



Hardware and Software
Engineered to Work Together



Oracle Database 12c: RAC and Data Guard Integration Workshop

Activity Guide

D94430GC10

Edition 1.0 | May 2016 | D95961

Learn more from Oracle University at oracle.com/education/

Author

Sean Kim

**Technical Contributors
and Reviewers**

Jerry Lee
Jim Womack
Peter Fusek

Editors

Chandrika Kennedy
Aju Kumar
Aishwarya Menon

Graphic Designer

Kavya Bellur

Publishers

Asief Baig
Veena Narasimhan

Copyright © 2016, 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

Practices for Lesson 1: Introduction to High Availability.....	1-1
Practices for Lesson 1: Overview.....	1-2
Practice 1-1: Laboratory Introduction	1-3
Practices for Lesson 2: Converting a Single Instance to an Oracle RAC Database	2-1
Practices for Lesson 2: Overview.....	2-2
Practice 2-1: Converting a Single Instance to a RAC Database on a Remote Cluster Using DBCA.....	2-3
Practice 2-2: Dropping the Existing RAC Database on cluster02.....	2-9
Practice 2-3: Converting a Single Instance to a RAC Database on the Same Cluster Using Rconfig.....	2-10
Practices for Lesson 3: Configuring Oracle Net Services in a Data Guard Environment with RAC	3-1
Practices for Lesson 3: Overview.....	3-2
Practice 3-1: Modifying the tnsnames.ora Configuration File on the Primary and Standby Hosts.....	3-3
Practice 3-2: Understanding Listener Management in an Oracle RAC Environment	3-7
Practice 3-3: Modifying the listener.ora Configuration File on the Primary and Standby Hosts.....	3-18
Practices for Lesson 4: Deploying a Physical Standby Database in an Oracle RAC environment by Using SQL and RMAN Commands	4-1
Practices for Lesson 4: Overview.....	4-2
Practice 4-1: Preparing the Primary Database to Create a Physical Standby Database in RAC	4-3
Practice 4-2: Preparing the Standby Hosts and Creating a Physical Standby Database Using RMAN.....	4-8
Practice 4-3: Completing a Physical Standby Database for RAC.....	4-20
Practice 4-4: Verifying Redo Transport and Redo Apply Operations in RAC.....	4-26
Practices for Lesson 5: Configuring Oracle Data Guard in an Oracle RAC Environment.....	5-1
Practices for Lesson 5: Overview.....	5-2
Practice 5-1: Using a Shared Password File in an ASM Disk Group.....	5-3
Practice 5-2: Creating a Data Guard Broker Configuration for RAC.....	5-9
Practice 5-3: Monitoring Redo Transport Service and Redo Apply Service in RAC	5-14
Practice 5-4: Configuring Flashback Database on the Primary and Physical Standby Databases	5-20
Practice 5-5: Enabling Fast-Start Failover for RAC.....	5-24
Practice 5-6: Testing Fast-Start Failover in RAC.....	5-28
Practice 5-7: Performing Switchover to Reinstated Database in RAC and Disabling Fast-Start Failover.....	5-31
Practices for Lesson 6: Managing Physical Standby Files After Structural Changes on the Primary Database	6-1
Practices for Lesson 6: Overview.....	6-3
Practice 6-1: Creating an Archival Backup.....	6-4
Practice 6-2: Using Transportable Tablespaces with a Physical Standby Database.....	6-6
Practice 6-3: Moving the Location of a Data File Online	6-16
Practice 6-4: Adding Online and Standby Log Groups.....	6-21
Practices for Lesson 7: Effective Client Failover Using Application Continuity	7-1
Practices for Lesson 7: Overview.....	7-2
Practice 7-1: Preparing the Data Guard Environment for Application Continuity	7-3
Practice 7-2: Using Application Continuity in RAC	7-6
Practice 7-3: Using Application Continuity in Data Guard	7-14
Practices for Lesson 8: Effective Service Failover and Workload Management Using Global Data Services	8-1
Practices for Lesson 8: Overview.....	8-2
Practice 8-1: Reconfiguring the Environment for GDS	8-3
Practice 8-2: Installing and Configuring Global Data Services	8-7

Practice 8-3: Global Service Failover	8-22
Practice 8-4: Role-Based Global Services	8-25
Practice 8-5: Replication Lag-Based Routing.....	8-31
Practices for Lesson 9: Performing Database Recovery in an Oracle Data Guard Environment	9-1
Practices for Lesson 9: Overview	9-2
Practice 9-1: Reconfiguring the Environment.....	9-3
Practice 9-3: Recovering Standby Control Files (Media Failure).....	9-12
Practice 9-4: Performing the Steps for Rolling Forward a Physical Standby Database Using RMAN Incremental Backup	9-22
Practice 9-5: Performing Incomplete Recovery Using Flashback Database (Logical Failure).....	9-36
Practices for Lesson 10: Performing Data Guard Standby-First Patch Apply.....	10-1
Practices for Lesson 10: Overview	10-2
Practice 10-1: Patching the RAC Database Homes in the Standby Hosts.....	10-3
Practice 10-2: Performing Switchover	10-13
Practice 10-3: Patching the RAC Database Homes in the New Standby Hosts.....	10-15
Practice 10-4: Performing Switchover to the Original Configuration.....	10-22
Practices for Lesson 11: Disassociating a Snapshot Standby Database from a Data Guard Configuration.....	11-1
Practices for Lesson 11: Overview.....	11-2
Practice 11-1a: Converting a Snapshot Standby Database into a New Independent Database by Using the DBNEWID (NID) utility	11-3
Practice 11-1b: Converting a Snapshot Standby Database into a New Independent Database by Using the DBNEWID (NID) Utility (Optional)	11-12
Practice 11-2: Reconfiguring the Environment for Rolling Upgrade	11-21
Practices for Lesson 12: Rolling Database Upgrade Using Transient Logical Standby.....	12-1
Practices for Lesson 12: Overview	12-2
Practice 12-1: Laboratory Introduction	12-3
Practice 12-2: Grid Infrastructure Upgrade	12-6
Practice 12-3: Database Preparation for Rolling Upgrade	12-38
Practice 12-4: Database Rolling Upgrade	12-41
Practice 12-5: Post-Upgrade Steps.....	12-80
Appendix.....	13-1
Appendix 13: Overview	13-2
Practice 13-1: Troubleshooting (Case 1).....	13-3
Practice 13-2: Troubleshooting (Case 2).....	13-7
Practice 13-3: Troubleshooting (Case 3).....	13-13
Practice 13-4: Re-creating Data Guard Broker Configuration (Case 4).....	13-14
Practice 13-5: Resynchronizing a Physical Standby Database by Using RMAN Incremental Backup (Case 5)	13-19
Practice 13-6: Recovering the Primary Database by Using Archival Backup (Case 6)	13-29
Practice 13-7: Re-creating the Physical Standby Database (Case 7)	13-34
Practice 13-8: Reconfiguring Your Practice Environment (Case 8).....	13-42

Practices for Lesson 1: Introduction to High Availability

Chapter 1

Practices for Lesson 1: Overview

Practice Overview

In this practice, you familiarize yourself with the laboratory environment for this course.

Practice Summary

- Practice 1-1: Laboratory Introduction
- Practice 1-2: Review Troubleshooting Cases in Appendix

Practice 1-1: Laboratory Introduction

Overview

In this practice, you will familiarize yourself with the laboratory environment for this course.

Tasks

Access to your laboratory environment will be through a graphical display running on your classroom workstation or hosted on a remote machine. Your instructor will provide you with instructions to access your practice environment.

The practice environment for this course is hosted on a server running Oracle Virtual Machine (OVM). In turn, OVM hosts numerous virtual machines (VMs). Each VM is a logically separate server that will be used to run the Oracle Database 12c software, including Global Data Services, Clusterware, Automatic Storage Management (ASM), and Real Application Clusters (RAC).

1. Open a terminal window and execute the command `xm list` as root OS user. The root user password is `oracle`. You should see output similar to the following example. It shows that your server is hosting five domains. Domain-0 is the OVM server. The other domains relate to four VMs, which you will use in the following practices.

```
[Classroom PC] $ su -
Password: <oracle>
[root@Classroom PC] # xm list
Name ID Mem VCPUs State Time(s)
Domain-0 0 1024 2 r---- 54609.2
enode01 1 3800 1 -b---- 98.0
enode02 2 3400 1 -b---- 97.2
wnode03 3 3400 1 -b---- 96.9
wnode04 4 3400 1 -b---- 96.1
[root@Classroom PC] #
```

2. Using SSH, connect to enode01 as the grid OS user. Enter `oracle` when you are prompted for the password. Note that you may see additional messages relating to server identities. Answer yes if you are prompted to acknowledge server authenticity.

```
[root@Classroom PC] # ssh grid@enode01
grid@enode01's password: <oracle>
[grid@enode01 ~]$
```

3. Make sure that you set up your environment variables correctly for the grid user to point to your grid installation.

```
[grid@enode01 ~]$ . oraenv
ORACLE_SID = [grid]? +ASM1
The Oracle base has been set to /u01/app/grid
[grid@enode01 ~]$
```

4. Your environment has been preconfigured with a two-node cluster named `cluster01`. Execute the following command to identify the members of `cluster01`:

```
[grid@enode01 ~]$ olsnodes -c
cluster01
[grid@enode01 ~]$ olsnodes -n
enode01 1
enode02 2
```

5. Execute the following command and verify all of the services in `cluster01`. Ensure that you have the `DATA` and `FRA` ASM disk groups and the single instance databases named `eastdb` and `gdscat` on `enode01`.

<code>[grid@enode01 ~]\$ crsctl stat res -t</code>							
Name	Target	State	Server	State details			
<hr/>							
Local Resources							
<code>ora.DATA.dg</code>	ONLINE	ONLINE	enode01	STABLE			
	ONLINE	ONLINE	enode02	STABLE			
<code>ora.FRA.dg</code>	ONLINE	ONLINE	enode01	STABLE			
	ONLINE	ONLINE	enode02	STABLE			
<code>ora.LISTENER.lsnr</code>	ONLINE	ONLINE	enode01	STABLE			
	ONLINE	ONLINE	enode02	STABLE			
<code>ora.asm</code>	ONLINE	ONLINE	enode01	Started, STABLE			
	ONLINE	ONLINE	enode02	Started, STABLE			
<code>ora.net1.network</code>	ONLINE	ONLINE	enode01	STABLE			
	ONLINE	ONLINE	enode02	STABLE			
<code>ora.ons</code>	ONLINE	ONLINE	enode01	STABLE			
	ONLINE	ONLINE	enode02	STABLE			
<hr/>							
Cluster Resources							
<code>ora.LISTENER_SCAN1.lsnr</code>	1	ONLINE	ONLINE	enode02	STABLE		
	1	ONLINE	ONLINE	enode01	STABLE		
<code>ora.LISTENER_SCAN2.lsnr</code>	1	ONLINE	ONLINE	enode01	STABLE		
	1	ONLINE	ONLINE	enode01	STABLE		
<code>ora.MGMTLSNR</code>	1	OFFLINE	OFFLINE		STABLE		
	1	ONLINE	ONLINE	enode01	STABLE		
<code>ora.cvu</code>	1	ONLINE	ONLINE	enode01	STABLE		
<code>ora.eastdb.db</code>	1	ONLINE	ONLINE	enode01	Open, STABLE		
<code>ora.enode01.vip</code>	1	ONLINE	ONLINE	enode01	STABLE		
<code>ora.enode02.vip</code>	1	ONLINE	ONLINE	enode02	STABLE		
<code>ora.gdscat.db</code>							

1	ONLINE	ONLINE	enode01	Open, STABLE
ora.mgmtdb	OFFLINE	OFFLINE		STABLE
1	ONLINE	ONLINE	enode01	STABLE
ora.oc4j	1	ONLINE	enode02	STABLE
ora.scan1.vip	1	ONLINE	enode01	STABLE
ora.scan2.vip	1	ONLINE	enode01	STABLE
ora.scan3.vip	1	ONLINE	enode01	STABLE

[grid@enode01 ~]\$				

6. Using the `srvctl` utility, verify the status of the clustered ASM instances in `cluster01`.

```
[grid@enode01 ~]$ srvctl status asm
ASM is running on enode01, enode02
[grid@enode01 ~]$
```

7. Using the `srvctl` utility, verify the status of the `eastdb` database.

```
[grid@enode01 ~]$ srvctl status database -db eastdb
Instance eastdb is running on node enode01
[grid@enode01 ~]$
```

So far, you have examined `cluster01`, which includes two cluster nodes (`enode01` and `enode02`), clustered ASM instances, and the single instance database (`eastdb`).

8. Open a new terminal window. Using SSH, connect to `wnode03` as the `grid` OS user. Enter `oracle` when you are prompted for the password. Note that you may see additional messages relating to server identities. Answer `yes` if you are prompted to acknowledge server authenticity.

```
[classroom PC]$ ssh grid@wnode03
grid@wnode03's password: <oracle>
[grid@wnode03 ~]$
```

9. Make sure that you set up your environment variables correctly for the `grid` user to point to your `grid` installation.

```
[grid@wnode03 ~]$ . oraenv
ORACLE_SID = [grid]? +ASM1
The Oracle base has been set to /u01/app/grid
[grid@wnode03 ~]$
```

10. Your environment has been preconfigured with another two-node cluster named `cluster02`. Execute the following command to identify the members of `cluster02`:

```
[grid@wnode03 ~]$ olsnodes -c
cluster02
[grid@wnode03 ~]$ olsnodes -n
wnode03 1
wnode04 2
[grid@wnode03 ~]$
```

11. Execute the following command and verify all of the services in cluster02. Ensure that you have the DATA and FRA ASM disk groups.

[grid@wnode03 ~]\$ crsctl stat res -t				
Name	Target	State	Server	State details
<hr/>				
Local Resources				
ora.DATA.dg	ONLINE	ONLINE	wnode03	STABLE
	ONLINE	ONLINE	wnode04	STABLE
ora.FRA.dg	ONLINE	ONLINE	wnode03	STABLE
	ONLINE	ONLINE	wnode04	STABLE
ora.LISTENER.lsnr	ONLINE	ONLINE	wnode03	STABLE
	ONLINE	ONLINE	wnode04	STABLE
ora.asm	ONLINE	ONLINE	wnode03	Started, STABLE
	ONLINE	ONLINE	wnode04	Started, STABLE
ora.net1.network	ONLINE	ONLINE	wnode03	STABLE
	ONLINE	ONLINE	wnode04	STABLE
ora.ons	ONLINE	ONLINE	wnode03	STABLE
	ONLINE	ONLINE	wnode04	STABLE
<hr/>				
Cluster Resources				
ora.LISTENER_SCAN1.lsnr	1	ONLINE ONLINE	wnode04	STABLE
ora.LISTENER_SCAN2.lsnr	1	ONLINE ONLINE	wnode04	STABLE
ora.LISTENER_SCAN3.lsnr	1	ONLINE ONLINE	wnode03	STABLE
ora.MGMTLSNR	1	OFFLINE OFFLINE		STABLE
ora.cvu	1	ONLINE ONLINE	wnode04	STABLE
ora.mgmtdb	1	OFFLINE OFFLINE		STABLE
ora.oc4j	1	ONLINE ONLINE	wnode03	STABLE
ora.scan1.vip	1	ONLINE ONLINE	wnode04	STABLE
ora.scan2.vip	1	ONLINE ONLINE	wnode04	STABLE
ora.scan3.vip	1	ONLINE ONLINE	wnode03	STABLE
ora.wnode03.vip	1	ONLINE ONLINE	wnode03	STABLE
ora.wnode04.vip	1	ONLINE ONLINE	wnode04	STABLE
<hr/>				
[grid@wnode03 ~]\$				

12. Using the `srvctl` utility, verify the status of the clustered ASM instances in `cluster02`.

```
[grid@wnode03 ~]$ srvctl status asm  
ASM is running on wnode03, wnode04  
[grid@wnode03 ~]$
```

So far, you have examined `cluster02`, which includes two cluster nodes (`wnode03` and `wnode04`) and clustered ASM instances. Note that there is no existing database in `cluster02`, but you will create a two-node RAC database in the following practices.

13. Exit all of your terminal sessions.

Practice 1-2: Review Troubleshooting Cases in Appendix

Overview

In this practice, you will quickly review the troubleshooting cases described in the appendix. When you encounter an issue during the practice activity, you can use one of the following troubleshooting cases, working with your instructors.

Tasks

- Review the following troubleshooting cases in the Appendix:

Note: This is for your reference only. **DO NOT PERFORM THE FOLLOWING TASKS AT THIS TIME.**

Appendix	Problems	Possible Workaround
13-1	<ul style="list-style-type: none"> FAL request rejected CORRUPTION DETECTED 	<ul style="list-style-type: none"> Manual Redo Log Gap Resolution
13-2	<ul style="list-style-type: none"> FAL request rejected CORRUPTION DETECTED 	<ul style="list-style-type: none"> Manual Redo Log Gap Resolution
13-3	<ul style="list-style-type: none"> Performance issue related 	<ul style="list-style-type: none"> Execute DBMS AQADM.QUEUE_TABLE
13-4	<ul style="list-style-type: none"> Data Guard Broker Configuration Issue 	<ul style="list-style-type: none"> Recreating Data Guard Broker Configuration
13-5	<ul style="list-style-type: none"> Unrecoverable Physical Standby Database Issue 	<ul style="list-style-type: none"> Resynchronizing the Physical Standby Database using RMAN Incremental Backup
13-6	<ul style="list-style-type: none"> Unrecoverable Primary Database Issue Flashback Database is available in the Physical Standby database 	<ul style="list-style-type: none"> Recovering the Primary Database Using Archival Backup
13-7	<ul style="list-style-type: none"> Unrecoverable Physical Standby Database Issue After Performing PITR in the primary database Flashback Database is NOT available in the Physical Standby database 	<ul style="list-style-type: none"> Recreating the Physical Standby Database
13-8	<ul style="list-style-type: none"> Unrecoverable Problems in the practice environment 	<ul style="list-style-type: none"> Reconfiguring your Practice Environment

Practices for Lesson 2: Converting a Single Instance to an Oracle RAC Database

Chapter 2

Practices for Lesson 2: Overview

Practice Overview

In this practice, you convert a single instance database into an Oracle RAC database using two methods. The first method shows how to convert a single instance *on a non-clustered environment* to an Oracle RAC database *using DBCA*. You create a two-node RAC database on cluster02 using a single instance database on cluster01 to simulate the first method. The second method shows how to convert a single instance *on a clustered environment* into an Oracle RAC database *using the Rconfig utility*. You leverage the existing database in the shared storage instead of moving the database files.

Practice Summary

- Practice 2-1: Converting a Single Instance to a RAC Database on a Remote Cluster Using DBCA
- Practice 2-2: Dropping the Existing RAC Database on cluster02
- Practice 2-3: Converting a Single Instance to a RAC Database on the Same Cluster Using Rconfig

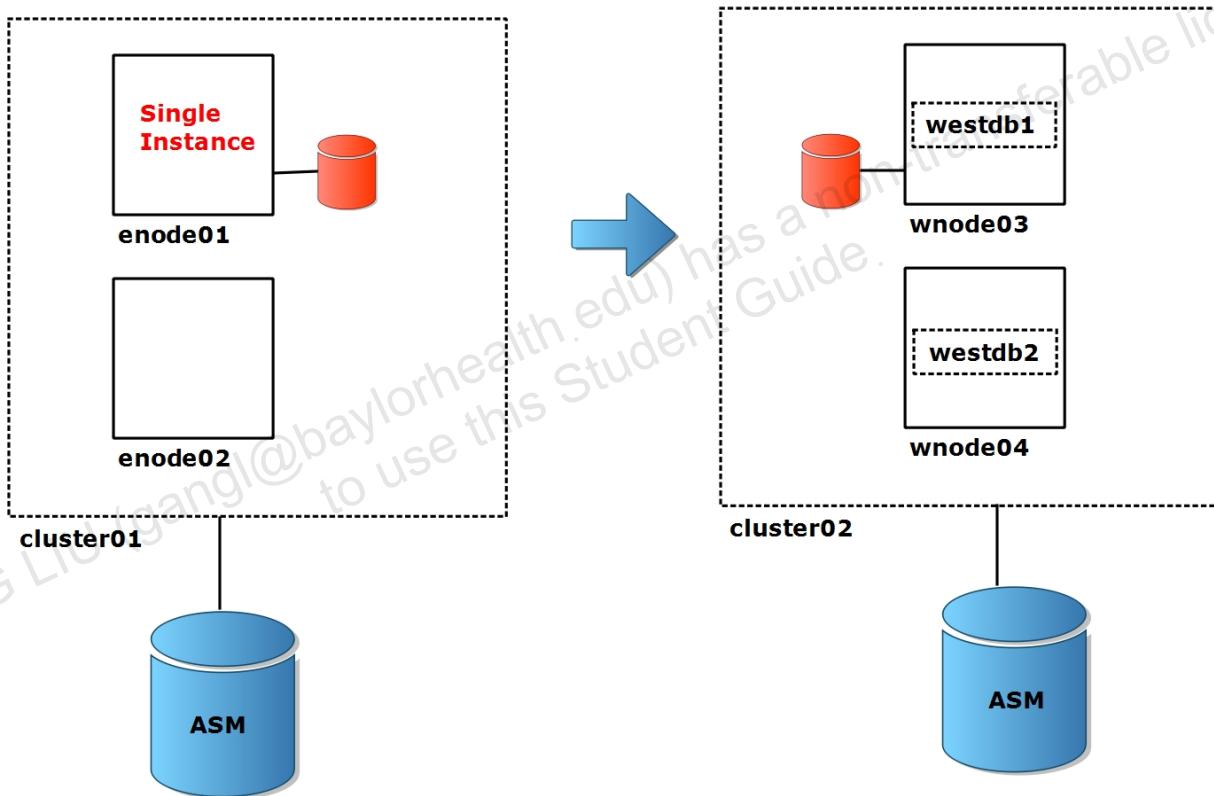
Practice 2-1: Converting a Single Instance to a RAC Database on a Remote Cluster Using DBCA

Overview

There are different ways to convert a single instance database to a RAC database. In this practice, you will convert a single instance database to RAC using DBCA.

Assumptions

- A single instance database named `eastdb` is running on `cluster01`.
- The database is converted into a two-node RAC database called `westdb` on `cluster02`.



Tasks

1. From your classroom PC desktop, execute `ssh -X oracle@enode01` to open a terminal session on `enode01` as the `oracle` user and configure the terminal environment as shown below.

```
[Classroom PC] $ ssh -X oracle@enode01
oracle@enode01's password: <oracle>
[oracle@enode01 ~]$ . oraenv
ORACLE_SID = [oracle] ? eastdb
The Oracle base has been set to /u01/app/oracle
[oracle@c01n01 ~]$
```

2. Start DBCA.

```
[oracle@enode01] $ dbca
```

Step	Screen/Page Description	Choices or Values
1	Database Operations	Select Manage Templates and click Next .
2	Template Management	Select Create a database template and From an existing database (structure as well as data) , and then click Next .
3	Source Database	Select <code>eastdb</code> in the Database instance list, and click Next .
4	Template Properties	Enter <code>Convert2RAC</code> in the Name field and click Next . By default, the template files are generated in the directory <code>\$ORACLE_HOME/assistants/dbca/templates</code> .
5	Database File Location	Select Maintain the file locations , so that you can restore the database to the current directory structure, and click Next .
6	Summary	Click Finish .
7	Progress	On the Finish page, click Close .

3. In the same terminal session, copy all template files starting with `Convert2RAC*` from the `$ORACLE_HOME/assistants/dbca/templates` directory to

`/u01/app/oracle/product/12.1.0/dbhome_1/assistants/dbca/templates` on `wnode03`.

```
[oracle@enode01 ~]$ cd $ORACLE_HOME/assistants/dbca/templates/
[oracle@enode01 templates]$ ls -l Convert2RAC*
-rw-r----- 1 oracle asmadmin 19251200 Sep  9 07:23 Convert2RAC.ctl
-rw-r----- 1 oracle oinstall    6594 Sep  9 07:26 Convert2RACdbc
-rw-r----- 1 oracle asmadmin 347021312 Sep  9 07:23 Convert2RACdfb
[oracle@enode01 templates]$ scp Convert2RAC*
wnode03:/u01/app/oracle/product/12.1.0/dbhome_1/assistants/dbca/templates/
oracle@wnode03's password: <oracle>
Convert2RAC.ctl                      100%   18MB   18.4MB/s   00:01
Convert2RACdbc                         100%   6594      6.4KB/s   00:00
Convert2RACdfb                        100%  331MB   10.0MB/s   00:33
[oracle@enode01 templates]$
```

4. Open a new terminal window, and execute `ssh -X oracle@wnode03` to connect to `wnode03` as the `oracle` OS user. Then navigate to

`/u01/app/oracle/product/12.1.0/dbhome_1/assistants/dbca/templates` to verify the template files.

```
[Classroom PC]$ ssh -X oracle@wnode03
oracle@enode01's password: <oracle>
[oracle@wnode03 ~]$ cd
/u01/app/oracle/product/12.1.0/dbhome_1/assistants/dbca/templates
[oracle@wnode03 templates]$ ls -l Convert2RAC*
-rw-r----- 1 oracle asmadmin 19251200 Sep  9 07:23 Convert2RAC.ctl
-rw-r----- 1 oracle oinstall    6594 Sep  9 07:26 Convert2RACdbc
-rw-r----- 1 oracle asmadmin 347021312 Sep  9 07:23 Convert2RACdfb
[oracle@wnode03 templates]$
```

5. Review the database structure file (`Convert2RAC.dbc`) to determine the location of the database preconfigured image file (`Convert2RAC.dfb`).

Note: If you copied the database preconfigured image file to a different location, you will have to modify the location accordingly. In the classroom, the database preconfigured image file is located in the default directory.

```
[oracle@wnode03 templates]$ cat Convert2RACdbc
<?xml version = '1.0'?>
<DatabaseTemplate name="Convert2RAC" description=""
version="12.1.0.2.0">
  <CommonAttributes>
    <option name="OMS" value="true"/>
    <option name="JSERVER" value="true"/>
    <option name="SPATIAL" value="true"/>
    <option name="IMEDIA" value="true"/>
    <option name="XDB_PROTOCOLS" value="true"/>
    <option name="ORACLE_TEXT" value="true"/>
    <option name="SAMPLE_SCHEMA" value="true"/>
    <option name="CWMLITE" value="true"/>
    <option name="APEX" value="true"/>
    <option name="DV" value="true"/>
  </CommonAttributes>
  <Variables/>
  <CustomScripts Execute="false"/>
  <InitParamAttributes>
    <InitParams>
      <initParam name="db_create_file_dest" value="+DATA"/>
      <initParam name="db_name" value="" />
      <initParam name="db_domain" value="" />
      <initParam name="dispatchers" value="(PROTOCOL=TCP)
(SERVICE={SID}XDB)"/>
      <initParam name="audit_file_dest"
value="/u01/app/oracle/admin/eastdb/adump"/>
      <initParam name="compatible" value="12.1.0.2.0"/>
      <initParam name="remote_login_passwordfile"
value="EXCLUSIVE"/>
      <initParam name="processes" value="300"/>
      <initParam name="control_files"
value="("+DATA/EASTDB/CONTROLFILE/current.288.890129189",
"+FRA/EASTDB/CONTROLFILE/current.261.890129189")" />
      <initParam name="diagnostic_dest" value="/u01/app/oracle"/>
      <initParam name="db_recovery_file_dest" value="+FRA"/>
      <initParam name="audit_trail" value="DB"/>
      <initParam name="memory_target" value="1048576000"/>
      <initParam name="db_block_size" value="8192"/>
      <initParam name="open_cursors" value="300"/>
      <initParam name="db_recovery_file_dest_size"
value="5242880000"/>
    </InitParams>
    <MiscParams>
      <dataVaultEnabled>false</dataVaultEnabled>
      <archiveLogMode>false</archiveLogMode>
    </MiscParams>
  </InitParamAttributes>
  <initParamFileName>{ORACLE_BASE}/admin/{DB_UNIQUE_NAME}/pfile/init.ora</initParamFileName>
  </MiscParams>
  <SPfile>
```

```

useSPFile="true">+DATA/EASTDB/PARAMETERFILE/spfile.293.890129783</SPfile>
</InitParamAttributes>
<StorageAttributes>
    <DataFiles>

        <Location>{ORACLE_HOME}/assistants/dbca/templates/Convert2RAC.dfb</Location>
            <SourceDBName cdb="false">eastdb</SourceDBName>
            <Name id="5" Tablespace="EXAMPLE" Contents="PERMANENT" Size="1243" autoextend="true" blocksize="8192" con_id="0">+DATA/EASTDB/DATAFILE/example.282.890129227</Name>
                <Name id="3" Tablespace="SYSAUX" Contents="PERMANENT" Size="650" autoextend="true" blocksize="8192" con_id="0">+DATA/EASTDB/DATAFILE/sysaux.280.890129055</Name>
                    <Name id="1" Tablespace="SYSTEM" Contents="PERMANENT" Size="790" autoextend="true" blocksize="8192" con_id="0">+DATA/EASTDB/DATAFILE/system.286.890129099</Name>
                        <Name id="4" Tablespace="UNDOTBS1" Contents="UNDO" Size="155" autoextend="true" blocksize="8192" con_id="0">+DATA/EASTDB/DATAFILE/undotbs1.295.890129147</Name>
                            <Name id="6" Tablespace="USERS" Contents="PERMANENT" Size="5" autoextend="true" blocksize="8192" con_id="0">+DATA/EASTDB/DATAFILE/users.296.890129145</Name>
                                </DataFiles>
                                <TempFiles>
                                    <Name id="1" Tablespace="TEMP" Contents="TEMPORARY" Size="197" con_id="0">+DATA/EASTDB/TEMPFILE/temp.299.890129215</Name>
                                </TempFiles>
                                <ControlfileAttributes id="Controlfile">
                                    <maxDatafiles>100</maxDatafiles>
                                    <maxLogfiles>16</maxLogfiles>
                                    <maxLogMembers>3</maxLogMembers>
                                    <maxLogHistory>292</maxLogHistory>
                                    <maxInstances>8</maxInstances>
                                    <image name="current.288.890129189" filepath="+DATA/EASTDB/CONTROLFILE/">
                                        <image name="current.261.890129189" filepath="+FRA/EASTDB/CONTROLFILE/">
                                    </ControlfileAttributes>
                                    <RedoLogGroupAttributes id="1">
                                        <reuse>false</reuse>
                                        <fileSize unit="KB">51200</fileSize>
                                        <Thread>1</Thread>
                                        <member ordinal="0" memberName="group_1.287.890129191" filepath="+DATA/EASTDB/ONLINELOG/">
                                            <member ordinal="0" memberName="group_1.271.890129193" filepath="+FRA/EASTDB/ONLINELOG/">
                                        </RedoLogGroupAttributes>
                                        <RedoLogGroupAttributes id="2">
                                            <reuse>false</reuse>
                                            <fileSize unit="KB">51200</fileSize>
                                            <Thread>1</Thread>
                                            <member ordinal="0" memberName="group_2.298.890129193" filepath="+DATA/EASTDB/ONLINELOG/">
                                                <member ordinal="0" memberName="group_2.270.890129197" filepath="+FRA/EASTDB/ONLINELOG/">
                                            </RedoLogGroupAttributes>

```

```

<RedoLogGroupAttributes id="3">
  <reuse>false</reuse>
  <fileSize unit="KB">51200</fileSize>
  <Thread>1</Thread>
  <member ordinal="0" memberName="group_3.283.890129199"
filepath="+DATA/EASTDB/ONLINELOG/">
  <member ordinal="0" memberName="group_3.269.890129201"
filepath="+FRA/EASTDB/ONLINELOG/">
</RedoLogGroupAttributes>
</StorageAttributes>
</DatabaseTemplate>
[oracle@wnode03 templates] $

```

6. Navigate to /u01/app/oracle/product/12.1.0/dbhome_1/bin and execute DBCA.

```

[oracle@wnode03 templates]$ cd
/u01/app/oracle/product/12.1.0/dbhome_1/bin
[oracle@wnode03 bin]$ ./dbca

```

Step	Screen/Page Description	Choices or Values
1	Database Operations	Select Create Database and click Next .
2	Creation Mode	Select Advanced Mode , and then click Next .
3	Database Template	Select Oracle Real Application Clusters (RAC) database for Database Type. Select Admin-Managed for Configuration Type. Select Convert2RAC in the template list. Click Next .
4	Database Identification	Enter westdb.example.com in the Global Database Name field. Click Next .
5	Database Placement	Select wnode03 and wnode04 . Click Next .
6	Management Options	Deselect Configure Enterprise Manager (EM) Database Express . Click Next .
7	Database Credentials	Select Use same Administrative password for All Accounts . Enter oracle_4U and confirm password. Click Next .
8	Storage Locations	Confirm Storage Type is Automatic Storage Management (ASM) . Confirm Use Common Location for All Database Files in +DATA diskgroup. Deselect Specify Fast Recovery Area . Click Next .
9	Database Options	Click Next .
10	Initialization Parameters	Set Memory Size (SGA and PGA) to 1000 MB . Select Use Automatic Memory Management . Select Character Sets . Select Use Unicode (AL32UTF8) . Click Next .
11	Creation Options	Select Create Database . Click Next .
12	Pre-Requisite Checks	Click Next .

13	Summary	Click Finish .
14	Progress Page	On the Finish page (for password management), click Exit . Click Close .

7. On wnode03, use SRVCTL to verify the converted RAC database and its instances.

```
[oracle@wnode03 bin]$ . oraenv
ORACLE_SID = [oracle]? westdb
The Oracle base has been set to /u01/app/oracle
[oracle@wnode03 ~]$ srvctl config database -db westdb
Database unique name: westdb
Database name: westdb
Oracle home: /u01/app/oracle/product/12.1.0/dbhome_1
Oracle user: oracle
Spfile: +DATA/WESTDB/PARAMETERFILE/spfile.303.890132899
Password file: +DATA/WESTDB/PASSWORD/pwdwestdb.298.890132395
Domain: example.com
Start options: open
Stop options: immediate
Database role: PRIMARY
Management policy: AUTOMATIC
Server pools:
Disk Groups: DATA
Mount point paths:
Services:
Type: RAC
Start concurrency:
Stop concurrency:
OSDBA group: dba
OSOPER group: oper
Database instances: westdb1,westdb2
Configured nodes: wnode03,wnode04
Database is administrator managed
[oracle@wnode03 ~]$
```

8. Exit all of your terminal sessions.

Practice 2-2: Dropping the Existing RAC Database on cluster02

Overview

The RAC database that you have created on `cluster02` is not needed for the following practices. So, you will drop the existing RAC database using DBCA.

