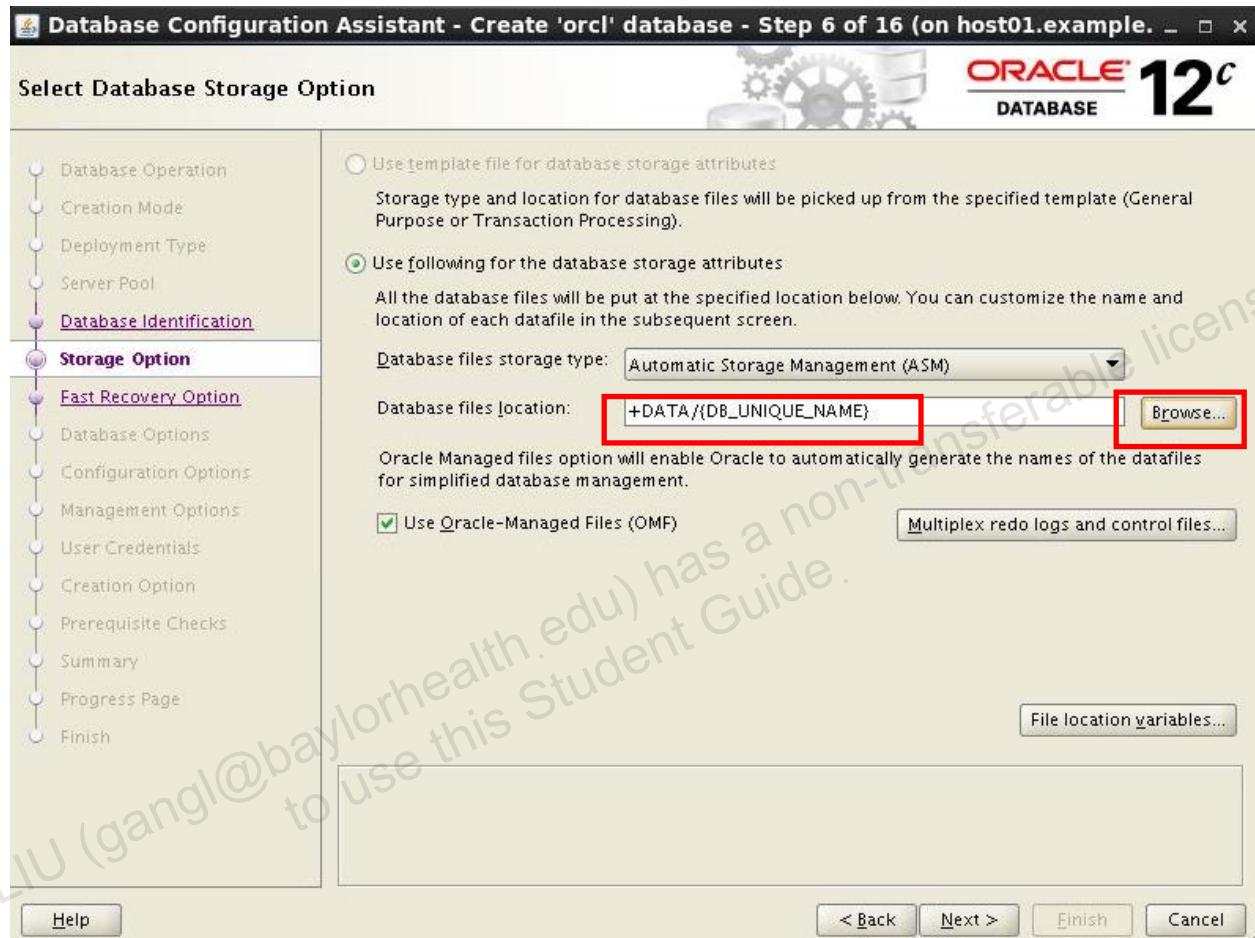


9. Ensure the Global database name is `cdb1`.
- Select “Create as Container Database.”
 - Select “Use Local Undo tablespace for PDBs
 - Select “Create a Container database with one or more PDBs”
 - Select 1 for Number of PDBs
 - Enter `pdb1` for PDB Name.

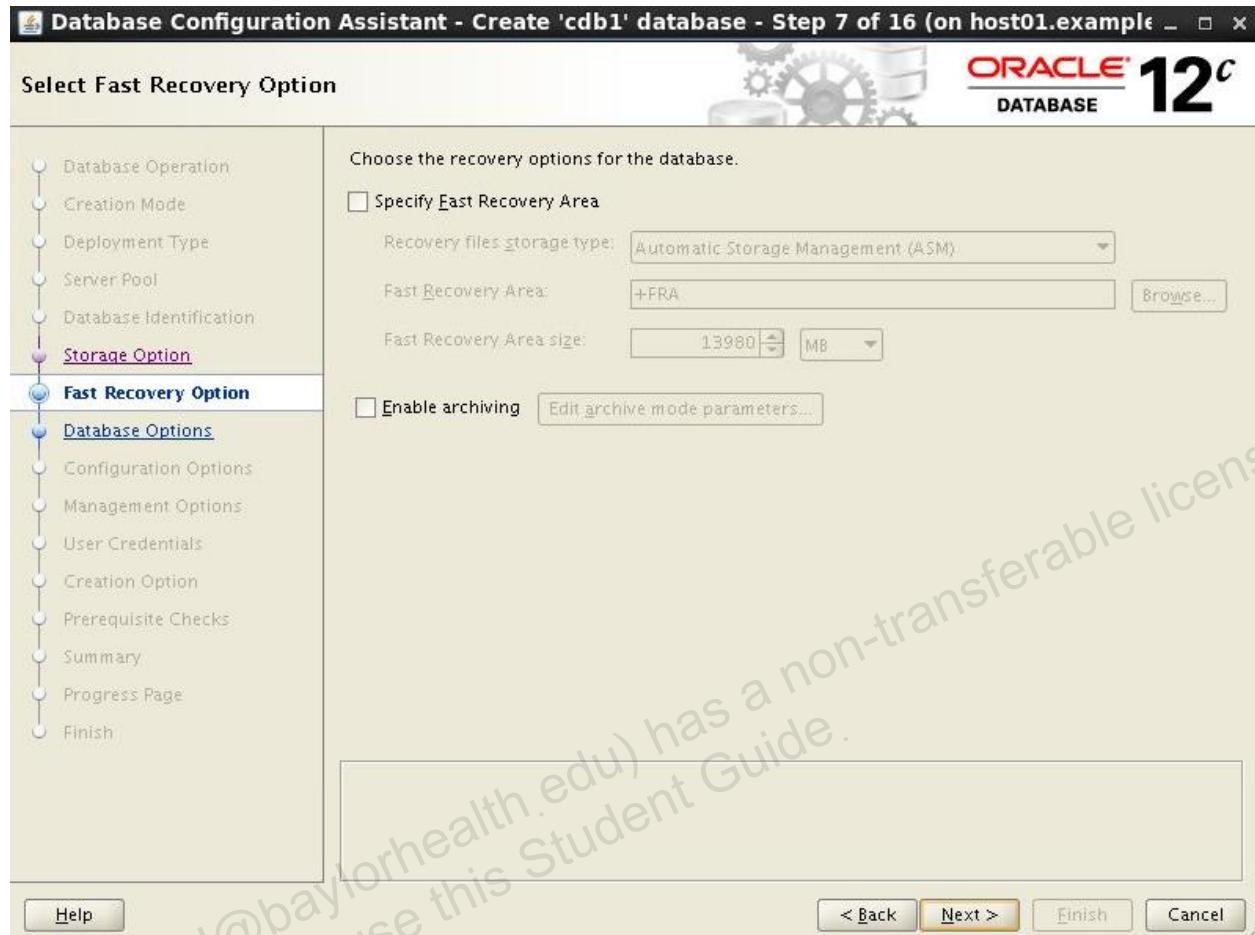
Click Next to continue.



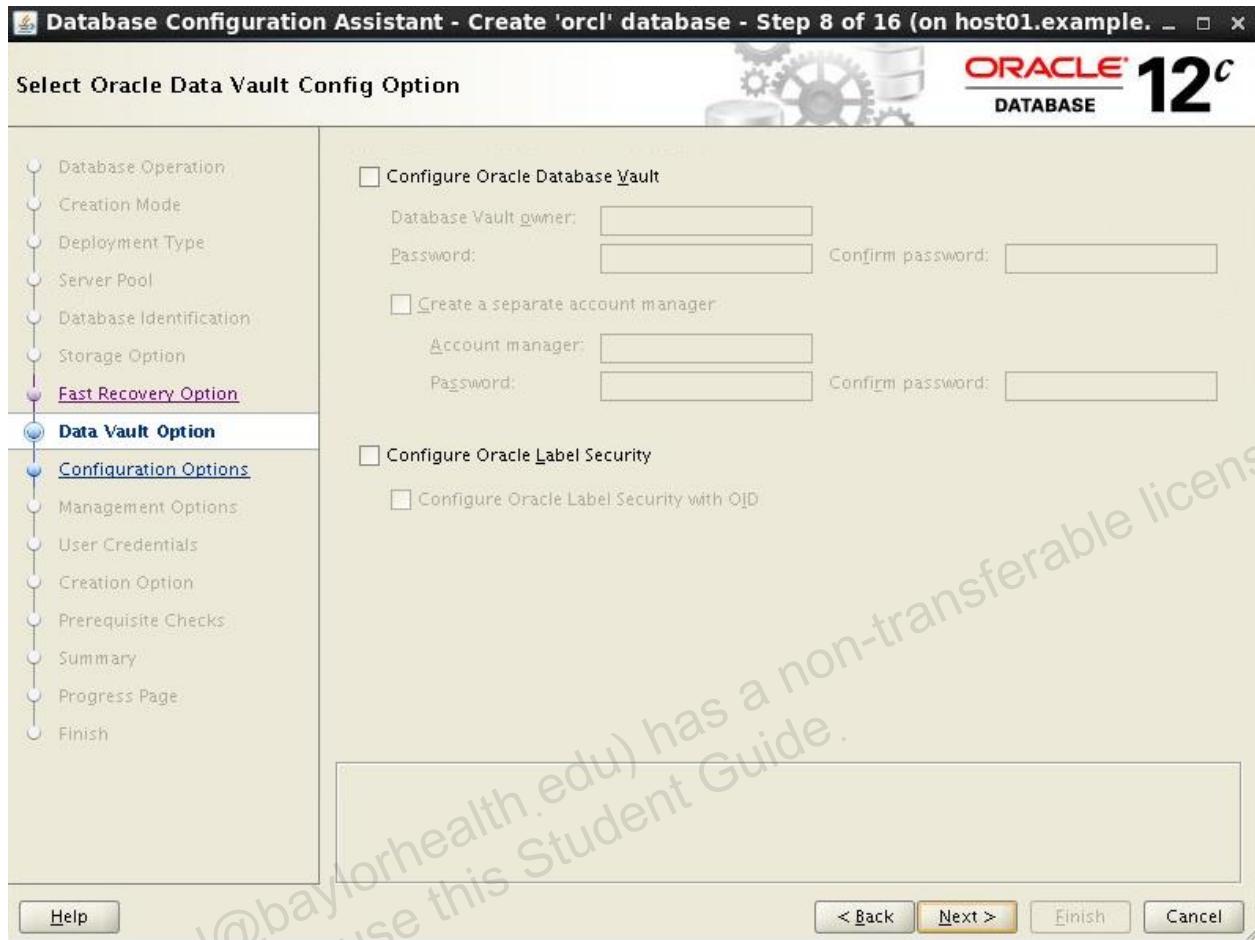
10. On the Select Database Storage Option page, click the “Use following for the database storage attributes” button. Make sure “Automatic Storage Management (ASM)” is selected from the Database files storage type pull down list and the Database files location is **+DATA/{DB_UNIQUE_NAME}**. Make sure the “Use Oracle-Managed Files (OMF)” check box is selected and click Next.



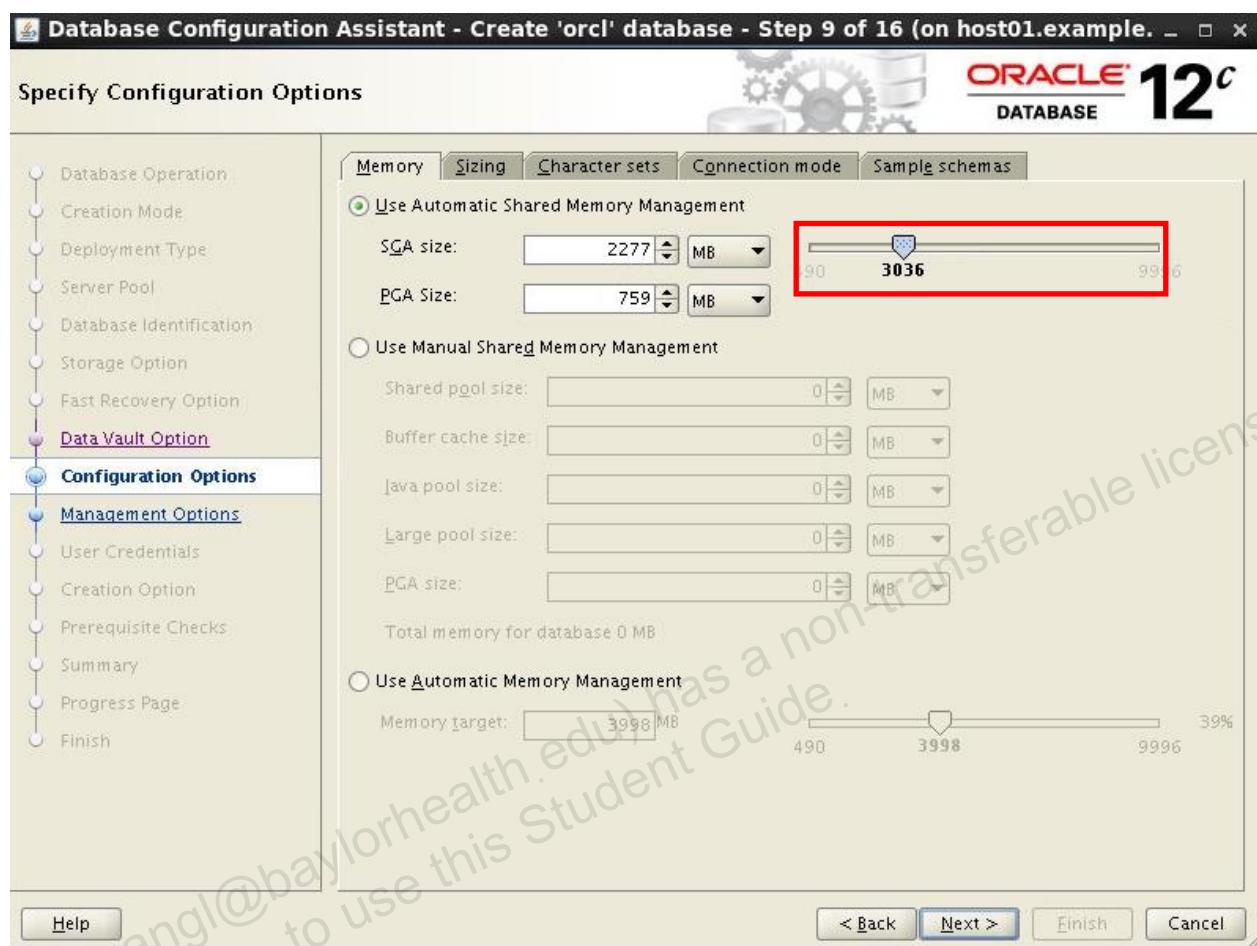
11. On the Select Fast Recovery page, click Next to continue.



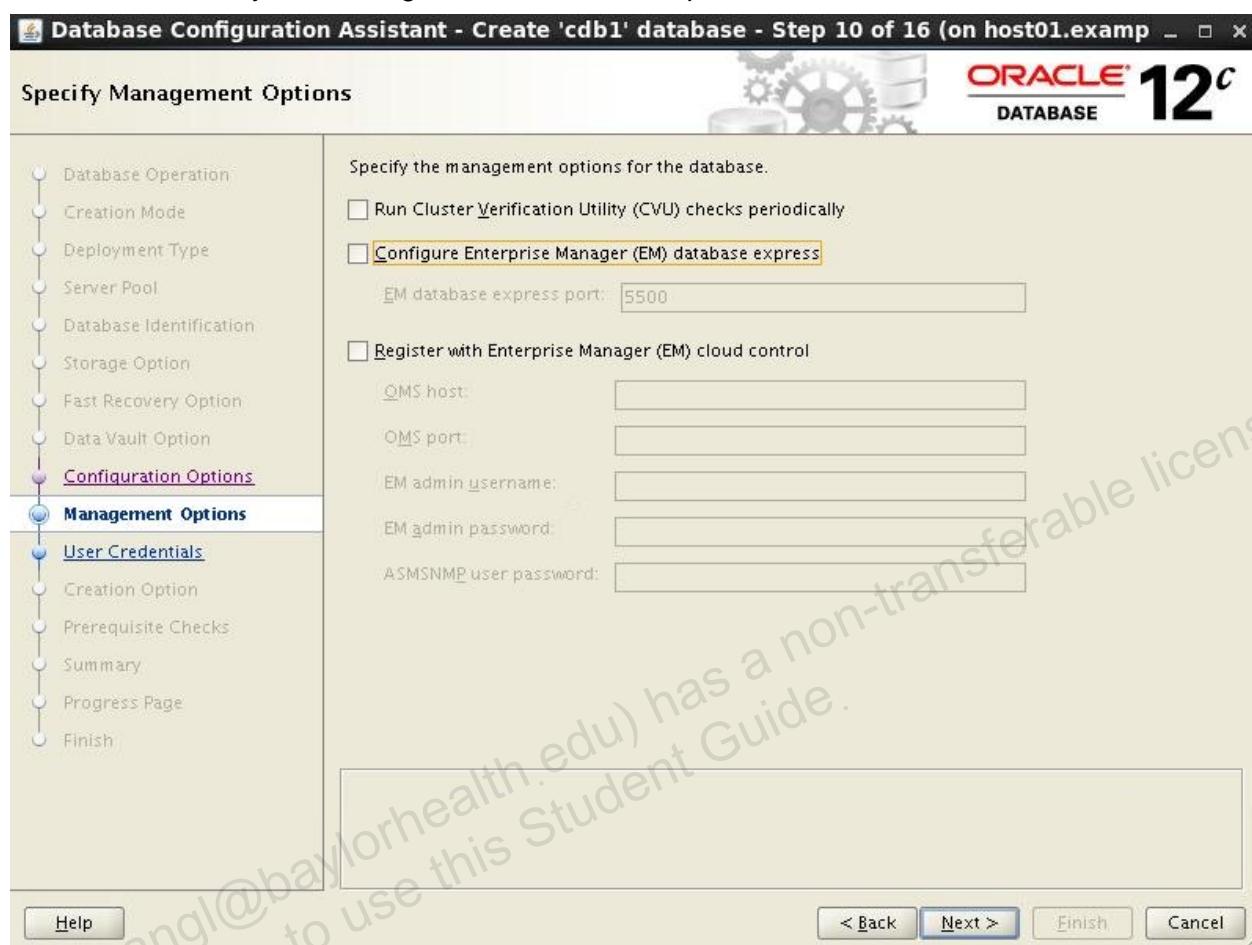
12. On the Select Oracle Data Vault Config Option page, click Next.