Tasks

- From your classroom PC desktop, execute `ssh -X oracle@wnode03` to open a terminal session on `wnode03` as the `oracle` user. Configure the terminal environment as shown below to execute DBCA.

```
[Classroom PC] $ ssh -X oracle@wnode03
oracle@node01's password: <oracle>
[oracle@wnode03 ~]$ . oraenv
ORACLE_SID = [oracle]? westdb
The Oracle base has been set to /u01/app/oracle
[oracle@wnode03 bin]$ dbca
```

Step	Screen/Page Description	Choices or Values
1	Database Operations	Select Delete Database and click Next .
2	Delete Database	Select <code>westdb</code> and click Next .
3	Management Options	Click Next .
4	Summary	Click Finish .
5	Database Configuration Assistant dialog box	You are informed that the instances and datafiles will be deleted. Click Yes to proceed.
6	Progress page	On the Finish page, click Close .

- Exit your terminal session.

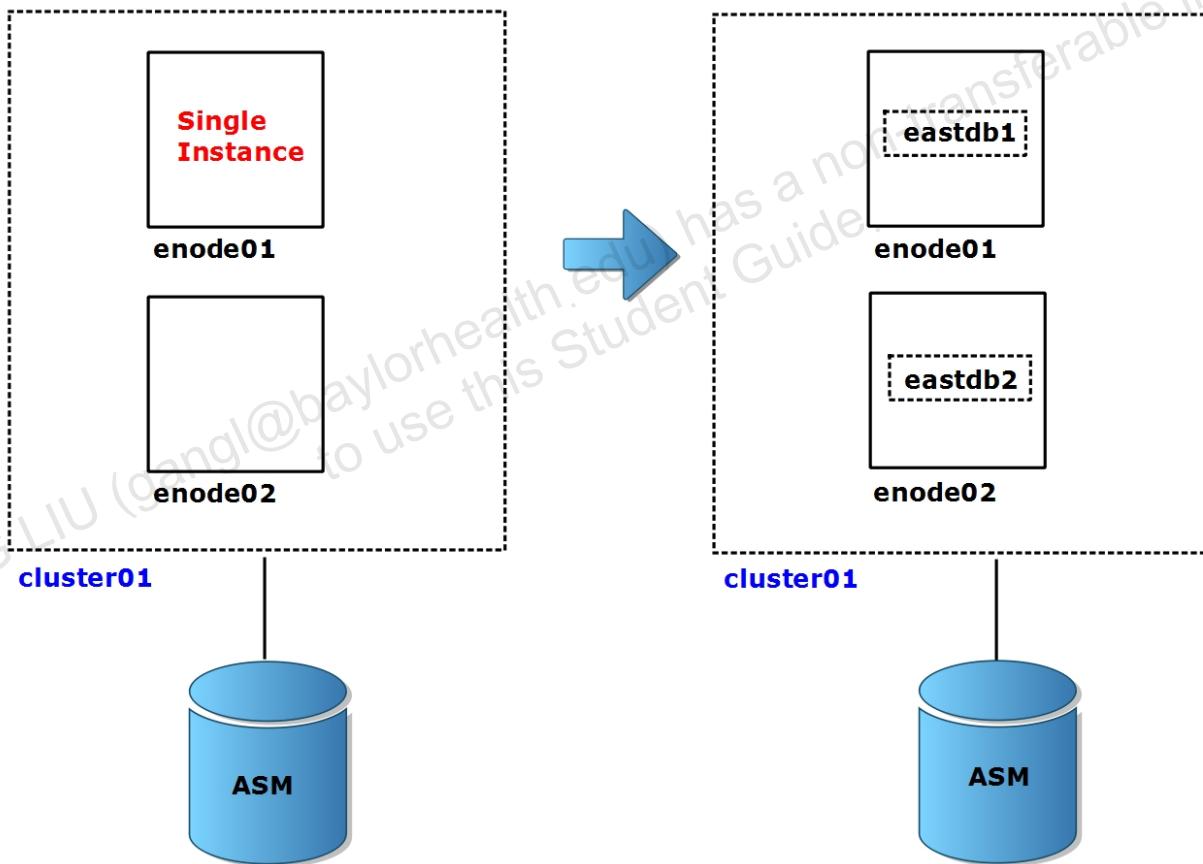
Practice 2-3: Converting a Single Instance to a RAC Database on the Same Cluster Using Rconfig

Overview

In this practice, you will convert a single instance database to RAC on the same cluster using the Rconfig utility.

Assumptions

- A single instance database named eastdb is running on cluster01.
- The database is converted into a two-node RAC database on the same cluster, cluster01.
- The RAC database must be created *without moving the database files* to leverage the same shared storage.



Tasks

1. From your classroom PC desktop, execute `ssh oracle@enode01` to open a terminal session on enode01 as the oracle user.

```
[Classroom PC] $ ssh oracle@enode01
oracle@enode01's password: <oracle>
[oracle@enode01 ~] $
```

2. Make sure that you set up your environment variables correctly for the oracle user.

```
[oracle@enode01 ~]$ . oraenv
ORACLE_SID = [oracle]? eastdb
The Oracle base has been set to /u01/app/oracle
[oracle@enode01 ~]$
```

3. Determine the location of the default listener and configure the default listener in Grid Infrastructure Home.

Note: The rconfig utility requires the default listener to be configured in Grid Infrastructure Home.

```
[oracle@enode01 ~]$ srvctl config listener
Name: LISTENER
Type: Database Listener
Network: 1, Owner: grid
Home: <CRS home>
End points: TCP:1521
Listener is enabled.
Listener is individually enabled on nodes:
Listener is individually disabled on nodes:
[oracle@enode01 ~]$
```

4. Execute `srvctl config database -db eastdb` to verify the database type.

Note: The database type must be SINGLE.

```
[oracle@enode01 ~]$ srvctl config database -db eastdb
Database unique name: eastdb
Database name: eastdb
Oracle home: /u01/app/oracle/product/12.1.0/dbhome_1
Oracle user: oracle
Spfile: +DATA/EASTDB/PARAMETERFILE/spfile.299.890151031
Password file:
Domain: example.com
Start options: open
Stop options: immediate
Database role: PRIMARY
Management policy: AUTOMATIC
Server pools:
Disk Groups: DATA,FRA
Mount point paths:
Services:
Type: SINGLE
OSDBA group: dba
OSOPER group: oper
Database instance: eastdb
Configured nodes: enode01
Database is administrator managed
[oracle@enode01 ~]$
```

5. In your real-life environment, you should create a full database backup of the single instance database before you continue. However, since this practice is for educational purposes, you will focus on the conversion steps. Continue to the next step.

6. In preparation for practice 2-3 and practice 4, set the ARCHIVE_LAG_TARGET parameter to 7200. This is not normally needed but it helps with labs running in the virtual machine architecture.

```
[oracle@enode01 ~]$ sqlplus / as sysdba
SQL> alter system set archive_lag_target=7200 scope=both;
System altered.
SQL> exit
[oracle@enode01 ~]$
```

Note: The ARCHIVE_LAG_TARGET parameter forces a log switch after a specified time interval in seconds. By default, the ARCHIVE_LAG_TARGET parameter is set to 0, indicating that the primary database does not perform a time-based redo switch. The recommended setting for this parameter is 1800 seconds (30 minutes), which informs the primary database that it must switch logfiles every 30 minutes during times of low or no activity.

7. Navigate to the directory \$ORACLE_HOME/assistants/rconfig/sampleXMLs and then create a copy of the ConvertToRAC_AdminManaged.xml file.

Note: If you want to create a Policy Managed database in your real environment, use ConvertToRAC_PolicyManaged.xml instead.

```
[oracle@enode01 ~]$ cd $ORACLE_HOME/assistants/rconfig/sampleXMLs
[oracle@enode01 sampleXMLs]$ cp ConvertToRAC_AdminManaged.xml
ConvertToRAC_AdminManaged.bkp
[oracle@enode01 sampleXMLs]$
```

8. Modify the ConvertToRAC_AdminManaged.xml file as required for your system and save the file.

Note: Because you do not need to move files, leave "TargetDatabaseArea" and "TargetFlashRecoveryArea" empty, so that rconfig will only convert the single instance database to RAC without migrating the locations of database files.

```
[oracle@enode01 sampleXMLs]$ vi ConvertToRAC_AdminManaged.xml
<?xml version="1.0" encoding="UTF-8"?>
<n:RConfig xmlns:n="http://www.oracle.com/rconfig"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="http://www.oracle.com/rconfig rconfig.xsd">
    <n:ConvertToRAC>
        <!-- Verify does a precheck to ensure all pre-requisites are met, before the conversion is attempted. Allowable values are: YES|NO|ONLY -->
        <n:Convert verify="ONLY">
        <!--Specify current OracleHome of non-rac database for SourceDBHome -->

        <n:SourceDBHome>/u01/app/oracle/product/12.1.0/dbhome_1</n:SourceDBHome>
        <!--Specify OracleHome where the rac database should be configured. It can be same as SourceDBHome -->

        <n:TargetDBHome>/u01/app/oracle/product/12.1.0/dbhome_1</n:TargetDBHome>
        <!--Specify SID of non-rac database and credential. User with sysdba role is required to perform conversion -->
        <n:SourceDBInfo SID="eastdb">
            <n:Credentials>
                <n:User>sys</n:User>
                <n:Password>oracle_4U</n:Password>
                <n:Role>sysdba</n:Role>
            </n:Credentials>
        <!--Specify Windows Secure Oracle Home Credentials if the Oracle Home was installed with Secure User option -->
```

```

<!--
    <n:Credentials>
        <n:User>UserDomain\SecureHomeUserName</n:User>
        <n:Password>oracle</n:Password>
        <n:Role>windows_svc</n:Role>
    </n:Credentials>
-->
    </n:SourceDBInfo>
<!--Specify the list of nodes that should have rac instances running for the Admin Managed Cluster Database. LocalNode should be the first node in this nodelist. -->
    <n:NodeList>
        <n:Node name="enode01" />
        <n:Node name="enode02" />
    </n:NodeList>
<!--Specify RacOneNode along with servicename to convert database to RACOne Node -->
    <!--n:RacOneNode servicename="salesrac1service"-->
<!--Instance Prefix tag is optional starting with 11.2. If left empty, it is derived from db_unique_name.-->
    <n:InstancePrefix>eastdb</n:InstancePrefix>
<!-- Listener details are no longer needed starting 11.2. Database is registered with default listener and SCAN listener running from Oracle Grid Infrastructure home. -->
<!--Specify the type of storage to be used by rac database. Allowable values are CFS|ASM. The non-rac database should have same storage type. ASM credentials are no needed for conversion. -->
    <n:SharedStorage type="ASM">
<!--Specify Database Area Location to be configured for rac database.If this field is left empty, current storage will be used for rac database. For CFS, this field will have directory path. -->
    <n:TargetDatabaseArea></n:TargetDatabaseArea>
<!--Specify Fast Recovery Area to be configured for rac database. If this field is left empty, current recovery area of non-rac database will be configured for rac database. If current database is not using recovery Area, the resulting rac database will not have a recovery area. -->
    <n:TargetFlashRecoveryArea></n:TargetFlashRecoveryArea>
        </n:SharedStorage>
    </n:Convert>
    </n:ConvertToRAC>
</n:RConfig>
:wq!
[oracle@enode01 sampleXMLs]$
```

- Run rconfig with `Convert verify="ONLY"` to perform a test conversion, to ensure that a conversion can be completed successfully.

```
[oracle@enode01 sampleXMLs]$ rconfig ConvertToRAC_AdminManaged.xml
<?xml version="1.0" ?>
<RConfig version="1.1" >
<ConvertToRAC>
    <Convert>
        <Response>
            <Result code="0" >
                Operation Succeeded
            </Result>
        </Response>
        <ReturnValue type="object">
            There is no return value for this step </ReturnValue>
    </Convert>
```

```
</ConvertToRAC></RConfig>
[oracle@enode01 sampleXMLs]$
```

10. Modify the ConvertToRAC_AdminManaged.xml file with `Convert verify="YES"`.

```
[oracle@enode01 sampleXMLs]$ vi ConvertToRAC_AdminManaged.xml
<n:Convert verify="YES">
...
<n:SourceDBHome>/u01/app/oracle/product/12.1.0/dbhome_1</n:SourceDBHome>
<n:TargetDBHome>/u01/app/oracle/product/12.1.0/dbhome_1</n:TargetDBHome>
<n:SourceDBInfo SID="eastdb">
<n:Node name="enode01"/>
<n:Node name="enode02"/>
<n:InstancePrefix>eastdb</n:InstancePrefix>
<n:Password>oracle_4U</n:Password>
<n:TargetDatabaseArea></n:TargetDatabaseArea>
<n:TargetFlashRecoveryArea></n:TargetFlashRecoveryArea>
...
:wq!
[oracle@enode01 sampleXMLs]$
```

11. Run rconfig with `Convert verify="YES"` to convert a single instance to a RAC database.

```
[oracle@enode01 sampleXMLs]$ rconfig ConvertToRAC_AdminManaged.xml
Converting Database "eastdb" to Cluster Database. Target Oracle Home:
/u01/app/oracle/product/12.1.0/dbhome_1. Database Role: PRIMARY.
Setting Data Files and Control Files
Adding Database Instances
Adding Redo Logs
Enabling threads for all Database Instances
Setting TEMP tablespace
Adding UNDO tablespaces
Adding Trace files
Setting Fast Recovery Area
Updating Oratab
Creating Password file(s)
Configuring Listeners
Configuring related CRS resources
Starting Cluster Database
<?xml version="1.0" ?>
<RConfig version="1.1" >
<ConvertToRAC>
<Convert>
<Response>
<Result code="0" >
  Operation Succeeded
</Result>
</Response>
<ReturnValue type="object">
<Oracle_Home>
  /u01/app/oracle/product/12.1.0/dbhome_1
</Oracle_Home>
<Database type="ADMIN_MANAGED" >
  <InstanceList>
    <Instance SID="eastdb1" Node="enode01" >
    </Instance>
    <Instance SID="eastdb2" Node="enode02" >
    </Instance>
```

```
</InstanceList>
</Database> </ReturnValue>
</Convert>
</ConvertToRAC></RConfig>
[oracle@enode01 sampleXMLs]$
```

12. Execute `srvctl config database -db eastdb` to verify the database type.

Note: The database type is RAC. The single instance database is successfully converted to a RAC database.

```
[oracle@enode01 sampleXMLs]$ srvctl config database -db eastdb
Database unique name: eastdb
Database name: eastdb
Oracle home: /u01/app/oracle/product/12.1.0/dbhome_1
Oracle user: oracle
Spfile: +DATA/EASTDB/PARAMETERFILE/spfile.299.890151031
Password file: +DATA/eastdb/orapweastdb
Domain: example.com
Start options: open
Stop options: immediate
Database role: PRIMARY
Management policy: AUTOMATIC
Server pools:
Disk Groups: DATA,FRA
Mount point paths:
Services:
Type: RAC
Start concurrency:
Stop concurrency:
OSDBA group: dba
OSOPER group: oper
Database instances: eastdb1,eastdb2
Configured nodes: enode01,enode02
Database is administrator managed
[oracle@enode01 sampleXMLs]$
```

13. Exit all of your terminal sessions.

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

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

Practices for Lesson 3: Configuring Oracle Net Services in a Data Guard Environment with RAC

Chapter 3

Practices for Lesson 3: Overview

Practice Overview

In this practice, you will use graphical utilities to create and modify the Oracle network configuration files. You will manually edit the files with a text editor when needed, and then propagate the resulting files to each server in the Data Guard environment.

Practice Summary

- Practice 3-1: Modifying the `tnsnames.ora` Configuration File on the Primary and Standby Hosts
- Practice 3-2: Understanding Listener Management in an Oracle RAC Environment
- Practice 3-3: Modifying the `listener.ora` Configuration File on the Primary and Standby Hosts

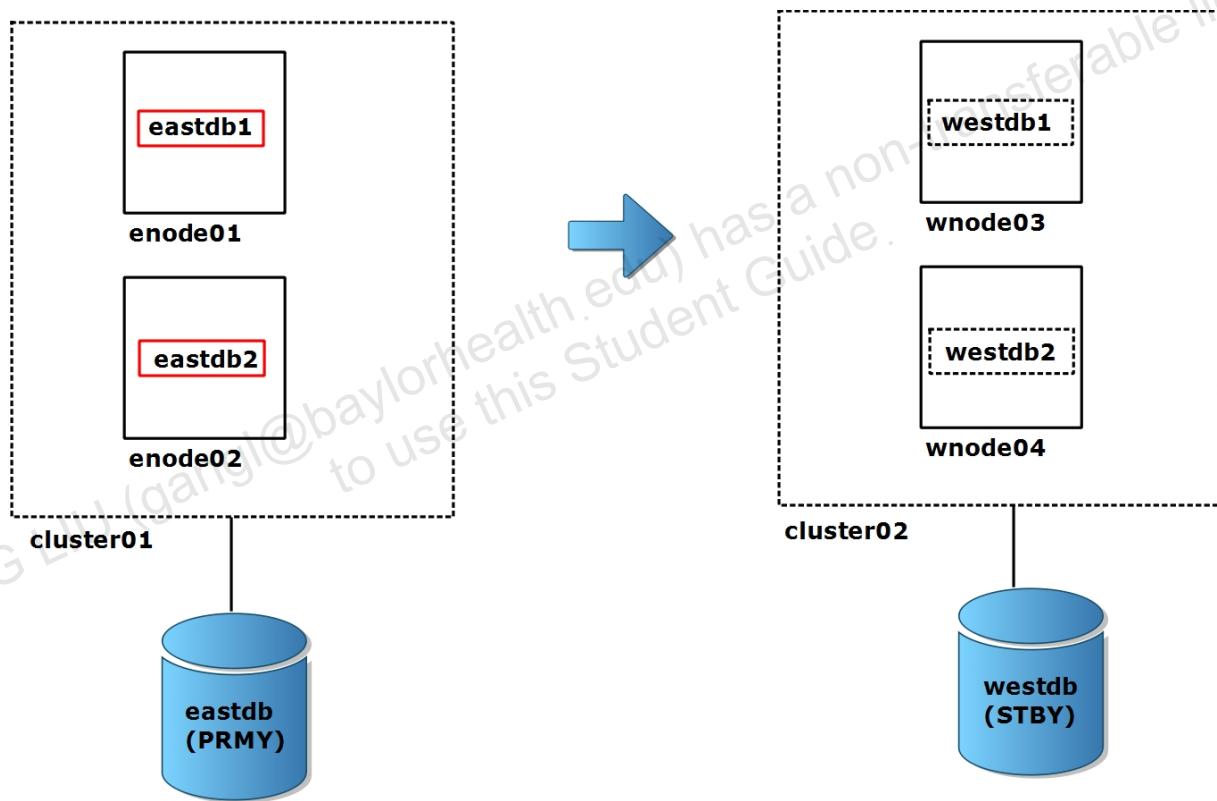
Practice 3-1: Modifying the tnsnames.ora Configuration File on the Primary and Standby Hosts

Overview

In this practice, you will modify the `tnsnames.ora` configuration file using the graphical Oracle Net Manager utility. You will create service name entries for each database instance that will be used in this course.

Assumptions

- The primary database is two-node RAC database.
- You will create service name entries to support the Data Guard creation and operations.
- You will create another two-node RAC database as a standby database later on.



Tasks

- From your classroom PC desktop, execute `ssh -X oracle@enode01` to open a terminal session on `enode01` as the `oracle` user.

```
[Classroom PC] $ ssh -X oracle@enode01
oracle@enode01's password: <oracle>
[oracle@enode01 ~] $
```

2. Make sure that you set up your environment variables correctly for the `oracle` user.

```
[oracle@enode01 ~]$ . oraenv  
ORACLE_SID = [oracle] ? eastdb  
The Oracle base has been set to /u01/app/oracle  
[oracle@enode01 ~]$
```

3. Invoke the Oracle Net Manager utility to prepare the networking files for Data Guard.

```
[oracle@enode01 ~]$ netmgr
```



4. Modify the existing Oracle Net alias to reach the `eastdb` database on the primary nodes.

Note: Make sure that the alias references the scan listener (`cluster01-scan`) and not the node-vip.

- In the navigator pane, expand the **Local** menu tree item.
- Highlight the `eastdb` entry under the **Service Naming** menu tree.
- Replace `enode01.example.com` for the Host Name in the **Address Configuration** section with `cluster01-scan`.
- Leave the port as its default value of **1521**.
- Confirm or enter `eastdb.example.com` as the Service Name.

5. Create an Oracle Net alias to reach the `westdb` database on the standby nodes.

Note: Make sure that the alias references the scan listener (`cluster02-scan`) and not the node-vip.

- Click the green "+" button at the top-left of the form to create a new Service Name entry.
- Enter `westdb` as the Net Service Name and click the **Next** button.
- Highlight **TCP/IP (Internet Protocol)** and click the **Next** button.
- Enter `cluster02-scan` for the Host Name.

- e. Leave the port as its default value of **1521** and click the **Next** button.
 - f. Enter **westdb.example.com** as the Service Name and click the **Next** button.
 - g. Click the **Finish** button.
6. Create a temporary Oracle Net alias named **clone** in the `tnsnames.ora` configuration file. This alias will be used by RMAN to create a physical standby database in RAC.
- Note:** Make sure that the alias references the local listener on `wnode03` where the physical standby instance will be running initially. According to Oracle Support Note (1144273.1), even when executing the RMAN `DUPLICATE` command in the standby host, you will still need the same alias in the primary database.
- a. Click the green "+" button at the top-left of the form to create a new Service Name entry.
 - b. Enter **clone** as the Net Service Name and click the **Next** button.
 - c. Highlight **TCP/IP (Internet Protocol)** and click the **Next** button.
 - d. Enter **wnode03** for the Host Name.
 - e. Leave the port as its default value of **1521** and click the **Next** button.
 - f. Enter **clone** as the Service Name and click the **Next** button.
 - g. Click the **Finish** button.
7. Click **File > Save Network Configuration** from the menu.
8. Do not exit the Oracle Net Manager utility. Open a new terminal window and use the SSH client to connect to `enode01` as the `oracle` OS user. Enter `oracle` when you are prompted for the password.

```
[Classroom PC] $ ssh oracle@enode01
oracle@enode01's password: <oracle>
[oracle@enode01 ~]$
```

9. Make sure that you set up your environment variables correctly for the `oracle` user.

```
[oracle@enode01 ~]$ . oraenv
ORACLE_SID = [oracle]? eastdb
The Oracle base has been set to /u01/app/oracle
[oracle@enode01 ~]$
```

10. Navigate to the `$ORACLE_HOME/network/admin` directory, examine the contents of `tnsnames.ora` network configuration file, and verify the changes made.

```
[oracle@enode01 ~]$ cd $ORACLE_HOME/network/admin
[oracle@enode01 admin]$ cat tnsnames.ora
# tnsnames.ora Network Configuration File:
/u01/app/oracle/product/12.1.0/dbhome_1/network/admin/tnsnames.ora
# Generated by Oracle configuration tools.

### Reference to the scan listener on cluster01 and cluster02

westdb =
(DESCRIPTION =
  (ADDRESS = (PROTOCOL = TCP) (HOST = cluster02-scan) (PORT = 1521))
```

```
(CONNECT_DATA =
  (SERVER = DEDICATED)
  (SERVICE_NAME = westdb.example.com)
)
)

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

### Reference to the local listener on wnode03

CLONE =
(DESCRIPTION =
(ADDRESS_LIST =
(ADDRESS = (PROTOCOL = TCP) (HOST = wnode03) (PORT = 1521))
)
(CONNECT_DATA =
  (SERVER = DEDICATED)
  (SERVICE_NAME = clone)
)
)

[oracle@enode01 admin]$
```

11. Copy the tnsnames.ora file to the remaining hosts (enode02, wnode03, and wnode04) in the Data Guard environment.

```
[oracle@enode01 admin]$ scp
$ORACLE_HOME/network/admin/tnsnames.ora
enode02:/u01/app/oracle/product/12.1.0/dbhome_1/network/admin
tnsnames.ora                                100%   546      0.5KB/s   00:00
[oracle@enode01 admin]$ scp
$ORACLE_HOME/network/admin/tnsnames.ora
wnode03:/u01/app/oracle/product/12.1.0/dbhome_1/network/admin
oracle@wnode03's password: <oracle>
tnsnames.ora                                100%   546      0.5KB/s   00:00
[oracle@enode01 admin]$ scp
$ORACLE_HOME/network/admin/tnsnames.ora
wnode04:/u01/app/oracle/product/12.1.0/dbhome_1/network/admin
oracle@wnode04's password: <oracle>
tnsnames.ora                                100%   546      0.5KB/s   00:00
[oracle@enode01 admin]$
```

12. Exit all of your terminal sessions.

Practice 3-2: Understanding Listener Management in an Oracle RAC Environment

Overview

The node listeners and scan listeners in Oracle Database 12c can be configured at the clusterware level. For example, with srvctl and netca, the configuration is propagated to the listeners accordingly. In this practice, you examine the implementation of listeners in a clustered environment, including the configuration of `listener.ora` and the role of the clusterware process `oraagent`.

Task Summary

- **Task 1:** Working with the default listener
- **Task 2:** Adding `lsnr_test01`
- **Task 3:** Adding `lsnr_test02`
- **Task 4:** Adding `lsnr_test03`

Terminal Sessions

- **Session 1:** `ssh grid@enode01`
- **Session 2:** `ssh oracle@enode01`

Task 1: Working with the default listener

1. **(Session 1)** From your classroom PC desktop, execute `ssh grid@enode01` to open a terminal session on enode01 as the grid user.

```
[Classroom PC] $ ssh grid@enode01
grid@enode01's password: <oracle>
[grid@enode01 ~]$
```

2. Make sure that you set up your environment variables correctly for the grid user.

```
[grid@enode01 ~]$ . oraenv
ORACLE_SID = [grid]? +ASM1
The Oracle base has been set to /u01/app/grid
[grid@enode01 ~]$
```

3. Display the clusterware resources for listeners.

Note: You should see one node listener in each node and three scan listeners in the cluster.

```
[grid@enode01 ~]$ crsctl stat res -t -w "NAME co LISTENER"
```

Name	Target	State	Server	State details
<hr/>				
Local Resources				
ora.INSTANCE.lsnr	ONLINE	ONLINE	enode01	STABLE
	ONLINE	ONLINE	enode02	STABLE

```

Cluster Resources
-----
ora.LISTENER_SCAN1.lsnr
  1 ONLINE ONLINE      enode02    STABLE
ora.LISTENER_SCAN2.lsnr
  1 ONLINE ONLINE      enode01    STABLE
ora.LISTENER_SCAN3.lsnr
  1 ONLINE ONLINE      enode01    STABLE
-----
[grid@enode01 ~]$
```

4. Display the resource profile of the default listener. The resource profile includes the following information:

- The resource (`ora.LISTENER.lsnr`) owner is the `grid` user.
- The resource can be managed by the privileged users based on the `ACL` setting, but it does not mean that all privileged users can start or stop the actual target application or process (`tnslnsr` for the listener).
- When a request comes, an appropriate `oraagent` manages the default listener process.
- The default listener process (`tnslnsr`) runs out of `CRS_HOME`.

```

[grid@enode01 ~]$ crsctl stat res ora.LISTENER.lsnr -p
NAME=ora.LISTENER.lsnr
TYPE=ora.listener.type
ACL=owner:grid:rwx,pgrp:oinstall:rwx,other::r--
ACTIONS=
ACTION_SCRIPT=%CRS_HOME%/bin/racgwrap%CRS_SCRIPT_SUFFIX%
ACTION_TIMEOUT=60
AGENT_FILENAME=%CRS_HOME%/bin/oraagent%CRS_EXE_SUFFIX%
AUTO_START=restore
CHECK_INTERVAL=60
CHECK_TIMEOUT=120
CLEAN_TIMEOUT=60
DEGREE=1
DELETE_TIMEOUT=60
DESCRIPTION=Oracle Listener resource
ENABLED=1
ENDPOINTS=TCP:1521
INSTANCE_FAILOVER=1
INTERMEDIATE_TIMEOUT=0
LOAD=1
LOGGING_LEVEL=1
MODIFY_TIMEOUT=60
NLS_LANG=
OFFLINE_CHECK_INTERVAL=0
ORACLE_HOME=%CRS_HOME%
PORT=1521
RESTART_ATTEMPTS=5
SCRIPT_TIMEOUT=60
SERVER_CATEGORY=ora.hub.category
START_CONCURRENCY=0
```

```
START_DEPENDENCIES=hard(type:ora.cluster_vip_net1.type)
pullup(type:ora.cluster_vip_net1.type)
START_TIMEOUT=180
STOP_CONCURRENCY=0
STOP_DEPENDENCIES=hard(intermediate:type:ora.cluster_vip_net1.type)
STOP_TIMEOUT=0
TYPE_VERSION=1.2
UPTIME_THRESHOLD=1d
USER_WORKLOAD=no
USR_ORA_ENV=
USR_ORA_OPI=false

[grid@enode01 ~]$
```

5. Display the oraagent processes running out of the current cluster.

Note: Oracle Clusterware started three oraagent processes in our practice environment.

- The first oraagent process is spawned by ohasd.
- The other oraagent processes are spawned by crsd.
 - oraagent for grid is responsible for managing the listeners configured in CRS HOME.
 - oraagent for oracle is responsible for managing the listeners configured in RDBMS HOME.

```
[grid@enode01 ~]$ ps -ef|grep oraagent|grep -v grep
grid 2153 1 0 Oct08 ? 01:32:00 /u01/app/12.1.0/grid/bin/oraagent.bin
grid 3834 1 0 Oct08 ? 01:41:53 /u01/app/12.1.0/grid/bin/oraagent.bin
oracle 4876 1 0 Oct08 ? 02:08:29 /u01/app/12.1.0/grid/bin/oraagent.bin
[grid@enode01 ~]$
```

Task 2: Adding lsnr_test01

6. (Session 2) Open a terminal window to connect as the oracle user.

```
[Classroom PC]$ ssh oracle@enode01
oracle@enode01's password: <oracle>
[oracle@enode01 ~]$
```

7. Make sure that you set up your environment variables correctly for the oracle user.

```
[oracle@enode01 ~]$ . oraenv
ORACLE_SID = [oracle]? eastdb
The Oracle base has been set to /u01/app/oracle
[oracle@enode01 ~]$
```

8. Using `srvctl`, add the listener named `lsnr_test01`.

```
[oracle@enode01 ~]$ srvctl add listener -listener lsnr_test01 -  
oraclehome /u01/app/oracle/product/12.1.0/dbhome_1  
[oracle@enode01 ~]$  
[oracle@enode01 ~]$
```

9. **(Session 1)** Return to the terminal session on `enode01` connected as `grid`. Display the clusterware resource profile for `lsnr_test01`.

Note: The listener named `LSNR_TEST01` has been added to run from RDBMS HOME and is managed by the `oraagent` process that runs as `oracle`.

```
[grid@enode01 ~]$ crsctl stat res ora.LSNR_TEST01.lsnr -p  
NAME=ora.LSNR_TEST01.lsnr  
TYPE=ora.listener.type  
ACL=owner:oracle:rwx,pgrp:oinstall:rwx,other::r--  
ACTIONS=  
ACTION_SCRIPT=/u01/app/oracle/product/12.1.0/dbhome_1/bin/racgwra  
p%CRS_SCRIPT_SUFFIX%  
ACTION_TIMEOUT=60  
AGENT_FILENAME=%CRS_HOME%/bin/oraagent%CRS_EXE_SUFFIX%  
AUTO_START=restore  
CHECK_INTERVAL=60  
CHECK_TIMEOUT=120  
CLEAN_TIMEOUT=60  
DEGREE=1  
DELETE_TIMEOUT=60  
DESCRIPTION=Oracle Listener resource  
ENABLED=1  
ENDPOINTS=TCP:1522  
INSTANCE_FAILOVER=1  
INTERMEDIATE_TIMEOUT=0  
LOAD=1  
LOGGING_LEVEL=1  
MODIFY_TIMEOUT=60  
NLS_LANG=  
OFFLINE_CHECK_INTERVAL=0  
ORACLE_HOME=/u01/app/oracle/product/12.1.0/dbhome_1  
PORT=1522  
RESTART_ATTEMPTS=5  
SCRIPT_TIMEOUT=60  
SERVER_CATEGORY=ora.hub.category  
START_CONCURRENCY=0  
START_DEPENDENCIES=hard(type:ora.cluster_vip_net1.type)  
pullup(type:ora.cluster_vip_net1.type)  
START_TIMEOUT=180  
STOP_CONCURRENCY=0  
STOP_DEPENDENCIES=hard(intermediate:type:ora.cluster_vip_net1.type)  
STOP_TIMEOUT=0  
TYPE_VERSION=1.2  
UPTIME_THRESHOLD=1d
```

```
USER_WORKLOAD=no
USR_ORA_ENV=
USR_ORA_OPI=false

[grid@enode01 ~]$
```

10. (**Session 2**) Return to the terminal session on enode01 connected as oracle. As an owner of the listener resource (ora.LSNR_TEST01.lsnr), start the newly added listener (LSNR_TEST01) from RDBMS_HOME.

Note: Since the owner of the listener resource is **oracle**, oraagent for **oracle** started LSNR_TEST01 from **RDBMS_HOME**.

```
[oracle@enode01 ~]$ srvctl start listener -l lsnr_test01
[oracle@enode01 ~]$
```

11. Review listener.ora in RDBMS_HOME.

Note: Oracle Clusterware (oraagent for oracle) added the new entry for the newly added listener in the configuration file. See the line marked with # line added by Agent.

```
[oracle@enode01 ~]$ cd $ORACLE_HOME/network/admin
[oracle@enode01 admin]$ cat listener.ora|grep "by Agent"
LSNR_TEST01=(DESCRIPTION=(ADDRESS_LIST=(ADDRESS=(PROTOCOL=IPC)(KEY=L
SNR_TEST01))))          # line added by Agent
ENABLE_GLOBAL_DYNAMIC_ENDPOINT_LSNR_TEST01=ON           # line added
by Agent
VALID_NODE_CHECKING_REGISTRATION_LSNR_TEST01=SUBNET      #
line added by Agent
[oracle@enode01 admin]$
```

12. Stop LSNR_TEST01 for next testing.

```
[oracle@enode01 admin]$ srvctl stop listener -l lsnr_test01
[oracle@enode01 admin]$
```

13. Remove the LSNR_TEST01 entry from the configuration file manually.

```
[oracle@enode01 admin]$ vi listener.ora

### Remove the following entry ###

LSNR_TEST01=(DESCRIPTION=(ADDRESS_LIST=(ADDRESS=(PROTOCOL=IPC)(KEY=L
SNR_TEST01))))          # line added by Agent

:wq
[oracle@enode01 admin]$
```

14. Restart the listener named LSNR_TEST01.

```
[oracle@enode01 admin]$ srvctl start listener -l lsnr_test01
[oracle@enode01 admin]$
```

15. Check if Oracle clusterware was able to add the removed entry.

Note: oraagent for oracle updated the configuration file automatically.

```
[oracle@enode01 admin]$ cat listener.ora|grep "by Agent"
LSNR_TEST01=(DESCRIPTION=(ADDRESS_LIST=(ADDRESS=(PROTOCOL=IPC)(KEY=L
SNR_TEST01))))                                     # line added by Agent
ENABLE_GLOBAL_DYNAMIC_ENDPOINT_LSNR_TEST01=ON          # line added
by Agent
VALID_NODE_CHECKING_REGISTRATION_LSNR_TEST01=SUBNET      #
line added by Agent
[oracle@enode01 admin]$
```

Task 3: Adding lsnr_test02

16. Using srvctl, add the listener named lsnr_test02 as oracle user.

```
[oracle@enode01 ~]$ srvctl add listener -listener lsnr_test02 -
oraclehome /u01/app/12.1.0/grid
[oracle@enode01 ~]$
```

17. (**Session 1**) Return to the terminal session on enode01 connected as grid. Display the clusterware resource profile for lsnr_test02.