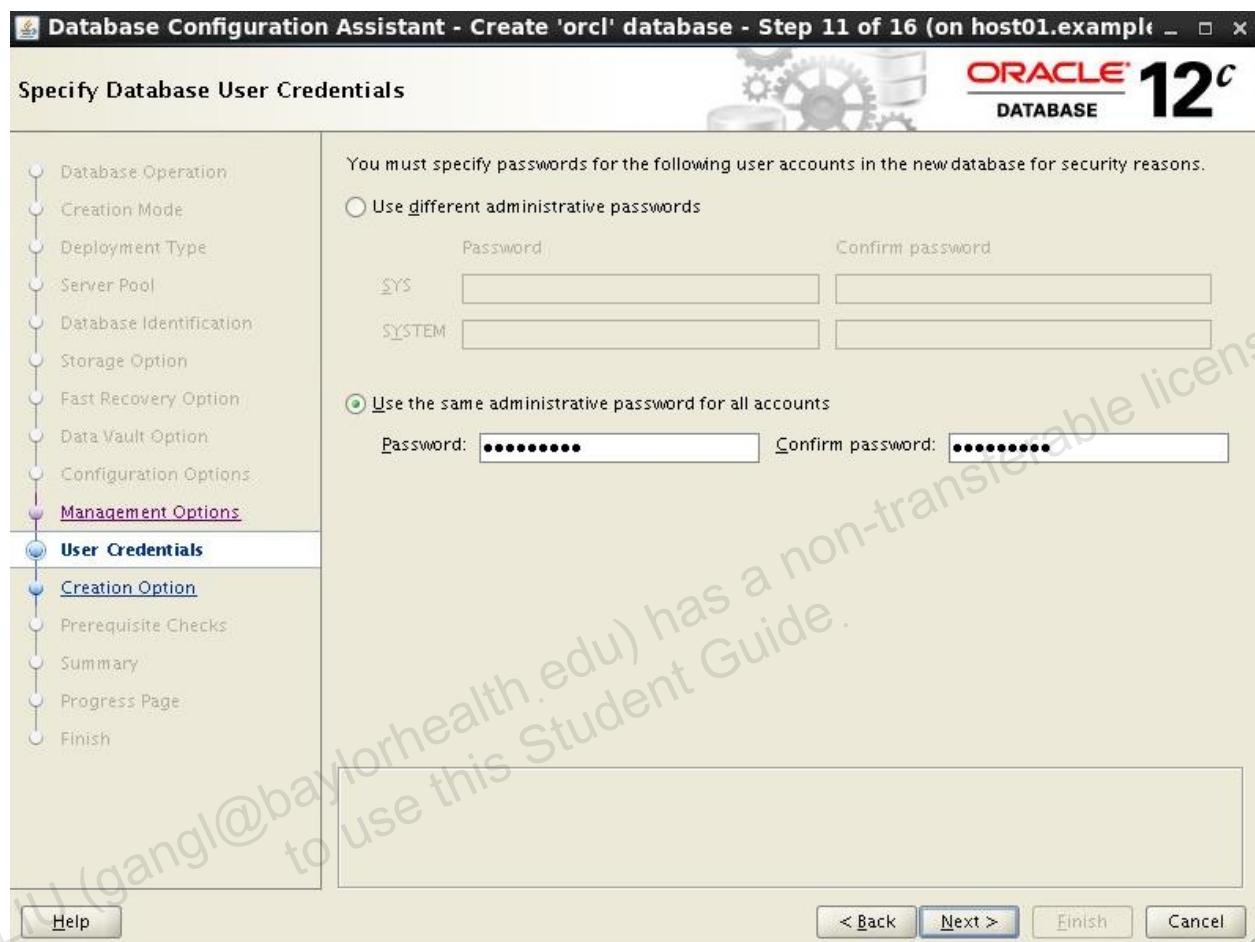
13. On the Specify Configuration Options page, set Memory Size (SGA and PGA) to around 3000. Click Next.



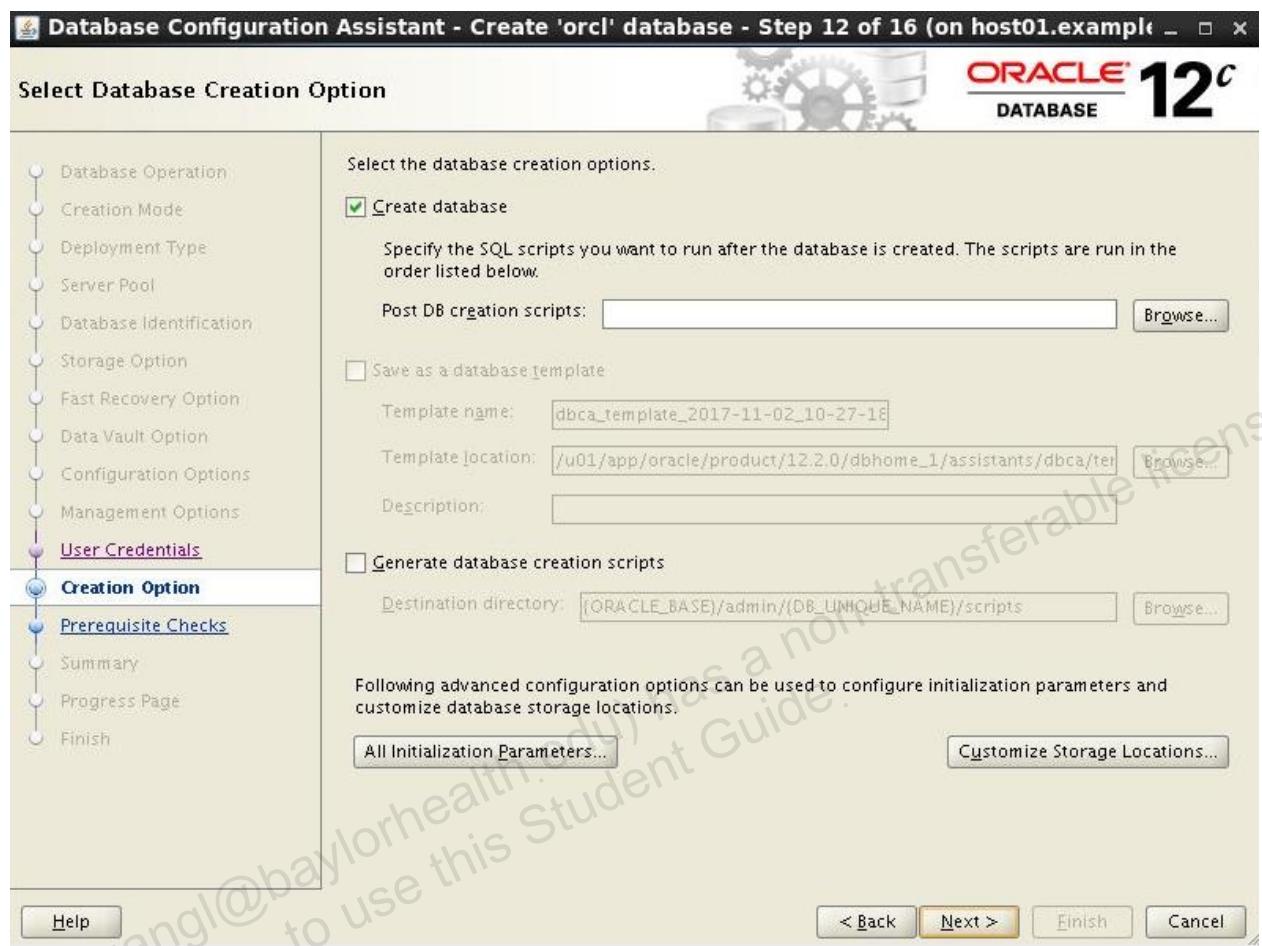
14. On the Management Options screen, uncheck both Run Cluster Verification Utility (CVU) Checks Periodically and Configure EM Database Express. Click next to continue.



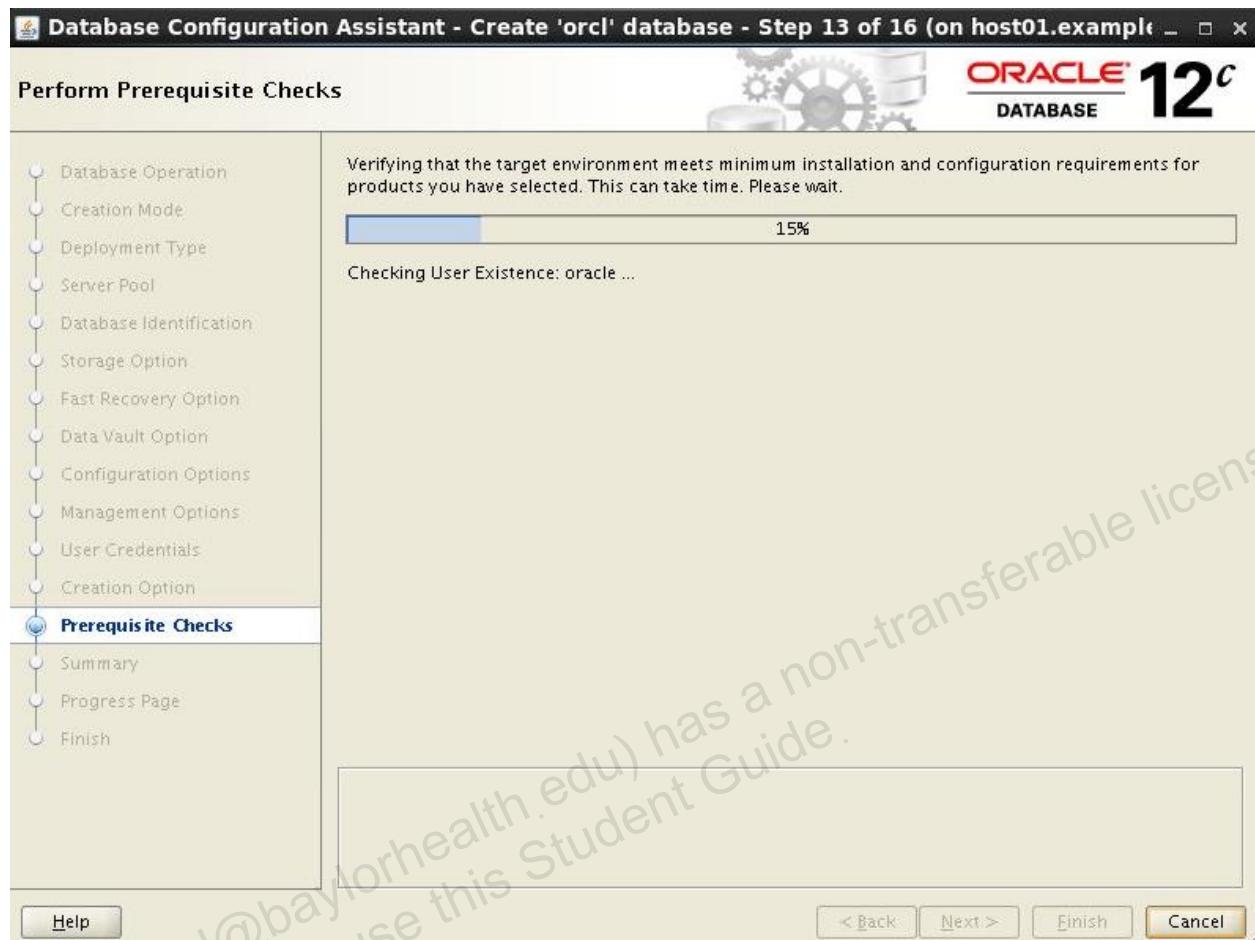
15. On the Specify database User Credentials page, click the “Use the same administrative password for all accounts” button. Enter the `sys` password (*please refer to the Course Practice Environment: Security Credentials page for account passwords*) in both the Password and Confirm Password fields and click Next.



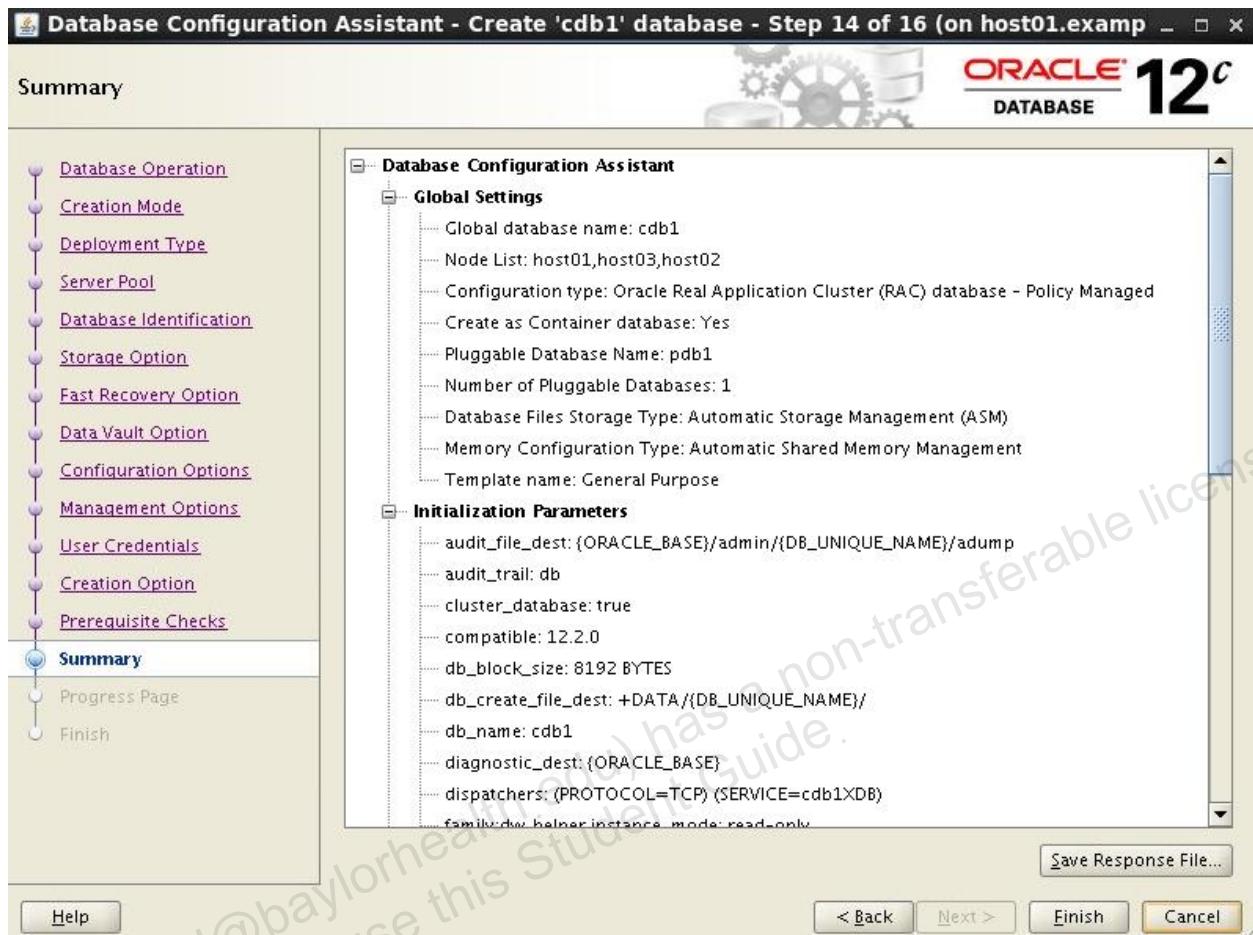
16. On the Select Database Creation Options page, make sure the “Create database” check box is selected and click Next.



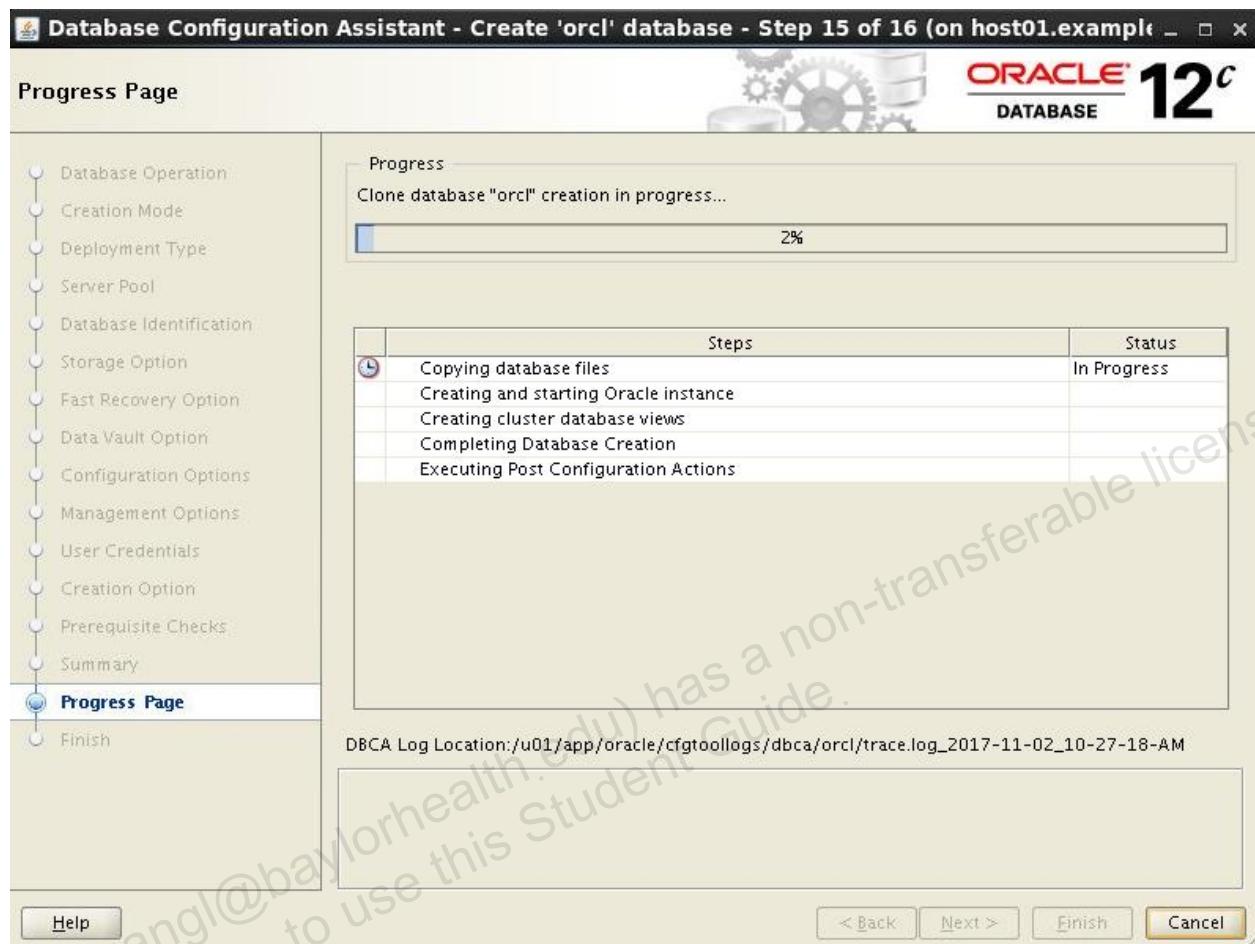
17. Wait while a series of prerequisite checks are performed.



18. On the Summary page, review the information and click Finish.



19. Wait a few moments and follow the database creation process on the Progress Page. *It takes about 20min to complete the rest of steps.*



20. Click Close to dismiss DBCA.
21. Close all terminal windows opened for this practice.

GANG LIU (gangli@baylorhealth.edu) has a non-transferable license
to use this Student Guide.

Practices for Lesson 13: Multitenant Architecture and RAC

Practices for Lesson 13: Overview

Overview

In this practice, you will explore the architecture and structures of `cdb1` and its pluggable databases in RAC.

In this practice, you will clone the `pdb1` PDB into a new PDB named `pdb2` in the `cdb1` CDB. This operation requires to close and open PDBs on multiple instances of the CDB.

In this practice, you will “affinitize” connections to a PDB to one or particular CDB instances. Because server pools determine which services run together or separately, you can configure and maintain required affinity or isolation.

In this practice, you will drop a PDB in the CDB and verify that the services and data files are deleted.

Practice 13-1: Exploring CDB Architecture and Structures in RAC

Overview

In this practice, you will explore the architecture and structures of `cdb1` and its pluggable databases in RAC.

1. Open a terminal window to connect to `host01` as `grid` user and set up the environment variables using the `oraenv` script.

```
[oracle@dns ~]$ ssh grid@host01
grid@host01's password:

[grid@host01 ~]$ . oraenv
ORACLE_SID = [grid] ? +ASM1
The Oracle base has been set to /u01/app/grid

[grid@host01 ~]$
```

2. Check the `cdb1pool` server pool and its cardinality.

```
[grid@host01 ~]$ srvctl status svrpool
Server pool name: Free
Active servers count: 0
Server pool name: Generic
Active servers count: 0
Server pool name: cdb1pool
Active servers count: 3

[grid@host01 ~]$ srvctl status svrpool -serverpool cdb1pool
Server pool name: cdb1pool
Active servers count: 3

[grid@host01 ~]$
```

3. Open a terminal window to connect to `host01` as `oracle` user. Use SRVCTL to know on which nodes the instances of the CDB are running, as you traditionally do for any non-CDB.

```
[oracle@dns ~]$ ssh oracle@host01
oracle@host01's Password:

[oracle@host01 ~]$ export
ORACLE_HOME=/u01/app/oracle/product/12.2.0/dbhome_1
[oracle@host01 ~]$ export PATH=$PATH:$ORACLE_HOME/bin
```

```
[oracle@host01 ~]$ srvctl status database -db cdb1
Instance cdb1_1 is running on node host03
Instance cdb1_2 is running on node host02
Instance cdb1_3 is running on node host01

[oracle@host01 ~]$ pgrep -l cdb1_3
23622 ora_pmon_cdb1_3
23630 ora_clmn_cdb1_3
23632 ora_psp0_cdb1_3
23635 ora_ipc0_cdb1_3
23637 ora_vktm_cdb1_3
23641 ora_gen0_cdb1_3
23643 ora_mman_cdb1_3
23647 ora_scmn_cdb1_3
23651 ora_diag_cdb1_3
23653 ora_scmn_cdb1_3
23657 ora_dbrm_cdb1_3
23659 ora_vkrm_cdb1_3
23661 ora_ping_cdb1_3
23663 ora_svcb_cdb1_3
23665 ora_acms_cdb1_3
23667 ora_pman_cdb1_3
23669 ora_dia0_cdb1_3
23671 ora_lmon_cdb1_3
23673 ora_lmd0_cdb1_3
23675 ora_lms0_cdb1_3
23679 ora_rmv0_cdb1_3
23681 ora_rms0_cdb1_3
23683 ora_lmhb_cdb1_3
23685 ora_lck1_cdb1_3
23687 ora_dbw0_cdb1_3
23689 ora_lgwr_cdb1_3
23691 ora_ckpt_cdb1_3
23693 ora_lg00_cdb1_3
23695 ora_smon_cdb1_3
23697 ora_lg01_cdb1_3
23699 ora_smco_cdb1_3
23701 ora_reco_cdb1_3
23703 ora_w000_cdb1_3
23705 ora_lreg_cdb1_3
23707 ora_w001_cdb1_3
23709 ora_pxmn_cdb1_3
```

```
23711 ora_rbal_cdb1_3  
23713 ora_asmb_cdb1_3  
23715 ora_fenc_cdb1_3  
23717 ora_mmon_cdb1_3  
23719 ora_mmn1_cdb1_3  
23721 ora_d000_cdb1_3  
23723 ora_s000_cdb1_3  
23725 ora_imr0_cdb1_3  
23728 ora_scm0_cdb1_3  
23730 ora_lck0_cdb1_3  
23734 ora_rsmn_cdb1_3  
23736 ora_tmon_cdb1_3  
23740 ora_mark_cdb1_3  
23763 ora_o000_cdb1_3  
24048 ora_tt00_cdb1_3  
24050 ora_tt01_cdb1_3  
24052 ora_tt02_cdb1_3  
24069 ora_gtx0_cdb1_3  
24071 ora_rcbg_cdb1_3  
24073 ora_aqpc_cdb1_3  
24077 ora_o001_cdb1_3  
24084 ora_p000_cdb1_3  
24086 ora_p001_cdb1_3  
24089 ora_p002_cdb1_3  
24097 ora_p003_cdb1_3  
24099 ora_p004_cdb1_3  
24101 ora_p005_cdb1_3  
24103 ora_p006_cdb1_3  
24105 ora_p007_cdb1_3  
24108 ora_cjq0_cdb1_3  
24240 ora_qm02_cdb1_3  
24251 ora_qm05_cdb1_3  
24256 ora_q002_cdb1_3  
24261 ora_qm03_cdb1_3  
24269 ora_q004_cdb1_3  
24807 ora_w002_cdb1_3  
24909 ora_gcr0_cdb1_3  
25577 ora_w003_cdb1_3  
25770 ora_w004_cdb1_3  
26737 ora_w005_cdb1_3  
28105 ora_ppa7_cdb1_3  
28116 ora_ppa6_cdb1_3
```

```
28325 ora_p008_cdb1_3
28327 ora_p009_cdb1_3

[oracle@host01 bin]$
```

4. Use LSNRCTL to list the CDB instances on two nodes of the server pool.
 a. Check the services on the first node.

```
[oracle@host01 ~]$ lsnrctl status

LSNRCTL for Linux: Version 12.2.0.1.0 - Production on 09-NOV-
2017 15:04:52

Copyright (c) 1991, 2016, Oracle. All rights reserved.

Connecting to (ADDRESS=(PROTOCOL=tcp) (HOST=) (PORT=1521))
STATUS of the LISTENER
-----
Alias                      LISTENER
Version                    TNSLSNR for Linux: Version 12.2.0.1.0
- Production
Start Date                 01-NOV-2017 14:29:38
Uptime                     8 days 0 hr. 35 min. 15 sec
Trace Level                off
Security                   ON: Local OS Authentication
SNMP                       OFF
Listener Parameter File    /u01/app/12.2.0/grid/network/admin/listener.ora
Listener Log File          /u01/app/grid/diag/tnslsnr/host01/listener/alert/log.x
ml
Listening Endpoints Summary...
  (DESCRIPTION=(ADDRESS=(PROTOCOL=ipc) (KEY=LISTENER)))

  (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp) (HOST=192.0.2.101) (PORT=1521
)))

  (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp) (HOST=192.0.2.254) (PORT=1521
)))
Services Summary...
Service "+ASM" has 1 instance(s).
  Instance "+ASM1", status READY, has 1 handler(s) for this
service...
Service "+ASM_DATA" has 1 instance(s).
```

```

Instance "+ASM1", status READY, has 1 handler(s) for this
service...
Service "+ASM_FRA" has 1 instance(s).
Instance "+ASM1", status READY, has 1 handler(s) for this
service...
Service "+ASM_MGMT" has 1 instance(s).
Instance "+ASM1", status READY, has 1 handler(s) for this
service...
Service "5d8efa95565c5e5ee053650200c04cd5" has 1 instance(s).
Instance "cdb1_3", status READY, has 1 handler(s) for this
service...
Service "cdb1" has 1 instance(s).
Instance "cdb1_3", status READY, has 1 handler(s) for this
service...
Service "cdb1XDB" has 1 instance(s).
Instance "cdb1_3", status READY, has 1 handler(s) for this
service...
Service "pdb1" has 1 instance(s).
Instance "cdb1_3", status READY, has 1 handler(s) for this
service...
The command completed successfully

[oracle@host01 ~]$
```

b. Check the services on the second node.

```

[oracle@host01 ~]$ ssh host02
Last login: Mon Sep  2 01:17:02 2014 from 192.0.2.1
[oracle@host02 ~]$ . oraenv
ORACLE_SID = [oracle] ? cdb1_2
ORACLE_HOME = [/home/oracle] ?
/u01/app/oracle/product/12.2.0/dbhome_1
The Oracle base has been set to /u01/app/oracle

[oracle@host02 ~]$ lsnrctl status

Connecting to (ADDRESS=(PROTOCOL=tcp) (HOST=) (PORT=1521))
STATUS of the LISTENER
-----
Alias                      LISTENER
Version                    TNSLSNR for Linux: Version 12.2.0.1.0
- Production
Start Date                 01-NOV-2017 14:40:09
Uptime                     8 days 0 hr. 27 min. 9 sec
Trace Level                off
```

```

Security          ON: Local OS Authentication
SNMP             OFF
Listener Parameter File
/u01/app/12.2.0/grid/network/admin/listener.ora
Listener Log File
/u01/app/grid/diag/tnslsnr/host02/listener/alert/log.xml
Listening Endpoints Summary...
  (DESCRIPTION=(ADDRESS=(PROTOCOL=ipc) (KEY=LISTENER)))

  (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp) (HOST=192.0.2.102) (PORT=1521
)))

  (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp) (HOST=192.0.2.231) (PORT=1521
)))

Services Summary...
Service "+ASM" has 1 instance(s).
  Instance "+ASM3", status READY, has 1 handler(s) for this
service...
Service "+ASM_DATA" has 1 instance(s).
  Instance "+ASM3", status READY, has 1 handler(s) for this
service...
Service "+ASM_FRA" has 1 instance(s).
  Instance "+ASM3", status READY, has 1 handler(s) for this
service...
Service "+ASM_MGMT" has 1 instance(s).
  Instance "+ASM3", status READY, has 1 handler(s) for this
service...
Service "5d8efa95565c5e5ee053650200c04cd5" has 1 instance(s).
  Instance "cdb1_2", status READY, has 1 handler(s) for this
service...
Service "cdb1" has 1 instance(s).
  Instance "cdb1_2", status READY, has 1 handler(s) for this
service...
Service "cdb1XDB" has 1 instance(s).
  Instance "cdb1_2", status READY, has 1 handler(s) for this
service...
Service "pdb1" has 1 instance(s).
  Instance "cdb1_2", status READY, has 1 handler(s) for this
service...
The command completed successfully

[oracle@host02 ~]$ exit
logout
Connection to host02 closed.
[oracle@host01 ~]$

```

5. Use SRVCTL to stop and restart the CDB as you traditionally would do for any non-CDB.

```
[oracle@host01 ~]$ . oraenv
ORACLE_SID = [oracle] ? cdb1_3
ORACLE_HOME = [/home/oracle] ?
/u01/app/oracle/product/12.2.0/dbhome_1
The Oracle base has been set to /u01/app/oracle

[oracle@host01 ~]$ srvctl stop database -db cdb1

[oracle@host01 ~]$ srvctl status database -db cdb1
Instance cdb1_1 is not running on node host03
Instance cdb1_2 is not running on node host02
Instance cdb1_3 is not running on node host01

[oracle@host01 ~]$ srvctl start database -db cdb1

[oracle@host01 ~]$ srvctl status database -db cdb1
Instance cdb1_1 is running on node host03
Instance cdb1_2 is running on node host02
Instance cdb1_3 is running on node host01

[oracle@host01 ~]$
```

6. Use SQL*Plus to connect to the instances of the `cdb1` CDB, check the UNDO tablespaces and the groups of redo log files, and verify the existence of the `pdb1` PDB.

- a. Check the UNDO tablespaces created in the CDB.

```
[oracle@host01 ~]$ sqlplus / as sysdba

SQL*Plus: Release 12.2.0.1.0 Production on Thu Nov 9 15:10:40
2017

Copyright (c) 1982, 2016, Oracle. All rights reserved.

Connected to:
Oracle Database 12c Enterprise Edition Release 12.2.0.1.0 -
64bit Production

SQL> SELECT name, cdb, con_id FROM v$database;

NAME          CDB      CON_ID
-----  -----  -----

```

```
CDB1      YES      0

SQL> SELECT instance_name, con_id FROM v$instance;

INSTANCE_NAME          CON_ID
-----
cdb1_3                  0

SQL> show con_name

CON_NAME
-----
CDB$ROOT

SQL> SELECT tablespace_name, con_id
  FROM   cdb_tablespaces
 WHERE  contents = 'UNDO';

TABLESPACE_NAME          CON_ID
-----
UNDOTBS1                  1
UNDOTBS2                  1
UNDOTBS3                  1

SQL>
```

- b. Check the groups of redo log files created for the three CDB instances.

```
SQL> SELECT group#, con_id FROM v$logfile order by 1;

GROUP#      CON_ID
-----
1            0
2            0
3            0
4            0
5            0
6            0
7            0
8            0
9            0
```

```
9 rows selected.
```

```
SQL>
```

- c. Check the PDB created in the CDB and its open mode. If the PDB is not opened, open it.

```
SQL> COL pdb_name format a10
SQL> SELECT pdb_id, pdb_name, guid, status FROM cdb_pdbs;
```

PDB_ID	PDB_NAME	GUID	STATUS
3	PDB1	5D8EFA95565C5E5EE053650200C04CD5	NORMAL
2	PDB\$SEED	5D8EDD1013E04888E053650200C02A7D	NORMAL

```
SQL> SELECT name, open_mode FROM v$pdbs;
```

NAME	OPEN_MODE
PDB\$SEED	READ ONLY
PDB1	MOUNTED

```
SQL> ALTER SESSION SET CONTAINER=pdb1;
```

```
Session altered.
```

```
SQL> show con_name
```

CON_NAME
PDB1

```
SQL> CONNECT / AS SYSDBA
```

```
Connected.
```

```
SQL> SELECT name FROM cdb_services;
```

NAME
SYS\$BACKGROUND
SYS\$USERS
cdb1XDB
cdb1

```
SQL> ALTER PLUGGABLE DATABASE pdb1 OPEN;

Pluggable database altered.

SQL> SELECT name, open_mode FROM v$pdbs;

NAME                      OPEN_MODE
-----
PDB$SEED                  READ ONLY
PDB1                       READ WRITE

SQL>
```

d. Check the services.

```
SQL> SELECT name FROM v$services;

NAME
-----
pdb1
cdb1XDB
cdb1
SYS$BACKGROUND
SYS$USERS

SQL> EXIT
Disconnected from Oracle Database 12c Enterprise Edition Release
12.2.0.1.0 - 64bit Production
[oracle@host01 ~]$
```

7. Switch to the second node to verify the open mode of the PDB in the second instance of the CDB.

```
[oracle@host01 ~]$ ssh host02
[oracle@host02 ~]$ . oraenv
ORACLE_SID = [oracle] ? cdb1_2
ORACLE_HOME = [/home/oracle] ?
/u01/app/oracle/product/12.2.0/dbhome_1
The Oracle base has been set to /u01/app/oracle

[oracle@host02 ~]$ sqlplus / as sysdba
```

```
SQL*Plus: Release 12.2.0.1.0 Production on Thu Nov 9 15:15:56  
2017
```

```
Copyright (c) 1982, 2016, Oracle. All rights reserved.
```

```
Connected to:
```

```
Oracle Database 12c Enterprise Edition Release 12.2.0.1.0 -  
64bit Production
```

```
SQL> SELECT name, cdb, con_id FROM v$database;
```

NAME	CDB	CON_ID
CDB1	YES	0

```
SQL> SELECT instance_name, con_id FROM v$instance;
```

INSTANCE_NAME	CON_ID
cdb1_2	0

```
SQL> show con_name
```

CON_NAME
CDB\$ROOT

```
SQL> SELECT name, open_mode FROM v$pdbs;
```

NAME	OPEN_MODE
PDB\$SEED	READ ONLY
PDB1	MOUNTED

```
SQL> ALTER SESSION SET CONTAINER=pdb1;
```

```
Session altered.
```

```
SQL> SELECT name FROM v$services;
```

NAME
pdb1

```
SQL> exit
[oracle@host02 ~]$ exit
logout
Connection to host02 closed.
[oracle@host01 ~]$
```

8. Verify that the `pdb1` service is accessible from instance `cdb1_3` on the first node but also from `cdb1_2` instance on the second node and from `cdb1_1` instance on the third node. First restart the listener.

```
[oracle@host01 ~]$ su - grid
Password:

[grid@host01 ~]$ . oraenv
ORACLE_SID = [grid] ? +ASM1
The Oracle base has been set to /u01/app/grid

[grid@host01 ~]$ srvctl stop listener -listener LISTENER

[grid@host01 ~]$ srvctl start listener -listener LISTENER

[grid@host01 ~]$ exit
logout

[oracle@host01 ~]$ sqlplus /nolog
```

```
SQL*Plus: Release 12.2.0.1.0 Production on Thu Nov 9 15:17:54
2017
```

```
Copyright (c) 1982, 2016, Oracle. All rights reserved.
```

```
SQL> CONNECT system@"host01:1521/pdb1"
Enter password:
Connected.
SQL> SELECT instance_name, con_id FROM v$instance;

INSTANCE_NAME          CON_ID
-----
cdb1_3                  0

SQL> show con_name

CON_NAME
```

```
PDB1

SQL> CONNECT system@"host02:1521/pdb1"
Enter password:
ERROR:
ORA-01033:ORACLE initialization or shutdown in progress
Process ID: 0
Session ID: 0 Serial Number: 0

Warning: You are no longer connected to ORACLE.
SQL>
```

Notice that the connection does not complete because pdb1 was opened for instance cdb1_3 on host01 only. Remember that the clause INSTANCES was not used in the ALTER PLUGGABLE DATABASE OPEN statement in the previous task.

```
SQL> CONNECT / AS SYSDBA
Connected.
SQL> ALTER PLUGGABLE DATABASE pdb1 OPEN INSTANCE=( 'cdb1_2' );

Pluggable database altered.
```

```
SQL> CONNECT system@"host02:1521/pdb1"
Enter password:
Connected.

SQL> SELECT instance_name, con_id FROM v$instance;

INSTANCE_NAME          CON_ID
-----
cdb1_2                  0

SQL> show con_name

CON_NAME
-----
PDB1

SQL> CONNECT system@"host03:1521/pdb1"
Enter password:
ERROR:
ORA-01033:ORACLE initialization or shutdown in progress
```

```
Process ID: 0
Session ID: 0 Serial Number: 0

Warning: You are no longer connected to ORACLE.

SQL> EXIT
```

The connection does not complete on host03 because `pdb1` was opened for instance `cdb1_3` on host01 and `cdb1_2` on host02 only.