Note: The listener named LSNR_TEST02 has been added to run from CRS HOME and is managed by the oraagent process that runs as oracle.

```
[grid@enode01 admin]$ crsctl stat res ora.LSNR_TEST02.lsnr -p
NAME=ora.LSNR_TEST02.lsnr
TYPE=ora.listener.type
ACL=owner:oracle:rwx,pgrp:oinstall:rwx,other::r--
ACTIONS=
ACTION_SCRIPT=%CRS_HOME%/bin/racgwrap%CRS_SCRIPT_SUFFIX%
ACTION_TIMEOUT=60
AGENT_FILENAME=%CRS_HOME%/bin/oraagent%CRS_EXE_SUFFIX%
AUTO_START=restore
CHECK_INTERVAL=60
CHECK_TIMEOUT=120
CLEAN_TIMEOUT=60
DEGREE=1
DELETE_TIMEOUT=60
DESCRIPTION=Oracle Listener resource
ENABLED=1
ENDPOINTS=TCP:1523
INSTANCE_FAILOVER=1
INTERMEDIATE_TIMEOUT=0
LOAD=1
LOGGING_LEVEL=1
MODIFY_TIMEOUT=60
NLS_LANG=
```

```

OFFLINE_CHECK_INTERVAL=0
ORACLE_HOME=%CRS_HOME%
PORT=1523
RESTART_ATTEMPTS=5
SCRIPT_TIMEOUT=60
SERVER_CATEGORY=ora.hub.category
START_CONCURRENCY=0
START_DEPENDENCIES=hard(type:ora.cluster_vip_net1.type)
pullup(type:ora.cluster_vip_net1.type)
START_TIMEOUT=180
STOP_CONCURRENCY=0
STOP_DEPENDENCIES=hard(intermediate:type:ora.cluster_vip_net1.type)
STOP_TIMEOUT=0
TYPE_VERSION=1.2
UPTIME_THRESHOLD=1d
USER_WORKLOAD=no
USR_ORA_ENV=
USR_ORA_OPI=false

[grid@enode01 admin] $

```

18. **(Session 2)** Return to the terminal session on enode01 connected as oracle. Using srvctl, attempt to start the newly added listener named LSNR_TEST02.

Note: Oracle clusterware failed to start the listener from CRS HOME because of the permission issue.

- As the resource owner (oracle), you attempted to start the listener.
- The resource owner (ora.LSNR_TEST02.lsnr) is oracle.
- Thus, oraagent for oracle attempted to start the LSNR_TEST02 process.
- However, the LSNR_TEST02 process home is CRS HOME (owned by grid).

```

[oracle@enode01 ~]$ srvctl start listener -listener lsnr_test02
PRCR-1079 : Failed to start resource ora.LSNR_TEST02.lsnr
CRS-5010: Update of configuration file
"/u01/app/12.1.0/grid/network/admin/listener.ora.new.enode02"
failed: details at "(:CLSN00014:)" in
"/u01/app/grid/diag/crs/enode02/crs/trace/crsd_oraagent_oracle.trc"
CRS-5010: Update of configuration file
"/u01/app/12.1.0/grid/network/admin/listener.ora.new.enode01"
failed: details at "(:CLSN00014:)" in
"/u01/app/grid/diag/crs/enode01/crs/trace/crsd_oraagent_oracle.trc"
CRS-2674: Start of 'ora.LSNR_TEST02.lsnr' on 'enode02' failed
CRS-2674: Start of 'ora.LSNR_TEST02.lsnr' on 'enode01' failed
[oracle@enode01 ~] $

```

19. (**Session 1**) Return to the terminal session on enode01 connected as grid user to attempt to start the listener.

Note: The command failed because of the permission issue.

- As the owner of the listener home, which is grid, you attempted to start the listener named LSNR_TEST02 that has been configured in CRS_HOME.
- However, the resource owner (ora.LSNR_TEST02.lsnr) is oracle.
- Thus, oraagent for oracle attempted to start the LSNR_TEST02 process.

```
[grid@enode01 ~]$ srvctl start listener -listener lsnr_test02
PRCR-1079 : Failed to start resource ora.LSNR_TEST02.lsnr
CRS-5010: Update of configuration file
"/u01/app/12.1.0/grid/network/admin/listener.ora.new.enode01"
failed: details at "(:CLSN00014:)" in
"/u01/app/grid/diag/crs/enode01/crs/trace/crsd_oraagent_oracle.trc"
CRS-5010: Update of configuration file
"/u01/app/12.1.0/grid/network/admin/listener.ora.new.enode02"
failed: details at "(:CLSN00014:)" in
"/u01/app/grid/diag/crs/enode02/crs/trace/crsd_oraagent_oracle.trc"
CRS-2674: Start of 'ora.LSNR_TEST02.lsnr' on 'enode02' failed
CRS-2674: Start of 'ora.LSNR_TEST02.lsnr' on 'enode01' failed
[grid@enode01 ~]$
```

Task 4: Adding lsnr_test03

20. Using srvctl, add the listener named lsnr_test03 as grid user.

```
[grid@enode01 ~]$ srvctl add listener -l lsnr_test03 -oraclehome
/u01/app/oracle/product/12.1.0/dbhome_1
[grid@enode01 ~]$
```

21. Display the clusterware resource profile for lsnr_test03.

Note: The listener named LSNR_TEST03 has been added to run from RDBMS_HOME and is managed by the oraagent process that runs as grid.

```
[grid@enode01 ~]$ crsctl stat res ora.LSNR_TEST03.lsnr -p
NAME=ora.LSNR_TEST03.lsnr
TYPE=ora.listener.type
ACL=owner:grid:rwx,pgrp:oinstall:rwx,other::r--
ACTIONS=
ACTION_SCRIPT=/u01/app/oracle/product/12.1.0/dbhome_1/bin/racgwra
p%CRS_SCRIPT_SUFFIX%
ACTION_TIMEOUT=60
AGENT_FILENAME=%CRS_HOME%/bin/oraagent%CRS_EXE_SUFFIX%
AUTO_START=restore
CHECK_INTERVAL=60
CHECK_TIMEOUT=120
CLEAN_TIMEOUT=60
DEGREE=1
```

```

DELETE_TIMEOUT=60
DESCRIPTION=Oracle Listener resource
ENABLED=1
ENDPOINTS=TCP:1523
INSTANCE_FAILOVER=1
INTERMEDIATE_TIMEOUT=0
LOAD=1
LOGGING_LEVEL=1
MODIFY_TIMEOUT=60
NLS_LANG=
OFFLINE_CHECK_INTERVAL=0
ORACLE_HOME=/u01/app/oracle/product/12.1.0/dbhome_1
PORT=1523
RESTART_ATTEMPTS=5
SCRIPT_TIMEOUT=60
SERVER_CATEGORY=ora.hub.category
START_CONCURRENCY=0
START_DEPENDENCIES=hard(type:ora.cluster_vip_net1.type)
pullup(type:ora.cluster_vip_net1.type)
START_TIMEOUT=180
STOP_CONCURRENCY=0
STOP_DEPENDENCIES=hard(intermediate:type:ora.cluster_vip_net1.type)
STOP_TIMEOUT=0
TYPE_VERSION=1.2
UPTIME_THRESHOLD=1d
USER_WORKLOAD=no
USR_ORA_ENV=
USR_ORA_OPI=false
[grid@enode01 ~]$

```

22. As the resource owner, attempt to start the listener named lsnr_test03.

Note: The command failed because of the permission issue.

- The resource owner (ora.LSNR_TEST03.lsnr) is grid.
- As the resource owner (grid), you attempted to start the listener that has been configured in RDBMS HOME.
- Even though you attempted to start the listener as the resource owner (grid), oraagent for grid had to start the LSNR_TEST03 process from RDBMS HOME.

```

[grid@enode01 ~]$ srvctl start listener -l lsnr_test03
PRCR-1079 : Failed to start resource ora.LSNR_TEST03.lsnr
CRS-5010: Update of configuration file
"/u01/app/oracle/product/12.1.0/dbhome_1/network/admin/listener.or
ra.new.enode02" failed: details at "(:CLSN00014:)" in
"/u01/app/grid/diag/crs/enode02/crs/trace/crsd_oraagent_grid.trc"
CRS-5010: Update of configuration file
"/u01/app/oracle/product/12.1.0/dbhome_1/network/admin/listener.or
ra.new.enode01" failed: details at "(:CLSN00014:)" in
"/u01/app/grid/diag/crs/enode01/crs/trace/crsd_oraagent_grid.trc"
CRS-2674: Start of 'ora.LSNR_TEST03.lsnr' on 'enode02' failed
CRS-2674: Start of 'ora.LSNR_TEST03.lsnr' on 'enode01' failed
[grid@enode01 ~]$

```

23. **(Session 2)** Return to the terminal session on enode01 connected as oracle to start the listener named lsnr_test03.

Note: The command failed because of the permission issue.

- The resource owner (ora.LSNR_TEST03.lsnr) is grid.
- As the owner of the listener home, which is grid, you attempted to start the listener.
- However, oraagent for grid had to start the LSNR_TEST03 process from RDBMS HOME.

```
[oracle@enode01 ~]$ srvctl start listener -l lsnr_test03
PRCR-1079 : Failed to start resource ora.LSNR_TEST03.lsnr
CRS-5010: Update of configuration file
"/u01/app/oracle/product/12.1.0/dbhome_1/network/admin/listener.o
ra.new.enode02" failed: details at "(:CLSN00014:)" in
"/u01/app/grid/diag/crs/enode02/crs/trace/crsd_oraagent_grid.trc"
CRS-5010: Update of configuration file
"/u01/app/oracle/product/12.1.0/dbhome_1/network/admin/listener.o
ra.new.enode01" failed: details at "(:CLSN00014:)" in
"/u01/app/grid/diag/crs/enode01/crs/trace/crsd_oraagent_grid.trc"
CRS-2674: Start of 'ora.LSNR_TEST03.lsnr' on 'enode02' failed
CRS-2674: Start of 'ora.LSNR_TEST03.lsnr' on 'enode01' failed
[oracle@enode01 ~]$
```

24. Remove all three listeners.

```
[oracle@enode01 ~]$ su -
Password: <oracle>
[root@enode01 ~]# . oraenv
ORACLE_SID = [root] ? +ASM1
The Oracle base has been set to /u01/app/grid
[root@enode01 ~]# srvctl remove listener -l lsnr_test01
PRCR-1025 : Resource ora.LSNR_TEST01.lsnr is still running
[root@enode01 ~]# srvctl stop listener -l lsnr_test01
[root@enode01 ~]# srvctl remove listener -l lsnr_test01
[root@enode01 ~]# srvctl remove listener -l lsnr_test02
[root@enode01 ~]# srvctl remove listener -l lsnr_test03
[root@enode01 ~]#
```

25. Exit all of your terminal sessions.

You learned that you have to add a LISTENER as an appropriate OS account, who can start the listener process from the LISTENER home directory, to manage at the Oracle clusterware level.

For example, if you want to add a listener that runs out of CRS_HOME, you should log in as an owner of CRS_HOME to make the owner of CRS_HOME the owner of the listener clusterware resource. In this way, oraagent for grid can start a listener from CRS_HOME with the proper permission. Otherwise, you have to specify the owner of the resource explicitly when adding a listener.

Practice 3-3: Modifying the listener.ora Configuration File on the Primary and Standby Hosts

Overview

In this practice, you will modify the `listener.ora` configuration files on each server in the Oracle Data Guard environment by using the graphical Oracle Net Manager utility. You will create static entries for the `eastdb` and `westdb` service names. A static entry is required for RMAN to make a connection after shutting down the instance during physical standby database creation. You will also create entries that are required later for Data Guard Broker operations.

Tasks on enode01

- From your classroom PC desktop, execute `ssh -X grid@enode01` to open a terminal session on `enode01` as the `grid` user.

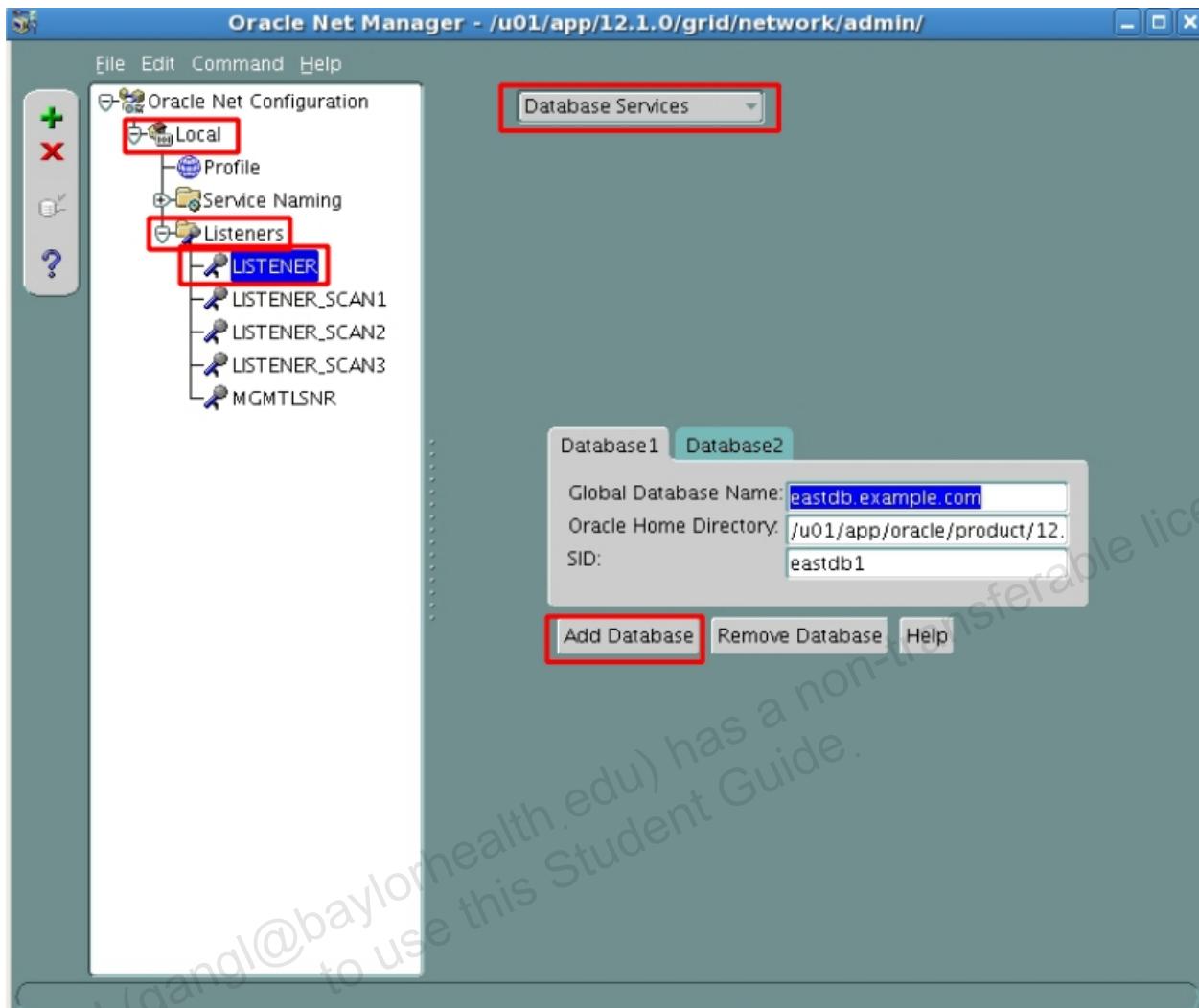
```
[Classroom PC] $ ssh -X grid@enode01
grid@enode01's password: <oracle>
[grid@enode01 ~]$
```

- Make sure that you set up your environment variables correctly for the `grid` user.

```
[grid@enode01 ~]$ . oraenv
ORACLE_SID = [grid]? +ASM1
The Oracle base has been set to /u01/app/grid
[grid@enode01 ~]$
```

- Invoke the Oracle Net Manager utility to prepare the networking files for Data Guard.

```
[grid@enode01 ~]$ netmgr
```



4. Create a static listener entry for enode01.
 - a. In the navigator pane, expand the **Local** menu tree item.
 - b. Expand the **Listeners** menu tree item.
 - c. Highlight the **LISTENER** entry.
 - d. Under the Listening locations drop-down menu, select the **Database Services** entry. Currently one service has been explicitly configured for this listener for Oracle Data Guard Broker.
Note: According to Oracle Support Note 1387859.1, Static "_DGMGRL" entries are no longer needed as of Oracle Database 12.1.0.2 in Oracle Data Guard Broker configurations that are managed by Oracle Restart, RAC One Node or RAC as the Broker will use the clusterware to restart an instance.
 - e. Click the **Add Database** button. Enter **eastdb.example.com** for the Global Database Name, **/u01/app/oracle/product/12.1.0/dbhome_1** for the Oracle Home Directory, and **eastdb1** for the SID. The Oracle Home Directory should default to the correct location.

5. Click **File > Save Network Configuration** from the menu.

6. Do not exit the Oracle Net Manager utility. Open a new terminal window and use the SSH client to connect to enode01 as the grid OS user. Enter grid when you are prompted for the password.

```
[Classroom PC]$ ssh grid@enode01
grid@enode01's password: <oracle>
[grid@enode01 ~]$
```

7. Make sure that you set up your environment variables correctly for the grid user.

```
[grid@enode01 ~]$ . oraenv
ORACLE_SID = [grid] ? +ASM1
The Oracle base has been set to /u01/app/grid
[grid@enode01 ~]$
```

8. Navigate to the \$ORACLE_HOME/network/admin directory, examine the contents of the listener.ora network configuration file, and verify the changes made.

```
[grid@enode01 ~]$ cd $ORACLE_HOME/network/admin
[grid@enode01 admin]$ cat listener.ora
# listener.ora Network Configuration File:
/u01/app/12.1.0/grid/network/admin/listener.ora
# Generated by Oracle configuration tools.

ENABLE_GLOBAL_DYNAMIC_ENDPOINT_LISTENER_SCAN3 = ON

ENABLE_GLOBAL_DYNAMIC_ENDPOINT_LISTENER_SCAN2 = ON

ENABLE_GLOBAL_DYNAMIC_ENDPOINT_LISTENER_SCAN1 = ON

VALID_NODE_CHECKING_REGISTRATION_MGMTLSNR = SUBNET

VALID_NODE_CHECKING_REGISTRATION_LISTENER_SCAN3 = OFF

VALID_NODE_CHECKING_REGISTRATION_LISTENER_SCAN2 = OFF

SID_LIST_LISTENER =
  (SID_LIST =
    (SID_DESC =
      (GLOBAL_DBNAME = eastdb_DGMGRL.example.com)
      (ORACLE_HOME = /u01/app/oracle/product/12.1.0/dbhome_1)
      (SID_NAME = eastdb1)
    )
    (SID_DESC =
      (GLOBAL_DBNAME = eastdb.example.com)
      (ORACLE_HOME = /u01/app/oracle/product/12.1.0/dbhome_1)
      (SID_NAME = eastdb1)
    )
  )
```

```
VALID_NODE_CHECKING_REGISTRATION_LISTENER_SCAN1 = OFF  
  
VALID_NODE_CHECKING_REGISTRATION_LISTENER = SUBNET  
  
MGMTLSNR =  
(DESCRIPTION =  
  (ADDRESS = (PROTOCOL = IPC) (KEY = MGMTLSNR))  
)  
  
ADR_BASE_MGMTLSNR = /u01/app/grid  
  
ENABLE_GLOBAL_DYNAMIC_ENDPOINT_MGMTLSNR = ON  
  
LISTENER =  
(DESCRIPTION =  
  (ADDRESS = (PROTOCOL = IPC) (KEY = LISTENER))  
)  
  
ADR_BASE_LISTENER = /u01/app/grid  
  
ENABLE_GLOBAL_DYNAMIC_ENDPOINT_LISTENER = ON  
  
LISTENER_SCAN3 =  
(DESCRIPTION =  
  (ADDRESS = (PROTOCOL = IPC) (KEY = LISTENER_SCAN3))  
)  
  
LISTENER_SCAN2 =  
(DESCRIPTION =  
  (ADDRESS = (PROTOCOL = IPC) (KEY = LISTENER_SCAN2))  
)  
  
ADR_BASE_LISTENER_SCAN3 = /u01/app/grid  
  
LISTENER_SCAN1 =  
(DESCRIPTION =  
  (ADDRESS = (PROTOCOL = IPC) (KEY = LISTENER_SCAN1))  
)  
  
ADR_BASE_LISTENER_SCAN2 = /u01/app/grid  
  
ADR_BASE_LISTENER_SCAN1 = /u01/app/grid  
  
[grid@enode01 admin]$
```

9. If the entries appear correct, then exit the Oracle Net Manager utility. Restart the listener to load the modified configuration.

Note: If you cannot start the listener, double check the information you entered in the previous steps, including ORACLE_HOME.

```
[grid@enode01 admin]$ srvctl stop listener -node enode01
[grid@enode01 admin]$ srvctl start listener -node enode01
[grid@enode01 admin]$ lsnrctl status

LSNRCTL for Linux: Version 12.1.0.2.0 - Production on 02-OCT-2015
08:20:42

Copyright (c) 1991, 2014, Oracle. All rights reserved.

Connecting to
(DESCRIPTION=(ADDRESS=(PROTOCOL=IPC)(KEY=LISTENER)))
STATUS of the LISTENER
-----
Alias                      LISTENER
Version                    TNSLSNR for Linux: Version 12.1.0.2.0 -
Production
Start Date                02-OCT-2015 08:19:43
Uptime                     0 days 0 hr. 0 min. 59 sec
Trace Level               off
Security                  ON: Local OS Authentication
SNMP                       OFF
Listener Parameter File   /u01/app/12.1.0/grid/network/admin/listener.ora
Listener Log File         /u01/app/grid/diag/tnslsnr/enode01/listener/alert/log.xml
Listening Endpoints Summary...
(DESCRIPTION=(ADDRESS=(PROTOCOL=ipc)(KEY=LISTENER)))

(DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=192.0.2.111)(PORT=1521))
)

(DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=192.0.2.161)(PORT=1521))
)

(DESCRIPTION=(ADDRESS=(PROTOCOL=tcps)(HOST=enode01.example.com)(PORT=5500))(Security=(my_wallet_directory=/u01/app/grid/product/12.1.0/dbhome_1/admin/eastdb/xdb_wallet))(Presentation=HTTP)(Session=RAW))
Services Summary...
Service "+ASM" has 1 instance(s).
  Instance "+ASM1", status READY, has 1 handler(s) for this service...
Service "eastdb.example.com" has 2 instance(s).
  Instance "eastdb1", status UNKNOWN, has 1 handler(s) for this service...
  Instance "eastdb1", status READY, has 1 handler(s) for this service...
Service "eastdbXDB.example.com" has 1 instance(s).
```

```

Instance "eastdb1", status READY, has 1 handler(s) for this
service...
Service "eastdb_DGMGRL.example.com" has 1 instance(s).
  Instance "eastdb1", status UNKNOWN, has 1 handler(s) for this
service...
Service "gdscat.example.com" has 1 instance(s).
  Instance "gdscat", status READY, has 1 handler(s) for this
service...
Service "gdscatXDB.example.com" has 1 instance(s).
  Instance "gdscat", status READY, has 1 handler(s) for this
service...
The command completed successfully

[grid@enode01 admin]$

```

10. Exit all of your terminal sessions.

Tasks on enode02

11. Open a new terminal window and use the SSH client to connect to enode02 as the grid user.

```

[Classroom PC] $ ssh -X grid@enode02
grid@enode02's password: <oracle>
[grid@enode02 ~]$

```

12. Make sure that you set up your environment variables correctly for the grid user.

```

[grid@enode02 ~]$ . oraenv
ORACLE_SID = [grid] ? +ASM2
The Oracle base has been set to /u01/app/grid
[grid@enode02 ~]$

```

13. Invoke the Oracle Net Manager utility to prepare the networking files for Data Guard.

```
[grid@enode02 ~]$ netmgr
```

14. Create a static listener entry for enode02.

- In the navigator pane, expand the **Local** menu tree item.
- Expand the **Listeners** menu tree item.
- Highlight the **LISTENER** entry.
- Under the Listening locations drop-down menu, select the **Database Services** entry. Currently, one service has been explicitly configured for this listener for Oracle Data Guard Broker.

Note: According to Oracle Support Note 1387859.1, Static "_DGMGRL" entries are no longer needed as of Oracle Database 12.1.0.2 in Oracle Data Guard Broker configurations that are managed by Oracle Restart, RAC One Node or RAC as the Broker will use the clusterware to restart an instance.

- Click the **Add Database** button. Enter **eastdb.example.com** for the Global Database Name, **/u01/app/oracle/product/12.1.0/dbhome_1** for the

Oracle Home Directory, and **eastdb2** for the SID. The Oracle Home Directory should default to the correct location.

15. Click **File > Save Network Configuration** from the menu.
16. Do not exit the Oracle Net Manager utility. Open a new terminal window and use the SSH client to connect to enode02 as the grid OS user. Enter grid when you are prompted for the password.

```
[Classroom PC]$ ssh grid@enode02
grid@enode02's password: <oracle>
[grid@enode02 ~]$
```

17. Make sure that you set up your environment variables correctly for the grid user.

```
[grid@enode02 ~]$ . oraenv
ORACLE_SID = [grid] ? +ASM2
The Oracle base has been set to /u01/app/grid
[grid@enode02 ~]$
```

18. Navigate to the \$ORACLE_HOME/network/admin directory, examine the contents of the listener.ora network configuration file, and verify the changes made.

```
[grid@enode02 ~]$ cd $ORACLE_HOME/network/admin
[grid@enode02 admin]$ cat listener.ora
# listener.ora Network Configuration File:
/u01/app/12.1.0/grid/network/admin/listener.ora
# Generated by Oracle configuration tools.

ENABLE_GLOBAL_DYNAMIC_ENDPOINT_LISTENER_SCAN3 = ON

ENABLE_GLOBAL_DYNAMIC_ENDPOINT_LISTENER_SCAN2 = ON

ENABLE_GLOBAL_DYNAMIC_ENDPOINT_LISTENER_SCAN1 = ON

VALID_NODE_CHECKING_REGISTRATION_MGMTLSNR = SUBNET

VALID_NODE_CHECKING_REGISTRATION_LISTENER_SCAN3 = OFF

VALID_NODE_CHECKING_REGISTRATION_LISTENER_SCAN2 = OFF

SID_LIST_LISTENER =
  (SID_LIST =
    (SID_DESC =
      (GLOBAL_DBNAME = eastdb_DGMGRL.example.com)
      (ORACLE_HOME = /u01/app/oracle/product/12.1.0/dbhome_1)
      (SID_NAME = eastdb2)
    )
    (SID_DESC =
```

```
(GLOBAL_DBNAME = eastdb.example.com)
(ORACLE_HOME = /u01/app/oracle/product/12.1.0/dbhome_1)
(SID_NAME = eastdb2)
)
)

VALID_NODE_CHECKING_REGISTRATION_LISTENER_SCAN1 = OFF

VALID_NODE_CHECKING_REGISTRATION_LISTENER = SUBNET

MGMTLSNR =
(DESCRIPTION =
(ADDRESS = (PROTOCOL = IPC) (KEY = MGMTLSNR))
)

ADR_BASE_MGMTLSNR = /u01/app/grid

ENABLE_GLOBAL_DYNAMIC_ENDPOINT_MGMTLSNR = ON

LISTENER =
(DESCRIPTION =
(ADDRESS = (PROTOCOL = IPC) (KEY = LISTENER))
)

ADR_BASE_LISTENER = /u01/app/grid

ENABLE_GLOBAL_DYNAMIC_ENDPOINT_LISTENER = ON

LISTENER_SCAN3 =
(DESCRIPTION =
(ADDRESS = (PROTOCOL = IPC) (KEY = LISTENER_SCAN3))
)

LISTENER_SCAN2 =
(DESCRIPTION =
(ADDRESS = (PROTOCOL = IPC) (KEY = LISTENER_SCAN2))
)

ADR_BASE_LISTENER_SCAN3 = /u01/app/grid

LISTENER_SCAN1 =
(DESCRIPTION =
(ADDRESS = (PROTOCOL = IPC) (KEY = LISTENER_SCAN1))
)

ADR_BASE_LISTENER_SCAN2 = /u01/app/grid
```

```
ADR_BASE_LISTENER_SCAN1 = /u01/app/grid

[grid@enode02 admin]$
```

19. If the entries appear correct, then exit the Oracle Net Manager utility. Restart the listener to load the modified configuration.

Note: If you cannot start the listener, double check the information you entered in the previous steps, including ORACLE_HOME.

```
[grid@enode02 admin]$ srvctl stop listener -node enode02
[grid@enode02 admin]$ srvctl start listener -node enode02
[grid@enode02 admin]$ lsnrctl status

LSNRCTL for Linux: Version 12.1.0.2.0 - Production on 02-OCT-2015
08:23:14

Copyright (c) 1991, 2014, Oracle. All rights reserved.

Connecting to
(DESCRIPTION=(ADDRESS=(PROTOCOL=IPC) (KEY=LISTENER)))
STATUS of the LISTENER
-----
Alias                      LISTENER
Version                    TNSLSNR for Linux: Version 12.1.0.2.0 -
Production
Start Date                02-OCT-2015 08:22:11
Uptime                     0 days 0 hr. 1 min. 3 sec
Trace Level               off
Security                   ON: Local OS Authentication
SNMP                       OFF
Listener Parameter File   /u01/app/12.1.0/grid/network/admin/listener.ora
Listener Log File         /u01/app/grid/diag/tnslsnr/enode02/listener/alert/log.xml
Listening Endpoints Summary...
  (DESCRIPTION=(ADDRESS=(PROTOCOL=ipc) (KEY=LISTENER)))

  (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp) (HOST=192.0.2.112) (PORT=1521)
))

  (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp) (HOST=192.0.2.162) (PORT=1521)
))

  (DESCRIPTION=(ADDRESS=(PROTOCOL=tcps) (HOST=enode02.example.com) (PORT=5500) (Security=(my_wallet_directory=/u01/app/grid/product/12.1.0/dbhome_1/admin/eastdb/xdb_wallet)) (Presentation=HTTP) (Session=RAW)))
Services Summary...
Service "+ASM" has 1 instance(s).
  Instance "+ASM2", status READY, has 1 handler(s) for this
service...
```

```

Service "eastdb.example.com" has 2 instance(s).
  Instance "eastdb2", status UNKNOWN, has 1 handler(s) for this
service...
  Instance "eastdb2", status READY, has 1 handler(s) for this
service...
Service "eastdbXDB.example.com" has 1 instance(s).
  Instance "eastdb2", status READY, has 1 handler(s) for this
service...
Service "eastdb_DGMGRL.example.com" has 1 instance(s).
  Instance "eastdb2", status UNKNOWN, has 1 handler(s) for this
service...
The command completed successfully
[grid@enode02 admin]$

```

20. Exit all of your terminal sessions.

Tasks on wnode03

21. Open a new terminal window to connect to wnode03 as the grid user.

```

[Classroom PC] $ ssh -X grid@wnode03
grid@wnode03's password: <oracle>
[grid@wnode01 ~]$

```

22. Make sure that you set up your environment variables correctly for the grid user.

```

[grid@wnode03 ~] $ . oraenv
ORACLE_SID = [grid]? +ASM1
The Oracle base has been set to /u01/app/grid
[grid@wnode03 ~] $

```

23. Invoke the Oracle Net Manager utility to prepare the networking files for Data Guard.

```
[grid@wnode03 ~] $ netmgr
```

24. Create static listener entries for wnode3.

- In the navigator pane, expand the **Local** menu tree item.
- Expand the **Listeners** menu tree item.
- Highlight the **LISTENER** entry.
- Under the Listening locations drop-down menu, select the **Database Services** entry. Currently one service has been explicitly configured for this listener for Oracle Data Guard Broker.

Note: According to Oracle Support Note 1387859.1, Static "_DGMGRL" entries are no longer needed as of Oracle Database 12.1.0.2 in Oracle Data Guard Broker configurations that are managed by Oracle Restart, RAC One Node or RAC as the Broker will use the clusterware to restart an instance.

- Click the **Add Database** button. Enter **westdb.example.com** for the Global Database Name, **/u01/app/oracle/product/12.1.0/dbhome_1** for the Oracle Home Directory, and **westdb1** for the SID. The Oracle Home Directory should default to the correct location.