9. Close all terminal windows opened for this practice.

Practice 13-2: Cloning a PDB in the RAC CDB

Overview

In this practice, you will clone the `pdb1` PDB into a new PDB named `pdb2` in the `cdb1` CDB. This operation requires to close and open PDBs on multiple instances of the CDB.

Tasks

1. Open a terminal window connected to `host01` as the `oracle` user.

```
[oracle@dns ~]$ ssh oracle@host01
oracle@host01's password:

[oracle@host01 ~]$
```

2. Connect to the root of the multitenant container database `cdb1` on any of the three instances.

```
[oracle@host01 ~]$ . oraenv
ORACLE_SID = [cdb1] ? cdb1_3
ORACLE_HOME = [/home/oracle] ?
/u01/app/oracle/product/12.2.0/dbhome_1
The Oracle base has been set to /u01/app/oracle

[oracle@host01 ~]$ sqlplus / as sysdba

SQL*Plus: Release 12.2.0.1.0 Production on Fri Nov 10 07:45:57
2017

Copyright (c) 1982, 2016, Oracle. All rights reserved.

Connected to:
Oracle Database 12c Enterprise Edition Release 12.2.0.1.0 -
64bit Production
SQL>
```

3. Use Oracle Managed Files to locate the data files of the new `pdb2`.

```
SQL> SHOW PARAMETER db_create_file_dest

NAME                                     TYPE        VALUE
-----
db_create_file_dest                      string      +DATA

SQL>
```

4. Create pdb2 from pdb1.

- a. Use the CREATE PLUGGABLE DATABASE command to create pdb2.

```
SQL> CREATE PLUGGABLE DATABASE pdb2 FROM pdb1;

Pluggable database created.

SQL> SELECT name, open_mode FROM v$pdbs;

NAME          OPEN_MODE
-----
PDB$SEED      READ ONLY
PDB1          READ WRITE
PDB2          MOUNTED

SQL>
```

- b. Now, open both PDBs in READ WRITE mode on all the CDB instances.

```
SQL> ALTER PLUGGABLE DATABASE pdb1 CLOSE IMMEDIATE
INSTANCES=ALL;

Pluggable database altered.

SQL> ALTER PLUGGABLE DATABASE ALL OPEN READ WRITE INSTANCES=ALL;

Pluggable database altered.

SQL> SELECT name, open_mode FROM v$pdbs;

NAME          OPEN_MODE
-----
PDB$SEED      READ ONLY
PDB1          READ WRITE
PDB2          READ WRITE

SQL> SELECT pdb_id, pdb_name, guid, status FROM cdb_pdbs;

PDB_ID  PDB_NAME    GUID                      STATUS
-----  -----
3       PDB1        5D8F3DBBF1449ACE053660200C0ED13  NORMAL
2       PDB$SEED    5D8F1F2AA1223683E053650200C0DCF8  NORMAL
4       PDB2        5D9D2399A0A130DBE053650200C0C9CD  NORMAL
```

```
SQL> ALTER SESSION SET CONTAINER=pdb2;

Session altered.

SQL> SELECT name FROM dba_services;

NAME
-----
pdb2

SQL> EXIT
```

- c. Use LSNRCTL to verify that the new pdb2 service associated to the new PDB in the CDB instance is automatically started after the PDB is opened. Because the PDB is opened in all the CDB instances, the pdb2 PDB service is started in all the CDB instances.

```
[oracle@host01 ~]$ lsnrctl status

LSNRCTL for Linux: Version 12.2.0.1.0 - Production on 10-NOV-2017 07:50:47

Copyright (c) 1991, 2016, Oracle. All rights reserved.

Connecting to (ADDRESS=(PROTOCOL=tcp) (HOST=) (PORT=1521))
STATUS of the LISTENER
-----
Alias                      LISTENER
Version                    TNSLSNR for Linux: Version 12.2.0.1.0
- Production
Start Date                 09-NOV-2017 12:42:01
Uptime                     0 days 19 hr. 8 min. 46 sec
Trace Level                off
Security                   ON: Local OS Authentication
SNMP                       OFF
Listener Parameter File    /u01/app/12.2.0/grid/network/admin/listener.ora
Listener Log File          /u01/app/grid/diag/tnslsnr/host01/listener/alert/log.xml
Listening Endpoints Summary...
(DESCRIPTION=(ADDRESS=(PROTOCOL=ipc) (KEY=LISTENER)))

(DESCRIPTION=(ADDRESS=(PROTOCOL=tcp) (HOST=192.0.2.101) (PORT=1521)))
```

```
(DESCRIPTION=(ADDRESS=(PROTOCOL=tcp) (HOST=192.0.2.242) (PORT=1521
)))
Services Summary...
Service "+ASM" has 1 instance(s).
  Instance "+ASM1", status READY, has 1 handler(s) for this
service...
Service "+ASM_DATA" has 1 instance(s).
  Instance "+ASM1", status READY, has 1 handler(s) for this
service...
Service "+ASM_MGMT" has 1 instance(s).
  Instance "+ASM1", status READY, has 1 handler(s) for this
service...
Service "5d8f3dbbf1449ace053660200c0ed13" has 1 instance(s).
  Instance "cdb1_3", status READY, has 1 handler(s) for this
service...
Service "5d9d2399a0a130dbe053650200c0c9cd" has 1 instance(s).
  Instance "cdb1_3", status READY, has 1 handler(s) for this
service...
Service "cdb1" has 1 instance(s).
  Instance "cdb1_3", status READY, has 1 handler(s) for this
service...
Service "cdb1XDB" has 1 instance(s).
  Instance "cdb1_3", status READY, has 1 handler(s) for this
service...
Service "pdb1" has 1 instance(s).
  Instance "cdb1_3", status READY, has 1 handler(s) for this
service...
Service "pdb2" has 1 instance(s).
  Instance "cdb1_3", status READY, has 1 handler(s) for this
service...
The command completed successfully

[oracle@host01 ~]$
```

5. Use the net service name to connect to `pdb2` as `system` user on any of the three instances of the CDB.

```
[oracle@host01 ~]$ sqlplus /nolog
SQL> CONNECT system@"host01:1521/pdb2"
Enter password:
Connected.

SQL> show con_name
CON_NAME
```

PDB2

```
SQL> CONNECT system@"host02:1521/pdb2"
```

```
Enter password:
```

```
Connected.
```

```
SQL> CONNECT system@"host03:1521/pdb2"
```

```
Enter password:
```

```
Connected.
```

```
SQL> EXIT
```

Practice 13-3: Affinitizing PDB Services to CDB Instances

Overview

In this practice, you will “affinitize” connections to a PDB to one or particular CDB instances. Because server pools determine which services run together or separately, you can configure and maintain required affinity or isolation.

Task 1: Affinize connections to pdb1 to all the CDB instances

- Check the configuration of the server pools.

```
[oracle@host01 ~]$ srvctl status svrpool
Server pool name: Free
Active servers count: 0
Server pool name: Generic
Active servers count: 0
Server pool name: cdb1pool
Active servers count: 3
[oracle@host01 ~]$

[oracle@host01 ~]$ srvctl config svrpool -serverpool cdb1pool
Server pool name: cdb1pool
Importance: 0, Min: 0, Max: 3
Category: hub
Candidate server names:
[oracle@host01 ~]$
```

- Check the services. You notice that the default services created at PDB creation are not managed by the clusterware.

```
[oracle@host01 ~]$ srvctl status service -db cdb1
[oracle@host01 ~]$ srvctl config service -db cdb1
[oracle@host01 ~]$
```

- Create a dynamic PDB service for the `pdb1` PDB in the CDB which will “affinitize” connections to `pdb1` to all the CDB instances uniformly.

- Create the service from the connection on the first node of the server pool.

```
[oracle@host01 ~]$ srvctl add service -db cdb1 -pdb pdb1 -
service mypdb1serv -policy automatic -serverpool cdb1pool -
cardinality uniform

[oracle@host01 ~]$ srvctl config service -db cdb1
Service name: mypdb1serv
```

```

Service is enabled
Server pool: cdb1pool
Cardinality: UNIFORM
Disconnect: false
Service role: PRIMARY
Management policy: AUTOMATIC
DTP transaction: false
AQ HA notifications: false
Global: false
Commit Outcome: false
Failover type:
Failover method:
TAF failover retries:
TAF failover delay:
Connection Load Balancing Goal: LONG
Runtime Load Balancing Goal: NONE
TAF policy specification: NONE
Edition:
Pluggable database name: pdb1
Maximum lag time: ANY
SQL Translation Profile:
Retention: 86400 seconds
Replay Initiation Time: 300 seconds
Session State Consistency:
Service is enabled on nodes:
Service is disabled on nodes:

[oracle@host01 ~]$

```

- b. Check that the PDB service is also created on the two other nodes of the server pool.

```

[oracle@host01 ~]$ ssh host02
[oracle@host02 ~]$
[oracle@host02 ~]$. oraenv
ORACLE_SID = [oracle] ? cdb1_2
ORACLE_HOME = [/home/oracle] ?
/u01/app/oracle/product/12.2.0/dbhome_1
The Oracle base has been set to /u01/app/oracle

```

```

[oracle@host02 ~]$ srvctl config service -db cdb1
Service name: mypdb1serv
Service is enabled

```

```

Server pool: cdb1pool
Cardinality: UNIFORM
Disconnect: false
Service role: PRIMARY
Management policy: AUTOMATIC
DTP transaction: false
AQ HA notifications: false
Global: false
Commit Outcome: false
Failover type:
Failover method:
TAF failover retries:
TAF failover delay:
Connection Load Balancing Goal: LONG
Runtime Load Balancing Goal: NONE
TAF policy specification: NONE
Edition:
Pluggable database name: pdb1
Maximum lag time: ANY
SQL Translation Profile:
Retention: 86400 seconds
Replay Initiation Time: 300 seconds
Session State Consistency:
Service is enabled on nodes:
Service is disabled on nodes:
[oracle@host02 ~]$ exit
logout
Connection to host02 closed.
[oracle@host01 ~]$
```

- c. You can reiterate the same verification on the third node of the server pool.

```

[oracle@host01 ~]$ ssh host03

[oracle@host03 ~]$ . oraenv
ORACLE_SID = [oracle] ? cdb1_1
ORACLE_HOME = [/home/oracle] ?
/u01/app/oracle/product/12.2.0/dbhome_1
The Oracle base has been set to /u01/app/oracle
[oracle@host03 ~]$ srvctl config service -db cdb1
Service name: mypdb1serv
Service is enabled
Server pool: cdb1pool
```

```

Cardinality: UNIFORM
Disconnect: false
Service role: PRIMARY
Management policy: AUTOMATIC
DTP transaction: false
AQ HA notifications: false
Global: false
Commit Outcome: false
Failover type:
Failover method:
TAF failover retries:
TAF failover delay:
Connection Load Balancing Goal: LONG
Runtime Load Balancing Goal: NONE
TAF policy specification: NONE
Edition:
Pluggable database name: pdb1
Maximum lag time: ANY
SQL Translation Profile:
Retention: 86400 seconds
Replay Initiation Time: 300 seconds
Session State Consistency:
Service is enabled on nodes:
Service is disabled on nodes:
[oracle@host03 ~]$ exit
Logout
Connection to host03 closed.

[oracle@host01 ~]$

```

4. Close the PDB. You will verify that restarting the CDB automatically starts the dynamic PDB service and opens the associated PDB.

```

[oracle@host01 ~]$ sqlplus / as sysdba

SQL*Plus: Release 12.2.0.1.0 Production on Fri Nov 10 08:00:30
2017

Copyright (c) 1982, 2016, Oracle. All rights reserved.

Connected to:
Oracle Database 12c Enterprise Edition Release 12.2.0.1.0 -
64bit Production

```

```

SQL> ALTER PLUGGABLE DATABASE pdb1 CLOSE IMMEDIATE
INSTANCES=ALL;
Pluggable database altered.

SQL> SELECT name, open_mode FROM v$pdbs;

NAME          OPEN_MODE
-----
PDB$SEED      READ ONLY
PDB1          MOUNTED
PDB2          READ WRITE

SQL> EXIT
Disconnected from Oracle Database 12c Enterprise Edition Release
12.2.0.1.0 - 64bit Production
[oracle@host01 ~]$

```

5. Stop and restart the CDB.

```

[oracle@host01 ~]$ srvctl stop database -db cdb1
[oracle@host01 ~]$
[oracle@host01 ~]$ srvctl start database -db cdb1 -eval
Database cdb1 will be started on nodes host03,host02,host01
Service mypdb1serv will be started on nodes host03,host02,host01

[oracle@host01 ~]$ srvctl start database -db cdb1

[oracle@host01 ~]$ srvctl status database -db cdb1
Instance cdb1_1 is running on node host02
Instance cdb1_2 is running on node host03
Instance cdb1_3 is running on node host01

[oracle@host01 ~]$

```

6. Verify that the new dynamic PDB service is started and the PDB opened automatically. It may take a few moments for the service to be started on all hosts. (If you do not wish to wait, use SRVCTL to start the service: `srvctl start service -d cdb1 -service mypdb1serv`).

```

[oracle@host01 ~]$ srvctl status service -db cdb1
Service mypdb1serv is running on nodes: host03,host02,host01.

[oracle@host01 ~]$ sqlplus / as sysdba

```

```
SQL*Plus: Release 12.2.0.1.0 Production on Fri Nov 10 08:04:10  
2017
```

```
Copyright (c) 1982, 2016, Oracle. All rights reserved.
```

```
Connected to:
```

```
Oracle Database 12c Enterprise Edition Release 12.2.0.1.0 -  
64bit Production
```

```
SQL> SELECT name, open_mode FROM v$pdbs;
```

NAME	OPEN_MODE
PDB\$SEED	READ ONLY
PDB1	READ WRITE
PDB2	MOUNTED

```
SQL> SELECT name FROM v$services;
```

NAME
mypdb1serv
pdb2
pdb1
cdb1XDB
cdb1
SYS\$BACKGROUND
SYS\$USERS

```
7 rows selected.
```

```
SQL> exit  
[oracle@host01 ~]$
```

Notice that PDBs are automatically opened by clusterware in all the instances in which the service is started. There is therefore no need to create a trigger AFTER STARTUP ON DATABASE to open PDBs as it is the case in non-RAC CDBs.