25. Click **File > Save Network Configuration** from the menu.
26. Do not exit the Oracle Net Manager utility. Open a new terminal window and use the SSH client to connect to `wnode03` as the `grid` OS user. Enter `grid` when you are prompted for the password.

```
[Classroom PC]$ ssh grid@wnode03
grid@wnode03's password: <oracle>
[grid@wnode03 ~]$
```

27. Make sure that you set up your environment variables correctly for the `grid` user.

```
[grid@wnode03 ~]$ . oraenv
ORACLE_SID = [grid]? +ASM1
The Oracle base has been set to /u01/app/grid
[grid@wnode03 ~]$
```

28. Navigate to the `$ORACLE_HOME/network/admin` directory, examine the contents of the `listener.ora` network configuration file, and verify the changes made.

```
[grid@enode03 ~]$ cd $ORACLE_HOME/network/admin
[grid@wnode03 admin]$ cat listener.ora
# listener.ora Network Configuration File:
/u01/app/12.1.0/grid/network/admin/listener.ora
# Generated by Oracle configuration tools.

ENABLE_DYNAMIC_ENDPOINT_LISTENER_SCAN3 = ON

ENABLE_DYNAMIC_ENDPOINT_LISTENER_SCAN2 = ON

ENABLE_DYNAMIC_ENDPOINT_LISTENER_SCAN1 = ON

VALID_NODE_CHECKING_REGISTRATION_MGMTLSNR = SUBNET

VALID_NODE_CHECKING_REGISTRATION_LISTENER_SCAN3 = OFF

VALID_NODE_CHECKING_REGISTRATION_LISTENER_SCAN2 = OFF

SID_LIST_LISTENER =
  (SID_LIST =
    (SID_DESC =
      (GLOBAL_DBNAME = westdb_DGMGRL.example.com)
      (ORACLE_HOME = /u01/app/oracle/product/12.1.0/dbhome_1)
      (SID_NAME = westdb1)
    )
    (SID_DESC =
      (GLOBAL_DBNAME = westdb.example.com)
      (ORACLE_HOME = /u01/app/oracle/product/12.1.0/dbhome_1)
      (SID_NAME = westdb1)
    )
  )
```

```
)  
)  
  
VALID_NODE_CHECKING_REGISTRATION_LISTENER_SCAN1 = OFF  
  
VALID_NODE_CHECKING_REGISTRATION_LISTENER = SUBNET  
  
MGMTLSNR =  
(DESCRIPTION =  
(ADDRESS = (PROTOCOL = IPC) (KEY = MGMTLSNR))  
)  
  
ADR_BASE_MGMTLSNR = /u01/app/grid  
  
ENABLE_GLOBAL_DYNAMIC_ENDPOINT_MGMTLSNR = ON  
  
LISTENER =  
(DESCRIPTION =  
(ADDRESS = (PROTOCOL = IPC) (KEY = LISTENER))  
)  
  
ADR_BASE_LISTENER = /u01/app/grid  
  
ENABLE_GLOBAL_DYNAMIC_ENDPOINT_LISTENER = ON  
  
LISTENER_SCAN3 =  
(DESCRIPTION =  
(ADDRESS = (PROTOCOL = IPC) (KEY = LISTENER_SCAN3))  
)  
  
LISTENER_SCAN2 =  
(DESCRIPTION =  
(ADDRESS = (PROTOCOL = IPC) (KEY = LISTENER_SCAN2))  
)  
  
ADR_BASE_LISTENER_SCAN3 = /u01/app/grid  
  
LISTENER_SCAN1 =  
(DESCRIPTION =  
(ADDRESS = (PROTOCOL = IPC) (KEY = LISTENER_SCAN1))  
)  
  
ADR_BASE_LISTENER_SCAN2 = /u01/app/grid  
  
ADR_BASE_LISTENER_SCAN1 = /u01/app/grid  
[grid@wnode03 admin]$
```

29. If the entries appear correct, then copy the configuration file. Restart the listener to load the modified configuration.

Note: If you cannot start the listener, double check the information you entered in the previous steps, including ORACLE_HOME.

```
[grid@wnode03 admin]$ srvctl stop listener -node wnode03
[grid@wnode03 admin]$ srvctl start listener -node wnode03
[grid@wnode03 admin]$ lsnrctl status
LSNRCTL for Linux: Version 12.1.0.2.0 - Production on 02-OCT-2015
08:26:58

Copyright (c) 1991, 2014, Oracle. All rights reserved.

Connecting to
(DESCRIPTION=(ADDRESS=(PROTOCOL=IPC) (KEY=LISTENER)))
STATUS of the LISTENER
-----
Alias                      LISTENER
Version                    TNSLSNR for Linux: Version 12.1.0.2.0 -
Production
Start Date                 02-OCT-2015 08:25:45
Uptime                     0 days 0 hr. 1 min. 13 sec
Trace Level                off
Security                   ON: Local OS Authentication
SNMP                       OFF
Listener Parameter File    /u01/app/12.1.0/grid/network/admin/listener.ora
Listener Log File          /u01/app/grid/diag/tnslsnr/wnode03/listener/alert/log.xml
Listening Endpoints Summary...
  (DESCRIPTION=(ADDRESS=(PROTOCOL=ipc) (KEY=LISTENER)))

  (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp) (HOST=192.0.2.121) (PORT=1521)
))

  (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp) (HOST=192.0.2.171) (PORT=1521)
))
Services Summary...
Service "+ASM" has 1 instance(s).
  Instance "+ASM1", status READY, has 1 handler(s) for this
service...
Service "westdb.example.com" has 1 instance(s).
  Instance "westdb1", status UNKNOWN, has 1 handler(s) for this
service...
Service "westdb_DGMGRL.example.com" has 1 instance(s).
  Instance "westdb1", status UNKNOWN, has 1 handler(s) for this
service...
The command completed successfully

[grid@wnode03 admin]$
```

Tasks on wnode04

30. Open a new terminal window to connect to wnode04 as the grid user.

```
[Classroom PC] $ ssh -X grid@wnode04  
grid@wnode04's password: <oracle>  
[grid@wnode04 ~] $
```

31. Make sure that you set up your environment variables correctly for the grid user.

```
[grid@wnode04 ~] $ . oraenv  
ORACLE_SID = [grid] ? +ASM2  
The Oracle base has been set to /u01/app/grid  
[grid@wnode04 ~] $
```

32. Invoke the Oracle Net Manager utility to prepare the networking files for Data Guard.

```
[grid@wnode04 ~] $ netmgr
```

33. Create a static listener entry for wnode04.

- In the navigator pane, expand the **Local** menu tree item.
- Expand the **Listeners** menu tree item.
- Highlight the **LISTENER** entry.
- Under the Listening locations drop-down menu, select the **Database Services** entry. Currently one service has been explicitly configured for this listener for Oracle Data Guard Broker.

Note: According to Oracle Support Note 1387859.1, Static "_DGMGR" entries are no longer needed as of Oracle Database 12.1.0.2 in Oracle Data Guard Broker configurations that are managed by Oracle Restart, RAC One Node or RAC as the Broker will use the clusterware to restart an instance.

- Click the **Add Database** button. Enter **westdb.example.com** for the Global Database Name, **/u01/app/oracle/product/12.1.0/dbhome_1** for the Oracle Home Directory, and **westdb2** for the SID. The Oracle Home Directory should default to the correct location.

34. Click **File > Save Network Configuration** from the menu.

35. Do not exit the Oracle Net Manager utility. Open a new terminal window and use the SSH client to connect to wnode04 as the grid OS user. Enter **grid** when you are prompted for the password.

```
[Classroom PC] $ ssh grid@wnode04  
grid@wnode04's password: <oracle>  
[grid@wnode04 ~] $
```

36. Make sure that you set up your environment variables correctly for the grid user.

```
[grid@wnode04 ~] $ . oraenv  
ORACLE_SID = [grid] ? +ASM2  
The Oracle base has been set to /u01/app/grid  
[grid@wnode04 ~] $
```

37. Navigate to the \$ORACLE_HOME/network/admin directory and examine the contents of the listener.ora network configuration file.

```
[grid@wnode04 ~]$ cd $ORACLE_HOME/network/admin
[grid@wnode04 admin]$ cat listener.ora
# listener.ora Network Configuration File:
/u01/app/12.1.0/grid/network/admin/listener.ora
# Generated by Oracle configuration tools.

ENABLE_GLOBAL_DYNAMIC_ENDPOINT_LISTENER_SCAN3 = ON

ENABLE_GLOBAL_DYNAMIC_ENDPOINT_LISTENER_SCAN2 = ON

ENABLE_GLOBAL_DYNAMIC_ENDPOINT_LISTENER_SCAN1 = ON

VALID_NODE_CHECKING_REGISTRATION_MGMTLSNR = SUBNET

VALID_NODE_CHECKING_REGISTRATION_LISTENER_SCAN3 = OFF

VALID_NODE_CHECKING_REGISTRATION_LISTENER_SCAN2 = OFF

SID_LIST_LISTENER =
  (SID_LIST =
    (SID_DESC =
      (GLOBAL_DBNAME = westdb_DGMGRL.example.com)
      (ORACLE_HOME = /u01/app/oracle/product/12.1.0/dbhome_1)
      (SID_NAME = westdb2)
    )
    (SID_DESC =
      (GLOBAL_DBNAME = westdb.example.com)
      (ORACLE_HOME = /u01/app/oracle/product/12.1.0/dbhome_1)
      (SID_NAME = westdb2)
    )
  )
)

VALID_NODE_CHECKING_REGISTRATION_LISTENER_SCAN1 = OFF

VALID_NODE_CHECKING_REGISTRATION_LISTENER = SUBNET

MGMTLSNR =
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL = IPC) (KEY = MGMTLSNR) )
  )

ADR_BASE_MGMTLSNR = /u01/app/grid

ENABLE_GLOBAL_DYNAMIC_ENDPOINT_MGMTLSNR = ON
```

```

LISTENER =
(DESCRIPTION =
  (ADDRESS = (PROTOCOL = IPC) (KEY = LISTENER))
)

ADR_BASE_LISTENER = /u01/app/grid

ENABLE_GLOBAL_DYNAMIC_ENDPOINT_LISTENER = ON

LISTENER_SCAN3 =
(DESCRIPTION =
  (ADDRESS = (PROTOCOL = IPC) (KEY = LISTENER_SCAN3))
)

LISTENER_SCAN2 =
(DESCRIPTION =
  (ADDRESS = (PROTOCOL = IPC) (KEY = LISTENER_SCAN2))
)

ADR_BASE_LISTENER_SCAN3 = /u01/app/grid

LISTENER_SCAN1 =
(DESCRIPTION =
  (ADDRESS = (PROTOCOL = IPC) (KEY = LISTENER_SCAN1))
)

ADR_BASE_LISTENER_SCAN2 = /u01/app/grid

ADR_BASE_LISTENER_SCAN1 = /u01/app/grid
[grid@wnode04 admin]$

```

38. If the entries appear correct, then copy the configuration file. Restart the listener to load the modified configuration.

Note: If you cannot start the listener, double check the information you entered in the previous steps, including ORACLE_HOME.

```

[grid@wnode04 admin]$ srvctl stop listener -node wnode04
[grid@wnode04 admin]$ srvctl start listener -node wnode04
[grid@wnode04 admin]$ lsnrctl status

LSNRCTL for Linux: Version 12.1.0.2.0 - Production on 02-OCT-2015
08:31:24

Copyright (c) 1991, 2014, Oracle. All rights reserved.

Connecting to
(DESCRIPTION=(ADDRESS=(PROTOCOL=IPC) (KEY=LISTENER)))

```

```
STATUS of the LISTENER
-----
Alias                      LISTENER
Version                   TNSLSNR for Linux: Version 12.1.0.2.0 -
Production
Start Date                02-OCT-2015 08:29:36
Uptime                     0 days 0 hr. 1 min. 47 sec
Trace Level               off
Security                  ON: Local OS Authentication
SNMP                      OFF
Listener Parameter File   /u01/app/12.1.0/grid/network/admin/listener.ora
Listener Log File         /u01/app/grid/diag/tnslnsr/wnode04/listener/alert/log.xml
Listening Endpoints Summary...
  (DESCRIPTION=(ADDRESS=(PROTOCOL=ipc) (KEY=LISTENER)))

  (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp) (HOST=192.0.2.122) (PORT=1521)
))

  (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp) (HOST=192.0.2.172) (PORT=1521)
))
Services Summary...
Service "+ASM" has 1 instance(s).
  Instance "+ASM2", status READY, has 1 handler(s) for this
service...
Service "westdb.example.com" has 1 instance(s).
  Instance "westdb2", status UNKNOWN, has 1 handler(s) for this
service...
Service "westdb_DGMGRL.example.com" has 1 instance(s).
  Instance "westdb2", status UNKNOWN, has 1 handler(s) for this
service...
The command completed successfully
[grid@wnode04 admin]$
```

39. Exit all of your terminal sessions.

Practices for Lesson 4: Deploying a Physical Standby Database in an Oracle RAC environment by Using SQL and RMAN Commands

Chapter 4

Practices for Lesson 4: Overview

Practice Overview

In this practice, you create a physical standby database in an Oracle RAC environment by using SQL and Oracle Recovery Manager (RMAN). You perform several tasks to create a 2-node RAC physical standby database named `westdb` in the cluster `cluster02` using the 2-node RAC primary database named `eastdb` in the cluster `cluster01`.

Practice Summary

- Practice 4-1: Preparing the Primary Database to Create a Physical Standby Database in RAC
- Practice 4-2: Preparing the Standby Hosts and Creating a Physical Standby Database Using RMAN
- Practice 4-3: Completing a Physical Standby Database for RAC
- Practice 4-4: Verifying Redo Transport and Redo Apply Operations in RAC

Practice 4-1: Preparing the Primary Database to Create a Physical Standby Database in RAC

Overview

In this practice, you will verify that the primary database in RAC is configured correctly to support a physical standby database in RAC.

Assumptions

- The primary database is utilizing ASM for data file storage.
- The standby hosts already have the existing Oracle Grid Infrastructure and Database software installed.
- The standby database storage will also utilize ASM.
- The primary and standby databases will be running on systems where the diskgroup names are the same.

All of the examples illustrated in this document use the following naming:

	Primary	Standby
Hosts	enode01, enode02	wnode03, wnode04
Database Unique Name	eastdb	westdb
Instance Names	eastdb1, eastdb2	westdb1, westdb2

Tasks

- From your classroom PC desktop, execute `ssh oracle@enode01` to open a terminal session on enode01 as the `oracle` user.

```
[Classroom PC] $ ssh oracle@enode01
oracle@enode01's password: <oracle>
[oracle@enode01 ~] $
```

- Make sure that you set up your environment variables correctly for the `oracle` user.

```
[oracle@enode01 ~] $ . oraenv
ORACLE_SID = [oracle]? eastdb
The Oracle base has been set to /u01/app/oracle
[oracle@enode01 ~] $ export ORACLE_SID=eastdb1
[oracle@enode01 ~] $
```

- Invoke SQL*Plus and connect as SYSDBA to your primary database (eastdb) to determine the current ARCHIVELOG mode. If it is the NOARCHIVELOG mode, configure the primary database in the ARCHIVELOG mode.

```
[oracle@enode01 ~] $ sqlplus / as sysdba
SQL> archive log list
Database log mode          No Archive Mode
Automatic archival        Disabled
Archive destination        USE_DB_RECOVERY_FILE_DEST
Oldest online log sequence 12
Current log sequence       14
SQL> exit
[oracle@enode01 ~] $ srvctl stop database -db eastdb
```

```
[oracle@enode01 ~]$ srvctl start instance -db eastdb -instance eastdb1
-startoption mount
[oracle@enode01 ~]$ sqlplus / as sysdba

SQL*Plus: Release 12.1.0.2.0 Production on Sat Sep 12 15:05:12 2015

Copyright (c) 1982, 2014, Oracle. All rights reserved.

Connected to:
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

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 13
Next log sequence to archive 15
Current log sequence       15
SQL> exit
[oracle@enode01 ~]$ srvctl stop database -db eastdb
[oracle@enode01 ~]$ srvctl start database -db eastdb
[oracle@enode01 ~]$
```

4. Invoke SQL*Plus and connect as SYSDBA to check if FORCE LOGGING is enabled. If it is not enabled, enable FORCE LOGGING mode.

```
[oracle@enode01 ~]$ sqlplus / as sysdba

SQL> SELECT force_logging FROM v$database;

FORCE_LOGGING
-----
NO

SQL> ALTER DATABASE FORCE LOGGING;
Database altered.

SQL> SELECT force_logging FROM v$database;

FORCE_LOGGING
-----
YES

SQL>
```

5. Determine the number of online redo log groups and their current size in each thread. Create standby redo log groups with one member for each group using the same size as the existing online redo logs.

Note: Oracle recommends having one more additional group than the number you have for online redo log groups in each thread. Verify the creation of the standby redo logs.

```
SQL> select thread#, group#, bytes/1024/1024 MB from v$log;
-----  
 THREAD#    GROUP#      MB  
-----  
      1          1      50  
      1          2      50  
      2          3      50  
      2          4      50  
  
SQL> alter database add standby logfile thread 1  
group 5 size 50M,  
group 6 size 50M,  
group 7 size 50M;  
Database altered.  
  
SQL> alter database add standby logfile thread 2  
group 8 size 50M,  
group 9 size 50M,  
group 10 size 50M;  
Database altered.  
  
SQL> select thread#, group#, bytes/1024/1024 MB from v$standby_log;
-----  
 THREAD#   GROUP#      MB  
-----  
      1          5      50  
      1          6      50  
      1          7      50  
      2          8      50  
      2          9      50  
      2         10      50  
  
6 rows selected.  
  
SQL>
```

6. Define the first log archive destination to use the fast recovery area and enable it. Ensure that the changes are done both in memory and also stored in the server parameter file. This location should be valid for any role and also valid for all types of log files.

```
SQL> alter system set  
log_archive_dest_1='location=USE_DB_RECOVERY_FILE_DEST  
valid_for=(ALL_LOGFILES,ALL_ROLES) db_unique_name=eastdb' scope=both  
sid='*';  
System altered.  
  
SQL> alter system set log_archive_dest_state_1='enable' scope=both  
sid='*';  
System altered.  
SQL>
```

7. Define the second log archive destination to start redo transport by defining LOG_ARCHIVE_DEST_2 pointing to the physical standby database.

Note: Setting LOG_ARCHIVE_DEST_STATE_2 to DEFER will avoid periodic errors (ORA-12514 and TNS-12564) in the primary instance alert log.

```
SQL> alter system set log_archive_dest_2='SERVICE=westdb ASYNC
valid_for=(ONLINE_LOGFILES,PRIMARY_ROLE) db_unique_name=westdb'
scope=both sid='*';
System altered.

SQL> alter system set log_archive_dest_state_2='DEFER' scope=both
sid='*';
System altered.
SQL>
```

8. Increase the maximum number of archive processes to 4.

Note: The documentation suggests that this parameter should be set to a value of 10. We are using a reduced number in this lab environment to reduce overhead.

```
SQL> alter system set log_archive_max_processes=4 scope=both sid='*';
System altered.
SQL>
```

9. Define the log_archive_config parameter to include entries for eastdb and westdb.

```
SQL> alter system set log_archive_config='dg_config=(eastdb,westdb)'
scope=both sid='*';
System altered.
SQL>
```

10. Enable automatic standby file management so that operating system file additions and deletions on the primary database are replicated to the standby database. This is normally set on the standby database. For the primary database, this is set for role reversals.

```
SQL> alter system set standby_file_management='auto' scope=both
sid='*';
System altered.
SQL>
```

11. Modify the FAL_SERVER parameter to support role reversals.

```
SQL> alter system set fal_server='westdb' scope=both sid='*';
System altered.
SQL> exit
[oracle@enode01 ~] $
```

12. Create an initial entry in the /etc/oratab file to facilitate setting environment variables.

Note: The following entries may be removed by the clusterware agent process eventually, but these entries are added to simplify the next few practices.

```
[oracle@enode01 ~]$ echo  
eastdb1:/u01/app/oracle/product/12.1.0/dbhome_1:N >> /etc/oratab  
[oracle@enode01 ~]$ ssh enode02  
[oracle@enode02 ~]$ echo  
eastdb2:/u01/app/oracle/product/12.1.0/dbhome_1:N >> /etc/oratab  
[oracle@enode02 ~]$ exit  
[oracle@enode01 ~]$
```

13. Exit all of your terminal sessions.

Practice 4-2: Preparing the Standby Hosts and Creating a Physical Standby Database Using RMAN

Overview

In this practice, you will prepare the standby hosts (wnode03 and wnode04) to receive the physical standby database and create the physical standby database using RMAN.

Terminal Sessions

- **Session 1:** ssh oracle@wnode03
- **Session 2:** ssh grid@enode01
- **Session 3:** ssh -X grid@wnode03
- **Session 4:** ssh grid@wnode03

Tasks

1. **(Session 1)** Use a terminal window to log in as oracle to wnode03 and wnode04 to create an initial entry in the /etc/oratab file to facilitate setting environment variables.

Note: The following entries may be removed by the clusterware agent process eventually, but these entries are added to simplify the next few practices.

```
[Classroom PC] $ ssh oracle@wnode03
oracle@wnode03's password: <oracle>
[oracle@wnode03 ~]$ echo
westdb1:/u01/app/oracle/product/12.1.0/dbhome_1:N >> /etc/oratab
[oracle@wnode03 ~]$ ssh wnode04
[oracle@wnode04 ~]$ echo
westdb2:/u01/app/oracle/product/12.1.0/dbhome_1:N >> /etc/oratab
[oracle@wnode04 ~]$ exit
logout
Connection to wnode04 closed.
[oracle@wnode03 ~]$
```

2. Create the audit directory on the standby hosts (wnode03 and wnode04), which is required for a physical standby database.

Note: If the primary database is using a multi-tenant architecture, additional directories are needed.

```
[oracle@wnode03 ~]$ mkdir -p /u01/app/oracle/admin/westdb/adump
[oracle@wnode03 ~]$ ssh wnode04
[oracle@wnode04 ~]$ mkdir -p /u01/app/oracle/admin/westdb/adump
[oracle@wnode04 ~]$ exit
logout
Connection to wnode04 closed.
[oracle@wnode03 ~]$
```

3. **(Session 2)** Open a new terminal to connect to enode01 as the grid OS user. Set up your environment variables correctly for the grid user.

```
[Classroom PC]$ ssh grid@enode01
grid@enode01's password: <oracle>
[grid@enode01 ~]$ . oraenv
ORACLE_SID = [grid]? +ASM1
The Oracle base has been set to /u01/app/grid
[grid@enode01 ~]$
```

4. Using asmcmd, copy the password file to the /tmp directory in the primary database.

```
[grid@enode01 ~]$ srvctl config database -db eastdb | grep Password
Password file: +DATA/eastdb/orapweastdb
[grid@enode01 ~]$ asmcmd pwcOPY +DATA/eastdb/orapweastdb
/tmp/orapweastdb
copying +DATA/eastdb/orapweastdb -> /tmp/orapweastdb
[grid@enode01 ~]$ ls -l /tmp/orapw*
-rw-r----- 1 grid oinstall 10240 Sep 12 14:02 /tmp/orapweastdb
[grid@enode01 ~]$
```

5. **(Session 1)** Return to the first terminal on wnode03 to copy the password file from the primary host to the physical standby hosts (wnode03 and wnode04). Rename the file accordingly.

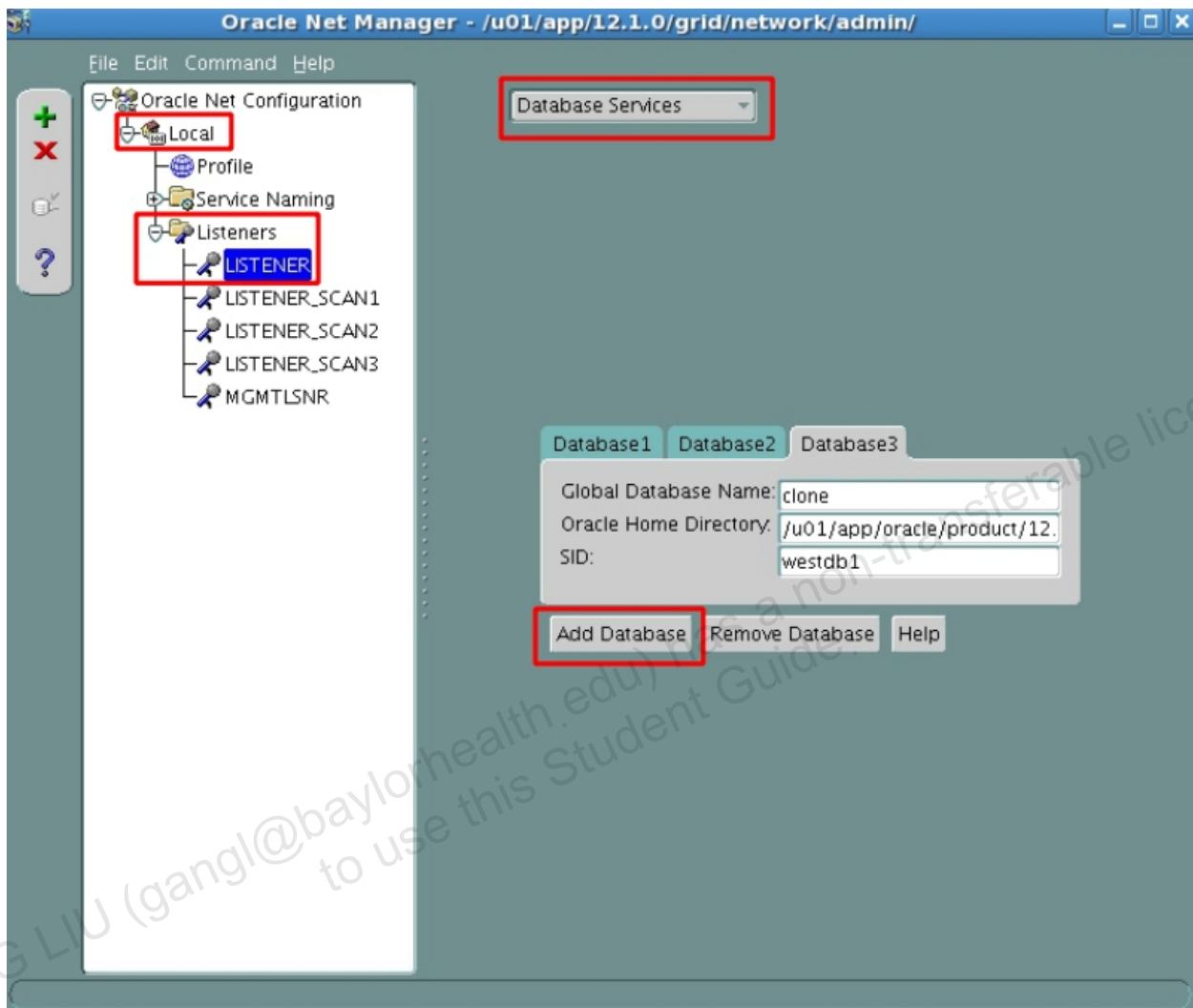
```
[oracle@wnode03 ~]$ scp enode01:/tmp/orapweastdb
wnode03:/u01/app/oracle/product/12.1.0/dbhome_1/dbs/orapwestdb1
oracle@enode01's password: <oracle>
oracle@wnode03's password: <oracle>
orapweastdb                                100%    10KB   10.0KB/s   00:00
Connection to enode01 closed.
[oracle@wnode03 ~]$ 
[oracle@wnode03 ~]$ scp enode01:/tmp/orapweastdb
wnode04:/u01/app/oracle/product/12.1.0/dbhome_1/dbs/orapwestdb2
oracle@enode01's password: <oracle>
oracle@wnode04's password: <oracle>
orapweastdb                                100%    10KB   10.0KB/s   00:00
Connection to enode01 closed.
[oracle@wnode03 ~]$
```

6. **(Session 3)** Open a new terminal on wnode03 as the grid user. Then add a static listener entry named clone in the listener.ora configuration file by using Oracle Net Manager. This will be used by RMAN to support the physical standby database creation.

```
[Classroom PC]$ ssh -X grid@wnode03
grid@wnode03's password: <oracle>
[grid@wnode03 ~]$ . oraenv
ORACLE_SID = [grid]? +ASM1
The Oracle base has been set to /u01/app/grid
[grid@wnode03 ~]$
```

7. Invoke the Oracle Net Manager utility to prepare the networking files for Data Guard.

```
[grid@wnode03 ~]$ netmgr
```



8. Create a temporary static listener entry on wnode3.

- In the navigator pane, expand the **Local** menu tree item.
- Expand the **Listeners** menu tree item.
- Highlight the **LISTENER** entry.
- Under the Listening locations drop-down menu, select the **Database Services** entry.
- Click the **Add Database** button. Enter **clone** for the Global Database Name, **/u01/app/oracle/product/12.1.0/dbhome_1** for the Oracle Home Directory, and **westdb1** for the SID. The Oracle Home Directory should default to the correct location.

9. Click **File > Save Network Configuration** from the menu.

10. (**Session 4**) Do not exit the Oracle Net Manager utility. Open a new terminal window and use the SSH client to connect to `wnode03` as the grid OS user. Enter `oracle` when you are prompted for the password.

```
[Classroom PC]$ ssh grid@wnode03
grid@wnode03's password: <oracle>
[grid@wnode03 ~]$
```

11. Make sure that you set up your environment variables correctly for the grid user.

```
[grid@wnode03 ~]$ . oraenv
ORACLE_SID = [grid] ? +ASM1
The Oracle base has been set to /u01/app/grid
[grid@wnode03 ~]$
```

12. Navigate to the `$ORACLE_HOME/network/admin` directory, examine the contents of the `listener.ora` network configuration file, and verify the changes made.

```
[grid@wnode03 ~]$ cd $ORACLE_HOME/network/admin
[grid@wnode03 admin]$ cat listener.ora
# listener.ora Network Configuration File:
/u01/app/12.1.0/grid/network/admin/listener.ora
# Generated by Oracle configuration tools.

ENABLE_GLOBAL_DYNAMIC_ENDPOINT_LISTENER_SCAN3 = ON
ENABLE_GLOBAL_DYNAMIC_ENDPOINT_LISTENER_SCAN2 = ON
ENABLE_GLOBAL_DYNAMIC_ENDPOINT_LISTENER_SCAN1 = ON
VALID_NODE_CHECKING_REGISTRATION_MGMTLSNR = SUBNET
VALID_NODE_CHECKING_REGISTRATION_LISTENER_SCAN3 = OFF
VALID_NODE_CHECKING_REGISTRATION_LISTENER_SCAN2 = OFF
VALID_NODE_CHECKING_REGISTRATION_LISTENER_SCAN1 = OFF

SID_LIST_LISTENER =
(SID_LIST =
(SID_DESC =
(GLOBAL_DBNAME = westdb_DGMGRL.example.com)
(ORACLE_HOME = /u01/app/oracle/product/12.1.0/dbhome_1)
(SID_NAME = westdb1)
)
(SID_DESC =
(GLOBAL_DBNAME = westdb.example.com)
(ORACLE_HOME = /u01/app/oracle/product/12.1.0/dbhome_1)
(SID_NAME = westdb1)
)
(SID_DESC =
(GLOBAL_DBNAME = clone)
(ORACLE_HOME = /u01/app/oracle/product/12.1.0/dbhome_1)
(SID_NAME = westdb1)
)
)
```

```

VALID_NODE_CHECKING_REGISTRATION_LISTENER = SUBNET

MGMTLSNR =
(DESCRIPTION =
(ADDRESS = (PROTOCOL = IPC) (KEY = MGMTLSNR))
)

ADR_BASE_MGMTLSNR = /u01/app/grid

ENABLE_GLOBAL_DYNAMIC_ENDPOINT_MGMTLSNR = ON

LISTENER =
(DESCRIPTION =
(ADDRESS = (PROTOCOL = IPC) (KEY = LISTENER))
)

ADR_BASE_LISTENER = /u01/app/grid

ENABLE_GLOBAL_DYNAMIC_ENDPOINT_LISTENER = ON

LISTENER_SCAN3 =
(DESCRIPTION =
(ADDRESS = (PROTOCOL = IPC) (KEY = LISTENER_SCAN3))
)

LISTENER_SCAN2 =
(DESCRIPTION =
(ADDRESS = (PROTOCOL = IPC) (KEY = LISTENER_SCAN2))
)

ADR_BASE_LISTENER_SCAN3 = /u01/app/grid

LISTENER_SCAN1 =
(DESCRIPTION =
(ADDRESS = (PROTOCOL = IPC) (KEY = LISTENER_SCAN1))
)

ADR_BASE_LISTENER_SCAN2 = /u01/app/grid
ADR_BASE_LISTENER_SCAN1 = /u01/app/grid

[grid@wnode03 admin]$

```

13. If the entries appear correct, then reload the modified configuration of the default LISTENER.

```

[grid@wnode03 admin]$ srvctl stop listener -node wnode03
[grid@wnode03 admin]$ srvctl start listener -node wnode03
[grid@wnode03 admin]$ lsnrctl status

LSNRCTL for Linux: Version 12.1.0.2.0 - Production on 02-OCT-2015
08:44:20

Copyright (c) 1991, 2014, Oracle. All rights reserved.

Connecting to (DESCRIPTION=(ADDRESS=(PROTOCOL=IPC) (KEY=LISTENER)))
STATUS of the LISTENER

```

Copyright © 2016, Oracle and/or its affiliates. All rights reserved.

```

-----
Alias                      LISTENER
Version                   TNSLSNR for Linux: Version 12.1.0.2.0 -
Production
Start Date                02-OCT-2015 08:43:22
Uptime                     0 days 0 hr. 0 min. 58 sec
Trace Level               off
Security                  ON: Local OS Authentication
SNMP                      OFF
Listener Parameter File   /u01/app/12.1.0/grid/network/admin/listener.ora
Listener Log File         /u01/app/grid/diag/tnslsnr/wnode03/listener/alert/log.xml
Listening Endpoints Summary...
  (DESCRIPTION=(ADDRESS=(PROTOCOL=ipc) (KEY=LISTENER)))
  (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp) (HOST=192.0.2.121) (PORT=1521)))
  (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp) (HOST=192.0.2.171) (PORT=1521)))
Services Summary...
Service "+ASM" has 1 instance(s).
  Instance "+ASM1", status READY, has 1 handler(s) for this service...
Service "clone" has 1 instance(s).
  Instance "westdb1", status UNKNOWN, has 1 handler(s) for this
service...
Service "westdb.example.com" has 1 instance(s).
  Instance "westdb1", status UNKNOWN, has 1 handler(s) for this
service...
Service "westdb_DGMGRL.example.com" has 1 instance(s).
  Instance "westdb1", status UNKNOWN, has 1 handler(s) for this
service...
The command completed successfully
[grid@wnode03 admin]$

```

14. **(Session 1)** Return to the first terminal session on wnode03. Set up your environment variables correctly for the oracle user.