7. You can also stop and restart the service manually.

```
[oracle@host01 ~]$ srvctl stop service -db cdb1 -service  
mypdb1serv
```

```
[oracle@host01 ~]$ srvctl status service -db cdb1 -service mypdb1serv
Service mypdb1serv is not running.

[oracle@host01 ~]$ srvctl start service -db cdb1 -service mypdb1serv

[oracle@host01 ~]$ srvctl status service -db cdb1 -service mypdb1serv
Service mypdb1serv is running on nodes: host03,host02,host01

[oracle@host01 ~]$ lsnrctl status

LSNRCTL for Linux: Version 12.2.0.1.0 - Production on 10-NOV-2017 08:06:05

Copyright (c) 1991, 2016, Oracle. All rights reserved.

Connecting to (ADDRESS=(PROTOCOL=tcp)(HOST=)(PORT=1521))
STATUS of the LISTENER
-----
Alias                      LISTENER
Version                    TNSLSNR for Linux: Version 12.2.0.1.0
- Production
Start Date                 09-NOV-2017 12:42:01
Uptime                     0 days 19 hr. 24 min. 4 sec
Trace Level                off
Security                   ON: Local OS Authentication
SNMP                       OFF
Listener Parameter File    /u01/app/12.2.0/grid/network/admin/listener.ora
Listener Log File          /u01/app/grid/diag/tnslsnr/host01/listener/alert/log.xml
Listening Endpoints Summary...
  (DESCRIPTION=(ADDRESS=(PROTOCOL=ipc) (KEY=LISTENER) ))

  (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp) (HOST=192.0.2.101) (PORT=1521)
))

  (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp) (HOST=192.0.2.242) (PORT=1521)
))

Services Summary...
Service "+ASM" has 1 instance(s).
  Instance "+ASM1", status READY, has 1 handler(s) for this
service...
```

```
Service "+ASM_DATA" has 1 instance(s).
  Instance "+ASM1", status READY, has 1 handler(s) for this
service...
Service "+ASM_MGMT" has 1 instance(s).
  Instance "+ASM1", status READY, has 1 handler(s) for this
service...
Service "5d8f3dbbbf1449ace053660200c0ed13" has 2 instance(s).
  Instance "cdb1_3", status READY, has 1 handler(s) for this
service...
  Instance "cdb1_3", status READY, has 2 handler(s) for this
service...
Service "5d9d2399a0a130dbe053650200c0c9cd" has 1 instance(s).
  Instance "cdb1_3", status READY, has 1 handler(s) for this
service...
Service "cdb1" has 1 instance(s).
  Instance "cdb1_3", status READY, has 1 handler(s) for this
service...
Service "cdb1XDB" has 1 instance(s).
  Instance "cdb1_3", status READY, has 1 handler(s) for this
service...
Service "mypdb1serv" has 2 instance(s).
  Instance "cdb1_3", status READY, has 1 handler(s) for this
service...
  Instance "cdb1_3", status READY, has 2 handler(s) for this
service...
Service "pdb1" has 2 instance(s).
  Instance "cdb1_3", status READY, has 1 handler(s) for this
service...
  Instance "cdb1_3", status READY, has 2 handler(s) for this
service...
Service "pdb2" has 1 instance(s).
  Instance "cdb1_3", status READY, has 1 handler

[oracle@host01 ~]$
```

8. Use the service to connect to the PDB on any of the CDB instances.

```
[oracle@host01 ~]$ sqlplus /nolog

SQL> CONNECT system@"host01/mypdb1serv"
Enter password:
Connected.

SQL> SELECT name, open_mode FROM v$pdbs;
```

NAME	OPEN_MODE
PDB1	READ WRITE

```
SQL> SELECT name FROM v$services;

NAME
-----
mypdb1serv
pdb1

SQL> CONNECT system@"host02/mypdb1serv"
Enter password:
Connected.

SQL> SELECT name FROM v$services;

NAME
-----
mypdb1serv
pdb1

SQL> CONNECT system@"host03/mypdb1serv"
Enter password:
Connected.

SQL> SELECT name FROM v$services;

NAME
-----
mypdb1serv
pdb1

SQL> exit
[oracle@host01 ~]$
```

Task 2: Affinize connections to pdb2 to a single CDB instance

1. Create and start the service for pdb2.

```
[oracle@host01 ~]$ srvctl add service -db cdb1 -pdb pdb2 -
service singpdb2serv -policy automatic -serverpool cdb1pool -
cardinality singleton

[oracle@host01 ~]$ srvctl start service -db cdb1 -service
singpdb2serv -eval
Service singpdb2serv will be started on node host01

[oracle@host01 ~]$ srvctl start service -db cdb1 -service
singpdb2serv

[oracle@host01 ~]$ srvctl status service -db cdb1 -service
singpdb2serv
Service singpdb2serv is running on nodes: host01
[oracle@host01 ~]$
```

2. Check that you can use the service to connect to pdb2 only on host01 and that the PDB is opened in the CDB instance on host01 only.

```
[oracle@host01 ~]$ sqlplus /nolog

SQL> CONNECT system@"host01/singpdb2serv"
Enter password:
Connected.

SQL> SELECT name, open_mode FROM v$pdbs;

NAME                      OPEN_MODE
-----
PDB2                      READ WRITE

SQL> SELECT name FROM v$services;

NAME
-----
singpdb2serv
pdb2

SQL>
```

3. Check that you cannot use the service to connect to pdb2 on host02 nor host03 and that the PDB is closed in the CDB instances on host02 and host03.

```
SQL> CONNECT system@"host02/singpdb2serv"
Enter password:
ERROR:
ORA-12514: TNS:listener does not currently know of service
requested in connect descriptor

Warning: You are no longer connected to ORACLE.

SQL> CONNECT system@"host03/singpdb2serv"
Enter password:
ERROR:
ORA-12514: TNS:listener does not currently know of service
requested in connect descriptor

Warning: You are no longer connected to ORACLE.

SQL> CONNECT system@"host02/cdb1"
Enter password:
Connected.

SQL> select name, open_mode from v$pdbs;

NAME          OPEN_MODE
-----
PDB$SEED      READ ONLY
PDB1          READ WRITE
PDB2          MOUNTED

SQL> CONNECT system@"host03/cdb1"
Enter password:
Connected.

SQL> select name, open_mode from v$pdbs;

NAME          OPEN_MODE
-----
PDB$SEED      READ ONLY
PDB1          READ WRITE
PDB2          MOUNTED

SQL> exit
[oracle@host01 ~]$
```

Practice 13-4: Dropping a PDB

Overview

In this practice, you will drop a PDB in the CDB and verify that the services and data files are deleted.

Tasks

1. To drop the pdb2, first stop and remove the service from the resources configuration.

```
[oracle@host01 ~]$ srvctl stop service -db cdb1 -service
singpdb2serv

[oracle@host01 ~]$ srvctl remove service -db cdb1 -service
singpdb2serv
[oracle@host01 ~]$
```

2. Drop the pdb2 PDB.

```
[oracle@host01 ~]$ ~]$ sqlplus /nolog

SQL> CONNECT system@"host01/pdb2"
Enter password:
Connected.

SQL> SELECT name FROM v$logfile;
NAME
-----
+DATA/CDB1/5D9D2399A0A130DBE053650200C0C9CD/DATAFILE/system.291.959672879
+DATA/CDB1/5D9D2399A0A130DBE053650200C0C9CD/DATAFILE/sysaux.292.959672879
+DATA/CDB1/5D9D2399A0A130DBE053650200C0C9CD/DATAFILE/undotbs1.288.959672879
+DATA/CDB1/5D9D2399A0A130DBE053650200C0C9CD/DATAFILE/undo_3.289.959672879
+DATA/CDB1/5D9D2399A0A130DBE053650200C0C9CD/DATAFILE/undo_4.290.959672879
+DATA/CDB1/5D9D2399A0A130DBE053650200C0C9CD/DATAFILE/users.287.959672879

6 rows selected.

SQL> CONNECT / AS SYSDBA
Connected.

SQL> SELECT name FROM v$services;
NAME
-----
```

```

mypdb1serv
pdb2
pdb1
cdb1XDB
cdb1
SYS$BACKGROUND
SYS$USERS

7 rows selected.

SQL> DROP PLUGGABLE DATABASE pdb2 INCLUDING DATAFILES;
*
ERROR at line 1:
ORA-65025: Pluggable database PDB2 is not closed on all
instances.

SQL> ALTER PLUGGABLE DATABASE pdb2 CLOSE INSTANCE=ALL;

Pluggable database altered.

SQL> DROP PLUGGABLE DATABASE pdb2 INCLUDING DATAFILES;

Pluggable database dropped.

SQL>

```

3. Verify that the data files are deleted.

```

SQL> SELECT name FROM v$datafile;

NAME
-----
+DATA/CDB1/DATAFILE/system.258.959612481
+DATA/CDB1/DATAFILE/sysaux.259.959612551
+DATA/CDB1/DATAFILE/undotbs1.260.959612587
+DATA/CDB1/4700A987085B3DFAE05387E5E50A8C7B/DATAFILE/system.268.959612679
+DATA/CDB1/4700A987085B3DFAE05387E5E50A8C7B/DATAFILE/sysaux.267.959612679
+DATA/CDB1/DATAFILE/users.261.959612589
+DATA/CDB1/4700A987085B3DFAE05387E5E50A8C7B/DATAFILE/undotbs1.269.959612679
+DATA/CDB1/DATAFILE/undotbs2.278.959613169
+DATA/CDB1/DATAFILE/undotbs3.279.959613171
+DATA/CDB1/5D8F3DBBBF1449ACE053660200C0ED13/DATAFILE/system.281.959613189
+DATA/CDB1/5D8F3DBBBF1449ACE053660200C0ED13/DATAFILE/sysaux.282.959613189

NAME
-----
+DATA/CDB1/5D8F3DBBBF1449ACE053660200C0ED13/DATAFILE/undotbs1.280.959613189

```

```
+DATA/CDB1/5D8F3DBBBF1449ACE053660200C0ED13/DATAFILE/undo_3.284.959613231
+DATA/CDB1/5D8F3DBBBF1449ACE053660200C0ED13/DATAFILE/undo_4.285.959613233
+DATA/CDB1/5D8F3DBBBF1449ACE053660200C0ED13/DATAFILE/users.286.959613235
```

15 rows selected.

SQL>

Note that all files related to `pdb2` are removed. The UNDO datafile is associated to the instance, and not to any PDB.

- Verify that the services are deleted. Check in `V$SERVICES` view and with `LSNRCTL`.

```
SQL> SELECT name FROM v$services;
```

NAME

```
-----
mypdb1serv
pdb1
cdb1XDB
cdb1
SYS$BACKGROUND
SYS$USERS
```

6 rows selected.

SQL> **exit**

```
[oracle@host01 ~]$ lsnrctl status
```

```
LSNRCTL for Linux: Version 12.2.0.1.0 - Production on 10-NOV-2017 08:19:46
```

```
Copyright (c) 1991, 2016, Oracle. All rights reserved.
```

```
Connecting to (ADDRESS=(PROTOCOL=tcp) (HOST=) (PORT=1521))
```

```
STATUS of the LISTENER
```

```
-----
```

Alias	LISTENER
Version	TNSLSNR for Linux: Version 12.2.0.1.0
- Production	
Start Date	09-NOV-2017 12:42:01
Uptime	0 days 19 hr. 37 min. 45 sec
Trace Level	off
Security	ON: Local OS Authentication
SNMP	OFF

```
Listener Parameter File
/u01/app/12.2.0/grid/network/admin/listener.ora
Listener Log File
/u01/app/grid/diag/tnslsnr/host01/listener/alert/log.x      ml
Listening Endpoints Summary...
  (DESCRIPTION=(ADDRESS=(PROTOCOL=ipc) (KEY=LISTENER)) )

  (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp) (HOST=192.0.2.101) (PORT=1521
  )))

  (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp) (HOST=192.0.2.242) (PORT=1521
  )))

Services Summary...
Service "+ASM" has 1 instance(s).
  Instance "+ASM1", status READY, has 1 handler(s) for this
service...
Service "+ASM_DATA" has 1 instance(s).
  Instance "+ASM1", status READY, has 1 handler(s) for this
service...
Service "+ASM_MGMT" has 1 instance(s).
  Instance "+ASM1", status READY, has 1 handler(s) for this
service...
Service "5d8f3dbbf1449ace053660200c0ed13" has 2 instance(s).
  Instance "cdb1_3", status READY, has 2 handler(s) for this
service...
  Instance "cdb1_3", status READY, has 1 handler(s) for this
service...
Service "cdb1" has 1 instance(s).
  Instance "cdb1_3", status READY, has 1 handler(s) for this
service...
Service "cdb1XDB" has 1 instance(s).
  Instance "cdb1_3", status READY, has 1 handler(s) for this
service...
Service "mypdb1serv" has 2 instance(s).
  Instance "cdb1_3", status READY, has 2 handler(s) for this
service...
  Instance "cdb1_3", status READY, has 1 handler(s) for this
service...
Service "pdb1" has 2 instance(s).
  Instance "cdb1_3", status READY, has 2 handler(s) for this
service...
  Instance "cdb1_3", status READY, has 1 handler(s) for this
service...
```

```
The command completed successfully  
[oracle@host01 ~] $
```

5. Close all terminal windows opened for this practice.

GANG LIU (gangli@baylorhealth.edu) has a non-transferable license
to use this Student Guide.

Practices for Appendix A: Cleanup and Catchup Procedures for Practices

Appendix A-1: Cleanup Script

Estimated Time: 20~25min

Assumption

- You have *NOT* configured the ASM disk groups with Oracle ASM Filter Driver (AFD). If you have configured, use `asmca` to fix the issue first working with your instructor.

1. Open a terminal window to connect to `host01` as root user. Run the `reset01.sh` script.

```
[oracle@dns ~] ssh root@host01
root@host01's Password:

[root@host01 ~]# cd /stage/RAC/solutions/less_01/catchup

[root@host01 catchup]# ./reset01.sh
#####
Task 1 of 6: Stop Clusterware on all nodes
#####
Fri Feb  2 06:58:11 UTC 2018

CRS-2673: Attempting to stop 'ora.crsd' on 'host05'
CRS-2673: Attempting to stop 'ora.crsd' on 'host04'
CRS-2790: Starting shutdown of Cluster Ready Services-managed
resources on server 'host05'
CRS-2673: Attempting to stop 'ora.chad' on 'host05'
CRS-2790: Starting shutdown of Cluster Ready Services-managed
resources on server 'host04'
CRS-2673: Attempting to stop 'ora.chad' on 'host04'
CRS-2677: Stop of 'ora.chad' on 'host05' succeeded
CRS-2792: Shutdown of Cluster Ready Services-managed resources
on 'host05' has completed
CRS-2677: Stop of 'ora.chad' on 'host04' succeeded
CRS-2792: Shutdown of Cluster Ready Services-managed resources
on 'host04' has completed
CRS-2677: Stop of 'ora.crsd' on 'host05' succeeded
CRS-2673: Attempting to stop 'ora.cluster_interconnect.haip' on
'host05'
CRS-2673: Attempting to stop 'ora.ctssd' on 'host05'
CRS-2673: Attempting to stop 'ora.evmd' on 'host05'
CRS-2673: Attempting to stop 'ora.storage' on 'host05'
CRS-2677: Stop of 'ora.crsd' on 'host04' succeeded
CRS-2673: Attempting to stop 'ora.cluster_interconnect.haip' on
'host04'
```

```
CRS-2673: Attempting to stop 'ora.ctssd' on 'host04'
CRS-2673: Attempting to stop 'ora.evmd' on 'host04'
CRS-2673: Attempting to stop 'ora.storage' on 'host04'
CRS-2677: Stop of 'ora.storage' on 'host05' succeeded
CRS-2677: Stop of 'ora.storage' on 'host04' succeeded
CRS-2677: Stop of 'ora.ctssd' on 'host05' succeeded
CRS-2677: Stop of 'ora.evmd' on 'host05' succeeded
CRS-2677: Stop of 'ora.ctssd' on 'host04' succeeded
CRS-2677: Stop of 'ora.evmd' on 'host04' succeeded
CRS-2677: Stop of 'ora.cluster_interconnect.haip' on 'host05'
succeeded
CRS-2673: Attempting to stop 'ora.cssd' on 'host05'
CRS-2677: Stop of 'ora.cluster_interconnect.haip' on 'host04'
succeeded
CRS-2673: Attempting to stop 'ora.cssd' on 'host04'
CRS-2677: Stop of 'ora.cssd' on 'host05' succeeded
CRS-2677: Stop of 'ora.cssd' on 'host04' succeeded
CRS-2673: Attempting to stop 'ora.crsd' on 'host03'
CRS-2673: Attempting to stop 'ora.crsd' on 'host02'
CRS-2673: Attempting to stop 'ora.crsd' on 'host01'
CRS-2790: Starting shutdown of Cluster Ready Services-managed
resources on server 'host03'
CRS-2673: Attempting to stop 'ora.LISTENER_SCAN1.lsnr' on
'host03'
CRS-2673: Attempting to stop 'ora.LISTENER.lsnr' on 'host03'
CRS-2673: Attempting to stop 'ora.DATA.dg' on 'host03'
CRS-2673: Attempting to stop 'ora.FRA.dg' on 'host03'
CRS-2673: Attempting to stop 'ora.MGMT.dg' on 'host03'
CRS-2673: Attempting to stop 'ora.chad' on 'host03'
CRS-2677: Stop of 'ora.FRA.dg' on 'host03' succeeded
CRS-2790: Starting shutdown of Cluster Ready Services-managed
resources on server 'host02'
CRS-2673: Attempting to stop 'ora.LISTENER_SCAN2.lsnr' on
'host02'
CRS-2673: Attempting to stop 'ora.LISTENER.lsnr' on 'host02'
CRS-2673: Attempting to stop 'ora.DATA.dg' on 'host02'
CRS-2673: Attempting to stop 'ora.FRA.dg' on 'host02'
CRS-2673: Attempting to stop 'ora.MGMT.dg' on 'host02'
CRS-2673: Attempting to stop 'ora.chad' on 'host02'
CRS-2677: Stop of 'ora.DATA.dg' on 'host03' succeeded
CRS-2677: Stop of 'ora.MGMT.dg' on 'host03' succeeded
CRS-2673: Attempting to stop 'ora.asm' on 'host03'
CRS-2677: Stop of 'ora.LISTENER_SCAN1.lsnr' on 'host03'
succeeded
```

```
CRS-2673: Attempting to stop 'ora.scan1.vip' on 'host03'
CRS-2677: Stop of 'ora.MGMT.dg' on 'host02' succeeded
CRS-2677: Stop of 'ora.FRA.dg' on 'host02' succeeded
CRS-2677: Stop of 'ora.LISTENER.lsnr' on 'host03' succeeded
CRS-2673: Attempting to stop 'ora.host03.vip' on 'host03'
CRS-2677: Stop of 'ora.DATA.dg' on 'host02' succeeded
CRS-2673: Attempting to stop 'ora.asm' on 'host02'
CRS-2677: Stop of 'ora.asm' on 'host03' succeeded
CRS-2673: Attempting to stop 'ora.ASMNET1LSNR_ASM.lsnr' on
'host03'
CRS-2677: Stop of 'ora.LISTENER_SCAN2.lsnr' on 'host02'
succeeded
CRS-2673: Attempting to stop 'ora.scan2.vip' on 'host02'
CRS-2790: Starting shutdown of Cluster Ready Services-managed
resources on server 'host01'
CRS-2673: Attempting to stop 'ora.qosmserver' on 'host01'
CRS-2673: Attempting to stop 'ora.gns' on 'host01'
CRS-2673: Attempting to stop 'ora.LISTENER_SCAN3.lsnr' on
'host01'
CRS-2673: Attempting to stop 'ora.cvu' on 'host01'
CRS-2677: Stop of 'ora.LISTENER.lsnr' on 'host02' succeeded
CRS-2673: Attempting to stop 'ora.host02.vip' on 'host02'
CRS-2677: Stop of 'ora.cvu' on 'host01' succeeded
CRS-2677: Stop of 'ora.LISTENER_SCAN3.lsnr' on 'host01'
succeeded
CRS-2673: Attempting to stop 'ora.scan3.vip' on 'host01'
CRS-2677: Stop of 'ora.chad' on 'host02' succeeded
CRS-2677: Stop of 'ora.asm' on 'host02' succeeded
CRS-2673: Attempting to stop 'ora.ASMNET1LSNR_ASM.lsnr' on
'host02'
CRS-2677: Stop of 'ora.gns' on 'host01' succeeded
CRS-2673: Attempting to stop 'ora.gns.vip' on 'host01'
CRS-2673: Attempting to stop 'ora.chad' on 'host01'
CRS-2677: Stop of 'ora.chad' on 'host03' succeeded
CRS-2677: Stop of 'ora.scan1.vip' on 'host03' succeeded
CRS-2677: Stop of 'ora.scan3.vip' on 'host01' succeeded
CRS-2677: Stop of 'ora.scan2.vip' on 'host02' succeeded
CRS-2677: Stop of 'ora.host03.vip' on 'host03' succeeded
CRS-2677: Stop of 'ora.host02.vip' on 'host02' succeeded
CRS-2677: Stop of 'ora.ASMNET1LSNR_ASM.lsnr' on 'host03'
succeeded
CRS-2677: Stop of 'ora.chad' on 'host01' succeeded
CRS-2673: Attempting to stop 'ora.mgmtdb' on 'host01'
CRS-2677: Stop of 'ora.gns.vip' on 'host01' succeeded
```

```
CRS-2673: Attempting to stop 'ora.ons' on 'host03'
CRS-2677: Stop of 'ora.ons' on 'host03' succeeded
CRS-2673: Attempting to stop 'ora.net1.network' on 'host03'
CRS-2677: Stop of 'ora.net1.network' on 'host03' succeeded
CRS-2792: Shutdown of Cluster Ready Services-managed resources
on 'host03' has completed
CRS-2677: Stop of 'ora.ASMNET1LSNR_ASM.lsnr' on 'host02'
succeeded
CRS-2677: Stop of 'ora.crsd' on 'host03' succeeded
CRS-2673: Attempting to stop 'ora.ctssd' on 'host03'
CRS-2673: Attempting to stop 'ora.evmd' on 'host03'
CRS-2673: Attempting to stop 'ora.storage' on 'host03'
CRS-2673: Attempting to stop 'ora.ons' on 'host02'
CRS-2677: Stop of 'ora.storage' on 'host03' succeeded
CRS-2673: Attempting to stop 'ora.asm' on 'host03'
CRS-2677: Stop of 'ora.ons' on 'host02' succeeded
CRS-2673: Attempting to stop 'ora.net1.network' on 'host02'
CRS-2677: Stop of 'ora.net1.network' on 'host02' succeeded
CRS-2792: Shutdown of Cluster Ready Services-managed resources
on 'host02' has completed
CRS-2677: Stop of 'ora.ctssd' on 'host03' succeeded
CRS-2677: Stop of 'ora.evmd' on 'host03' succeeded
CRS-2677: Stop of 'ora.crsd' on 'host02' succeeded
CRS-2673: Attempting to stop 'ora.ctssd' on 'host02'
CRS-2673: Attempting to stop 'ora.evmd' on 'host02'
CRS-2673: Attempting to stop 'ora.storage' on 'host02'
CRS-2677: Stop of 'ora.storage' on 'host02' succeeded
CRS-2677: Stop of 'ora.qosmserver' on 'host01' succeeded
CRS-2677: Stop of 'ora_mgmtdb' on 'host01' succeeded
CRS-2673: Attempting to stop 'ora.MGMTLSNR' on 'host01'
CRS-2673: Attempting to stop 'ora.DATA.dg' on 'host01'
CRS-2673: Attempting to stop 'ora.FRA.dg' on 'host01'
CRS-2673: Attempting to stop 'ora.MGMT.dg' on 'host01'
CRS-2677: Stop of 'ora.DATA.dg' on 'host01' succeeded
CRS-2677: Stop of 'ora.MGMT.dg' on 'host01' succeeded
CRS-2677: Stop of 'ora.FRA.dg' on 'host01' succeeded
CRS-2673: Attempting to stop 'ora.asm' on 'host01'
CRS-2673: Attempting to stop 'ora.asm' on 'host02'
CRS-2677: Stop of 'ora.ctssd' on 'host02' succeeded
CRS-2677: Stop of 'ora.evmd' on 'host02' succeeded
CRS-2677: Stop of 'ora.MGMTLSNR' on 'host01' succeeded
CRS-2673: Attempting to stop 'ora.LISTENER.lsnr' on 'host01'
CRS-2677: Stop of 'ora.LISTENER.lsnr' on 'host01' succeeded
```

```
CRS-2673: Attempting to stop 'ora.host01.vip' on 'host01'
CRS-2677: Stop of 'ora.host01.vip' on 'host01' succeeded
CRS-2677: Stop of 'ora.asm' on 'host03' succeeded
CRS-2673: Attempting to stop 'ora.cluster_interconnect.haip' on
'host03'
CRS-2677: Stop of 'ora.cluster_interconnect.haip' on 'host03'
succeeded
CRS-2673: Attempting to stop 'ora.cssd' on 'host03'
CRS-2677: Stop of 'ora.asm' on 'host02' succeeded
CRS-2673: Attempting to stop 'ora.cluster_interconnect.haip' on
'host02'
CRS-2677: Stop of 'ora.cssd' on 'host03' succeeded
CRS-2677: Stop of 'ora.cluster_interconnect.haip' on 'host02'
succeeded
CRS-2673: Attempting to stop 'ora.cssd' on 'host02'
CRS-2677: Stop of 'ora.cssd' on 'host02' succeeded
CRS-2677: Stop of 'ora.asm' on 'host01' succeeded
CRS-2673: Attempting to stop 'ora.ASMNET1LSNR_ASM.lsnr' on
'host01'
CRS-2677: Stop of 'ora.ASMNET1LSNR_ASM.lsnr' on 'host01'
succeeded
CRS-2673: Attempting to stop 'ora.ons' on 'host01'
CRS-2677: Stop of 'ora.ons' on 'host01' succeeded
CRS-2673: Attempting to stop 'ora.net1.network' on 'host01'
CRS-2677: Stop of 'ora.net1.network' on 'host01' succeeded
CRS-2792: Shutdown of Cluster Ready Services-managed resources
on 'host01' has completed
CRS-2677: Stop of 'ora.crsd' on 'host01' succeeded
CRS-2673: Attempting to stop 'ora.ctssd' on 'host01'
CRS-2673: Attempting to stop 'ora.evmd' on 'host01'
CRS-2673: Attempting to stop 'ora.storage' on 'host01'
CRS-2677: Stop of 'ora.storage' on 'host01' succeeded
CRS-2673: Attempting to stop 'ora.asm' on 'host01'
CRS-2677: Stop of 'ora.evmd' on 'host01' succeeded
CRS-2677: Stop of 'ora.ctssd' on 'host01' succeeded
CRS-2677: Stop of 'ora.asm' on 'host01' succeeded
CRS-2673: Attempting to stop 'ora.cluster_interconnect.haip' on
'host01'
CRS-2677: Stop of 'ora.cluster_interconnect.haip' on 'host01'
succeeded
CRS-2673: Attempting to stop 'ora.cssd' on 'host01'
CRS-2677: Stop of 'ora.cssd' on 'host01' succeeded
CRS-2791: Starting shutdown of Oracle High Availability
Services-managed resources on 'host01'
```

```
CRS-2673: Attempting to stop 'ora.drivers.acfs' on 'host01'
CRS-2673: Attempting to stop 'ora.gpnpd' on 'host01'
CRS-2673: Attempting to stop 'ora.mdnsd' on 'host01'
CRS-2673: Attempting to stop 'ora.crf' on 'host01'
CRS-2677: Stop of 'ora.drivers.acfs' on 'host01' succeeded
CRS-2677: Stop of 'ora.gpnpd' on 'host01' succeeded
CRS-2677: Stop of 'ora.crf' on 'host01' succeeded
CRS-2673: Attempting to stop 'ora.gipcd' on 'host01'
CRS-2677: Stop of 'ora.mdnsd' on 'host01' succeeded
CRS-2677: Stop of 'ora.gipcd' on 'host01' succeeded
CRS-2793: Shutdown of Oracle High Availability Services-managed
resources on 'host01' has completed
CRS-4133: Oracle High Availability Services has been stopped.
CRS-2791: Starting shutdown of Oracle High Availability
Services-managed resources on 'host02'
CRS-2673: Attempting to stop 'ora.drivers.acfs' on 'host02'
CRS-2673: Attempting to stop 'ora.gpnpd' on 'host02'
CRS-2673: Attempting to stop 'ora.crf' on 'host02'
CRS-2673: Attempting to stop 'ora.mdnsd' on 'host02'
CRS-2677: Stop of 'ora.drivers.acfs' on 'host02' succeeded
CRS-2677: Stop of 'ora.gpnpd' on 'host02' succeeded
CRS-2677: Stop of 'ora.crf' on 'host02' succeeded
CRS-2673: Attempting to stop 'ora.gipcd' on 'host02'
CRS-2677: Stop of 'ora.mdnsd' on 'host02' succeeded
CRS-2677: Stop of 'ora.gipcd' on 'host02' succeeded
CRS-2793: Shutdown of Oracle High Availability Services-managed
resources on 'host02' has completed
CRS-4133: Oracle High Availability Services has been stopped.
CRS-2791: Starting shutdown of Oracle High Availability
Services-managed resources on 'host03'
CRS-2673: Attempting to stop 'ora.gpnpd' on 'host03'
CRS-2673: Attempting to stop 'ora.drivers.acfs' on 'host03'
CRS-2673: Attempting to stop 'ora.mdnsd' on 'host03'
CRS-2673: Attempting to stop 'ora.crf' on 'host03'
CRS-2677: Stop of 'ora.drivers.acfs' on 'host03' succeeded
CRS-2677: Stop of 'ora.gpnpd' on 'host03' succeeded
CRS-2677: Stop of 'ora.mdnsd' on 'host03' succeeded
CRS-2677: Stop of 'ora.crf' on 'host03' succeeded
CRS-2673: Attempting to stop 'ora.gipcd' on 'host03'
CRS-2677: Stop of 'ora.gipcd' on 'host03' succeeded
CRS-2793: Shutdown of Oracle High Availability Services-managed
resources on 'host03' has completed
CRS-4133: Oracle High Availability Services has been stopped.
```

```

CRS-2791: Starting shutdown of Oracle High Availability
Services-managed resources on 'host04'
CRS-2673: Attempting to stop 'ora.crf' on 'host04'
CRS-2673: Attempting to stop 'ora.mdnsd' on 'host04'
CRS-2673: Attempting to stop 'ora.gpnpd' on 'host04'
CRS-2673: Attempting to stop 'ora.drivers.acfs' on 'host04'
CRS-2677: Stop of 'ora.crf' on 'host04' succeeded
CRS-2673: Attempting to stop 'ora.gipcd' on 'host04'
CRS-2677: Stop of 'ora.drivers.acfs' on 'host04' succeeded
CRS-2677: Stop of 'ora.gpnpd' on 'host04' succeeded
CRS-2677: Stop of 'ora.gipcd' on 'host04' succeeded
CRS-2677: Stop of 'ora.mdnsd' on 'host04' succeeded
CRS-2793: Shutdown of Oracle High Availability Services-managed
resources on 'host04' has completed
CRS-4133: Oracle High Availability Services has been stopped.
CRS-2791: Starting shutdown of Oracle High Availability
Services-managed resources on 'host05'
CRS-2673: Attempting to stop 'ora.crf' on 'host05'
CRS-2673: Attempting to stop 'ora.gpnpd' on 'host05'
CRS-2673: Attempting to stop 'ora.drivers.acfs' on 'host05'
CRS-2673: Attempting to stop 'ora.mdnsd' on 'host05'
CRS-2677: Stop of 'ora.drivers.acfs' on 'host05' succeeded
CRS-2677: Stop of 'ora.crf' on 'host05' succeeded
CRS-2677: Stop of 'ora.gpnpd' on 'host05' succeeded
CRS-2673: Attempting to stop 'ora.gipcd' on 'host05'
CRS-2677: Stop of 'ora.mdnsd' on 'host05' succeeded
CRS-2677: Stop of 'ora.gipcd' on 'host05' succeeded
CRS-2793: Shutdown of Oracle High Availability Services-managed
resources on 'host05' has completed
CRS-4133: Oracle High Availability Services has been stopped.

#####
Task 2 of 6: Blunt force execution of ohasd
#####
Fri Feb  2 07:00:49 UTC 2018

sh: cannot set terminal process group (-1): Inappropriate ioctl
for device
sh: no job control in this shell
sh: cannot set terminal process group (-1): Inappropriate ioctl
for device
sh: no job control in this shell
sh: cannot set terminal process group (-1): Inappropriate ioctl
for device

```

```
sh: no job control in this shell
sh: cannot set terminal process group (-1): Inappropriate ioctl
for device
sh: no job control in this shell

#####
Task 3 of 6: Blast the Grid Home and asst. directories
#####
Fri Feb  2 07:00:50 UTC 2018

#####
Task 4 of 6: remove OHASD init files
#####
Fri Feb  2 07:01:22 UTC 2018

#####
Task 5 of 6: remove /etc/oratab /etc/oracle/* if they exist
#####
Fri Feb  2 07:01:23 UTC 2018

#####
Task 6 of 6: Reconfigure ASM Disks
#####
Fri Feb  2 07:01:31 UTC 2018

File is /dev/c1_DATA1_dsk1
500+0 records in
500+0 records out
524288000 bytes (524 MB) copied, 17.515 s, 29.9 MB/s
File is /dev/c1_DATA1_dsk10
500+0 records in
500+0 records out
524288000 bytes (524 MB) copied, 14.5236 s, 36.1 MB/s
File is /dev/c1_DATA1_dsk11
500+0 records in
500+0 records out
524288000 bytes (524 MB) copied, 15.849 s, 33.1 MB/s
File is /dev/c1_DATA1_dsk12
500+0 records in
500+0 records out
```

```
524288000 bytes (524 MB) copied, 14.9201 s, 35.1 MB/s
File is /dev/c1_DATA1_dsk13
500+0 records in
500+0 records out
524288000 bytes (524 MB) copied, 15.412 s, 34.0 MB/s
File is /dev/c1_DATA1_dsk14
500+0 records in
500+0 records out
524288000 bytes (524 MB) copied, 14.6569 s, 35.8 MB/s
File is /dev/c1_DATA1_dsk2
500+0 records in
500+0 records out
524288000 bytes (524 MB) copied, 16.1129 s, 32.5 MB/s
File is /dev/c1_DATA1_dsk3
500+0 records in
500+0 records out
524288000 bytes (524 MB) copied, 15.522 s, 33.8 MB/s
File is /dev/c1_DATA1_dsk4
500+0 records in
500+0 records out
524288000 bytes (524 MB) copied, 15.8241 s, 33.1 MB/s
File is /dev/c1_DATA1_dsk5
500+0 records in
500+0 records out
524288000 bytes (524 MB) copied, 15.9136 s, 32.9 MB/s
File is /dev/c1_DATA1_dsk6
500+0 records in
500+0 records out
524288000 bytes (524 MB) copied, 15.836 s, 33.1 MB/s
File is /dev/c1_DATA1_dsk7
500+0 records in
500+0 records out
524288000 bytes (524 MB) copied, 15.1377 s, 34.6 MB/s
File is /dev/c1_DATA1_dsk8
500+0 records in
500+0 records out
524288000 bytes (524 MB) copied, 15.6603 s, 33.5 MB/s
File is /dev/c1_DATA1_dsk9
500+0 records in
500+0 records out
524288000 bytes (524 MB) copied, 15.3633 s, 34.1 MB/s
File is /dev/c1_FRA_dsk1
```

```
500+0 records in
500+0 records out
524288000 bytes (524 MB) copied, 16.0668 s, 32.6 MB/s
File is /dev/c1_FRA_dsk10
500+0 records in
500+0 records out
524288000 bytes (524 MB) copied, 15.0152 s, 34.9 MB/s
File is /dev/c1_FRA_dsk11
500+0 records in
500+0 records out
524288000 bytes (524 MB) copied, 16.1954 s, 32.4 MB/s
File is /dev/c1_FRA_dsk12
500+0 records in
500+0 records out
524288000 bytes (524 MB) copied, 15.6662 s, 33.5 MB/s
File is /dev/c1_FRA_dsk13
500+0 records in
500+0 records out
524288000 bytes (524 MB) copied, 16.0708 s, 32.6 MB/s
File is /dev/c1_FRA_dsk14
500+0 records in
500+0 records out
524288000 bytes (524 MB) copied, 15.3648 s, 34.1 MB/s
File is /dev/c1_FRA_dsk2
500+0 records in
500+0 records out
524288000 bytes (524 MB) copied, 15.8046 s, 33.2 MB/s
File is /dev/c1_FRA_dsk3
500+0 records in
500+0 records out
524288000 bytes (524 MB) copied, 15.8509 s, 33.1 MB/s
File is /dev/c1_FRA_dsk4
500+0 records in
500+0 records out
524288000 bytes (524 MB) copied, 15.1068 s, 34.7 MB/s
File is /dev/c1_FRA_dsk5
500+0 records in
500+0 records out
524288000 bytes (524 MB) copied, 16.2474 s, 32.3 MB/s
File is /dev/c1_FRA_dsk6
500+0 records in
500+0 records out
```

```
524288000 bytes (524 MB) copied, 15.1595 s, 34.6 MB/s
File is /dev/c1_FRA_dsk7
500+0 records in
500+0 records out
524288000 bytes (524 MB) copied, 16.0715 s, 32.6 MB/s
File is /dev/c1_FRA_dsk8
500+0 records in
500+0 records out
524288000 bytes (524 MB) copied, 15.2109 s, 34.5 MB/s
File is /dev/c1_FRA_dsk9
500+0 records in
500+0 records out
524288000 bytes (524 MB) copied, 15.8876 s, 33.0 MB/s
File is /dev/c1_SPARE_dsk1
500+0 records in
500+0 records out
524288000 bytes (524 MB) copied, 15.6668 s, 33.5 MB/s
File is /dev/c1_SPARE_dsk10
500+0 records in
500+0 records out
524288000 bytes (524 MB) copied, 15.924 s, 32.9 MB/s
File is /dev/c1_SPARE_dsk11
500+0 records in
500+0 records out
524288000 bytes (524 MB) copied, 15.4294 s, 34.0 MB/s
File is /dev/c1_SPARE_dsk12
500+0 records in
500+0 records out
524288000 bytes (524 MB) copied, 16.2076 s, 32.3 MB/s
File is /dev/c1_SPARE_dsk13
500+0 records in
500+0 records out
524288000 bytes (524 MB) copied, 14.9914 s, 35.0 MB/s
File is /dev/c1_SPARE_dsk14
500+0 records in
500+0 records out
524288000 bytes (524 MB) copied, 15.7853 s, 33.2 MB/s
File is /dev/c1_SPARE_dsk15
500+0 records in
500+0 records out
524288000 bytes (524 MB) copied, 15.9972 s, 32.8 MB/s
File is /dev/c1_SPARE_dsk2
```

```
500+0 records in
500+0 records out
524288000 bytes (524 MB) copied, 16.051 s, 32.7 MB/s
File is /dev/c1_SPARE_dsk3
500+0 records in
500+0 records out
524288000 bytes (524 MB) copied, 14.903 s, 35.2 MB/s
File is /dev/c1_SPARE_dsk5
500+0 records in
500+0 records out
524288000 bytes (524 MB) copied, 15.6917 s, 33.4 MB/s
File is /dev/c1_SPARE_dsk6
500+0 records in
500+0 records out
524288000 bytes (524 MB) copied, 15.2228 s, 34.4 MB/s
File is /dev/c1_SPARE_dsk7
500+0 records in
500+0 records out
524288000 bytes (524 MB) copied, 16.0492 s, 32.7 MB/s
File is /dev/c1_SPARE_dsk8
500+0 records in
500+0 records out
524288000 bytes (524 MB) copied, 15.3851 s, 34.1 MB/s
File is /dev/c1_SPARE_dsk9
500+0 records in
500+0 records out
524288000 bytes (524 MB) copied, 15.9991 s, 32.8 MB/s
File is /dev/c1_MGMT_dsk1
500+0 records in
500+0 records out
524288000 bytes (524 MB) copied, 15.3728 s, 34.1 MB/s
File is /dev/c1_MGMT_dsk10
500+0 records in
500+0 records out
524288000 bytes (524 MB) copied, 15.4536 s, 33.9 MB/s
File is /dev/c1_MGMT_dsk11
500+0 records in
500+0 records out
524288000 bytes (524 MB) copied, 15.1812 s, 34.5 MB/s
File is /dev/c1_MGMT_dsk12
500+0 records in
500+0 records out
```

```
524288000 bytes (524 MB) copied, 15.8596 s, 33.1 MB/s
File is /dev/c1_MGMT_dsk13
500+0 records in
500+0 records out
524288000 bytes (524 MB) copied, 14.9365 s, 35.1 MB/s
File is /dev/c1_MGMT_dsk14
500+0 records in
500+0 records out
524288000 bytes (524 MB) copied, 121.667 s, 4.3 MB/s
File is /dev/c1_MGMT_dsk2
500+0 records in
500+0 records out
524288000 bytes (524 MB) copied, 15.3671 s, 34.1 MB/s
File is /dev/c1_MGMT_dsk3
500+0 records in
500+0 records out
524288000 bytes (524 MB) copied, 15.4773 s, 33.9 MB/s
File is /dev/c1_MGMT_dsk4
500+0 records in
500+0 records out
524288000 bytes (524 MB) copied, 15.2824 s, 34.3 MB/s
File is /dev/c1_MGMT_dsk5
500+0 records in
500+0 records out
524288000 bytes (524 MB) copied, 15.9853 s, 32.8 MB/s
File is /dev/c1_MGMT_dsk6
500+0 records in
500+0 records out
524288000 bytes (524 MB) copied, 15.3184 s, 34.2 MB/s
File is /dev/c1_MGMT_dsk7
500+0 records in
500+0 records out
524288000 bytes (524 MB) copied, 15.623 s, 33.6 MB/s
File is /dev/c1_MGMT_dsk8
500+0 records in
500+0 records out
524288000 bytes (524 MB) copied, 15.1113 s, 34.7 MB/s
File is /dev/c1_MGMT_dsk9
500+0 records in
500+0 records out
524288000 bytes (524 MB) copied, 15.8357 s, 33.1 MB/s
```

```
#####
All 6 Tasks Completed
#####
Fri Feb  2 07:17:51 UTC 2018

Elapsed Real Time: 19:40.14

[root@host01 catchup]#
```