Note: We have set ORACLE_SID to westdb in the previous step. Please reset it to **westdb1** for the following steps.

```

[oracle@wnode03 ~]$ . oraenv
ORACLE_SID = [oracle] ? westdb1
The Oracle base has been set to /u01/app/oracle
[oracle@wnode03 ~]$

```

15. Create a starter initialization file for the physical standby instance with the following minimal parameters.

```

[oracle@wnode03 ~]$ cd $ORACLE_HOME/dbs
[oracle@wnode03 dbs]$ vi initwestdb.ora

### Add the following entries

db_name=eastdb
db_unique_name=westdb
db_domain=example.com

:wq!
[oracle@wnode03 dbs]$

```

16. Start up the temporary standby instance on wnode03 in unmount mode. Exit SQL*Plus when done.

```
[oracle@wnode03 dbs]$ sqlplus / as sysdba

SQL*Plus: Release 12.1.0.2.0 Production on Sat Sep 12 14:16:27 2015

Copyright (c) 1982, 2014, Oracle. All rights reserved.

Connected to an idle instance.

SQL> startup nomount pfile=$ORACLE_HOME/dbs/initwestdb.ora
ORACLE instance started.

Total System Global Area 243269632 bytes
Fixed Size          2923000 bytes
Variable Size       184550920 bytes
Database Buffers   50331648 bytes
Redo Buffers        5464064 bytes
SQL> exit
[oracle@wnode03 dbs]$
```

17. Create a physical standby using the RMAN utility.

```
[oracle@wnode03 dbs]$ rman target sys/oracle_4U@eastdb auxiliary
sys/oracle_4U@clone

Recovery Manager: Release 12.1.0.2.0 - Production on Sat Sep 12
14:20:34 2015

Copyright (c) 1982, 2014, Oracle and/or its affiliates. All rights
reserved.

connected to target database: EASTDB (DBID=812282532)
connected to auxiliary database: EASTDB (not mounted)

RMAN> run {
allocate channel prmy1 type disk;
allocate channel prmy2 type disk;
allocate channel prmy3 type disk;
allocate channel prmy4 type disk;
allocate auxiliary channel stby1 type disk;
duplicate target database for standby from active database
spfile
set db_unique_name='westdb'
set cluster_database='false'
set control_files='+DATA','+FRA'
set remote_listener='cluster02-scan:1521'
set fal_server='eastdb'
set audit_file_dest='/u01/app/oracle/admin/westdb/adump'
nofilenamecheck;
allocate auxiliary channel stby type disk;
sql channel stby "alter database recover managed standby database
disconnect";
}

using target database control file instead of recovery catalog
allocated channel: prmy1
```

Copyright © 2016, Oracle and/or its affiliates. All rights reserved.

```

channel prmy1: SID=40 instance=eastdb1 device type=DISK

allocated channel: prmy2
channel prmy2: SID=75 instance=eastdb1 device type=DISK

allocated channel: prmy3
channel prmy3: SID=83 instance=eastdb2 device type=DISK

allocated channel: prmy4
channel prmy4: SID=74 instance=eastdb2 device type=DISK

allocated channel: stby1
channel stby1: SID=29 device type=DISK

Starting Duplicate Db at 02-OCT-15

contents of Memory Script:
{
  backup as copy reuse
  targetfile '+DATA/eastdb/orapweastdb' auxiliary format
  '/u01/app/oracle/product/12.1.0/dbhome_1/dbs/orapwestdb1'
targetfile
  '+DATA/EASTDB/PARAMETERFILE/spfile.287.892019959' auxiliary format
  '/u01/app/oracle/product/12.1.0/dbhome_1/dbs/spfilewestdb1.ora' ;
  sql clone "alter system set spfile=
  ''/u01/app/oracle/product/12.1.0/dbhome_1/dbs/spfilewestdb1.ora'''";
}
executing Memory Script

Starting backup at 02-OCT-15
Finished backup at 02-OCT-15

sql statement: alter system set spfile=
  ''/u01/app/oracle/product/12.1.0/dbhome_1/dbs/spfilewestdb1.ora'''

contents of Memory Script:
{
  sql clone "alter system set db_unique_name =
  ''westdb'' comment=
  ''' scope=spfile";
  sql clone "alter system set cluster_database =
  false comment=
  ''' scope=spfile";
  sql clone "alter system set control_files =
  '''+DATA'', ''+FRA'' comment=
  ''' scope=spfile";
  sql clone "alter system set remote_listener =
  ''cluster02-scan:1521'' comment=
  ''' scope=spfile";
  sql clone "alter system set fal_server =
  ''eastdb'' comment=
  ''' scope=spfile";
  sql clone "alter system set audit_file_dest =
  ''/u01/app/oracle/admin/westdb/adump'' comment=
  ''' scope=spfile";
  shutdown clone immediate;
  startup clone nomount;
}

```

```

}

executing Memory Script

sql statement: alter system set db_unique_name = ''westdb'' comment=
'''' scope=spfile

sql statement: alter system set cluster_database = false comment=
'''' scope=spfile

sql statement: alter system set control_files = '''+DATA'', '''+FRA''
comment= ''''' scope=spfile

sql statement: alter system set remote_listener = ''cluster02-
scan:1521'' comment= ''''' scope=spfile

sql statement: alter system set fal_server = ''eastdb'' comment= '''
scope=spfile

sql statement: alter system set audit_file_dest =
''/u01/app/oracle/admin/westdb/adump'' comment= ''''' scope=spfile

Oracle instance shut down

connected to auxiliary database (not started)
Oracle instance started

Total System Global Area      524288000 bytes

Fixed Size                  2926320 bytes
Variable Size                436209936 bytes
Database Buffers             79691776 bytes
Redo Buffers                 5459968 bytes
allocated channel: stby1
channel stby1: SID=34 device type=DISK

contents of Memory Script:
{
  sql clone "alter system set control_files =
    '+DATA/WESTDB/CONTROLFILE/current.292.892025937',
    '+FRA/WESTDB/CONTROLFILE/current.328.892025937'' comment=
    ''Set by RMAN'' scope=spfile";
    backup as copy current controlfile for standby auxiliary format
    '+DATA/WESTDB/CONTROLFILE/current.293.892025937';
    restore clone primary controlfile to
    '+FRA/WESTDB/CONTROLFILE/current.327.892025937' from
    '+DATA/WESTDB/CONTROLFILE/current.293.892025937';
    sql clone "alter system set control_files =
      '+DATA/WESTDB/CONTROLFILE/current.293.892025937',
      '+FRA/WESTDB/CONTROLFILE/current.327.892025937'' comment=
      ''Set by RMAN'' scope=spfile";
      shutdown clone immediate;
      startup clone nomount;
}
executing Memory Script

sql statement: alter system set control_files =
  '+DATA/WESTDB/CONTROLFILE/current.292.892025937',

```

```
' '+FRA/WESTDB/CONTROLFILE/current.328.892025937'' comment= ''Set by
RMAN'' scope=spfile

Starting backup at 02-OCT-15
channel prmy1: starting datafile copy
copying standby control file
output file name=+DATA/WESTDB/CONTROLFILE/current.293.892025937
tag=TAG20151002T085857
channel prmy1: datafile copy complete, elapsed time: 00:00:03
Finished backup at 02-OCT-15

Starting restore at 02-OCT-15

channel stby1: copied control file copy
Finished restore at 02-OCT-15

sql statement: alter system set control_files =
' '+DATA/WESTDB/CONTROLFILE/current.293.892025937',
' '+FRA/WESTDB/CONTROLFILE/current.327.892025937'' comment= ''Set by
RMAN'' scope=spfile

Oracle instance shut down

connected to auxiliary database (not started)
Oracle instance started

Total System Global Area      524288000 bytes

Fixed Size                  2926320 bytes
Variable Size                436209936 bytes
Database Buffers             79691776 bytes
Redo Buffers                 5459968 bytes
allocated channel: stby1
channel stby1: SID=34 device type=DISK

contents of Memory Script:
{
  sql clone 'alter database mount standby database';
}
executing Memory Script

sql statement: alter database mount standby database

contents of Memory Script:
{
  set newname for clone tempfile 1 to new;
  switch clone tempfile all;
  set newname for clone datafile 1 to new;
  set newname for clone datafile 2 to new;
  set newname for clone datafile 3 to new;
  set newname for clone datafile 4 to new;
  set newname for clone datafile 5 to new;
  set newname for clone datafile 6 to new;
  backup as copy reuse
  datafile 1 auxiliary format new
  datafile 2 auxiliary format new
  datafile 3 auxiliary format new
```

```
datafile 4 auxiliary format new
datafile 5 auxiliary format new
datafile 6 auxiliary format new
;
sql 'alter system archive log current';
}
executing Memory Script

executing command: SET NEWNAME

renamed tempfile 1 to +DATA in control file

executing command: SET NEWNAME

Starting backup at 02-OCT-15
channel prmy1: starting datafile copy
input datafile file number=00005
name=+DATA/EASTDB/DATAFILE/example.282.892019073
channel prmy2: starting datafile copy
input datafile file number=00001
name=+DATA/EASTDB/DATAFILE/system.276.892018907
channel prmy3: starting datafile copy
input datafile file number=00003
name=+DATA/EASTDB/DATAFILE/sysaux.279.892018831
channel prmy4: starting datafile copy
input datafile file number=00002
name=+DATA/EASTDB/DATAFILE/undotbs2.285.892021195
output file name=+DATA/WESTDB/DATAFILE/undotbs2.286.892026023
tag=TAG20151002T085950
channel prmy4: datafile copy complete, elapsed time: 00:02:28
channel prmy4: starting datafile copy
input datafile file number=00004
name=+DATA/EASTDB/DATAFILE/undotbs1.290.892018975
output file name=+DATA/WESTDB/DATAFILE/undotbs1.276.892026173
tag=TAG20151002T085950
channel prmy4: datafile copy complete, elapsed time: 00:01:36
channel prmy4: starting datafile copy
input datafile file number=00006
name=+DATA/EASTDB/DATAFILE/users.289.892018973
output file name=+DATA/WESTDB/DATAFILE/users.294.892026273
tag=TAG20151002T085950
channel prmy4: datafile copy complete, elapsed time: 00:00:36
output file name=+DATA/WESTDB/DATAFILE/sysaux.287.892026011
tag=TAG20151002T085950
channel prmy3: datafile copy complete, elapsed time: 00:06:33
output file name=+DATA/WESTDB/DATAFILE/system.284.892026001
tag=TAG20151002T085950
```

```
channel prmy2: datafile copy complete, elapsed time: 00:06:50
output file name=+DATA/WESTDB/DATAFILE/example.281.892025993
tag=TAG20151002T085950
channel prmy1: datafile copy complete, elapsed time: 00:07:21
Finished backup at 02-OCT-15

sql statement: alter system archive log current

contents of Memory Script:
{
    switch clone datafile all;
}
executing Memory Script

datafile 1 switched to datafile copy
input datafile copy RECID=1 STAMP=892026441 file
name=+DATA/WESTDB/DATAFILE/system.284.892026001
datafile 2 switched to datafile copy
input datafile copy RECID=2 STAMP=892026441 file
name=+DATA/WESTDB/DATAFILE/undotbs2.286.892026023
datafile 3 switched to datafile copy
input datafile copy RECID=3 STAMP=892026441 file
name=+DATA/WESTDB/DATAFILE/sysaux.287.892026011
datafile 4 switched to datafile copy
input datafile copy RECID=4 STAMP=892026442 file
name=+DATA/WESTDB/DATAFILE/undotbs1.276.892026173
datafile 5 switched to datafile copy
input datafile copy RECID=5 STAMP=892026442 file
name=+DATA/WESTDB/DATAFILE/example.281.892025993
datafile 6 switched to datafile copy
input datafile copy RECID=6 STAMP=892026442 file
name=+DATA/WESTDB/DATAFILE/users.294.892026273
Finished Duplicate Db at 02-OCT-15

allocated channel: stby
channel stby: SID=46 device type=DISK

sql statement: alter database recover managed standby database
disconnect
released channel: prmy1
released channel: prmy2
released channel: prmy3
released channel: prmy4
released channel: stby1
released channel: stby

RMAN> exit
[oracle@wnode03 ~] $
```

18. Exit all of your terminal sessions.

Practice 4-3: Completing a Physical Standby Database for RAC

Overview

You have created the physical standby database in practice 4-2. In this practice, you will perform several tasks to finish the RAC configuration on the physical standby database.

Terminal Sessions

- **Session 1:** ssh oracle@enode01
- **Session 2:** ssh oracle@wnode03



Tasks

1. **(Session 1)** Open a terminal window for enode01 logged in as oracle with the environment variables set to eastdb1.

```
[Classroom PC] $ ssh oracle@enode01
oracle@enode01's password: <oracle>
[oracle@enode01 ~]$ . oraenv
ORACLE_SID = [oracle]? eastdb
The Oracle base has been set to /u01/app/oracle
[oracle@enode01 ~]$ export ORACLE_SID=eastdb1
[oracle@enode01 ~]$
```

2. Determine the last sequence number archived on the primary database (eastdb). Your number may be different in the lab.

```
[oracle@enode01 ~]$ sqlplus / as sysdba
SQL> SELECT THREAD#, MAX(SEQUENCE#) FROM V$ARCHIVED_LOG GROUP BY
THREAD#;

THREAD# MAX (SEQUENCE#)
-----
1           11
2            2

SQL>
```

3. **(Session 2)** Open a new terminal to connect to wnode03 as the oracle user. Set up your environment variables correctly for the oracle user.

```
[Classroom PC] $ ssh oracle@wnode03
oracle@wnode03's password: <oracle>
[oracle@wnode03 ~]$ . oraenv
ORACLE_SID = [oracle]? westdb1
The Oracle base has been set to /u01/app/oracle
[oracle@wnode03 ~]$
```

4. Check if the managed recovery process (MRP) is running on the physical standby database or not.

```
[oracle@wnode03 ~]$ pgrep -lf mrp
25968 ora_mrp0_westdb1
[oracle@wnode03 dbs]$
```

5. Check if the archived log files have been received and applied on the physical standby database (westdb). **Note:** The archive log files have not been transported to the physical standby database.

```
[oracle@wnode03 ~]$ sqlplus / as sysdba
SQL> SELECT THREAD#, SEQUENCE#, APPLIED FROM V$ARCHIVED_LOG ORDER BY
THREAD#;

no rows selected

SQL>
```

6. (**Session 1**) Return to the first terminal session on enode01 to examine the status of the remote log archive destination (LOG_ARCHIVE_DEST_2). **Note:** You may observe an error in the ERROR column. Ignore it and move to the next step.

```
SQL> col destination format a10
SQL> col error format a10
SQL> select dest_id, status, destination, archiver, error, valid_now
from v$archive_dest where dest_id = 2;

DEST_ID STATUS      DESTINATIO ARCHIVER      ERROR          VALID_NOW
----- -----        -----
2 DEFERRED    westdb     LGWR        ORA-12514    UNKNOWN
SQL>
```

7. Enable LOG_ARCHIVE_DEST_2.

Note: You have set LOG_ARCHIVE_DEST_STATE_2 to DEFER to avoid periodic errors (ORA-12514 and TNS-12564) in practice 4-1. If an error appears in the ERROR column, resolve the issue first, working with your instructor.

```
SQL> alter system set log_archive_dest_state_2=ENABLE;
System altered.
SQL> select dest_id, status, destination, archiver, error, valid_now
from v$archive_dest where dest_id = 2;

DEST_ID STATUS      DESTINATIO ARCHIVER      E VALID_NOW
----- -----        -----
2 VALID       westdb     LGWR          YES
SQL>
```

Possible Workarounds

Try one of the following methods:

- Restart the physical standby and primary databases.
- Review practice 13-1 and 13-2 in Appendix.
- Perform step 7 again.

8. **(Session 2)** Return to the terminal session on wnode03 to check the redo transport and apply status. **Note:** If none of the archived logs have been applied, then perform step 7 again and apply the possible workaround above if necessary.

```
SQL> SELECT THREAD#, SEQUENCE#, APPLIED FROM V$ARCHIVED_LOG ORDER BY
  THREAD#;
```

THREAD#	SEQUENCE#	APPLIED
1	11	YES
2	2	YES

```
SQL>
```

Note: As you can see, everything is back to normal. Now, you will perform the additional tasks to configure your physical standby database in a RAC environment.

9. In the same terminal session on wnode03, create a temporary pfile from spfile on wnode03. Exit SQL*Plus when done.

```
SQL> create pfile='/tmp/init.ora' from spfile;
File created.
SQL> exit
[oracle@wnode03 ~] $
```

10. Modify the temporary pfile to match the standby instance names.

```
[oracle@wnode03 ~] $ vi /tmp/init.ora
```

a. Remove the westdb1_* entries.

```
:g/westdb1._/d
10 fewer lines
```

b. Replace the eastdb_* entries with the westdb_* entries.

```
:%s/eastdb._/westdb._/g
10 substitutions on 10 lines
```

c. Replace the eastdb1 entries with the westdb1 entries.

```
:%s/eastdb1/westdb1/g
13 substitutions on 13 lines
```

d. Replace the eastdb2 entries with the westdb2 entries.

```
:%s/eastdb2/westdb2/g
13 substitutions on 13 lines
```

e. Update the highlighted entries and save the changes.

```
*.audit_file_dest='/u01/app/oracle/admin/westdb/adump'
*.audit_trail='none'
*.cluster_database=TRUE
*.compatible='12.1.0.2.0'
*.control_files='+DATA/WESTDB/CONTROLFILE/current.293.890740483', '+FRA/
WESTDB/CONTROLFILE/current.327.890740483'#Set by RMAN
*.db_block_size=8192
*.db_create_file_dest='+DATA'
*.db_domain='example.com'
*.db_name='eastdb'
*.db_recovery_file_dest='+FRA'
*.db_recovery_file_dest_size=5000m
*.db_unique_name='westdb'
```

```
*.diagnostic_dest='/u01/app/oracle'
*.dispatchers='(PROTOCOL=TCP)  (SERVICE=westdbXDB)'
*.fal_server='eastdb'
westdb1.instance_number=1
westdb2.instance_number=2
*.log_archive_config='dg_config=(eastdb,westdb)'
*.log_archive_dest_1='location=USE_DB_RECOVERY_FILE_DEST
valid_for=(ALL_LOGFILES,ALL_ROLES) db_unique_name=westdb'
*.log_archive_dest_2='SERVICE=eastdb ASYNC
valid_for=(ONLINE_LOGFILES,PRIMARY_ROLE) db_unique_name=eastdb'
*.log_archive_dest_state_1='enable'
*.log_archive_dest_state_2='enable'
*.log_archive_max_processes=4
*.memory_target=500m
*.open_cursors=300
*.processes=300
*.remote_listener='cluster02-scan:1521'
*.remote_login_passwordfile='EXCLUSIVE'
*.standby_file_management='auto'
westdb1.thread=1
westdb2.thread=2
westdb1.undo_tablespace='UNDOTBS1'
westdb2.undo_tablespace='UNDOTBS2'

:wq!
[oracle@wnode03 ~] $
```

11. Create an spfile in a shared storage for the standby database in RAC.

```
[oracle@wnode03 ~]$ sqlplus / as sysdba
SQL> create spfile='+DATA/westdb/spfilewestdb.ora' from
pfile='/tmp/init.ora';
File created.
SQL> exit
[oracle@wnode03 ~] $
```

12. Modify the \$ORACLE_HOME/dbs/initwestdb1.ora file on wnode03 to point to the spfile created in the shared storage.

```
[oracle@wnode03 ~]$ vi $ORACLE_HOME/dbs/initwestdb1.ora

### And Add the following entry

spfile='+DATA/westdb/spfilewestdb.ora'

:wq!
[oracle@wnode03 ~] $
```

13. Create the \$ORACLE_HOME/dbs/initwestdb2.ora file on wnode04 to point to the spfile created in the shared storage.

```
[oracle@wnode03 ~]$ scp $ORACLE_HOME/dbs/initwestdb1.ora
wnode04:/u01/app/oracle/product/12.1.0/dbhome_1/dbs/initwestdb2.ora
initwestdb1.ora                                100%   39      0.0KB/s   00:00
[oracle@wnode03 ~] $
```

14. Shut down the first standby instance named `westdb1` to restart with the modified initialization parameter file.

```
[oracle@wnode03 ~]$ sqlplus / as sysdba
SQL> shutdown immediate
ORA-01109: database not open

Database dismounted.
ORACLE instance shut down.
SQL> exit
[oracle@wnode03 ~] $
```

15. Using the SRVCTL utility, create the clusterware entries for the newly created standby database.

```
[oracle@wnode03 ~]$ srvctl add database -db westdb -oraclehome
/u01/app/oracle/product/12.1.0/dbhome_1 -dbtype RAC -spfile
'+DATA/westdb/spfilewestdb.ora' -role physical_standby -diskgroup
"DATA,FRA" -dbname eastdb -domain example.com
[oracle@wnode03 ~]$ srvctl add instance -db westdb -instance westdb1 -
node wnode03
[oracle@wnode03 ~]$ srvctl add instance -db westdb -instance westdb2 -
node wnode04
[oracle@wnode03 ~] $
```

16. Using the SRVCTL utility, start the physical standby database.

```
[oracle@wnode03 ~]$ srvctl start database -db westdb -startoption mount
[oracle@wnode03 ~]$ srvctl status database -db westdb -verbose
Instance westdb1 is running on node wnode03. Instance status: Mounted
(Closed).
Instance westdb2 is running on node wnode04. Instance status: Mounted
(Closed).
[oracle@wnode03 ~] $
```

17. Start the managed recovery process (MRP).

```
[oracle@wnode03 ~]$ sqlplus / as sysdba
SQL> alter database recover managed standby database disconnect;
Database altered.
SQL> exit
SQL>
```

18. Verify the standby database in RAC.

```
[oracle@wnode03 ~]$ srvctl config database -db westdb
Database unique name: westdb
Database name: eastdb
Oracle home: /u01/app/oracle/product/12.1.0/dbhome_1
Oracle user: oracle
Spfile: +DATA/westdb/spfilewestdb.ora
Password file:
Domain: example.com
Start options: open
Stop options: immediate
Database role: PHYSICAL_STANDBY
Management policy: AUTOMATIC
Server pools:
Disk Groups: DATA,FRA
Mount point paths:
Services:
```

```
Type: RAC
Start concurrency:
Stop concurrency:
Database is enabled
Database is individually enabled on nodes:
Database is individually disabled on nodes:
OSDBA group: dba
OSOPER group: oper
Database instances: westdb1,westdb2
Configured nodes: wnode03,wnode04
Database is administrator managed
[oracle@wnode03 ~] $
```

19. Exit all of your terminal sessions.

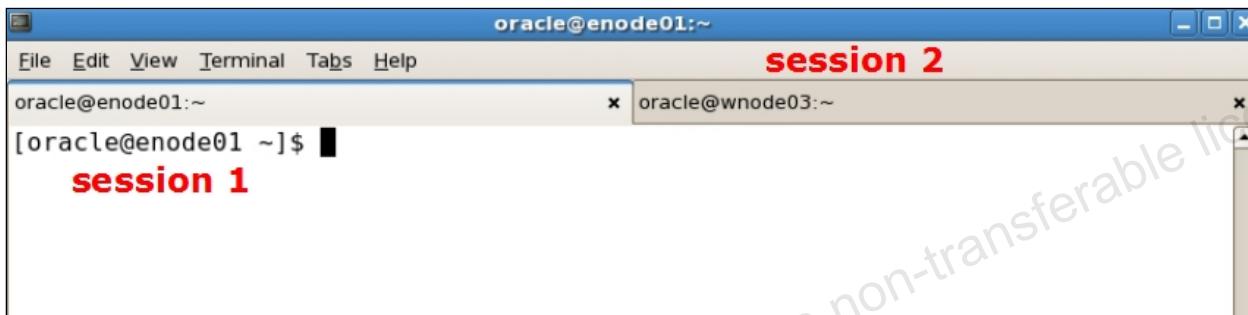
Practice 4-4: Verifying Redo Transport and Redo Apply Operations in RAC

Overview

In this practice, you will perform a small test to verify the redo transport and redo apply operations in RAC.

Terminal Sessions

- **Session 1:** ssh oracle@enode01
- **Session 2:** ssh oracle@wnode03



Tasks

1. **(Session 1)** Open a terminal window for enode01 logged in as oracle with the environment variables set to eastdb1.

```
[Classroom PC]$ ssh oracle@enode01
oracle@enode01's password: <oracle>
[oracle@enode01 ~]$ . oraenv
ORACLE_SID = [oracle]? eastdb
The Oracle base has been set to /u01/app/oracle
[oracle@enode01 ~]$ export ORACLE_SID=eastdb1
[oracle@enode01 ~]$
```

2. Determine the last sequence number archived on the primary database (eastdb). Your number may be different in the lab.

```
[oracle@enode01 ~]$ sqlplus / as sysdba
SQL> SELECT THREAD#, MAX(SEQUENCE#) FROM GV$ARCHIVED_LOG WHERE
RESETLOGS_CHANGE# = (SELECT MAX(RESETLOGS_CHANGE#) FROM
GV$ARCHIVED_LOG) GROUP BY THREAD#;

      THREAD#  MAX (SEQUENCE#)
-----  -----
          1            13
          2              4

SQL>
```

3. **(Session 2)** Open a new terminal to connect to wnode03 as the oracle user. Set up your environment variables correctly for the oracle user.

```
[Classroom PC]$ ssh oracle@wnode03
oracle@wnode03's password: <oracle>
[oracle@wnode03 ~]$ . oraenv
```

```
ORACLE_SID = [oracle]? westdb1
The Oracle base has been set to /u01/app/oracle
[oracle@wnode03 ~]$
```

4. Check if the managed recovery process (MRP) is running on the physical standby database or not.

```
[oracle@wnode03 ~]$ pgrep -lf mrp
25968 ora_mrp0_westdb1
[oracle@wnode03 dbs]$
```

5. Check if the archived log files have been received and applied on the physical standby database (westdb).

```
[oracle@wnode03 ~]$ sqlplus / as sysdba
SQL> SELECT THREAD#, SEQUENCE#, APPLIED FROM V$ARCHIVED_LOG ORDER BY
THREAD#;

THREAD#    SEQUENCE# APPLIED
-----
1          12 YES
1          13 YES
1          11 YES
2          4 IN-MEMORY
2          3 YES
2          2 YES

6 rows selected.

SQL>
```

6. **(Session 1)** Return to the first terminal session on enode01 and force a log switch by issuing ALTER SYSTEM ARCHIVE LOG CURRENT to advance the online redo log sequence number. Verify that the sequence number has increased. **Note:** The ALTER SYSTEM ARCHIVE LOG CURRENT command switches the logs on all RAC instances synchronously, whereas the ALTER SYSTEM SWITCH LOGFILE command switches the log file on the current instance where you issue the switch command.

```
SQL> alter system archive log current;
System altered.

SQL> SELECT THREAD#, MAX(SEQUENCE#) FROM GV$ARCHIVED_LOG WHERE
RESETLOGS_CHANGE# = (SELECT MAX(RESETLOGS_CHANGE#) FROM
GV$ARCHIVED_LOG) GROUP BY THREAD#;

THREAD# MAX (SEQUENCE#)
-----
1          16
2            7

SQL>
```

7. (**Session 2**) Return to the second terminal session on wnode03 and check if the archived log files have been received and applied on the physical standby database (westdb).

```
SQL> SELECT THREAD#, SEQUENCE#, APPLIED FROM V$ARCHIVED_LOG ORDER BY THREAD#;
```

THREAD#	SEQUENCE#	APPLIED
1	16	YES
1	15	YES
1	13	YES
1	11	YES
1	14	YES
1	12	YES
2	6	YES
2	5	YES
2	7	IN-MEMORY
2	3	YES
2	2	YES

THREAD#	SEQUENCE#	APPLIED
2	4	YES

12 rows selected.

As you can see, the redo transport and redo apply operations are normal in RAC environment. Now, you will perform a small test.

8. (**Session 1**) Return to the first terminal session on enode01 to create a simple table named hr.test and insert a few rows to test the Redo Transport and Redo Apply operations.

```
SQL> create table hr.test (col number);

Table created.

SQL> insert into hr.test values (1);

1 row created.

SQL> /

1 row created.

SQL> /

1 row created.

SQL> commit;

Commit complete.

SQL> select count(*) from hr.test;

COUNT(*)
-----
3

SQL>
```

9. **(Session 2)** Return to the terminal session on wnode03 to check the status of the managed recovery process (MRP). **Note:** MRP0 in the process column indicates MRP is running.

```
SQL> select inst_id, process, status, sequence#, thread# from gv$managed_standby;
```

INST_ID	PROCESS	STATUS	SEQUENCE#	THREAD#
1	ARCH	CLOSING	13	1
1	ARCH	CONNECTED	0	0
1	ARCH	CLOSING	7	2
1	ARCH	CLOSING	5	2
1	RFS	IDLE	8	2
1	RFS	IDLE	0	0
1	RFS	IDLE	0	0
1	MRP0	APPLYING_LOG	8	2
2	ARCH	CLOSING	16	1
2	ARCH	CONNECTED	0	0
2	ARCH	CLOSING	14	1
INST_ID	PROCESS	STATUS	SEQUENCE#	THREAD#
2	ARCH	CLOSING	15	1
2	RFS	IDLE	0	0
2	RFS	IDLE	0	0
2	RFS	IDLE	17	1

15 rows selected.

```
SQL> -- If MRP is not running, execute the following command
SQL> alter database recover managed standby database disconnect;
SQL>
```

10. Stop the managed recovery process.

```
SQL> alter database recover managed standby database cancel;
Database altered.
SQL>
```

11. Open the physical standby database instance in READ ONLY mode to check if the changes have been applied.

```
SQL> alter database open read only;
Database altered.
```

```
SQL> select count(*) from hr.test;
```

```
COUNT (*)
-----
3
```

```
SQL> exit
[oracle@wnode03 ~] $
```

Note: As you can see, the changes made to the primary database have been transported and applied on the physical standby database successfully.

12. Mount the physical standby database.

```
[oracle@wnode03 ~]$ srvctl stop database -db westdb
[oracle@wnode03 ~]$ srvctl start database -db westdb -startoption mount
[oracle@wnode03 ~]$
```

13. Start the managed recovery process.

```
[oracle@wnode03 ~]$ sqlplus / as sysdba
SQL> alter database recover managed standby database disconnect;
Database altered.
SQL>
```

14. Exit all of your terminal sessions.

Practices for Lesson 5: Configuring Oracle Data Guard in an Oracle RAC Environment

Chapter 5

Practices for Lesson 5: Overview

Practices Overview

In this practice, you use the password file in a disk group and copy it from the primary site to the standby site when needed. You also test the local and remote connection with DGMGRL, and then create a Data Guard Broker configuration for RAC. Additionally, you configure and test the role transition services, including switchover, failover, and fast-start failover in RAC. The successful completion of these practices is required for the following practices.

Practice Summary

- Practice 5-1: Using a Shared Password File in an ASM Disk Group
- Practice 5-2: Creating a Data Guard Broker Configuration for RAC
- Practice 5-3: Monitoring Redo Transport Service and Redo Apply Service in RAC
- Practice 5-4: Configuring Flashback Database on the Primary and Physical Standby Databases
- Practice 5-5: Enabling Fast-Start Failover for RAC
- Practice 5-6: Testing Fast-Start Failover in RAC
- Practice 5-7: Performing Switchover to Reinstated Database in RAC and Disabling Fast-Start Failover

Practice 5-1: Using a Shared Password File in an ASM Disk Group

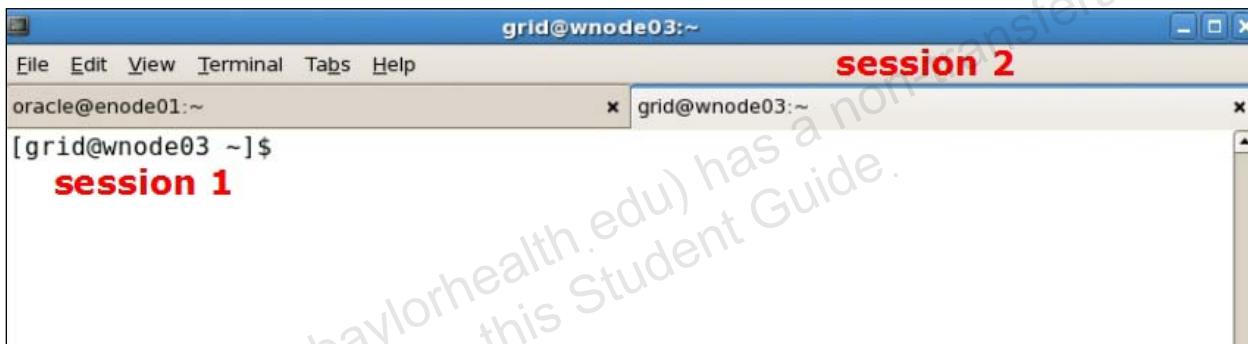
Overview

During the creation of a standby database or after having altered the password for any user who is granted the SYSDBA, SYSOPER, or SYSDG privileges, the password file must be copied from the primary site to the standby site. Oracle Database Enterprise Edition 12.1.0.1 and above can now store password files in ASM disk groups. By default, if DBCA is used to create a RAC database, the password file is created in an ASM disk group on the primary site.

In this practice, you will create the password file in an ASM disk group for the standby database.

Terminal Sessions

- **Session 1:** ssh oracle@enode01
- **Session 2:** ssh grid@wnode03



Tasks

1. **(Session 1)** From your classroom PC desktop, execute ssh oracle@enode01 to open a terminal session on enode01 as the oracle user.

```
[Classroom PC]$ ssh oracle@enode01
oracle@enode01's password: <oracle>
[oracle@enode01 ~]$
```

2. Make sure that you set up your environment variables correctly for the oracle user.

```
[oracle@enode01 ~]$ . oraenv
ORACLE_SID = [oracle]? eastdb
The Oracle base has been set to /u01/app/oracle
[oracle@enode01 ~]$ export ORACLE_SID=eastdb1
[oracle@enode01 ~]$
```

3. Determine the current location of the password file for eastdb.

Note: The current location of the password file on the primary database is the shared storage +DATA disk group.

```
[oracle@enode01 ~]$ srvctl config database -db eastdb
Database unique name: eastdb
Database name: eastdb
Oracle home: /u01/app/oracle/product/12.1.0/dbhome_1
Oracle user: oracle
```

```
Spfile: +DATA/EASTDB/PARAMETERFILE/spfile.299.890151031
Password file: +DATA/eastdb/orapweastdb
Domain: example.com
Start options: open
Stop options: immediate
Database role: PRIMARY
Management policy: AUTOMATIC
Server pools:
Disk Groups: DATA,FRA
Mount point paths:
Services:
Type: RAC
Start concurrency:
Stop concurrency:
OSDBA group: dba
OSOPER group: oper
Database instances: eastdb1,eastdb2
Configured nodes: enode01,enode02
Database is administrator managed
[oracle@enode01 ~]$
```

4. Launch the DGMGRL utility and verify that you are able to connect as the sysdg user with operating system authentication. (The oracle OS user is in the dgdba group.)

Note: With operating system authentication, any password will work for local connections. However, during switchover and failover operations to the remote site, you must use the correct password.

```
[oracle@enode01 ~]$ dgmgrl
DGMGRL for Linux: Version 12.1.0.2.0 - 64bit Production
Copyright (c) 2000, 2012, Oracle. All rights reserved.
Welcome to DGMGRL, type "help" for information.
DGMGRL> connect sysdg
Password: ldjladsjf1sjdkljaldsjflds → Any password will work
Connected as SYSDG.
DGMGRL>
```

5. Attempt to make a remote connection to the physical standby database westdb as the sysdg user. You must use the password that is in the password file. Exit DGMGRL.

```
DGMGRL> connect sysdg@eastdb
Password: <oracle_4U>
ORA-01017: invalid username/password; logon denied

Warning: You are no longer connected to ORACLE.
DGMGRL> exit
[oracle@enode01 ~]$
```

Note: During the creation of the database, the option to use the same password for all administrative accounts was chosen. However, this only applied to the SYS and SYSTEM database accounts.

6. Use SQL*Plus on enode01 connected as SYSDBA to reset the SYSDG password and unlock the account. Exit SQL*Plus.

```
[oracle@enode01 ~]$ sqlplus / as sysdba
SQL*Plus: Release 12.1.0.2.0 Production on Wed Sep 11 06:17:09
2013
Copyright (c) 1982, 2013, Oracle. All rights reserved.
Connected to:
Oracle Database 12c Enterprise Edition Release 12.1.0.1.0 -
64bit Production
With the Partitioning, OLAP, Advanced Analytics and Real
Application Testing options
SQL> alter user sysdg identified by oracle_4U;
User altered.
SQL> alter user sysdg account unlock;
User altered.
SQL> exit;
[oracle@enode01 ~]$
```

7. Attempt to make a remote connection to the physical standby database westdb as the sysdg user again. Exit DGMGRL.

```
[oracle@enode01 ~]$ dgmgrl
DGMGRL> connect sysdg@eastadb
Password: <oracle_4U>
ORA-01017: invalid username/password; logon denied

Warning: You are no longer connected to ORACLE.
DGMGRL> exit
[oracle@enode01 ~]$
```

Note: The remote connection failed again because the standby database password file has not been updated yet.

8. In the same terminal window, switch to the grid OS user and then set up your environment variables correctly for the grid user.

```
[oracle@enode01 ~]$ su - grid
Password: <oracle>
[grid@enode01 ~]$ . oraenv
ORACLE_SID = [grid]? +ASM1
The Oracle base has been set to /u01/app/grid
[grid@enode01 ~]$
```

9. Using the asmcmd command, copy the shared password file in +DATA to a file system-based directory /tmp. Exit out of the grid session.

```
[grid@enode01 ~]$ asmcmd pwcOPY +DATA/eastdb/orapweastdb
/tmp/orapweastdb
copying +DATA/eastdb/orapweastdb -> /tmp/orapwestdb
[grid@enode01 ~]$ exit
[oracle@enode01 ~]$
```

10. Copy the password file in /tmp/orapweastdb to one of the standby hosts (wnode03).

```
[oracle@enode01 ~]$ scp /tmp/orapweastdb wnode03:/tmp/orapwestdb
oracle@wnode03's password: <oracle>
orapweastdb                                         100% 7680      7.5KB/s   00:00
[oracle@enode01 ~]$
```

11. (**Session 2**) Open a terminal to connect to wnode03 as the grid user and then set up your environment variables correctly for the grid user.

```
[Classroom PC]$ ssh grid@wnode03
[grid@wnode03 ~]$ . oraenv
ORACLE_SID = [grid]? +ASM1
The Oracle base has been set to /u01/app/grid
[grid@wnode03 ~]$
```

12. Determine the current location of the password file for westdb.

Note: The Password file entry is empty, which indicates that the local password file is used to authenticate the administrative users.

```
[grid@wnode03 ~]$ srvctl config database -db westdb
Database unique name: westdb
Database name: eastdb
Oracle home: /u01/app/oracle/product/12.1.0/dbhome_1
Oracle user: oracle
Spfile: +DATA/westdb/spfilewestdb.ora
Password file:
Domain: example.com
Start options: open
Stop options: immediate
Database role: PHYSICAL_STANDBY
Management policy: AUTOMATIC
Server pools:
Disk Groups: DATA,FRA
Mount point paths:
Services:
Type: RAC
Start concurrency:
Stop concurrency:
OSDBA group: dba
OSOPER group: oper
Database instances: westdb1,westdb2
Configured nodes: wnode03,wnode04
Database is administrator managed
[grid@wnode03 ~]$
```

13. Using the asmcmd command, copy the password file in the /tmp directory to +DATA.

```
[grid@wnode03 ~]$ asmcmd pwcopy /tmp/orapwestdb
+data/westdb/orapwestdb
copying /tmp/orapwestdb -> +data/westdb/orapwestdb [grid@wnode03 ~]$
```

14. Attempt to update the clusterware resource for the database and set the location of the password file to be used by the database using the `srvctl modify database` command.

Note: The grid OS user does not have enough privileges to update the password file location of the standby database.

```
[grid@wnode03 ~]$ srvctl modify database -db westdb -pwfile
+data/westdb/orapwwestdb
PRCD-1163 : Failed to modify database westdb
PRCR-1071 : Failed to register or update resource ora.westdb.db
CRS-0245: User doesn't have enough privilege to perform the operation
[grid@wnode03 ~]$
```

15. As the owner of the RDBMS software for the database, update the clusterware resource for the database and set the location of the password file.

```
[grid@wnode03 ~]$ su - oracle
Password: <oracle>
[oracle@wnode03 ~]$ . oraenv
ORACLE_SID = [oracle] ? westdb
The Oracle base has been set to /u01/app/oracle
[oracle@wnode03 ~]$ export ORACLE_SID=westdb1
[oracle@wnode03 ~]$ srvctl modify database -db westdb -pwfile
+data/westdb/orapwwestdb
[oracle@wnode03 ~]$
```

16. Verify the updated password file location for `westdb`.

```
[oracle@wnode03 ~]$ srvctl config database -db westdb
Database unique name: westdb
Database name: eastdb
Oracle home: /u01/app/oracle/product/12.1.0/dbhome_1
Oracle user: oracle
Spfile: +DATA/westdb/spfilewestdb.ora
Password file: +data/westdb/orapwwestdb
Domain: example.com
Start options: open
Stop options: immediate
Database role: PHYSICAL_STANDBY
Management policy: AUTOMATIC
Server pools:
Disk Groups: DATA,FRA
Mount point paths:
Services:
Type: RAC
Start concurrency:
Stop concurrency:
OSDBA group: dba
OSOPER group: oper
Database instances: westdb1,westdb2
Configured nodes: wnode03,wnode04
Database is administrator managed
[oracle@wnode03 ~]$
```

17. Launch the DGMGRL utility to verify that you are now able to establish a remote connection as sysdg to the physical standby database. Exit DGMGRL when done.

```
[oracle@wnode03 ~]$ dgmgrl
DGMGRL for Linux: Version 12.1.0.2.0 - 64bit Production
Copyright (c) 2000, 2012, Oracle. All rights reserved.
Welcome to DGMGRL, type "help" for information.
DGMGRL> connect sysdg/oracle_4U@westdb
Connected as SYSDG.
DGMGRL> connect sysdg/oracle_4U@eastdb
Connected as SYSDG.
DGMGRL> exit
[oracle@wnode03 ~]$
```

18. Exit all of your terminal sessions.

Practice 5-2: Creating a Data Guard Broker Configuration for RAC

Overview

In this practice, you will create the Data Guard configuration for RAC. The physical standby database will be added to the configuration. You will also enable the configuration.

Terminal Sessions

- **Session 1:** ssh oracle@enode01
- **Session 2:** ssh oracle@wnode03



Tasks

1. **(Session 1)** From your classroom PC desktop, execute ssh oracle@enode01 to open a terminal session on enode01 as the oracle user.

```
[Classroom PC]$ ssh oracle@enode01
oracle@enode01's password: <oracle>
[oracle@enode01 ~]$
```

2. Make sure that you set up your environment variables correctly for the oracle user.

```
[oracle@enode01 ~]$ . oraenv
ORACLE_SID = [oracle]? eastdb
The Oracle base has been set to /u01/app/oracle
[oracle@enode01 ~]$ export ORACLE_SID=eastdb1
[oracle@enode01 ~]$
```

3. Connect to the primary database using SQL*Plus to perform the following tasks:

- a. Reset the LOG_ARCHIVE_DEST_2 parameter.

Note: As of Oracle Database 12c Release 1 (12.1), for all databases to be added to a broker configuration, any LOG_ARCHIVE_DEST_n parameters that have the SERVICE attribute set, but not the NOREGISTER attribute, must be cleared.

- b. Configure the DG_BROKER_CONFIG_FILEn initialization parameters for that database so that they point to the same shared files for all instances of that database.

Note: These parameters can only be set or changed when the Data Guard Broker is not running (DG_BROKER_START=FALSE).

- c. Start the Data Guard Broker process. Make sure the changes are persistent. Exit SQL*Plus.

```
[oracle@enode01 ~]$ sqlplus / as sysdba
SQL> alter system set log_archive_dest_2='' scope=both sid='*';
System altered.
SQL> alter system set
dg_broker_config_file1='+data/eastdb/dr1config.dat' scope=both sid='*';
System altered.
SQL> alter system set
dg_broker_config_file2='+fra/eastdb/dr2config.dat' scope=both sid='*';
System altered.
SQL> alter system set dg_broker_start=true scope=both sid='*';
System altered.
SQL> exit
[oracle@enode01 ~]$
```

4. **(Session 2)** Open a terminal window on wnode03 connected as oracle with the environment variables set to westdb1.

```
[Classroom PC]$ ssh oracle@wnode03
oracle@wnode03's password: <oracle>
[oracle@wnode03 ~]$ . oraenv
ORACLE_SID = [oracle]? westdb
The Oracle base has been set to /u01/app/oracle
[oracle@wnode03 ~]$ export ORACLE_SID=westdb1
[oracle@wnode03 ~]$
```

5. Connect to the physical standby using SQL*Plus to perform the following tasks:

- Stop managed recovery.
- Reset the LOG_ARCHIVE_DEST_2 parameter.

Note: As of Oracle Database 12c Release 1 (12.1), for all databases to be added to a broker configuration, any LOG_ARCHIVE_DEST_n parameters that have the SERVICE attribute set, but not the NOREGISTER attribute, must be cleared.

- Configure the DG_BROKER_CONFIG_FILEn initialization parameters for that database so that they point to the same shared files for all instances of that database.

Note: These parameters can only be set or changed when the Data Guard Broker is not running (DG_BROKER_START=FALSE).

- Start the Data Guard Broker process. Make sure the changes are persistent. Exit SQL*Plus.

```
[oracle@wnode03 ~]$ sqlplus / as sysdba
SQL> alter database recover managed standby database cancel;
Database altered.
SQL> alter system set log_archive_dest_2='' scope=both sid='*';
System altered.
SQL> alter system set
dg_broker_config_file1='+data/westdb/dr1config.dat' scope=both sid='*';
System altered.
SQL> alter system set
dg_broker_config_file2='+fra/westdb/dr2config.dat' scope=both sid='*';
System altered.
```

```
SQL> alter system set dg_broker_start=true scope=both sid='*';
System altered.
SQL> exit
[oracle@wnode03 ~]$
```

6. **(Session 1)** Return to the terminal window on enode01 to the primary database. Launch DGMGRL and attempt to show the configuration.

```
[oracle@enode01 ~]$ dgmgrl
DGMGRL for Linux: Version 12.1.0.2.0 - 64bit Production
Copyright (c) 2000, 2012, Oracle. All rights reserved.
Welcome to DGMGRL, type "help" for information.
DGMGRL> connect sysdg/oracle_4U@eastdb
Connected as SYSDG.
DGMGRL> show configuration
ORA-16532: Data Guard broker configuration does not exist

Configuration details cannot be determined by DGMGRL
DGMGRL>
```

7. Create the Data Guard broker configuration and then show the configuration.

```
DGMGRL> create configuration 'dg_config' as primary database is
'eastdb' connect identifier is eastdb;
Configuration "dg_config" created with primary database "eastdb"

DGMGRL> show configuration

Configuration - dg_config

Protection Mode: MaxPerformance
Members:
eastdb - Primary database

Fast-Start Failover: DISABLED

Configuration Status:
DISABLED

DGMGRL>
```

Note: Since the Data Guard broker is a distributed framework, the DGMGRL utility can be launched from any host machine that participates in the Data Guard configuration. The labs will continue to display the machine name (enode01), and therefore the terminal session window being used, for which the DGMGRL utility was launched. It would be acceptable though to launch it from another terminal session connected to another virtual host machine.

8. Add the physical standby database westdb to the configuration and show the results.

```
DGMGRL> add database 'westdb' as connect identifier is westdb;
Database "westdb" added
DGMGRL> show configuration

Configuration - dg_config

Protection Mode: MaxPerformance
Members:
```

```

eastdb - Primary database
westdb - Physical standby database

Fast-Start Failover: DISABLED

Configuration Status:
DISABLED

DGMGRL>

```

- To prepare for the future practices, define redo routing rules for the configuration and show the results. The current primary database 'eastdb' should transport redo to the physical standby database 'westdb' synchronously. Additional redo routing rules should be created for role reversal. After role reversal, the primary database will be 'westdb' and should transport redo to the physical standby database 'eastdb' synchronously.

```

DGMGRL> EDIT DATABASE 'eastdb' SET PROPERTY 'RedoRoutes' =
'(eastdb:westdb SYNC)';
Property "RedoRoutes" updated
DGMGRL> EDIT DATABASE 'westdb' SET PROPERTY 'RedoRoutes' =
'(westdb:eastdb SYNC)';
Property "RedoRoutes" updated
DGMGRL>

```

Note: By default, a primary database sends the redo that it generates to every other redo transport destination in the configuration. You can use the `RedoRoutes` property to create a more complex redo transport topology, such as one in which a physical standby database or a far sync instance forwards redo received from the primary database to one or more destinations, or one in which the redo transport mode used for a given destination is dependent on which database is in the primary role.

Note: If a database receives redo from a database where the `RedoRoutes` property has been configured with a redo transport mode, then the mode specified by that `RedoRoutes` property value overrides the value of the `LogXptMode` property (`LOG_ARCHIVE_DEST_n`).

- Set the `MaxConnections` property for the primary and standby databases to transmit redo data from a single archived redo log on the primary database to the archived redo log at the remote site for gap resolution in parallel.

```

DGMGRL> edit database westdb set property MaxConnections=4;
DGMGRL> edit database eastdb set property MaxConnections=4;
DGMGRL>

```

- Set the `CommunicationTimeout` property to a value of 300.

```

DGMGRL> edit configuration set property CommunicationTimeout = 300;
Property "communicationtimeout" updated
DGMGRL>

```

Note: The `CommunicationTimeout` configuration property allows you to decide how many seconds the broker should wait before timing out its network communication between the databases in the configuration. A value of zero indicates that a network communication should never be timed out. This is not normally needed but it is used to avoid the Data Guard broker issues that may be caused by the low internal timeout (default 180 sec) for labs running in the virtual machine architecture.

12. Set the OperationTimeout property to the maximum value of 600.

```
DGMGRL> edit configuration set property OperationTimeout = 600;
Property "operationtimeout" updated
DGMGRL>
```

Note: The OperationTimeout configuration property specifies the maximum amount of time the broker should wait for health check, get monitorable property, and set property operations to complete. This is not normally needed but it is used to avoid the Data Guard broker issues that may be caused by the low internal timeout (default 30 sec) for labs running in the virtual machine architecture.

13. Upgrade the data protection mode to maximum availability.

Note: This data protection mode will be used in a later practice.

```
DGMGRL> edit configuration set protection mode as MaxAvailability;
Succeeded.
DGMGRL>
```

14. Enable the Data Guard broker configuration and show the results.

Note: You may or may not see warning messages like ORA-16525, ORA-16629, ORA-16665, ORA-16810, or ORA-16817 due to the lag and overall performance of the virtual machine environment. The warning messages will be cleared in a few minutes. **If your output does not match, do not proceed with labs until all issues have been resolved. Please consult with your instructor if you need to troubleshoot any issues.**

```
DGMGRL> enable configuration
Enabled.
DGMGRL> show configuration

Configuration - dg_config

Protection Mode: MaxAvailability
Members:
  eastdb - Primary database
  westdb - Physical standby database

Fast-Start Failover: DISABLED

Configuration Status:
SUCCESS  (status updated 75 seconds ago)

DGMGRL>
```

15. Exit all of your terminal sessions.

Practice 5-3: Monitoring Redo Transport Service and Redo Apply Service in RAC

Overview

In this practice, you will monitor the redo transport status on the primary database and the redo apply status in the physical standby database.

Terminal Sessions

- **Session 1:** ssh oracle@enode01
- **Session 2:** ssh oracle@wnode03



Tasks

1. **(Session 1)** From your classroom PC desktop, execute ssh oracle@enode01 to open a terminal session on enode01 as the oracle user.

```
[Classroom PC]$ ssh oracle@enode01
oracle@enode01's password: <oracle>
[oracle@enode01 ~]$
```

2. Make sure that you set up your environment variables correctly for the oracle user.

```
[oracle@enode01 ~]$ . oraenv
ORACLE_SID = [oracle]? eastdb
The Oracle base has been set to /u01/app/oracle
[oracle@enode01 ~]$ export ORACLE_SID=eastdb1
[oracle@enode01 ~]$
```

3. Launch the DGMGRL utility to show the Data Guard configuration.

```
[oracle@enode01 ~]$ dgmgrl
DGMGRL for Linux: Version 12.1.0.2.0 - 64bit Production
Copyright (c) 2000, 2012, Oracle. All rights reserved.
Welcome to DGMGRL, type "help" for information.
DGMGRL> connect sysdg/oracle_4U@eastdb
Connected as SYSDG.
DGMGRL> show configuration

Configuration - dg_config

Protection Mode: MaxAvailability
Members:
eastdb - Primary database
```

```

westdb - Physical standby database

Fast-Start Failover: DISABLED

Configuration Status:
SUCCESS      (status updated 46 seconds ago)
DGMGRL>

```

4. Using DGMGRL, show the primary database instances that are responsible for redo transport service. Exit DGMGRL.

Note: You may or may not see warning messages like ORA-16737 and ORA-16629 due to the lag and overall performance of the virtual machine environment. The warning messages will be cleared in a few minutes.

```

DGMGRL> show database eastdb

Database - eastdb

Role:          PRIMARY
Intended State: TRANSPORT-ON
Instance(s):
eastdb1
eastdb2

Database Status:
SUCCESS
DGMGRL> exit
[oracle@enode01 ~]$ 

```

5. Using SQL*Plus, determine the THREAD that has been assigned to each primary instance.

```

[oracle@enode01 ~]$ sqlplus / as sysdba
SQL> col name format a10
SQL> col value format a10
SQL> select inst_id, name, value from gv$parameter where name =
'thread';

INST_ID NAME        VALUE
----- -----
1       thread      1
2       thread      2
SQL>

```

6. Perform the following query to determine the most recently archived sequence number for each thread (thus by each instance). Your number may be different in the lab.

```

SQL> SELECT THREAD#, MAX(SEQUENCE#) FROM GV$ARCHIVED_LOG WHERE
RESETLOGS_CHANGE# = (SELECT MAX(RESETLOGS_CHANGE#) FROM
GV$ARCHIVED_LOG) GROUP BY THREAD#;

THREAD# MAX(SEQUENCE#)
-----
1           18
2            9
SQL>

```

7. Execute the alter system archive log current command to determine the most recently archived sequence number again.

```
SQL> alter system archive log current;
System altered.

SQL> SELECT THREAD#, MAX(SEQUENCE#) FROM GV$ARCHIVED_LOG WHERE
RESETLOGS_CHANGE# = (SELECT MAX(RESETLOGS_CHANGE#) FROM
GV$ARCHIVED_LOG) GROUP BY THREAD#;

THREAD# MAX(SEQUENCE#)
-----
1          19
2          10
```

8. **(Session 2)** Open another terminal window to connect to wnode03 as the oracle user.

```
[Classroom PC]$ ssh oracle@wnode03
oracle@wnode03's password: <oracle>
[oracle@wnode03 ~]$ . oraenv
ORACLE_SID = [oracle] ? westdb
The Oracle base has been set to /u01/app/oracle
[oracle@wnode03 ~]$ export ORACLE_SID=westdb1
[oracle@wnode03 ~]$
```

9. Using SQL*Plus, determine the most recent archived logfile received in the physical standby database.

Note: The sequence number is the same as the one in step 7. As you observed, both primary instances named eastdb1 and eastdb2 were able to transport the redo to the physical standby database.

```
[oracle@wnode03 ~]$ sqlplus / as sysdba
SQL> SELECT THREAD#, MAX(SEQUENCE#) FROM GV$ARCHIVED_LOG WHERE
RESETLOGS_CHANGE# = (SELECT MAX(RESETLOGS_CHANGE#) FROM
GV$ARCHIVED_LOG) GROUP BY THREAD#;

THREAD# MAX(SEQUENCE#)
-----
1          19
2          10

SQL>
```

10. Execute the following query to display Redo Apply (MRP) and redo transport status on a physical standby database.

```
SQL> select inst_id, thread#, client_process, process, sequence# from
gv$managed_standby where thread# in (1,2) order by 1;
```

INST_ID	THREAD#	CLIENT_P	PROCESS	SEQUENCE#
1	2	LGWR	RFS	13
1	2	ARCH	ARCH	11
1	2	ARCH	ARCH	9
1	2	ARCH	ARCH	12
2	2	N/A	MRPO	13
2	1	ARCH	ARCH	21
2	1	ARCH	ARCH	22
2	1	LGWR	RFS	23

2

2 ARCH

ARCH

10

```
9 rows selected.
```

```
SQL> exit
[oracle@wnode03 ~]$
```

Note: Your result might be slightly different. Depending on your environment, the output may show that a single physical instance or both physical standby instances named westdb1 (INST_ID=1) and westdb2 (INST_ID=2) have received redo from the primary database instances. However, only one instance named westdb2 is responsible for recovering the standby database as an apply instance in our example.

- In the same terminal session on wnode03, show the status of the physical standby database using DGMGRL.

Note: The current apply instance is westdb2. If you have no preference about which instance should be the apply instance in an Oracle RAC standby database, the broker randomly picks an apply instance. Your result may be different.

```
[oracle@wnode03 ~]$ dgmgrl
DGMGRL> connect sysdg/oracle_4U@westdb
Connected as SYSDG.
DGMGRL>
DGMGRL> show database westdb

Database - westdb

Role: PHYSICAL STANDBY
Intended State: APPLY-ON
Transport Lag: 0 seconds (computed 1 second ago)
Apply Lag: 0 seconds (computed 1 second ago)
Average Apply Rate: 8.00 KByte/s
Real Time Query: OFF
Instance(s):
  westdb1
  westdb2 (apply instance)

Database Status:
SUCCESS

DGMGRL>
```

- Switch the apply instance to westdb1 and make it as the preferred instance. Exit DGMGRL.

```
DGMGRL> edit database westdb set state='APPLY-ON' with apply instance =
westdb1;
Succeeded.
DGMGRL>
DGMGRL> show database westdb

Database - westdb

Role: PHYSICAL STANDBY
Intended State: APPLY-ON
Transport Lag: 0 seconds (computed 0 seconds ago)
Apply Lag: 0 seconds (computed 1 second ago)
```

```

Average Apply Rate: 8.00 KByte/s
Real Time Query: OFF
Instance(s):
westdb1 (apply instance)
westdb2

Database Status:
SUCCESS

DGMGRL> exit
[oracle@wnode03 ~]$
```

13. Restart the physical standby database to see if the broker chooses the preferred instance that has been set as the apply instance.

Note: The output shows that the preferred instance became the apply instance even after restarting the database instead of choosing an instance randomly.

```

[oracle@wnode03 ~]$ srvctl stop database -db westdb
[oracle@wnode03 ~]$ srvctl start database -db westdb -startoption mount
[oracle@wnode03 ~]$ dgmgrl
DGMGRL> connect sysdg/oracle_4U@westdb
Connected as SYSDG.
DGMGRL> show database westdb

Database - westdb

Role: PHYSICAL STANDBY
Intended State: APPLY-ON
Transport Lag: 0 seconds (computed 1 second ago)
Apply Lag: 0 seconds (computed 1 second ago)
Average Apply Rate: 580.00 KByte/s
Real Time Query: OFF
Instance(s):
westdb1 (apply instance)
westdb2

Database Status:
SUCCESS
DGMGRL>
```

14. Configure the Data Guard broker to wait for 10 seconds after detecting that the current apply instance has failed before initiating the apply instance failover.

```

DGMGRL> edit database westdb set property 'ApplyInstanceTimeout' =
'10';
Property "ApplyInstanceTimeout" updated
DGMGRL> show database westdb ApplyInstanceTimeout
ApplyInstanceTimeout = '10'
DGMGRL>
```

Note: After the broker initiates an apply instance failover, the broker selects a new apply instance according to the following rule: If the PreferredApplyInstance property indicates an instance that is currently running, select it as the new apply instance. Otherwise, pick a random instance that is currently running to be the new apply instance.

15. Review the current settings of the TransportLagThreshold and ApplyLagThreshold database properties. You can configure the Data Guard broker to generate a health check warning when a transport lag or an apply lag exceeds a user-defined value.

```
DGMGRL> show database eastdb TransportLagThreshold  
TransportLagThreshold = '0'  
DGMGRL> show database westdb TransportLagThreshold  
TransportLagThreshold = '0'  
DGMGRL> show database eastdb ApplyLagThreshold  
ApplyLagThreshold = '0'  
DGMGRL> show database westdb ApplyLagThreshold  
ApplyLagThreshold = '0'  
DGMGRL>
```

Note: The TransportLagThreshold configurable property can be used to generate a warning status for a logical, physical, or snapshot standby, or a far sync instance when the member's transport lag exceeds the value specified by the property. The property value is expressed in seconds. A value of 0 seconds results in no warnings being generated when a transport lag exists.

Note: The ApplyLagThreshold configurable property generates a warning status for a logical or physical standby when the member's apply lag exceeds the value specified by the property. The property value is expressed in seconds. A value of 0 seconds results in no warnings being generated when an apply lag exists.

16. Exit all of your terminal sessions.

Practice 5-4: Configuring Flashback Database on the Primary and Physical Standby Databases

Overview

In this practice, you will configure Flashback Database on the primary and standby databases, and verify that it has been enabled.

Terminal Sessions

- **Session 1:** ssh oracle@enode01
- **Session 2:** ssh oracle@wnode03



Tasks

1. **(Session 1)** Use a terminal window on enode01 connected as oracle with the environment variables set to eastdb1. Launch SQL*Plus and determine the current state of the Flashback Database.

```
[Classroom PC]$ ssh oracle@enode01
oracle@enode01's password: <oracle>
[oracle@enode01 ~]$ . oraenv
ORACLE_SID = [oracle]? eastdb
The Oracle base has been set to /u01/app/oracle
[oracle@enode01 ~]$ export ORACLE_SID=eastdb1
[oracle@enode01 ~]$ sqlplus / as sysdba
SQL> select flashback_on from v$database;

FLASHBACK_ON
-----
NO
SQL>
```

2. Verify that the primary database is in archive log mode, a prerequisite to Flashback Database.

```
SQL> archive log list
Database log mode          Archive Mode
Automatic archival        Enabled
Archive destination        USE_DB_RECOVERY_FILE_DEST
Oldest online log sequence 44
Next log sequence to archive 45
Current log sequence       45
SQL>
```

3. Verify that the fast recovery area has been configured for the primary database, a prerequisite to Flashback Database.

```
SQL> show parameter db_recovery

NAME                           TYPE        VALUE
-----
db_recovery_file_dest          string      +FRA
db_recovery_file_dest_size     big integer 7000M

SQL>
```

4. Determine the current amount of time in minutes for the Flashback window.

```
SQL> show parameter flashback

NAME                           TYPE        VALUE
-----
db_flashback_retention_target integer    1440

SQL>
```

5. Enable Flashback Database for the whole database.

```
SQL> alter database flashback on;
Database altered

SQL>
```

6. Verify that Flashback Database has been enabled.

```
SQL> select flashback_on from v$database;

FLASHBACK_ON
-----
YES

SQL> exit
[oracle@enode01 ~]$
```

7. **(Session 2)** Use a terminal window on wnode03 connected as oracle with the environment variables set to westdb1. Launch SQL*Plus and determine the current state of Flashback Database.

```
[Classroom PC]$ ssh oracle@wnode03
oracle@wnode03's password: <oracle>
[oracle@wnode03 ~]$ . oraenv
ORACLE_SID = [oracle]? westdb1
The Oracle base has been set to /u01/app/oracle
[oracle@wnode03 ~]$ sqlplus / as sysdba
SQL> select flashback_on from v$database;

FLASHBACK_ON
-----
NO

SQL>
```

8. Verify that the physical standby database is in archive log mode, a prerequisite to Flashback Database.

```
SQL> archive log list
Database log mode           Archive Mode
Automatic archival          Enabled
Archive destination          USE_DB_RECOVERY_FILE_DEST
Oldest online log sequence   0
Next log sequence to archive 0
Current log sequence        0
SQL>
```

9. Verify that the fast recovery area has been configured for the physical standby database, a prerequisite to Flashback Database.

```
SQL> show parameter db_recovery
NAME                           TYPE      VALUE
-----
db_recovery_file_dest          string    +FRA
db_recovery_file_dest_size     big integer 7000M
(SQL>
```

10. Determine the current amount of time in minutes for the Flashback window.

```
SQL> show parameter flashback
NAME                           TYPE      VALUE
-----
db_flashback_retention_target integer  1440
SQL>
```

11. Enable Flashback Database for the whole database. Note the error message that is returned.

```
SQL> alter database flashback on;
alter database flashback on
*
ERROR at line 1:
ORA-01153: an incompatible media recovery is active
SQL>
```

12. **(Session 1)** Return to the terminal session on enode01 connected as oracle with the environment variables set to eastdb1. Launch the DGMGRL utility and connect as the sysdg user with operating system authentication.

```
[oracle@enode01 ~]$ dgmgrl
DGMGRL> connect sysdg/oracle_4U@eastdb
Connected as SYSDG.
DGMGRL>
```

13. Stop the managed recovery mode for the physical standby database.

```
DGMGRL> edit database westdb set state='APPLY-OFF';
Succeeded.
DGMGRL>
```

14. **(Session 2)** Return to the SQL*Plus session on wnode03 connected to the 'westdb' physical standby database and enable Flashback Database a second time.

```
SQL> alter database flashback on;
Database altered.
SQL>
```

15. Verify that Flashback Database has been enabled.

```
SQL> select flashback_on from v$database;

FLASHBACK_ON
-----
YES

SQL> exit;
[oracle@wnode03 ~]$
```

16. **(Session 1)** Return to the terminal window on enode01 that is running DGMGRL and restart the managed recovery mode for the 'westdb' physical standby database. Exit DGMGRL when done.

```
DGMGRL> edit database westdb set state='APPLY-ON';
Succeeded.
DGMGRL> exit
[oracle@enode01 ~]$
```

17. Exit all of your terminal sessions.

Practice 5-5: Enabling Fast-Start Failover for RAC

Overview

In this practice, you will enable fast-start failover in RAC. After enabling fast-start failover, you will start the observer process. This practice is required to perform the "Application Continuity" practice at a later point.

Terminal Session

- Session 1: `ssh oracle@wnode03`

Tasks

- (Session 1) Use a terminal window on wnode03 connected as `oracle` with the environment variables set to `westdb01`. Launch the DGMGRL utility and connect as the `sysdg` user.

```
[Classroom PC]$ ssh oracle@wnode03
oracle@wnode03's password: <oracle>
[oracle@wnode03 ~]$ . oraenv
ORACLE_SID = [oracle]? westdb
The Oracle base has been set to /u01/app/oracle
[oracle@wnode03 ~]$ export ORACLE_SID=westdb1
[oracle@wnode03 ~]$ dgmgrl
DGMGRL> connect sysdg/oracle_4U@westdb
Connected as SYSDG.
[oracle@wnode03 ~]$
```

- Wait until the physical standby database is synchronized. It may take a few moments for the Lag for the standby database to clear.

Note: Do not continue with labs until the Apply Lag is cleared.

```
DGMGRL> show database westdb

Database - westdb

Role: PHYSICAL STANDBY
Intended State: APPLY-ON
Transport Lag: 0 seconds (computed 1 second ago)
Apply Lag: 0 seconds (computed 2 seconds ago)
Average Apply Rate: 53.00 KByte/s
Real Time Query: OFF
Instance(s):
    westdb1 (apply instance)
    westdb2

Database Status:
SUCCESS

DGMGRL>
```

- Display the current configuration and note the current state of fast-start failover.

```
DGMGRL> show configuration

Configuration - dg_config
```

```
Protection Mode: MaxAvailability
Members:
  eastdb - Primary database
  westdb - Physical standby database

Fast-Start Failover: DISABLED

Configuration Status:
  SUCCESS

DGMGRL>
```

4. Display a detailed status of the current fast-start failover settings.

```
DGMGRL> show fast_start failover

Fast-Start Failover: DISABLED

Threshold:          30 seconds
Target:            (none)
Observer:          (none)
Lag Limit:         30 seconds
Shutdown Primary: TRUE
Auto-reinstate:   TRUE
Observer Reconnect: (none)
Observer Override: FALSE

Configurable Failover Conditions
  Health Conditions:
    Corrupted Controlfile      YES
    Corrupted Dictionary       YES
    Inaccessible Logfile      NO
    Stuck Archiver             NO
    Datafile Offline           YES

  Oracle Error Conditions:
    (none)

DGMGRL>
```

5. Set up the `FastStartFailoverTarget` configuration property on the '`eastdb`' primary database to indicate the desired '`westdb`' target standby database.

```
DGMGRL> edit database eastdb set property FastStartFailoverTarget =
westdb;
Property "FastStartFailoverTarget" updated
DGMGRL>
```

6. Display the `FastStartFailoverTarget` configuration property for both the primary database and the physical standby database.

```
DGMGRL> show database eastdb faststartfailovertarget;
  FastStartFailoverTarget = 'westdb'
DGMGRL> show database westdb FastStartFailoverTarget;
  FastStartFailoverTarget = ''
DGMGRL>
```

7. Define the reciprocal fast-start failover target for when the 'westdb' physical standby database becomes the primary database. This would be set automatically by the broker.

```
DGMGRL> edit database westdb set property FastStartFailoverTarget =
eastdb;
Property "FastStartFailoverTarget" updated
DGMGRL>
```

8. Set the FSFO threshold via the broker configuration property FastStartFailoverThreshold to avoid a false fast-start failover to the physical standby when a RAC node fails.

```
DGMGRL> edit configuration set property FastStartFailoverThreshold=50;
Property "FastStartFailoverThreshold" updated
DGMGRL>
```

Note: The value of this property should be set to the time it takes to reconfigure the cluster (CSS Miscount, 30 sec by default) + 20 seconds as per the MAA white paper recommendation regarding RAC configurations and Fast Start Failover. This setting will need to be tested to determine how well it avoids false failover from occurring. If 50 seconds still sees a false failover occur, then larger values may need to be tested. Review the Oracle Support Note titled "*How do you avoid a false fast start failover to the physical standby when a RAC node fails (Doc ID 1319917.1)*."