2. Optionally, delete the large files in the /var/log directory.

```
[root@host01 catchup]# ssh host01 rm -f /var/log/mess*
[root@host01 catchup]# ssh host02 rm -f /var/log/mess*
[root@host01 catchup]# ssh host03 rm -f /var/log/mess*
[root@host01 catchup]# ssh host04 rm -f /var/log/mess*
[root@host01 catchup]# ssh host05 rm -f /var/log/mess*
[root@host01 catchup]#
[root@host01 catchup]# ssh host01 rm -rf /tmp/*
[root@host01 catchup]# ssh host02 rm -rf /tmp/*
[root@host01 catchup]# ssh host03 rm -rf /tmp/*
[root@host01 catchup]# ssh host04 rm -rf /tmp/*
[root@host01 catchup]# ssh host05 rm -rf /tmp/*
[root@host01 catchup]#
```

3. Optionally, purge the deleted files that are held open to reclaim disk space.

```
[root@host01 catchup]# lsof |grep delete
java      4322      root    txt      REG            202,2
7734    745648          /u01/app/12.2.0/grid/jdk/jre/bin/java
(deleted)
...
[root@host01 catchup]# kill -9 4322
[root@host01 catchup]# lsof |grep delete
[root@host01 catchup]#
[root@host01 catchup]# ssh host02 lsof|grep delete
java      29466      root    txt      REG            202,2
7734    986836 /u01/app/12.2.0/grid/jdk/jre/bin/java (deleted)
...
[root@host01 catchup]# ssh host02 kill -9 29466
[root@host01 catchup]# ssh host02 lsof |grep delete
[root@host01 catchup]#
[root@host01 catchup]# ssh host03 lsof|grep delete
java      20160      root    txt      REG            202,2
7734   1518544 /u01/app/12.2.0/grid/jdk/jre/bin/java (deleted)
...
[root@host01 catchup]# ssh host03 kill -9 20160
[root@host01 catchup]# ssh host03 lsof|grep delete
[root@host01 catchup]#
[root@host01 catchup]# ssh host04 lsof|grep delete
nautilus  3002    oracle   19r      REG            202,2
524    1581093 /home/oracle/.local/share/gvfs-metadata/home
```

```
(deleted)
nautilus    3002      oracle   20r      REG          202,2
32768     1583605 /home/oracle/.local/share/gvfs-metadata/home-
cdd03e8e.log (deleted)
java       23432      root     txt      REG          202,2
7734      360449 /u01/app/12.2.0/grid/jdk/jre/bin/java (deleted)
...
[root@host01 catchup]# ssh host04 kill -9 3002 23432
[root@host01 catchup]# ssh host04 lsof|grep delete
[root@host01 catchup]#
[root@host01 catchup]# ssh host05 lsof|grep delete
java      14322      root     txt      REG          202,2
7734      114768 /u01/app/12.2.0/grid/jdk/jre/bin/java (deleted)
...
tail      21883      grid     3r      REG          202,2
5800      1430917
/u01/app/12.2.0/grid/install/root_host05.example.com_2018-01-
27_11-36-01-428452867.log (deleted)
[root@host01 catchup]# ssh host05 kill -9 14322 21883
[root@host01 catchup]# ssh host05 lsof|grep delete
[root@host01 catchup]#
```

4. Make sure the root file system (/) has enough space. The available space should be close enough to the one in the output.

```
[root@host01 catchup]# ssh host01 df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/xvda2     128G  7.2G  114G  6% /
tmpfs          6.0G  76K   6.0G  1% /dev/shm
/dev/xvda1     488M  133M  330M  29% /boot
/dev/xvdf1      12G   6.5G  4.7G  58% /stage

[root@host01 catchup]# ssh host02 df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/xvda2     29G   5.6G  22G  21% /
tmpfs          6.0G   0    6.0G  0% /dev/shm
/dev/xvda1     488M  133M  330M  29% /boot
/dev/xvdf1      12G   6.5G  4.7G  58% /stage

[root@host01 catchup]# ssh host03 df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/xvda2     29G   5.4G  23G  20% /
tmpfs          4.9G   0    4.9G  0% /dev/shm
/dev/xvda1     488M  107M  356M  24% /boot
/dev/xvdf1      12G   6.5G  4.7G  58% /stage

[root@host01 catchup]# ssh host04 df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/xvda2     29G   5.4G  23G  20% /
tmpfs          4.9G  76K   4.9G  1% /dev/shm
/dev/xvda1     488M  108M  356M  24% /boot
/dev/xvdf1      12G   6.5G  4.7G  58% /stage
```

```
[root@host01 catchup]# ssh host05 df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/xvda2       29G   5.5G   22G  20% /
tmpfs           4.8G   76K   4.8G   1% /dev/shm
/dev/xvda1      488M  117M  346M  26% /boot
/dev/xvdf1       12G   6.5G   4.7G  58% /stage

[root@host01 catchup]#
```

5. Now, you can run the catchup script (`less01_catchup.sh`) to install and configure the Grid Infrastructure software.
 - Refer to **Appendix A-2: Catchup for Practice 1**

Appendix A-2: Catchup Script for Practice 1

Estimated Time: 60min

1. Open a terminal window to connect to host01 as root user. Make sure the /u01/app/12.2.0/grid directory is empty.

```
[oracle@dns ~] ssh root@host01
root@host01's Password:

[root@host01 ~]# ssh host01 ls -l /u01/app/12.2.0/grid
ls: cannot access /u01/app/12.2.0/grid: No such file or directory

[root@host01 ~]# ssh host02 ls -l /u01/app/12.2.0/grid
ls: cannot access /u01/app/12.2.0/grid: No such file or directory

[root@host01 ~]# ssh host03 ls -l /u01/app/12.2.0/grid
ls: cannot access /u01/app/12.2.0/grid: No such file or directory

[root@host01 ~]# ssh host04 ls -l /u01/app/12.2.0/grid
ls: cannot access /u01/app/12.2.0/grid: No such file or directory

[root@host01 ~]# ssh host05 ls -l /u01/app/12.2.0/grid
ls: cannot access /u01/app/12.2.0/grid: No such file or directory

[root@host01 ~]#
```

2. Run the less01_catchup.sh script.

```
[root@host01 ~]# cd /stage/RAC/solutions/less_01/catchup
[root@host01 ~]# ./less01_catchup.sh

#####
Task 1 of 7: Creating software directories
#####
Fri Feb 2 17:42:03 UTC 2018

#####
Task 2 of 7: Configuring NSCD
#####
Fri Feb 2 17:42:08 UTC 2018

#####
Task 3 of 7: Configuring shell limits
#####
Fri Feb 2 17:42:11 UTC 2018

bash_profile          100%   194      0.2KB/s  00:00
bash_profile          100%   194      0.2KB/s  00:00
limits.conf           100%  1400     1.4KB/s  00:00
```

```

profile          100% 1308     1.3KB/s  00:00
profile          100% 1308     1.3KB/s  00:00
bash_profile    100% 194      0.2KB/s  00:00
bash_profile    100% 194      0.2KB/s  00:00
limits.conf     100% 1400     1.4KB/s  00:00
profile          100% 1308     1.3KB/s  00:00
profile          100% 1308     1.3KB/s  00:00
bash_profile    100% 194      0.2KB/s  00:00
bash_profile    100% 194      0.2KB/s  00:00
limits.conf     100% 1400     1.4KB/s  00:00
profile          100% 1308     1.3KB/s  00:00
profile          100% 1308     1.3KB/s  00:00
bash_profile    100% 194      0.2KB/s  00:00
bash_profile    100% 194      0.2KB/s  00:00
limits.conf     100% 1400     1.4KB/s  00:00
profile          100% 1308     1.3KB/s  00:00
profile          100% 1308     1.3KB/s  00:00
bash_profile    100% 194      0.2KB/s  00:00
bash_profile    100% 194      0.2KB/s  00:00
limits.conf     100% 1400     1.4KB/s  00:00
profile          100% 1308     1.3KB/s  00:00
profile          100% 1308     1.3KB/s  00:00
bash_profile    100% 194      0.2KB/s  00:00
bash_profile    100% 194      0.2KB/s  00:00
limits.conf     100% 1400     1.4KB/s  00:00
profile          100% 1308     1.3KB/s  00:00
profile          100% 1308     1.3KB/s  00:00
bash_profile    100% 194      0.2KB/s  00:00
bash_profile    100% 194      0.2KB/s  00:00
limits.conf     100% 1400     1.4KB/s  00:00
profile          100% 1308     1.3KB/s  00:00
profile          100% 1308     1.3KB/s  00:00
#####
Task 4 of 7: Unzipping the Grid Infrastructure SW
#####
Fri Feb  2 17:42:16 UTC 2018

Archive: /stage/clusterware.zip
  creating: /u01/app/12.2.0/grid/addnode/
  inflating: /u01/app/12.2.0/grid/addnode/addnode_oraparam.ini.sbs
  inflating: /u01/app/12.2.0/grid/addnode/addnode.pl
  inflating: /u01/app/12.2.0/grid/addnode/addnode.sh
  inflating: /u01/app/12.2.0/grid/addnode/addnode_oraparam.ini
  creating: /u01/app/12.2.0/grid/assistants/
  creating: /u01/app/12.2.0/grid/assistants/jlib/
  inflating: /u01/app/12.2.0/grid/assistants/jlib/initconfig
  inflating: /u01/app/12.2.0/grid/assistants/jlib/appfwk.jar
...
/u01/app/12.2.0/grid/lib/libodm12.so -> libodmd12.so
/u01/app/12.2.0/grid/lib/libagtsh.so -> libagtsh.so.1.0
/u01/app/12.2.0/grid/lib/libclntsh.so -> libclntsh.so.12.1
/u01/app/12.2.0/grid/lib/libocci.so -> libocci.so.12.1

#####
Task 5 of 7: Installing cvuqdisk RPM
#####
Fri Feb  2 17:44:29 UTC 2018

Preparing...
#####
[100%]
package cvuqdisk-1.0.10-1.x86_64 is already installed

```

```

Preparing...
#####
    package cvuqdisk-1.0.10-1.x86_64 is already installed

```

3. After a few moments, the Oracle Grid Infrastructure Setup Wizard is launched.

```

#####
Task 6 of 7: Run OUI to create the cluster
#####
Fri Feb  2 17:44:34 UTC 2018

Launching Oracle Grid Infrastructure Setup Wizard...

You can find the log of this install session at:
 /tmp/GridSetupActions2018-02-02_05-44-34PM/gridSetupActions2018-02-
02_05-44-34PM.log

```

4. Open a terminal window connected to host01 as the root user and monitor the installation progress. The following tasks are performed in this stage.

- Installing Oracle Grid Infrastructure on host01 to host05
- Creating the MGMT database
- Running the CLUVFY utility

```

[oracle@dns ~]$ ssh root@host01
root@host01's password:

[root@host01 ~]# tail -f /tmp/GridSetupActions2018-02-02_05-44-
34PM/gridSetupActions2018-02-02_05-44-34PM.log

...

INFO: [Timestamp] Successfully executed the flow in SILENT mode
INFO: [Timestamp] Finding the most appropriate exit status for the
current application
INFO: [Timestamp] inventory location is/u01/app/oraInventory
INFO: [Timestamp] Finding the most appropriate exit status for the
current application
INFO: [Timestamp] Exit Status is 0

```

```
INFO: [Timestamp] Shutdown Oracle Grid Infrastructure 12c Release 2  
Installer
```

5. Switch to the first terminal window to monitor the rest of tasks.

```
You can find the log of this install session at:  
/tmp/GridSetupActions2018-02-02_05-44-34PM/gridSetupActions2018-02-  
02_05-44-34PM.log
```

```
Successfully Setup Software  
Moved the install session logs to :  
/u01/app/oraInventory/logs/GridSetupActions2017-12-07_01-16-57PM
```

```
As a root user, execute the following script(s):  
1. /u01/app/oraInventory/orainstRoot.sh  
2. /u01/app/12.2.0/grid/root.sh
```

```
Execute /u01/app/oraInventory/orainstRoot.sh on the following nodes:  
[host01, host02, host03, host04, host05]  
Execute /u01/app/12.2.0/grid/root.sh on the following nodes:  
[host01, host02, host03, host04, host05]
```

```
Execute the script on the local node first. After successful  
completion, you can execute the script in parallel on all other nodes.  
Oracle recommends that the script executes on all HB nodes before  
executing the script on LEAF nodes.
```

```
Successfully Setup Software.
```

```
As install user, execute the following command to complete the  
configuration.
```

```
/u01/app/12.2.0/grid/gridSetup.sh -executeConfigTools -  
responseFile /stage/RAC/solutions/less_01/catchup/less01_grid.rsp [-  
silent]
```

```
Moved the install session logs to:
```

```
/u01/app/oraInventory/logs/GridSetupActions2018-02-02_05-44-34PM  
Changing permissions of /u01/app/oraInventory.
```

```
Adding read,write permissions for group.
```

```
Removing read,write,execute permissions for world.
```

```
Changing groupname of /u01/app/oraInventory to oinstall.
```

```
The execution of the script is complete.
```

```
Changing permissions of /u01/app/oraInventory.
```

```
Adding read,write permissions for group.
```

```
Removing read,write,execute permissions for world.
```

```
Changing groupname of /u01/app/oraInventory to oinstall.
```

```
The execution of the script is complete.
```

```
Changing permissions of /u01/app/oraInventory.
```

```
Adding read,write permissions for group.
```

```
Removing read,write,execute permissions for world.
```

```
Changing groupname of /u01/app/oraInventory to oinstall.  
The execution of the script is complete.  
Changing permissions of /u01/app/oraInventory.  
Adding read,write permissions for group.  
Removing read,write,execute permissions for world.  
  
Changing groupname of /u01/app/oraInventory to oinstall.  
The execution of the script is complete.  
Changing permissions of /u01/app/oraInventory.  
Adding read,write permissions for group.  
Removing read,write,execute permissions for world.  
  
Changing groupname of /u01/app/oraInventory to oinstall.  
The execution of the script is complete.  
Check /u01/app/12.2.0/grid/install/root_host01.example.com_2018-02-  
02_18-00-22-350262475.lo for the output of root script  
Check /u01/app/12.2.0/grid/install/root_host02.example.com_2018-02-  
02_18-13-48-077925368.lo for the output of root script  
Check /u01/app/12.2.0/grid/install/root_host03.example.com_2018-02-  
02_18-19-28-852096296.lo for the output of root script  
Check /u01/app/12.2.0/grid/install/root_host04.example.com_2018-02-  
02_18-25-05-782707676.lo for the output of root script  
Check /u01/app/12.2.0/grid/install/root_host05.example.com_2018-02-  
02_18-28-50-031415660.lo for the output of root script  
  
#####  
Task 7 of 7: Run the config tools to complete cluster config  
#####  
Fri Feb  2 18:32:32 UTC 2018  
  
Setting the invPtrLoc to /u01/app/12.2.0/grid/oraInst.loc  
  
perform - mode is starting for action: configure  
  
perform - mode finished for action: configure  
  
You can see the log file:  
/u01/app/12.2.0/grid/cfgtoollogs/oui/configActions2018-02-02_06-3-34-  
PM.log  
  
Post configuration completed successfully  
  
Fri Feb  2 18:34:50 UTC 2018  
Elapsed Real Time: 52:46.75  
  
[root@host01 catchup]#
```