9. Display the MaxConnections property for the primary and standby databases.

```
DGMGRL> show database westdb MaxConnections
MaxConnections = '4'
DGMGRL> show database eastdb MaxConnections
MaxConnections = '4'
DGMGRL>
```

10. Enable fast-start failover.

```
DGMGRL> enable fast_start failover;
Enabled.
DGMGRL>
```

11. Display a detailed status of the current fast-start failover settings.

```
DGMGRL> show fast_start failover;

Fast-Start Failover: ENABLED

Threshold:          50 seconds
Target:             westdb
Observer:           (none)
Lag Limit:          30 seconds (not in use)
Shutdown Primary:   TRUE
Auto-reinstate:     TRUE
Observer Reconnect: (none)
Observer Override:  FALSE

Configurable Failover Conditions
Health Conditions:
Corrupted Controlfile      YES
Corrupted Dictionary        YES
Inaccessible Logfile        NO
```

Stuck Archiver	NO
Datafile Offline	YES
Oracle Error Conditions:	
(none)	
DGMGRL>	

12. Start the observer process.

Note: The prompt will not return after starting the observer. Keep this terminal window open with the observer running in it.

DGMGRL> start observer
Observer started

13. **Do not close the terminal session.**

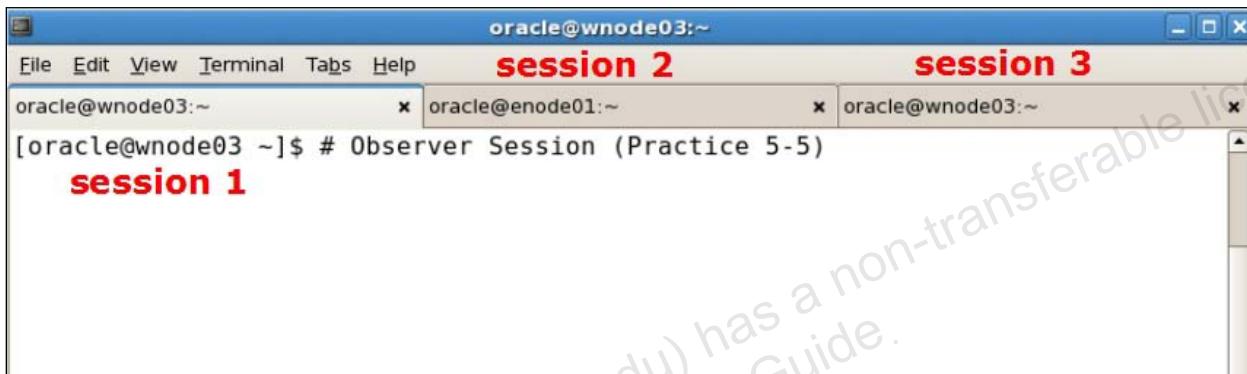
Practice 5-6: Testing Fast-Start Failover in RAC

Overview

In this practice, you will simulate a failure on the primary database and observe the automatic failover to the standby database.

Terminal Sessions

- **Session 1:** Observer Session (Practice 5-5)
- **Session 2:** ssh oracle@enode01
- **Session 3:** ssh oracle@wnode03



Tasks

1. **(Session 2)** Use a terminal window on enode01 connected as oracle with the environment variables set to 'eastbdb1'. Simulate a failure by issuing the `srvctl stop database -db eastdb -stopoption abort` command.

```
[Classroom PC]$ ssh oracle@enode01
oracle@enode01's password: <oracle>
[oracle@enode01 ~]$ . oraenv
ORACLE_SID = [oracle]? eastbdb
The Oracle base has been set to /u01/app/oracle
[oracle@enode01 ~]$ export ORACLE_SID=eastbdb1
[oracle@enode01 ~]$ srvctl stop database -db eastdb -stopoption abort
[oracle@enode01 ~]$
```

2. **(Session 1)** Observe the status output in the terminal session connected to wnode03 running the observer process from the previous lab step.

Note: It may take a minute before the failover is initiated.

```
(wnode03) DGMGRL> start observer
Observer started

17:50:49.15 Tuesday, September 22, 2015
Initiating Fast-Start Failover to database "westdb"...
Performing failover NOW, please wait...
Failover succeeded, new primary is "westdb"
17:52:13.69 Tuesday, September 22, 2015
```

3. **(Session 3)** Open another terminal window on wnode03 connected as oracle with the environment variables set to westdb1. Launch the DGMGRL utility and connect as the sysdg user with operating system authentication.

```
[Classroom PC]$ ssh oracle@wnode03
oracle@wnode03's password: <oracle>
[oracle@wnode03 ~]$ . oraenv
ORACLE_SID = [oracle]? westdb
The Oracle base has been set to /u01/app/oracle
[oracle@wnode03 ~]$ export ORACLE_SID=westdb1
[oracle@wnode03 ~]$ dgmgrl

DGMGRL for Linux: Version 12.1.0.2.0 - 64bit Production

Copyright (c) 2000, 2012, Oracle. All rights reserved.

Welcome to DGMGRL, type "help" for information.

DGMGRL> connect sysdg/oracle_4U@westdb
Connected as SYSDG.
DGMGRL>
DGMGRL>
```

4. Display the current configuration and note the current state of fast-start failover.

Note: You may or may not see warning messages like ORA-16525, ORA-16665, ORA-16810, or ORA-16817 due to the lag and overall performance of the virtual machine environment. The warning messages will be cleared in a few minutes. **Do not wait and move to the next step.**

```
DGMGRL> show configuration

Configuration - dg_config

Protection Mode: MaxAvailability
Members:
westdb - Primary database
      Warning: ORA-16817: unsynchronized fast-start failover
      configuration

eastdb - (*) Physical standby database (disabled)
      ORA-16661: the standby database needs to be reinstated

Fast-Start Failover: ENABLED

Configuration Status:
WARNING   (status updated 9 seconds ago)

DGMGRL>
```

5. **(Session 2)** Return to the terminal session on enode01 connected as oracle to mount the former primary database (the new physical standby database).

```
[oracle@enode01 ~]$ srvctl start database -db eastdb -startoption mount
[oracle@enode01 ~]$
```

6. **(Session 1)** Observe the output in the terminal session on wnode03 running the observer process.

```
18:01:22.40 Tuesday, September 22, 2015
Initiating reinstatement for database "eastdb"...
Reinstating database "eastdb", please wait...
Reinstatement of database "eastdb" succeeded
18:02:32.61 Tuesday, September 22, 2015
```

7. **(Session 3)** Return to the DGMGRL session running on wnode03 and display the configuration.

Note: You may or may not see warning messages like ORA-16525, ORA-16629, ORA-16665, ORA-16810, or ORA-16817 due to the lag and overall performance of the virtual machine environment. The warning messages will be cleared in a few minutes.

```
DGMGRL> show configuration

Configuration - dg_config

Protection Mode: MaxAvailability
Members:
westdb - Primary database
eastdb - (*) Physical standby database

Fast-Start Failover: ENABLED

Configuration Status:
SUCCESS      (status updated 25 seconds ago)

DGMGRL>
```

8. Wait until the physical standby database is synchronized. It may take a few moments for the Lag for the standby database to clear.

Note: Do not continue with labs until the Apply Lag is cleared.

```
DGMGRL> show database eastdb

Database - eastdb

Role:          PHYSICAL STANDBY
Intended State: APPLY-ON
Transport Lag: 0 seconds (computed 1 second ago)
Apply Lag:    0 seconds (computed 1 second ago)
Average Apply Rate: 40.00 KByte/s
Real Time Query: OFF
Instance(s):
eastdb1 (apply instance)
eastdb2

Database Status:
SUCCESS

DGMGRL>
```

9. Do not close the terminal sessions.

Practice 5-7: Performing Switchover to Reinstated Database in RAC and Disabling Fast-Start Failover

Overview

In this practice, you will perform a switchover to return the configuration to the state that it was before the failover.

Terminal Sessions (Same as Practice 5-6)

- **Session 1:** Observer Session (Practice 5-5)
- **Session 2:** ssh oracle@enode01
- **Session 3:** ssh oracle@wnode03

The screenshot shows a terminal window titled "oracle@wnode03:~". It has three tabs labeled "session 1", "session 2", and "session 3". The "session 3" tab is active, showing the command "[oracle@wnode03 ~]\$ # Observer Session (Practice 5-5)". The "session 1" tab is labeled "session 1" in red text. The "session 2" tab is labeled "session 2" in red text.

Tasks

1. **(Session 3)** Validate that the 'westdb' primary database is ready for switchover.

```
DGMGRL> validate database westdb
Database Role: Primary database
Ready for Switchover: Yes
DGMGRL>
```

2. Validate that the 'eastdb' standby database is ready for switchover.

```
DGMGRL> validate database eastdb
Database Role: Physical standby database
Primary Database: westdb
Ready for Switchover: Yes
Ready for Failover: Yes (Primary Running)

Transport-Related Property Settings:
Property      westdb Value          eastdb Value
RedoRoutes    (westdb:eastdb SYNC)  (eastdb:westdb SYNC)
DGMGRL>
```

3. Switch over to the 'eastdb' database.

Note: As of Oracle Database 12c Release 1, when you perform a switchover from an Oracle RAC primary database to a physical standby database, *it is no longer necessary* to shut down all but one primary database instance.

```
DGMGRL> switchover to eastdb
Performing switchover NOW, please wait...
Operation requires a connection to instance "eastdb2" on database
"eastdb"
Connecting to instance "eastdb2"...
Connected as SYSDBA.
New primary database "eastdb" is opening...
Oracle Clusterware is restarting database "westdb" ...
Switchover succeeded, new primary is "eastdb"
DGMGRL>
```

4. Display the resulting configuration.

Note: You may or may not see warning messages like ORA-16525, ORA-16665, ORA-16810, or ORA-16817 due to the lag and overall performance of the virtual machine environment. The warning messages will be cleared in a few minutes.

```
DGMGRL> show configuration

Configuration - dg_config

Protection Mode: MaxAvailability
Members:
  eastdb - Primary database
  westdb - Physical standby database

Fast-Start Failover: DISABLED

Configuration Status:
  SUCCESS

DGMGRL>
```

5. Stop the observer process.

Note: The prompt should now be returned in the terminal window that was connected to wnode03 running the observer process.

```
DGMGRL> stop observer
Done.
DGMGRL>
```

6. Disable fast-start failover.

```
DGMGRL> disable fast_start failover;
Disabled.
DGMGRL>
```

- Wait until the physical standby database is cleared. It may take a few moments for the Lag for the standby database to clear.

Note: Do not continue with labs until the Apply Lag is cleared.

```
DGMGRL> show database westdb
Database - westdb

Role: PHYSICAL STANDBY
Intended State: APPLY-ON
Transport Lag: 0 seconds (computed 1 second ago)
Apply Lag: 0 seconds (computed 1 second ago)
Average Apply Rate: 40.00 KByte/s
Real Time Query: OFF
Instance(s):
westdb1 (apply instance)
westdb2

Database Status:
SUCCESS

DGMGRL>
```

- Reset your environment for the next practices.

```
DGMGRL> EDIT CONFIGURATION SET PROTECTION MODE AS MaxPerformance;
Succeeded.
DGMGRL> EDIT DATABASE 'eastdb' SET PROPERTY 'RedoRoutes' =
'(eastdb:westdb ASYNC)';
Property "RedoRoutes" updated
DGMGRL> EDIT DATABASE 'westdb' SET PROPERTY 'RedoRoutes' =
'(westdb:eastdb ASYNC)';
Property "RedoRoutes" updated
DGMGRL> EDIT DATABASE eastdb SET PROPERTY MaxConnections=1;
Property "maxconnections" updated
DGMGRL> EDIT DATABASE westdb SET PROPERTY MaxConnections=1;
Property "maxconnections" updated
DGMGRL>
```

- Exit all of your terminal sessions.

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

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

Practices for Lesson 6: Managing Physical Standby Files After Structural Changes on the Primary Database

Chapter 6

Practices for Lesson 6: Overview

Practices Overview

In this practice, you perform several tasks to synchronize the physical standby database after making several structural changes to the primary database instead of recreating the physical standby database.

Practice Summary

- Practice 6-1: Creating an Archival Backup
- Practice 6-2: Using Transportable Tablespaces with a Physical Standby Database
- Practice 6-3: Moving the Location of a Data File Online
- Practice 6-4: Adding Online and Standby Log Groups

Practice 6-1: Creating an Archival Backup

Overview

In this practice, you create an archival backup, which is a backup that is not under your normal retention policies and it is not in your normal FRA destination. This `KEEP UNTIL TIME` database backup could be used for recovery if some of your later hands-on activities were to require it. Archival backups can be created with an open database or a mounted database.

Tasks

- From your classroom PC desktop, execute `ssh oracle@enode01` to open a terminal session on enode01 as the `oracle` user.

```
[Classroom PC]$ ssh oracle@enode01
oracle@enode01's password: <oracle>
[oracle@enode01 ~]$
```

- Make sure that you set up your environment variables correctly for the `oracle` user.

```
[oracle@enode01 ~]$ . oraenv
ORACLE_SID = [oracle]? eastdb
The Oracle base has been set to /u01/app/oracle
[oracle@enode01 ~]$ export ORACLE_SID=eastdb1
[oracle@enode01 ~]$
```

- Connect to the target database (`eastdb`) using RMAN.

```
[oracle@enode01 ~]$ rman target /
Recovery Manager: Release 12.1.0.2.0 - Production on Sat Mar 26
19:19:39 2016

Copyright (c) 1982, 2014, Oracle and/or its affiliates. All rights
reserved.

connected to target database: EASTDB (DBID=827681294)

RMAN>
```

- Create an archival backup with a `FORMAT` clause that creates the backup in the `/home/oracle/backup` directory.

```
RMAN> BACKUP DATABASE FORMAT '/home/oracle/backup/%U' TAG KEEPDB KEEP
UNTIL TIME 'SYSDATE+7' RESTORE POINT KEEPDB6_1;

Starting backup at 26-MAR-16

using channel ORA_DISK_1
backup will be obsolete on date 02-APR-16
archived logs required to recover from this backup will be backed up
channel ORA_DISK_1: starting full datafile backup set
channel ORA_DISK_1: specifying datafile(s) in backup set
input datafile file number=00005
name=+DATA/EASTDB/DATAFILE/example.279.905549011
input datafile file number=00001
name=+DATA/EASTDB/DATAFILE/system.290.905548821
input datafile file number=00003
name=+DATA/EASTDB/DATAFILE/sysaux.291.905548767
```

```

input datafile file number=00002
name=+DATA/EASTDB/DATAFILE/undotbs2.285.907524789
input datafile file number=00004
name=+DATA/EASTDB/DATAFILE/undotbs1.276.905548889
input datafile file number=00006
name=+DATA/EASTDB/DATAFILE/users.289.905548887
channel ORA_DISK_1: starting piece 1 at 26-MAR-16
channel ORA_DISK_1: finished piece 1 at 26-MAR-16
piece handle=/home/oracle/backup/0gr1fck_1_1 tag=KEEPDB comment=NONE
channel ORA_DISK_1: backup set complete, elapsed time: 00:05:17
current log archived

using channel ORA_DISK_1
backup will be obsolete on date 02-APR-16
archived logs required to recover from this backup will be backed up
channel ORA_DISK_1: starting archived log backup set
channel ORA_DISK_1: specifying archived log(s) in backup set
input archived log thread=2 sequence=9 RECID=35 STAMP=907532004
input archived log thread=1 sequence=25 RECID=36 STAMP=907532004
channel ORA_DISK_1: starting piece 1 at 26-MAR-16
channel ORA_DISK_1: finished piece 1 at 26-MAR-16
piece handle=/home/oracle/backup/0hrlfln6_1_1 tag=KEEPDB comment=NONE
channel ORA_DISK_1: backup set complete, elapsed time: 00:00:03

using channel ORA_DISK_1
backup will be obsolete on date 02-APR-16
archived logs required to recover from this backup will be backed up
channel ORA_DISK_1: starting full datafile backup set
channel ORA_DISK_1: specifying datafile(s) in backup set
including current SPFILE in backup set
channel ORA_DISK_1: starting piece 1 at 26-MAR-16
channel ORA_DISK_1: finished piece 1 at 26-MAR-16
piece handle=/home/oracle/backup/0irlflna_1_1 tag=KEEPDB comment=NONE
channel ORA_DISK_1: backup set complete, elapsed time: 00:00:01

using channel ORA_DISK_1
backup will be obsolete on date 02-APR-16
archived logs required to recover from this backup will be backed up
channel ORA_DISK_1: starting full datafile backup set
channel ORA_DISK_1: specifying datafile(s) in backup set
including current control file in backup set
channel ORA_DISK_1: starting piece 1 at 26-MAR-16
channel ORA_DISK_1: finished piece 1 at 26-MAR-16
piece handle=/home/oracle/backup/0jr1flnb_1_1 tag=KEEPDB comment=NONE
channel ORA_DISK_1: backup set complete, elapsed time: 00:00:03
Finished backup at 26-MAR-16

RMAN> exit
[oracle@enode01 ~]$

```

Note: Use the recovery steps for possible **later** use of the archival backup in Appendix (Practice 13-6 Case 6). In the Data Guard environments, the physical standby database must be synchronized.

5. Exit all of your terminal sessions.

Practice 6-2: Using Transportable Tablespaces with a Physical Standby Database

Overview

In this practice, you will transport a tablespace into the primary database with all the steps that it would take to transfer it across different platforms (**although in your training environment, you are using only one host on one platform**). After the tablespace is transported, you will perform the additional tasks to synchronize the physical standby database.

- **Task 1:** Transporting a tablespace into the primary database using RMAN
- **Task 2:** Performing actions on the physical standby database

Assumptions

- The database named gdscat is your source database, and eastdb is your destination database.
- The data file is transported into the **+DATA** diskgroup for eastdb using RMAN backup sets.
- The same file must be plugged into **+FRA** for westdb.
- You will use four terminal sessions during the practice activity.

Terminal Sessions

- **Session 1:** ssh oracle@enode01
- **Session 2:** ssh oracle@enode01
- **Session 3:** ssh oracle@wnode03
- **Session 4:** ssh grid@enode01



Task 1: Transporting a tablespace using RMAN on the primary database

1. **(Session 1)** Open a terminal window to connect to enode01 as the oracle user.

```
[Classroom PC]$ ssh oracle@enode01
oracle@enode01's password: <oracle>
[oracle@enode01 ~]$
```

2. Make sure that you set up your environment variables correctly for the gdscat database.

```
[oracle@enode01 ~]$ . oraenv
ORACLE_SID = [oracle]? gdscat
The Oracle base has been set to /u01/app/oracle
[oracle@enode01 ~]$
```

3. Set up for this practice by executing the setup_06_01.sh script from the /home/oracle/labs directory. In this script:

- A new tablespace named BARTBS and user are created.
- The user creates a table and populates it.

```
[oracle@enode01 ~]$ cd /home/oracle/labs
[oracle@enode01 labs]$ ./setup_06_01.sh
[oracle@enode01 labs]$
```

4. Start a SQL*Plus session on your gdscat source database and verify the prerequisites for transporting a tablespace across platforms.

- a. Log in as the SYS user and verify that the source database is in read/write mode.

```
[oracle@enode01 labs]$ sqlplus / as sysdba
SQL> SELECT NAME, LOG_MODE, OPEN_MODE, CURRENT_SCN FROM
V$DATABASE;
NAME      LOG_MODE      OPEN_MODE          CURRENT_SCN
-----  -----  -----  -----
GDSCAT    NOARCHIVELOG  READ WRITE        2071800
SQL>
```

- b. For performing cross-platform tablespace transport, you must know the exact name of the destination platform to which you are transporting data.

```
SQL> col platform_name format a30
SQL> SELECT PLATFORM_ID, PLATFORM_NAME, ENDIAN_FORMAT
FROM V$TRANSPORTABLE_PLATFORM
WHERE UPPER(PLATFORM_NAME) LIKE '%LINUX%';

PLATFORM_ID PLATFORM_NAME          ENDIAN_FORMAT
-----  -----  -----
10 Linux IA (32-bit)           Little
11 Linux IA (64-bit)           Little
9 IBM zSeries Based Linux     Big
13 Linux x86 64-bit           Little
18 IBM Power Based Linux      Big
SQL>
```

- c. Make the BARTBS tablespace read-only. This is required for the export of the tablespace metadata. Then exit.

```
SQL> ALTER TABLESPACE bartbs READ ONLY;
Tablespace altered.
SQL> exit
[oracle@enode01 labs]$
```

- In the same terminal session, start an RMAN session and connect to your GDSCAT source database as the target instance. Then, back up the source tablespace by using the BACKUP command with the TO PLATFORM clause. Use the DATAPUMP clause to indicate that an export dump file for the tablespaces must be created for the tablespace metadata.

Note: You are performing the "Cross-Platform Tablespace Transport with Backup Sets" method. You can also use the traditional implementation using Data Pump Export.

```
[oracle@enode01 labs]$ rman target ''/ as sysbackup"

Recovery Manager: Release 12.1.0.2.0 - Production on Wed Sep 30
13:12:59 2015

Copyright (c) 1982, 2014, Oracle and/or its affiliates. All rights
reserved.

connected to target database: GDSCAT (DBID=3218565307)

RMAN> BACKUP TO PLATFORM 'Linux x86 64-bit' FORMAT
'./home/oracle/backup/test.bck' DATAPUMP FORMAT
'./home/oracle/backup/test.dmp' TABLESPACE bartbs;

Starting backup at 30-SEP-15
using target database control file instead of recovery catalog
allocated channel: ORA_DISK_1
channel ORA_DISK_1: SID=31 device type=DISK
Running TRANSPORT_SET_CHECK on specified tablespaces
TRANSPORT_SET_CHECK completed successfully

Performing export of metadata for specified tablespaces...
EXPDP> Starting "SYSBACKUP"."TRANSPORT_EXP_GDSCAT_Foai":
EXPDP> Processing object type TRANSPORTABLE_EXPORT/PLUGTS_BLK
EXPDP> Processing object type TRANSPORTABLE_EXPORT/STATISTICS/MARKER
EXPDP> Processing object type
TRANSPORTABLE_EXPORT/POST_INSTANCE/PLUGTS_BLK
EXPDP> Master table "SYSBACKUP"."TRANSPORT_EXP_GDSCAT_Foai"
successfully loaded/unloaded
EXPDP>
*****
EXPDP> Dump file set for SYSBACKUP.TRANSPORT_EXP_GDSCAT_Foai is:
EXPDP>
/u01/app/oracle/product/12.1.0/dbhome_1/dbs/backup_tts_GDSCAT_17352.dmp
EXPDP>
*****
EXPDP> Datafiles required for transportable tablespace BARTBS:
EXPDP> /home/oracle/bartbs.dbf
EXPDP> Job "SYSBACKUP"."TRANSPORT_EXP_GDSCAT_Foai" successfully
completed at Wed Sep 30 13:15:06 2015 elapsed 0 00:00:33
Export completed

channel ORA_DISK_1: starting full datafile backup set
channel ORA_DISK_1: specifying datafile(s) in backup set
input datafile file number=00005 name=/home/oracle/bartbs.dbf
channel ORA_DISK_1: starting piece 1 at 30-SEP-15
channel ORA_DISK_1: finished piece 1 at 30-SEP-15
piece handle=/home/oracle/backup/test.bck tag=TAG20150930T131404
```

```

comment=NONE
channel ORA_DISK_1: backup set complete, elapsed time: 00:00:02
channel ORA_DISK_1: starting full datafile backup set
input Data Pump dump
file=/u01/app/oracle/product/12.1.0/dbhome_1/dbs/backup_tts_GDSCAT_1735
2.dmp
channel ORA_DISK_1: starting piece 1 at 30-SEP-15
channel ORA_DISK_1: finished piece 1 at 30-SEP-15
piece handle=/home/oracle/backup/test.dmp tag=TAG20150930T131404
comment=NONE
channel ORA_DISK_1: backup set complete, elapsed time: 00:00:01
Finished backup at 30-SEP-15
RMAN>

```

Note: Normally, after you disconnect from the source database, you move the backup sets and the Data Pump export dump file to the destination host with operating system utilities. In this training example, you do not need to do it because the source (GDSCAT) and destination database (EASTDB) are on the same host.

6. **(Session 2)** Open a new terminal window to connect to enode01 as oracle. Then start an RMAN session and connect to your primary database named eastdb1 as the target instance.

```

[Classroom PC]$ ssh oracle@enode01
oracle@enode01's password: <oracle>
[oracle@enode01 ~]$ . oraenv
ORACLE_SID = [oracle]? eastdb
The Oracle base has been set to /u01/app/oracle
[oracle@enode01 ~]$ export ORACLE_SID=eastdb1
[oracle@enode01 ~]$ rman target ''/ as sysbackup''
Recovery Manager: Release 12.1.0.2.0 - Production on Wed Sep 30
13:20:46 2015

Copyright (c) 1982, 2014, Oracle and/or its affiliates. All rights
reserved.

connected to target database: EASTDB (DBID=813839866)

RMAN>

```

7. Use the RESTORE command with the FOREIGN TABLESPACE clause. The FORMAT clause specifies the file destination. Use the DUMP FILE FROM BACKUPSET clause to restore the metadata from the dump file, which is required to plug the tablespace into the destination database.

Note: You are performing the "**Cross-Platform Tablespace Transport with Backup Sets**" method. You can also use the traditional implementation.

```

RMAN> RESTORE FOREIGN TABLESPACE bartbs FORMAT
'+DATA/eastdb/bartbs.dbf' FROM BACKUPSET
'/home/oracle/backup/test.bck' DUMP FILE FROM BACKUPSET
'/home/oracle/backup/test.dmp';

Starting restore at 30-SEP-15
using target database control file instead of recovery catalog
allocated channel: ORA_DISK_1
channel ORA_DISK_1: SID=65 instance=eastdb1 device type=DISK

```

```

channel ORA_DISK_1: starting datafile backup set restore
channel ORA_DISK_1: specifying datafile(s) to restore from backup set
channel ORA_DISK_1: restoring all files in foreign tablespace BARTBS
channel ORA_DISK_1: reading from backup piece
/home/oracle/backup/test.bck
channel ORA_DISK_1: restoring foreign file 5 to +DATA/eastdb/bartbs.dbf
channel ORA_DISK_1: foreign piece handle=/home/oracle/backup/test.bck
channel ORA_DISK_1: restored backup piece 1
channel ORA_DISK_1: restore complete, elapsed time: 00:00:02
channel ORA_DISK_1: starting datafile backup set restore
channel ORA_DISK_1: specifying datafile(s) to restore from backup set
channel ORA_DISK_1: restoring Data Pump dump file to
/u01/app/oracle/product/12.1.0/dbhome_1/dbs/backup_tts_EASTDB_54272.dmp
channel ORA_DISK_1: reading from backup piece
/home/oracle/backup/test.dmp
channel ORA_DISK_1: foreign piece handle=/home/oracle/backup/test.dmp
channel ORA_DISK_1: restored backup piece 1
channel ORA_DISK_1: restore complete, elapsed time: 00:00:01

Performing import of metadata...
IMPDP> Master table "SYSBACKUP"."TSPITR_IMP_EASTDB_wknf"
successfully loaded/unloaded
IMPDP> Starting "SYSBACKUP"."TSPITR_IMP_EASTDB_wknf":
IMPDP> Processing object type TRANSPORTABLE_EXPORT/PLUGTS_BLK
IMPDP> Processing object type
TRANSPORTABLE_EXPORT/POST_INSTANCE/PLUGTS_BLK
IMPDP> Job "SYSBACKUP"."TSPITR_IMP_EASTDB_wknf" successfully
completed at Wed Sep 30 13:23:51 2015 elapsed 0 00:00:31
Import completed

Finished restore at 30-SEP-15
RMAN>

```

- In the same terminal session, confirm that the tablespace exists in your destination database, which is your primary database. Then exit.

```

RMAN> select tablespace_name, status from dba_tablespaces;

TABLESPACE_NAME          STATUS
-----
SYSTEM                  ONLINE
SYSAUX                 ONLINE
UNDOTBS1                ONLINE
TEMP                   ONLINE
USERS                  ONLINE
UNDOTBS2                ONLINE
EXAMPLE                 ONLINE
BARTBS                  READ ONLY

8 rows selected
RMAN> exit
[oracle@enode01 ~]$

```

Note: So far, you have performed the cross-platform transportable tablespace using backup sets successfully. However, in order to move the tablespace into a primary database when a physical standby is being used, you have to perform additional tasks.

Task 2: Performing Actions on the Physical Standby Database

9. **(Session 3)** Open a new terminal window on wnode03 to connect to the westdb1 physical standby instance.

```
[Classroom PC]$ ssh oracle@wnode03
oracle@wnode03's password: <oracle>
[oracle@wnode03 ~]$ . oraenv
ORACLE_SID = [oracle]? westdb
The Oracle base has been set to /u01/app/oracle
[oracle@wnode03 ~]$ export ORACLE_SID=westdb1
[oracle@wnode03 ~]$
```

10. Using SQL*Plus, connect to the westdb1 instance as SYSDBA. Then attempt to open the instance in READ ONLY mode to check if the tablespace has been transported to the physical standby database successfully or not.

```
[oracle@wnode03 ~]$ sqlplus / as sysdba
SQL> alter database open read only;
alter database open read only
*
ERROR at line 1:
ORA-10458: standby database requires recovery
ORA-01157: cannot identify/lock data file 7 - see DBWR trace file
ORA-01110: data file 7: '+DATA/eastdb/bartbs.dbf'

SQL>
```

Note: As you can see above, the data file was not copied over to the physical standby database. So, you must copy the tablespace datafile to the physical standby database manually as well when performing the transportable tablespace operation in the primary database. This is true even when STANDBY_FILE_MANAGEMENT is set to AUTO.

11. Set STANDBY_FILE_MANAGEMENT to MANUAL.

Note: The STANDBY_FILE_MANAGEMENT initialization parameter must be set to MANUAL before renaming the data files, and should be reset to the previous value after renaming the data files.

```
SQL> show parameter standby_file_management

NAME                      TYPE        VALUE
-----
standby_file_management    string      AUTO

SQL> alter system set standby_file_management=MANUAL sid='*';
System altered.

SQL>
```

12. **(Session 4)** Open a terminal session on enode01 as the grid user. Make sure that you set up your environment variables correctly for the grid user.

```
[Classroom PC]$ ssh grid@enode01
grid@enode01's password: <oracle>
[grid@enode01 ~]$ . oraenv
ORACLE_SID = [grid]? +ASM1
The Oracle base has been set to /u01/app/grid
[grid@enode01 ~]$
```

13. Copy the data file to your desired location (+FRA). For the purpose of this demonstration, you will copy the data file to a different location from the primary database.

Note: You can use different methods to copy the data file such as FTP proxy, ASM cp command, and DBMS_FILE_TRANSFER.

```
[grid@enode01 ~]$ asmcmd cp +data/eastdb/bartbs.dbf /tmp/bartbs.dbf
copying +data/eastdb/bartbs.dbf -> /tmp/bartbs.dbf
[grid@enode01 ~]$ scp /tmp/bartbs.dbf wnode03:/tmp/
grid@wnode03's password: <oracle>
bartbs.dbf                                         100%    10MB   10.0MB/s   00:00

[grid@enode01 ~]$ ssh wnode03
grid@wnode03's password: <oracle>
[grid@wnode03 ~]$ . oraenv
ORACLE_SID = [grid]? +ASM1
The Oracle base has been set to /u01/app/grid
[grid@wnode03 ~]$ asmcmd cp /tmp/bartbs.dbf +fra/westdb/bartbs.dbf
copying /tmp/bartbs.dbf -> +fra/westdb/bartbs.dbf
[grid@wnode03 ~]$
```

14. **(Session 1)** Return to the terminal session on enode01 to allow read/write operations on the BARTBS tablespace. Then exit RMAN.

Note: This step could be done along with the earlier step if you copied the source data file into both the primary and standby databases. However, the source data file is copied into the primary and standby databases separately for demonstration purposes.

```
(GDSCAT) RMAN> alter tablespace bartbs READ WRITE;
Statement processed
(GDSCAT) RMAN>
```

15. **(Session 3)** Return to the terminal session on wnode03 to determine the value of the DB_FILE_NAME_CONVERT parameter.

```
SQL> show parameter db_file_name_convert
NAME          TYPE        VALUE
-----
db_file_name_convert  string
SQL>
```

16. Issue the ALTER DATABASE RENAME FILE statement to rename the data files. Do this after Redo Apply has failed to apply the redo generated by plugging the tablespace into the primary database like step 10. Remember the following:

- DB_FILE_NAME_CONVERT has *not* been configured.
- The path name of the data file is not the same on the primary (+DATA) and standby databases (+FRA).

```
SQL> alter database rename file '+DATA/eastdb/bartbs.dbf' to
'+FRA/westdb/bartbs.dbf';
Database altered.
SQL>
```

17. Reset STANDBY_FILE_MANAGEMENT to AUTO.

```
SQL> alter system set standby_file_management=AUTO sid='*';
System altered.
SQL>
```

18. Start Redo Apply Service.

```
SQL> alter database recover managed standby database disconnect;
Database altered.
SQL>
```

19. Open the physical standby database instance named westdb1 in READ ONLY mode.

```
SQL> alter database open read only;
Database altered.
SQL>
```

20. Confirm that the tablespace named BARTBS has been transported into the physical standby database successfully.

```
SQL> select tablespace_name, status from dba tablespaces;

TABLESPACE_NAME          STATUS
-----
SYSTEM                  ONLINE
SYSAUX                 ONLINE
UNDOTBS1                ONLINE
TEMP                   ONLINE
USERS                  ONLINE
UNDOTBS2                ONLINE
EXAMPLE                 ONLINE
BARTBS                 READ ONLY

8 rows selected.

SQL>
```

21. **(Session 2)** Return to the terminal session on enode01 that has been configured to connect to eastdb1. Make the BARTBS tablespace READ WRITE.

```
[oracle@enode01 ~]$ sqlplus / as sysdba

SQL> alter tablespace bartbs READ WRITE;
Tablespace altered.

SQL>
```

22. **(Session 3)** Return to the terminal session on wnode03 to determine the status of the BARTBS tablespace. Then exit.