6. Open a new terminal window connected to host01 as the grid user. Check the status of the cluster. Ensure that all the listed services are online on all cluster nodes.

```
[oracle@dns ~]$ ssh grid@host01
grid@host01's password:

[grid@host01 ~]$ . oraenv
ORACLE_SID = [grid] ? +ASM1
The oracle base has been set to /u01/app/grid

[grid@host01 ~]$ crsctl check cluster -all

*****
host01:
CRS-4537: Cluster Ready Services is online
CRS-4529: Cluster Synchronization Services is online
CRS-4533: Event Manager is online
*****
host02:
CRS-4537: Cluster Ready Services is online
CRS-4529: Cluster Synchronization Services is online
CRS-4533: Event Manager is online
*****
host03:
CRS-4537: Cluster Ready Services is online
CRS-4529: Cluster Synchronization Services is online
CRS-4533: Event Manager is online
*****
host04:
CRS-4537: Cluster Ready Services is online
CRS-4529: Cluster Synchronization Services is online
CRS-4533: Event Manager is online
*****
host05:
CRS-4537: Cluster Ready Services is online
CRS-4529: Cluster Synchronization Services is online
CRS-4533: Event Manager is online
*****
```

[grid@host01 ~]\$ crsctl stat res -t

Name	Target	State	Server	State details
<hr/> Local Resources <hr/>				
ora.ASMNET1LSNR_ASM.lsnr	ONLINE	ONLINE	host01	STABLE
	ONLINE	ONLINE	host02	STABLE
	ONLINE	ONLINE	host03	STABLE
ora.DATA.dg	ONLINE	ONLINE	host01	STABLE
	ONLINE	ONLINE	host02	STABLE

	ONLINE	ONLINE	host03	STABLE
ora.LISTENER.lsnr				
	ONLINE	ONLINE	host01	STABLE
	ONLINE	ONLINE	host02	STABLE
	ONLINE	ONLINE	host03	STABLE
ora.LISTENER_LEAF.lsnr				
	OFFLINE	OFFLINE	host04	STABLE
	OFFLINE	OFFLINE	host05	STABLE
ora.MGMT.dg				
	ONLINE	ONLINE	host01	STABLE
	ONLINE	ONLINE	host02	STABLE
	ONLINE	ONLINE	host03	STABLE
ora.chad				
	ONLINE	ONLINE	host01	STABLE
	ONLINE	ONLINE	host02	STABLE
	ONLINE	ONLINE	host03	STABLE
	ONLINE	ONLINE	host04	STABLE
	ONLINE	ONLINE	host05	STABLE
ora.net1.network				
	ONLINE	ONLINE	host01	STABLE
	ONLINE	ONLINE	host02	STABLE
	ONLINE	ONLINE	host03	STABLE
ora.ons				
	ONLINE	ONLINE	host01	STABLE
	ONLINE	ONLINE	host02	STABLE
	ONLINE	ONLINE	host03	STABLE
ora.proxy_advm				
	OFFLINE	OFFLINE	host01	STABLE
	OFFLINE	OFFLINE	host02	STABLE
	OFFLINE	OFFLINE	host03	STABLE
<hr/>				
Cluster Resources				
<hr/>				
ora.LISTENER_SCAN1.lsnr				
	1	ONLINE	ONLINE	host03
ora.LISTENER_SCAN2.lsnr				
	1	ONLINE	ONLINE	host02
ora.LISTENER_SCAN3.lsnr				
	1	ONLINE	ONLINE	host01
ora.MGMLSRR				
	1	ONLINE	ONLINE	host01
				169.254.2.192
				192.168.1.101
				192.168.2.101
				, STABLE
ora.asm				
	1	ONLINE	ONLINE	host01
	2	ONLINE	ONLINE	host03
	3	ONLINE	ONLINE	host02
ora.cvu				
	1	ONLINE	ONLINE	host01
ora.gns				
	1	ONLINE	ONLINE	host01
ora.gns.vip				

1	ONLINE	ONLINE	host01	STABLE
ora.host01.vip				
1	ONLINE	ONLINE	host01	STABLE
ora.host02.vip				
1	ONLINE	ONLINE	host02	STABLE
ora.host03.vip				
1	ONLINE	ONLINE	host03	STABLE
ora.mgmtdb				
1	ONLINE	ONLINE	host01	Open, STABLE
ora.qosmserver				
1	ONLINE	ONLINE	host01	STABLE
ora.scan1.vip				
1	ONLINE	ONLINE	host03	STABLE
ora.scan2.vip				
1	ONLINE	ONLINE	host02	STABLE
ora.scan3.vip				
1	ONLINE	ONLINE	host01	STABLE
<hr/>				
[grid@host01 ~]\$				

7. Now, you can run the catchup script (`less03_catchup.sh`) to install Oracle database SW and create the `orcl` database.

- Refer to **Appendix A-3: Catchup for Practice 3**

Appendix A-3: Catchup Script for Practice 3

Estimated Time: 40min

1. Open a terminal window to connect to host01 as root user. Make sure that the root file system (/) has at least 8GB of free space.

```
[oracle@dns ~] ssh root@host01
root@host01's Password:
[root@host01 ~]# ssh host01 df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/xvda2     128G   18G  104G  15% /
tmpfs          6.0G  641M  5.4G  11% /dev/shm
/dev/xvda1     488M  133M  330M  29% /boot
/dev/xvdf1      12G   6.5G  4.7G  58% /stage
[root@host01 ~]#
[root@host01 ~]# ssh host02 df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/xvda2     29G   15G  14G  52% /
tmpfs          6.0G  640M  5.4G  11% /dev/shm
/dev/xvda1     488M  133M  330M  29% /boot
/dev/xvdf1      12G   6.5G  4.7G  58% /stage
[root@host01 ~]#
[root@host01 ~]# ssh host03 df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/xvda2     29G   15G  14G  52% /
tmpfs          4.9G  640M  4.3G  13% /dev/shm
/dev/xvda1     488M  107M  356M  24% /boot
/dev/xvdf1      12G   6.5G  4.7G  58% /stage
[root@host01 ~]#
[root@host01 ~]# ssh host04 df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/xvda2     29G   15G  14G  52% /
tmpfs          4.9G  76K   4.9G   1% /dev/shm
/dev/xvda1     488M  108M  356M  24% /boot
/dev/xvdf1      12G   6.5G  4.7G  58% /stage
[root@host01 ~]#
[root@host01 ~]# ssh host05 df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/xvda2     29G   15G  14G  52% /
tmpfs          4.8G  76K   4.8G   1% /dev/shm
/dev/xvda1     488M  117M  346M  26% /boot
/dev/xvdf1      12G   6.5G  4.7G  58% /stage
[root@host01 ~]#
```

2. Run the less03_catchup.sh script.

```
[root@host01 ~]# cd /stage/RAC/solutions/less_03/catchup
[root@host01 catchup]# ./less03_catchup.sh
#####
#####
```

```
Task 1 of 5: Create and mount FRA disk group
#####
Fri Dec  8 09:13:34 UTC 2017

#####
Task 2 of 5: Change the permission on /u01/app
#####
Fri Dec  8 09:13:38 UTC 2017

#####
Task 3 of 5: Run OUI to install DB software
#####
Fri Dec  8 09:13:39 UTC 2017

Starting Oracle Universal Installer...

Checking Temp space: must be greater than 500 MB.    Actual 106913 MB
Passed
Checking swap space: must be greater than 150 MB.    Actual 10699 MB
Passed
Preparing to launch Oracle Universal Installer from
/tmp/OraInstall2017-12-08_09-13-39AM. Please wait ...You can find the
log of this install session at:
/u01/app/oraInventory/logs/installActions2017-12-08_09-13-39AM.log

Prepare in progress.
..... 7% Done.

Prepare successful.

Copy files in progress.
..... 14% Done.
..... 20% Done.
..... 25% Done.
..... 30% Done.
..... 36% Done.
..... 45% Done.
..... 50% Done.
..... 55% Done.
..... 60% Done.
..... 65% Done.
.....
Copy files successful.

Link binaries in progress.
.....
Link binaries successful.

Setup files in progress.
.....
Setup files successful.
```

Setup Inventory in progress.

Setup Inventory successful.

Finish Setup successful.

The installation of Oracle Database 12c was successful.

Please check '/u01/app/oraInventory/logs/silentInstall2017-12-08_09-13-39AM.log' for more details.

Copy Files to Remote Nodes in progress.

Copy Files to Remote Nodes successful.

..... 85% Done.

Prepare in progress.

Prepare successful.

.....

Setup in progress.

.....

Setup successful.

The Cluster Node Addition of /u01/app/oracle/product/12.2.0/dbhome_1 was successful.

Please check '/u01/app/oraInventory/logs/silentInstall2017-12-08_09-13-39AM.log' for more details.

Setup Oracle Base in progress.

Setup Oracle Base successful.

..... 97% Done.

As a root user, execute the following script(s):

1. /u01/app/oracle/product/12.2.0/dbhome_1/root.sh

Execute /u01/app/oracle/product/12.2.0/dbhome_1/root.sh on the following nodes:

[host01, host03, host02, host04, host05]

..... 100% Done.

Successfully Setup Software.

Starting Oracle Universal Installer...

Checking Temp space: must be greater than 500 MB. Actual 99149 MB
Passed

Checking swap space: must be greater than 150 MB. Actual 10684 MB
Passed

Preparing to launch Oracle Universal Installer from /tmp/OraInstall2017-12-08_09-33-46AM. Please wait ...

#####

Task 4 of 5: Run root scripts

```
#####
Fri Dec  8 09:33:59 UTC 2017

Check
/u01/app/oracle/product/12.2.0/dbhome_1/install/root_host01.example.co
m_2017-12-08_09-33-59-216859428.log for the output of root script
Check
/u01/app/oracle/product/12.2.0/dbhome_1/install/root_host02.example.co
m_2017-12-08_09-34-00-356267198.log for the output of root script
Check
/u01/app/oracle/product/12.2.0/dbhome_1/install/root_host03.example.co
m_2017-12-08_09-34-00-809833542.log for the output of root script
Check
/u01/app/oracle/product/12.2.0/dbhome_1/install/root_host04.example.co
m_2017-12-08_09-34-01-219757557.log for the output of root script
Check
/u01/app/oracle/product/12.2.0/dbhome_1/install/root_host05.example.co
m_2017-12-08_09-34-01-674218372.log for the output of root script

#####
Task 5 of 5: Run DBCA to create database
#####
Fri Dec  8 09:34:01 UTC 2017

You can find the logs of this session at:
/u01/app/oraInventory/logs

[FATAL] [INS-32601] The Installer has detected that there are no
config tools to execute for the specified Oracle home.
[WARNING] [DBT-12504] For Policy Managed database, database instance
may come up on local node depending on node availability. Sample
schema on local node will be configured only when database instance
comes up on local node.
Copying database files
1% complete
2% complete
15% complete
27% complete
Creating and starting Oracle instance
29% complete
32% complete
36% complete
40% complete
43% complete
44% complete
45% complete
47% complete
Creating cluster database views
49% complete
65% complete
Completing Database Creation
66% complete
67% complete
```

```

70% complete
71% complete
73% complete
74% complete
Executing Post Configuration Actions
100% complete
Look at the log file "/u01/app/oracle/cfgtoollogs/dbca/orcl/orcl.log"
for further details.

Fri Dec  8 09:46:10 UTC 2017

[root@host01 catchup]#

```

3. Optionally, delete the large files in the /var/log directory.

```

[root@host01 catchup]# ssh host01 rm -f /var/log/mess*
[root@host01 catchup]# ssh host02 rm -f /var/log/mess*
[root@host01 catchup]# ssh host03 rm -f /var/log/mess*
[root@host01 catchup]# ssh host04 rm -f /var/log/mess*
[root@host01 catchup]# ssh host05 rm -f /var/log/mess*
[root@host01 catchup]#
[root@host01 catchup]# ssh host01 rm -rf /tmp/*
[root@host01 catchup]# ssh host02 rm -rf /tmp/*
[root@host01 catchup]# ssh host03 rm -rf /tmp/*
[root@host01 catchup]# ssh host04 rm -rf /tmp/*
[root@host01 catchup]# ssh host05 rm -rf /tmp/*
[root@host01 catchup]#

```

4. Optionally, purge the deleted files that are held open to reclaim disk space.

```

[root@host01 catchup]# lsof |grep delete
java      4322      root    txt      REG            202,2
7734      745648                      /u01/app/12.2.0/grid/jdk/jre/bin/java
(deleted)
...
[root@host01 catchup]# kill -9 4322
[root@host01 catchup]# lsof |grep delete
[root@host01 catchup]#
[root@host01 catchup]# ssh host02 lsof|grep delete
java      29466      root    txt      REG            202,2
7734      986836 /u01/app/12.2.0/grid/jdk/jre/bin/java (deleted)
...
[root@host01 catchup]# ssh host02 kill -9 29466
[root@host01 catchup]# ssh host02 lsof |grep delete
[root@host01 catchup]#
[root@host01 catchup]# ssh host03 lsof|grep delete
java      20160      root    txt      REG            202,2      7734
1518544 /u01/app/12.2.0/grid/jdk/jre/bin/java (deleted)
...
[root@host01 catchup]# ssh host03 kill -9 20160
[root@host01 catchup]# ssh host03 lsof|grep delete
[root@host01 catchup]#
[root@host01 catchup]# ssh host04 lsof|grep delete

```

```

nautilus 3002 oracle 19r REG 202,2
524 1581093 /home/oracle/.local/share/gvfs-metadata/home
(deleted)
nautilus 3002 oracle 20r REG 202,2
32768 1583605 /home/oracle/.local/share/gvfs-metadata/home-
cdd03e8e.log (deleted)
java 23432 root txt REG 202,2
7734 360449 /u01/app/12.2.0/grid/jdk/jre/bin/java (deleted)
...
[root@host01 catchup]# ssh host04 kill -9 3002 23432
[root@host01 catchup]# ssh host04 lsof|grep delete
[root@host01 catchup]#
[root@host01 catchup]# ssh host05 lsof|grep delete
java 14322 root txt REG 202,2 7734
114768 /u01/app/12.2.0/grid/jdk/jre/bin/java (deleted)
...
tail 21883 grid 3r REG 202,2 5800
1430917 /u01/app/12.2.0/grid/install/root_host05.example.com_2018-
01-27_11-36-01-428452867.log (deleted)
[root@host01 catchup]# ssh host05 kill -9 14322 21883
[root@host01 catchup]# ssh host05 lsof|grep delete
[root@host01 catchup]#

```

5. Make sure the root file system (/) has enough free space (5GB+). The available space should be close enough to the one in the output.

```

[root@host01 ~]# ssh host01 df -h
Filesystem Size Used Avail Use% Mounted on
/dev/xvda2 128G 22G 100G 18% /
tmpfs 6.0G 641M 5.4G 11% /dev/shm
/dev/xvda1 488M 133M 330M 29% /boot
/dev/xvdf1 12G 6.5G 4.7G 58% /stage
[root@host01 ~]#
[root@host01 ~]# ssh host02 df -h
Filesystem Size Used Avail Use% Mounted on
/dev/xvda2 29G 22G 6.4G 77% /
tmpfs 6.0G 641M 5.4G 11% /dev/shm
/dev/xvda1 488M 133M 330M 29% /boot
/dev/xvdf1 12G 6.5G 4.7G 58% /stage
[root@host01 ~]#
[root@host01 ~]# ssh host03 df -h
Filesystem Size Used Avail Use% Mounted on
/dev/xvda2 29G 23G 5.2G 82% /
tmpfs 4.9G 641M 4.3G 13% /dev/shm
/dev/xvda1 488M 107M 356M 24% /boot
/dev/xvdf1 12G 6.5G 4.7G 58% /stage
[root@host01 ~]#
[root@host01 ~]# ssh host04 df -h
Filesystem Size Used Avail Use% Mounted on
/dev/xvda2 29G 23G 5.2G 82% /
tmpfs 4.9G 76K 4.9G 1% /dev/shm
/dev/xvda1 488M 108M 356M 24% /boot
/dev/xvdf1 12G 6.5G 4.7G 58% /stage

```

```
[root@host01 ~]#
[root@host01 ~]# ssh host05 df -h
Filesystem      Size   Used  Avail Use% Mounted on
/dev/xvda2       29G   23G   5.2G  82% /
tmpfs           4.8G   76K   4.8G   1% /dev/shm
/dev/xvda1     488M  117M  346M  26% /boot
/dev/xvdf1      12G   6.5G   4.7G  58% /stage

[root@host01 catchup]#
```

6. Open a terminal window connected to host01 as the oracle user and check the database status and configuration details.

```
[oracle@dns ~]$ ssh oracle@host01
oracle@host01's password:

[oracle@host01 ~]$ export
ORACLE_HOME=/u01/app/oracle/product/12.2.0/dbhome_1

[oracle@host01 ~]$ export PATH=$PATH:$ORACLE_HOME/bin

[oracle@host01 ~]$ srvctl status database -db orcl
Instance orcl_1 is running on node host02
Instance orcl_2 is running on node host03
Instance orcl_3 is running on node host01

[oracle@host01 ~]$ srvctl config database -db orcl
Database unique name: orcl
Database name: orcl
Oracle home: /u01/app/oracle/product/12.2.0/dbhome_1
Oracle user: oracle
Spfile: +DATA/ORCL/PARAMETERFILE/spfile.297.958993403
Password file: +DATA/ORCL/PASSWORD/pwdorcl.284.958993065
Domain:
Start options: open
Stop options: immediate
Database role: PRIMARY
Management policy: AUTOMATIC
Server pools: orcldb
Disk Groups: FRA, DATA
Mount point paths:
Services:
Type: RAC
Start concurrency:
Stop concurrency:
OSDBA group: dba
OSOPER group: oper
Database instances:
Configured nodes:
CSS critical: no
CPU count: 0
Memory target: 0
Maximum memory: 0
```

```
Default network number for database services:  
Database is policy managed  
[oracle@host01 ~] $
```

7. Now, you can move to Practice 4.
8. Close all terminals opened for this practice.

GANG LIU (gangli@baylorhealth.edu) has a non-transferable license
to use this Student Guide.