Note: It may take a few seconds to bring the BARTBS tablespace ONLINE.

```
SQL> select tablespace_name, status from dba tablespaces;

TABLESPACE_NAME          STATUS
-----
SYSTEM                  ONLINE
SYSAUX                 ONLINE
```

```

UNDOTBS1           ONLINE
TEMP              ONLINE
USERS             ONLINE
UNDOTBS2          ONLINE
EXAMPLE           ONLINE
BARTBS            ONLINE

8 rows selected.

SQL> exit
[oracle@wnode03 ~]$
```

23. Mount the physical standby database. Make sure that the managed recovery process is running.

```
[oracle@wnode03 ~]$ srvctl stop database -db westdb
[oracle@wnode03 ~]$ srvctl start database -db westdb -startoption mount
[oracle@wnode03 ~]$ pgrep -lf mrp
5547 ora_mrp0_westdb
[oracle@wnode03 ~]$
```

24. Determine the status of the Data Guard configuration.

Note: You may or may not see warning messages like ORA-16525, ORA-16665, ORA-16810, or ORA-16817 due to the lag and overall performance of the virtual machine environment. The warning messages will be cleared in a few minutes. Make sure that your Data Guard configuration status is still fine.

```
[oracle@wnode03 ~]$ dgmgrl
DGMGRL> connect sysdg/oracle_4U@westdb
Connected as SYSDG.
DGMGRL> show configuration

Configuration - dg_config

Protection Mode: MaxPerformance
Members:
  eastdb - Primary database
  westdb - Physical standby database

Fast-Start Failover: DISABLED

Configuration Status:
  SUCCESS  (status updated 17 seconds ago)

DGMGRL>
```

25. Wait until the physical standby database is synchronized. It may take a few moments for the Lag for the standby database to clear.

Note: Do not continue with labs until the Apply Lag is cleared.

```
DGMGRL> show database westdb

Database - westdb

Role: PHYSICAL STANDBY
Intended State: APPLY-ON
Transport Lag: 0 seconds (computed 1 second ago)
Apply Lag: 0 seconds (computed 1 second ago)
Average Apply Rate: 40.00 KByte/s
Real Time Query: OFF
Instance(s):
    westdb1 (apply instance)
    westdb2

Database Status:
SUCCESS

DGMGRL>
```

26. Exit all of your terminal sessions.

Practice 6-3: Moving the Location of a Data File Online

Overview

In this practice, you will move a data file to another location online on the primary database. After the data file is moved, you will perform additional tasks to synchronize the physical standby database.

- **Task 1:** Moving the location of a data file online on the primary database
- **Task 2:** Performing actions on the physical standby database with Active Data Guard

Terminal Sessions

- **Session 1:** ssh oracle@enode01
- **Session 2:** ssh oracle@wnode03



Task 1: Moving the location of a data file online on the primary database

1. **(Session 1)** Open a terminal window on enode01 as the oracle user.

```
[Classroom PC]$ ssh oracle@enode01
oracle@enode01's password: <oracle>
[oracle@enode01 ~]$
```

2. Make sure that you set up your environment variables correctly for the eastdb1 instance.

```
[oracle@enode01 ~]$ . oraenv
ORACLE_SID = [oracle]? eastdb
The Oracle base has been set to /u01/app/oracle
[oracle@enode01 ~]$ export ORACLE_SID=eastdb1
[oracle@enode01 ~]$
```

3. Create a tablespace ONLINE_TBS and find the newly created datafile from the v\$logfile view.

```
[oracle@enode01 ~]$ sqlplus / as sysdba
SQL> create tablespace ONLINE_TBS datafile
  '+data/eastdb/online_tbs01.dbf' size 10m;

Tablespace created.

SQL> col name format a60
```

```
SQL> select name from v$datafile;

NAME
-----
+DATA/EASTDB/DATAFILE/system.290.891707387
+DATA/EASTDB/DATAFILE/undotbs2.285.891710019
+DATA/EASTDB/DATAFILE/sysaux.291.891707331
+DATA/EASTDB/DATAFILE/undotbs1.276.891707463
+DATA/EASTDB/DATAFILE/example.279.891707573
+DATA/EASTDB/DATAFILE/users.289.891707461
+DATA/eastdb/bartbs.dbf
+DATA/eastdb/online_tbs01.dbf

8 rows selected.
```

SQL>

- Move the data file named +DATA/eastdb/online_tbs01.dbf to the +FRA/eastdb/online_tbs01.dbf destination without taking it offline.

Note: The "Online Move Data File" feature in Oracle Database 12c provides the capability to move an online data file from one kind of storage system to another while the database is open and accessing the file.

```
SQL> ALTER DATABASE MOVE DATAFILE '+data/eastdb/online_tbs01.dbf' TO
  '+fra/eastdb/online_tbs01.dbf';
Database altered.

SQL>
```

- Determine the new location of the data file.

```
SQL> select name from v$datafile;

NAME
-----
+DATA/EASTDB/DATAFILE/system.290.891707387
+DATA/EASTDB/DATAFILE/undotbs2.285.891710019
+DATA/EASTDB/DATAFILE/sysaux.291.891707331
+DATA/EASTDB/DATAFILE/undotbs1.276.891707463
+DATA/EASTDB/DATAFILE/example.279.891707573
+DATA/EASTDB/DATAFILE/users.289.891707461
+DATA/eastdb/bartbs.dbf
+FRA/eastdb/online_tbs01.dbf

8 rows selected.
```

SQL>

Note: When you move or rename a data file in the primary database, the changes are not propagated to the standby database. Therefore, if you want to rename the same data files on the standby database, you must manually make the equivalent modifications on the standby database because the modifications are not performed automatically, even if the STANDBY_FILE_MANAGEMENT initialization parameter is set to AUTO.

Task 2: Performing actions on the physical standby database with Active Data Guard

6. (**Session 2**) Open a terminal session on wnode03 as the oracle user. Make sure that you set up your environment variables correctly for the westdb1 instance.

```
[Classroom PC]$ ssh oracle@wnode03
oracle@wnode03's password: <oracle>
[oracle@wnode03 ~]$ . oraenv
ORACLE_SID = [oracle] ? westdb
The Oracle base has been set to /u01/app/oracle
[oracle@wnode03 ~]$ export ORACLE_SID=westdb1
[oracle@wnode03 ~]$
```

7. Determine the value of STANDBY_FILE_MANAGEMENT.

NAME	TYPE	VALUE
standby_file_management	string	AUTO

8. Open the physical standby database in READ ONLY mode to determine the location of the ONLINE_TBS datafile in the physical standby database. **Note:** Even though STANDBY_FILE_MANAGEMENT has been set to AUTO, the tablespace named ONLINE_TBS is still located in the same location.

```
SQL> alter database open read only;
Database altered.
SQL> col name format a60
SQL> select name from v$datafile;

NAME
-----
+DATA/WESTDB/DATAFILE/system.284.891712419
+DATA/WESTDB/DATAFILE/undotbs2.286.891712435
+DATA/WESTDB/DATAFILE/sysaux.287.891712431
+DATA/WESTDB/DATAFILE/undotbs1.276.891712539
+DATA/WESTDB/DATAFILE/example.281.891712403
+DATA/WESTDB/DATAFILE/users.294.891712603
+FRA/westdb/bartbs.dbf
+DATA/WESTDB/DATAFILE/online_tbs.299.891791057

8 rows selected.
SQL>
```

9. Set STANDBY_FILE_MANAGEMENT to MANUAL to move the data file manually in the physical standby database.

```
SQL> alter system set standby_file_management=MANUAL sid='*';
System altered.
SQL>
```

10. Move the data file online in the physical standby database. Please use your data file name obtained in step 8 instead of the data file name shown in our example.

Note: The online data file move feature:

- Requires the Active Data Guard option
- Cannot be executed on the physical standby while standby recovery is running in a mounted, but not open instance

```
SQL> alter database move datafile
' +DATA/WESTDB/DATAFILE/online_tbs.299.891791057' to
' +FRA/WESTDB/online_tbs01.dbf';
Database altered.

SQL>
```

11. Check if the data file has been moved to the new location or not.

```
SQL> select name from v$datafile;

NAME
-----
+DATA/WESTDB/DATAFILE/system.284.891712419
+DATA/WESTDB/DATAFILE/undotbs2.286.891712435
+DATA/WESTDB/DATAFILE/sysaux.287.891712431
+DATA/WESTDB/DATAFILE/undotbs1.276.891712539
+DATA/WESTDB/DATAFILE/example.281.891712403
+DATA/WESTDB/DATAFILE/users.294.891712603
+FRA/westdb/bartbs.dbf
+FRA/WESTDB/online_tbs01.dbf

8 rows selected.

SQL>
```

12. Reset STANDBY_FILE_MANAGEMENT to AUTO. Then, exit.

```
SQL> alter system set standby_file_management=AUTO sid='*';
System altered.

SQL> exit
[oracle@wnode03 ~]$
```

13. If the real-time query feature is not used, restart database mount.

```
[oracle@wnode03 ~]$ srvctl stop database -db westdb
[oracle@wnode03 ~]$ srvctl start database -db westdb -startoption mount
[oracle@wnode03 ~]$
```

14. Check if the managed recovery process (MRP) is running.

```
[oracle@wnode03 ~]$ pgrep -lf mrp
1505 ora_mrp0_westdb1
[oracle@wnode03 ~]$
```

15. Determine the status of the Data Guard configuration.

Note: You may or may not see warning messages like ORA-16525, ORA-16665, ORA-16810, or ORA-16817 due to the lag and overall performance of the virtual machine environment. The warning messages will be cleared in a few minutes. Make sure that your Data Guard configuration status is still fine.

```
[oracle@wnode03 ~]$ dgmgrl
DGMGRL> connect sysdg/oracle_4U@westdb
Connected as SYSDG.
DGMGRL> show configuration

Configuration - dg_config

Protection Mode: MaxPerformance
Members:
  eastdb - Primary database
  westdb - Physical standby database

Fast-Start Failover: DISABLED

Configuration Status:
SUCCESS (status updated 17 seconds ago)

DGMGRL>
```

16. Wait until the physical standby database is synchronized. It may take a few moments for the Lag for the standby database to clear.

Note: Do not continue with labs until the Apply Lag is cleared.

```
DGMGRL> show database westdb

Database - westdb

Role: PHYSICAL STANDBY
Intended State: APPLY-ON
Transport Lag: 0 seconds (computed 1 second ago)
Apply Lag: 0 seconds (computed 1 second ago)
Average Apply Rate: 40.00 KByte/s
Real Time Query: OFF
Instance(s):
  westdb1 (apply instance)
  westdb2

Database Status:
SUCCESS

DGMGRL>
```

17. Exit all of your terminal sessions.

Practice 6-4: Adding Online and Standby Log Groups

Overview

In this practice, you will add online and standby redo groups to the primary database and synchronize the physical standby database accordingly.

- **Task 1:** Adding online and standby log groups in the primary database
- **Task 2:** Performing actions on the physical standby database

Terminal Sessions

- **Session 1:** ssh oracle@enode01
- **Session 2:** ssh oracle@wnode03



Task 1: Adding online and standby log groups in the primary database

1. **(Session 1)** Open a terminal window on enode01 as the oracle user to connect to the eastdb1 instance.

```
[Classroom PC]$ ssh oracle@enode01
oracle@enode01's password: <oracle>
[oracle@enode01 ~]$ . oraenv
ORACLE_SID = [oracle]? eastdb
The Oracle base has been set to /u01/app/oracle
[oracle@enode01 ~]$ export ORACLE_SID=eastdb1
[oracle@enode01 ~]$
```

2. Determine the current online and standby redo files in the primary database.

```
[oracle@enode01 ~]$ sqlplus / as sysdba
SQL> SELECT THREAD#, GROUP#, SEQUENCE#,BYTES/1024/1024 "SIZE MB",
ARCHIVED,STATUS FROM V$LOG order by 1;

        THREAD#      GROUP#    SEQUENCE#      SIZE MB ARC STATUS
-----  -----  -----  -----
          1          1          7          50  NO  CURRENT
          1          2          6          50  YES INACTIVE
          2          3          7          50  NO  CURRENT
          2          4          6          50  YES INACTIVE

SQL> SELECT THREAD#, GROUP#, SEQUENCE#,BYTES/1024/1024 "SIZE MB",
ARCHIVED,STATUS FROM V$STANDBY_LOG order by 1;
```

THREAD#	GROUP#	SEQUENCE#	SIZE MB	ARC	STATUS
1	5	0	50	YES	UNASSIGNED
1	6	0	50	YES	UNASSIGNED
1	7	0	50	YES	UNASSIGNED
2	8	0	50	YES	UNASSIGNED
2	9	0	50	YES	UNASSIGNED
2	10	0	50	YES	UNASSIGNED

6 rows selected.

SQL>

3. **(Session 2)** Open a new terminal window on wnode03 as the oracle user to connect to the westdb1 instance.

```
[Classroom PC]$ ssh oracle@wnode03
oracle@wnode03's password: <oracle>
[oracle@wnode03 ~]$ . oraenv
ORACLE_SID = [oracle]? westdb
The Oracle base has been set to /u01/app/oracle
[oracle@wnode03 ~]$ export ORACLE_SID=westdb1
[oracle@wnode03 ~]$
```

4. Determine the current online and standby redo files in the physical standby database.

```
[oracle@wnode03 ~]$ sqlplus / as sysdba
SQL> SELECT THREAD#, GROUP#, SEQUENCE#,BYTES/1024/1024 "SIZE MB",
ARCHIVED,STATUS FROM V$LOG order by 1;

THREAD#      GROUP#      SEQUENCE#      SIZE MB      ARC      STATUS
-----      -----      -----      -----      -----      -----
1            1            0            50      YES      UNUSED
1            2            0            50      YES      UNUSED
2            3            0            50      YES      UNUSED
2            4            0            50      YES      UNUSED

SQL> SELECT THREAD#, GROUP#, SEQUENCE#,BYTES/1024/1024 "SIZE MB",
ARCHIVED,STATUS FROM V$STANDBY_LOG order by 1;

THREAD#      GROUP#      SEQUENCE#      SIZE MB      ARC      STATUS
-----      -----      -----      -----      -----      -----
1            5            0            50      NO       UNASSIGNED
1            6            7            50      YES       ACTIVE
1            7            0            50      YES       UNASSIGNED
2            8            0            50      NO       UNASSIGNED
2            9            7            50      YES       ACTIVE
2            10           0            50      YES      UNASSIGNED

6 rows selected.

SQL>
```

5. Using SQL*Plus, determine the current setting of STANDBY_FILE_MANAGEMENT.

```
SQL> show parameter standby_file_management

NAME                      TYPE        VALUE
-----
standby_file_management    string     AUTO

SQL>
```

6. **(Session 1)** Return to the terminal session on enode01 to add the online and standby redo files in the primary database.

```
SQL> ALTER DATABASE ADD LOGFILE THREAD 1 GROUP 11 size 50M;
Database altered.

SQL> ALTER DATABASE ADD LOGFILE THREAD 2 GROUP 12 size 50M;
Database altered.

SQL> ALTER DATABASE ADD STANDBY LOGFILE THREAD 1 GROUP 13 size 50M;
Database altered.

SQL> ALTER DATABASE ADD STANDBY LOGFILE THREAD 2 GROUP 14 size 50M;
Database altered.

SQL>
```

7. Check the result of step 6 in the primary database.

```
SQL> SELECT THREAD#, GROUP#, SEQUENCE#,BYTES/1024/1024 "SIZE MB",
ARCHIVED,STATUS FROM V$LOG order by 1;

THREAD#  GROUP#  SEQUENCE#  SIZE MB ARC STATUS
-----
1        11      0          50 YES UNUSED
1        2       6          50 YES INACTIVE
1        1       7          50 NO  CURRENT
2        4       6          50 YES INACTIVE
2        12      0          50 YES UNUSED
2        3       7          50 NO  CURRENT

6 rows selected.

SQL> SELECT THREAD#, GROUP#, SEQUENCE#,BYTES/1024/1024 "SIZE MB",
ARCHIVED,STATUS FROM V$STANDBY_LOG order by 1;

THREAD#  GROUP#  SEQUENCE#  SIZE MB ARC STATUS
-----
1        13      0          50 YES UNASSIGNED
1        7       0          50 YES UNASSIGNED
1        6       0          50 YES UNASSIGNED
1        5       0          50 YES UNASSIGNED
2        10      0          50 YES UNASSIGNED
2        9       0          50 YES UNASSIGNED
2        14      0          50 YES UNASSIGNED
2        8       0          50 YES UNASSIGNED

8 rows selected.

SQL>
```

8. **(Session 2)** Return to the terminal session on wnode03 to check if the changes made to the primary database have been applied to the physical standby database.

Note: The result indicates that nothing has changed in the physical standby database.

```
SQL> SELECT THREAD#, GROUP#, SEQUENCE#,BYTES/1024/1024 "SIZE MB",
ARCHIVED,STATUS FROM V$LOG order by 1;
```

THREAD#	GROUP#	SEQUENCE#	SIZE MB	ARC	STATUS
1	1	0	50	YES	UNUSED
1	2	0	50	YES	UNUSED
2	3	0	50	YES	UNUSED
2	4	0	50	YES	UNUSED

```
SQL> SELECT THREAD#, GROUP#, SEQUENCE#,BYTES/1024/1024 "SIZE MB",
ARCHIVED,STATUS FROM V$STANDBY_LOG order by 1;
```

THREAD#	GROUP#	SEQUENCE#	SIZE MB	ARC	STATUS
1	5	0	50	NO	UNASSIGNED
1	6	7	50	YES	ACTIVE
1	7	0	50	YES	UNASSIGNED
2	8	0	50	NO	UNASSIGNED
2	9	7	50	YES	ACTIVE
2	10	0	50	YES	UNASSIGNED

6 rows selected.

```
SQL>
```

Task 2: Performing actions on the physical standby database

9. Attempt to add the online redo file and standby log file to synchronize the physical standby database with the primary database.

Note: You cannot add or drop log files while the managed recovery process (MRP) is running.

```
SQL> ALTER DATABASE ADD LOGFILE THREAD 1 GROUP 11 size 50M;
ALTER DATABASE ADD LOGFILE THREAD 1 GROUP 11 size 50M
*
ERROR at line 1:
ORA-01156: recovery or flashback in progress may need access to files

SQL>
```

10. Stop the managed recovery process.

```
SQL> alter database recover managed standby database cancel;
Database altered.

SQL>
```

11. Attempt to add the online redo file and standby log file again.

Note: You cannot add or drop log files when STANDBY_FILE_MANAGEMENT is set to AUTO.

```
SQL> ALTER DATABASE ADD LOGFILE THREAD 1 GROUP 11 size 50M;
ALTER DATABASE ADD LOGFILE THREAD 1 GROUP 11 size 50M
*
ERROR at line 1:
ORA-01275: Operation ADD LOGFILE is not allowed if standby file
management is automatic.

SQL>
```

12. Set STANDBY_MANAGEMENT_FILE to MANUAL.

```
SQL> alter system set standby_file_management=MANUAL sid='*';
System altered.

SQL>
```

13. Try to add the online redo file and standby log file again.

```
SQL> ALTER DATABASE ADD LOGFILE THREAD 1 GROUP 11 size 50M;
Database altered.

SQL> ALTER DATABASE ADD LOGFILE THREAD 2 GROUP 12 size 50M;
Database altered.

SQL> ALTER DATABASE ADD STANDBY LOGFILE THREAD 1 GROUP 13 size 50M;
Database altered.

SQL> ALTER DATABASE ADD STANDBY LOGFILE THREAD 2 GROUP 14 size 50M;
Database altered.

SQL>
```

14. In the same terminal window, verify if the changes made to the primary database are applied in the physical standby database.

```
SQL> SELECT THREAD#, GROUP#, SEQUENCE#, BYTES/1024/1024 "SIZE MB",
ARCHIVED, STATUS FROM V$LOG order by 1;
```

THREAD#	GROUP#	SEQUENCE#	SIZE MB	ARC	STATUS
1	11	0	50	YES	UNUSED
1	2	0	50	YES	UNUSED
1	1	0	50	YES	UNUSED
2	4	0	50	YES	UNUSED
2	12	0	50	YES	UNUSED
2	3	0	50	YES	UNUSED

6 rows selected.

```
SQL> SELECT THREAD#, GROUP#, SEQUENCE#, BYTES/1024/1024 "SIZE MB",
ARCHIVED, STATUS FROM V$STANDBY_LOG order by 1;
```

THREAD#	GROUP#	SEQUENCE#	SIZE MB	ARC	STATUS
1	13	0	50	YES	UNASSIGNED

```

1      7      0      50 YES UNASSIGNED
1      6      7      50 YES ACTIVE
1      5      0      50 NO UNASSIGNED
2     10      0      50 YES UNASSIGNED
2      9      7      50 YES ACTIVE
2     14      0      50 YES UNASSIGNED
2      8      0      50 NO UNASSIGNED

8 rows selected.

SQL>
```

Note: The result indicates the physical standby database is now synchronized with the primary database.

15. Reset STANDBY_MANAGEMENT_FILE to AUTO.

```

SQL> alter system set standby_file_management=AUTO sid='*';
System altered.

SQL>
```

16. Start the managed recovery process.

```

SQL> ALTER DATABASE RECOVER MANAGED STANDBY DATABASE DISCONNECT;
Database altered.

SQL> exit
[oracle@wnode03 ~]$
```

17. Determine the status of the Data Guard configuration.

Note: You may or may not see warning messages like ORA-16525, ORA-16665, ORA-16810, or ORA-16817 due to the lag and overall performance of the virtual machine environment. The warning messages will be cleared in a few minutes. Make sure that your Data Guard configuration status is still fine.

```

[oracle@wnode03 ~]$ dgmgrl
DGMGRL> connect sysdg/oracle_4U@westdb
Connected as SYSDG.
DGMGRL> show configuration

Configuration - dg_config

Protection Mode: MaxPerformance
Members:
  eastdb - Primary database
  westdb - Physical standby database

Fast-Start Failover: DISABLED

Configuration Status:
  SUCCESS  (status updated 17 seconds ago)

DGMGRL>
```

18. Wait until the physical standby database is synchronized. It may take a few moments for the Lag for the standby database to clear.

Note: Do not continue with labs until the Apply Lag is cleared.

```
DGMGRL> show database westdb

Database - westdb

Role: PHYSICAL STANDBY
Intended State: APPLY-ON
Transport Lag: 0 seconds (computed 1 second ago)
Apply Lag: 0 seconds (computed 1 second ago)
Average Apply Rate: 40.00 KByte/s
Real Time Query: OFF
Instance(s):
    westdb1 (apply instance)
    westdb2

Database Status:
SUCCESS

DGMGRL>
```

19. Exit all of your terminal sessions.

Practices for Lesson 7: Effective Client Failover Using Application Continuity

Chapter 7

Practices for Lesson 7: Overview

Practice Summary

- Practice 7-1: Preparing the Data Guard Environment for Application Continuity
- Practice 7-2: Using Application Continuity in RAC
- Practice 7-3: Using Application Continuity in Data Guard

Practice 7-1: Preparing the Data Guard Environment for Application Continuity

Overview

In this practice, you will prepare the practice environment for Application Continuity.

Terminal Sessions

- **Session 1:** ssh oracle@enode01 (Observer Session)
- **Session 2:** ssh oracle@wnode03

Task 1: Preparing the Data Guard environment for Application Continuity

1. **(Session 1)** Establish a terminal session connected to enode01 as the oracle user.

```
[Classroom PC]$ ssh oracle@enode01
[oracle@enode01 ~]$ . oraenv
ORACLE_SID = [oracle] ? eastdb
The Oracle base has been set to /u01/app/oracle
[oracle@enode01 ~]$ export ORACLE_SID=eastdb1
[oracle@enode01 ~]$
```

2. Using DGMGRL, edit the RedoRoutes property of the primary and standby databases.

```
[oracle@enode01 ~]$ dgmgrl
DGMGRL> connect sysdg/oracle_4U@eastdb
Connected as SYSDG.

DGMGRL> EDIT DATABASE 'eastdb' SET PROPERTY 'RedoRoutes' =
'(eastdb:westdb SYNC)';
Property "RedoRoutes" updated

DGMGRL> EDIT DATABASE 'westdb' SET PROPERTY 'RedoRoutes' =
'(westdb:eastdb SYNC)';
Property "RedoRoutes" updated

DGMGRL>
```

3. Set the MaxConnections property for the primary and standby databases to transmit redo data from a single archived redo log on the primary database to the archived redo log at the remote site for gap resolution in parallel.

```
DGMGRL> edit database westdb set property MaxConnections=4;
Property "maxconnections" updated

DGMGRL> edit database eastdb set property MaxConnections=4;
Property "maxconnections" updated

DGMGRL>
```

4. Upgrade the data protection mode to maximum availability. Application Continuity can be used in a Data Guard environment with Fast-Start Failover (FSFO). The data protection mode must be Maximum Availability for FSFO.

Note: You may or may not see warning messages like ORA-16525, ORA-16665, ORA-16810, or ORA-16817 due to the lag and overall performance of the virtual machine environment. The warning messages will be cleared in a few minutes.

```
DGMGRL> EDIT CONFIGURATION SET PROTECTION MODE AS MaxAvailability;
Succeeded.

DGMGRL> show configuration

Configuration - dg_config

Protection Mode: MaxAvailability
Members:
  eastdb - Primary database
  westdb - Physical standby database

Fast-Start Failover: DISABLED

Configuration Status:
SUCCESS      (status updated 34 seconds ago)

DGMGRL>
```

5. Start the observer process.

Note: The prompt will not return after starting the observer. Keep this terminal window open with the observer running in it.

```
DGMGRL> start observer
Observer started
```

6. **(Session 2)** Establish a terminal session connected to wnode03 as the oracle user and configure the terminal environment as shown below.

```
[Classroom PC]$ ssh oracle@wnode03
oracle@wnode03's password: <oracle>
[oracle@wnode03 ~]$ . oraenv
ORACLE_SID = [oracle] ? westdb
The Oracle base has been set to /u01/app/oracle
[oracle@wnode03 ~]$ export ORACLE_SID=westdb1
[oracle@wnode03 ~]$
```

Task 2: Enabling Fast-Start Failover

7. In the same terminal on wnode03, enable fast-start failover.

```
[oracle@wnode03 ~]$ dgmgrl
DGMGRL> connect sysdg/oracle_4U@westdb
Connected as SYSDG.
DGMGRL> enable fast_start failover;
Enabled.
DGMGRL>
```

8. Set the FSFO threshold via the broker configuration property `FastStartFailoverThreshold` to a lower value for testing purpose.

```
DGMGRL> edit configuration set property FastStartFailoverThreshold=10;
Property "faststartfailoverthreshold" updated
DGMGRL>
```

9. Verify the `fast_start failover` settings.

```
DGMGRL> show fast_start failover;

Fast-Start Failover: ENABLED

Threshold:          10 seconds
Target:             westdb
Observer:           enode01.example.com
Lag Limit:          30 seconds (not in use)
Shutdown Primary:   TRUE
Auto-reinstate:     TRUE
Observer Reconnect: (none)
Observer Override:  FALSE

Configurable Failover Conditions
  Health Conditions:
    Corrupted Controlfile      YES
    Corrupted Dictionary       YES
    Inaccessible Logfile       NO
    Stuck Archiver             NO
    Datafile Offline           YES

  Oracle Error Conditions:
    (none)

DGMGRL> exit
[oracle@wnode03 ~] $
```

10. Exit out of the terminal **Session 2**, but keep Session 1 (Observer).

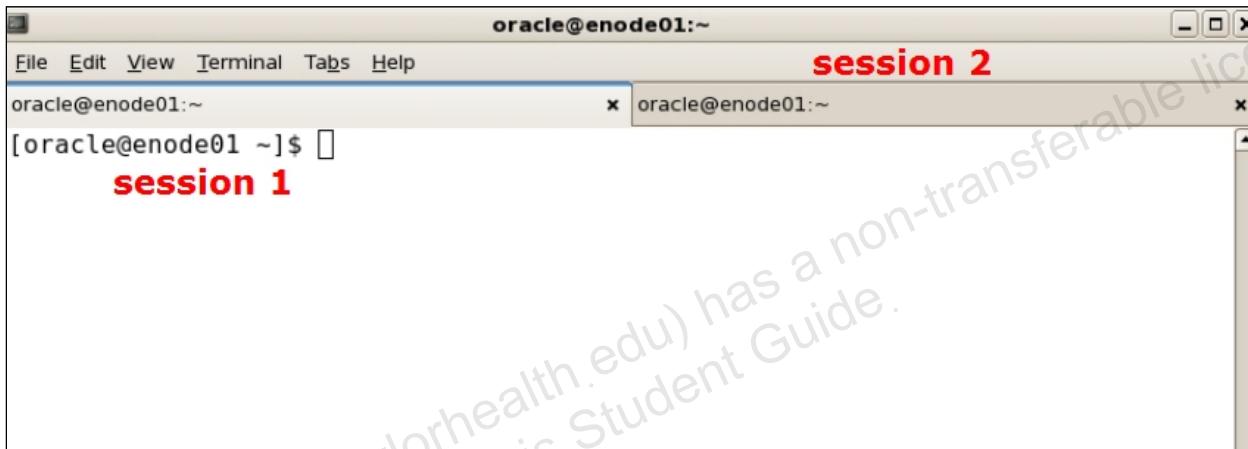
Practice 7-2: Using Application Continuity in RAC

Overview

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

Terminal Sessions

- **Observer Session**
- **Session 1: ssh oracle@enode01**
- **Session 2: ssh oracle@enode01**



Tasks

1. **(Session 1)** Establish a terminal session connected to enode01 as the oracle user and configure the terminal environment as shown below.

```
[Classroom PC]$ ssh oracle@enode01
oracle@enode01's password: <oracle>
[oracle@enode01 ~]$ . oraenv
ORACLE_SID = [oracle] ? eastdb
The Oracle base has been set to /u01/app/oracle
[oracle@enode01 ~]$ export ORACLE_SID=eastdb1
[oracle@enode01 ~]$
```

2. Display the primary database configuration.

Note: Make sure that Start option is open and Database role is PRIMARY. Occasionally, the srvctl config database command reports the wrong information.

```
[oracle@enode01 ~]$ srvctl config database -db eastdb
Database unique name: eastdb
Database name: eastdb
Oracle home: /u01/app/oracle/product/12.1.0/dbhome_1
Oracle user: oracle
Spfile: +DATA/EASTDB/PARAMETERFILE/spfile.287.892719121
Password file: +DATA/eastdb/orapwasteastdb
Domain: example.com
Start options: open
Stop options: immediate
```

```
Database role: PRIMARY
Management policy: AUTOMATIC
Server pools:
Disk Groups: DATA, FRA
Mount point paths:
Services: actest
Type: RAC
Start concurrency:
Stop concurrency:
OSDBA group: dba
OSOPER group: oper
Database instances: eastdb1,eastdb2
Configured nodes: enode01,enode02
Database is administrator managed
[oracle@enode01 ~]$
```

3. Confirm that both instances of the RAC database are up and running.

Note: Make sure that both instances are running for the practice.

```
[oracle@enode01 ~]$ srvctl status database -db eastdb -verbose
Instance eastdb1 is running on node enode01. Instance status: Open.
Instance eastdb2 is running on node enode02. Instance status: Open.
[oracle@enode01 ~]$
```

4. Using SQL*Plus, change the password of the scott user to 'tiger' for the practice.

```
[oracle@enode01 ~]$ sqlplus / as sysdba
SQL> alter user scott account unlock identified by tiger;
User altered.
SQL> exit
[oracle@enode01 ~]$
```

5. Change into the /home/oracle/labs directory that contains the files for this practice.

```
[oracle@enode01 ~]$ cd /home/oracle/labs
[oracle@enode01 labs]$
```

6. Create a database service on the eastdb database. This service is configured for use in conjunction with Application Continuity.

Note: Services that support Application Continuity must specify

FAILOVER_TYPE=TRANSACTION and COMMIT_OUTCOME=TRUE. The remaining service attribute settings used in this example are based on general recommendations for Application Continuity in conjunction with Oracle RAC; however, they may be adjusted to suit different circumstances.

```
[oracle@enode01 labs]$ srvctl add service -db eastdb \
    -service actest \
    -preferred eastdb1 \
    -available eastdb2 \
    -clbgoal SHORT \
    -rlbgoal SERVICE_TIME \
    -failovertype TRANSACTION \
    -commit_outcome TRUE \
    -failoverretry 30 \
    -failoverdelay 5 \
    -retention 86400 \
    -replay_init_time 1800 \
```

7. Display the actest service configuration.

```
[oracle@enode01 labs]$ srvctl config service -db eastdb -service actest
Service name: actest
Server pool:
Cardinality: 1
Disconnect: false
Service role: PRIMARY
Management policy: AUTOMATIC
DTP transaction: false
AQ HA notifications: true
Global: false
Commit Outcome: true
Failover type: TRANSACTION
Failover method:
TAF failover retries: 30
TAF failover delay: 5
Connection Load Balancing Goal: SHORT
Runtime Load Balancing Goal: SERVICE_TIME
TAF policy specification: NONE
Edition:
Pluggable database name:
Maximum lag time: ANY
SQL Translation Profile:
Retention: 86400 seconds
Replay Initiation Time: 1800 seconds
Session State Consistency: DYNAMIC
GSM Flags: 0
Service is enabled
Preferred instances: eastdb1
Available instances: eastdb2
[oracle@enode01 labs]$
```

8. (**Session 2**) Establish another terminal session connected to enode01 as the oracle user and configure the terminal environment as shown below. To differentiate this session from your primary session, it will be referred to as the ADMIN session for the rest of the practice.

```
[Classroom PC]$ ssh oracle@enode01
oracle@enode01's password: <oracle>
[oracle@enode01 ~]$ . oraenv
ORACLE_SID = [oracle] ? eastdb
The Oracle base has been set to /u01/app/oracle
[oracle@enode01 ~]$ export ORACLE_SID=eastdb1
[oracle@enode01 ~]$ export PS1='ADMIN $ '
ADMIN $
```

9. (**Session 1**) Back in your primary session, 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@enode01 labs]$ cat runnoreplay.sh
java -classpath
./actest.jar:$ORACLE_HOME/ucp/lib/ucp.jar:$ORACLE_HOME/jdbc/lib/
ojdbc6.jar actest ACTest actest_noreplay.properties
```

```
[oracle@enode01 labs]$ cat runreplay.sh
java -classpath
./actest.jar:$ORACLE_HOME/ucp/lib/ucp.jar:$ORACLE_HOME/jdbc/lib/
ojdbc6.jar actest ACTest actest_replay.properties
[oracle@enode01 labs]$
```

10. Examine the properties files. Notice that the only difference is the datasource specification.

```
[oracle@enode01 labs]$ cat actest_noreplay.properties
username=scott
password=tiger
autoCommit=false
# Use standard 12.1 datasource no replay
datasource=oracle.jdbc.pool.OracleDataSource
url=jdbc:oracle:thin:@(DESCRIPTION=(ADDRESS=(PROTOCOL=tcp) (HOST=
cluster01-
scan) (PORT=1521)) (CONNECT_DATA=(SERVICE_NAME=actest.
example.com)))
# 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@enode01 labs]$ cat actest_replay.properties
username=scott
password=tiger
autoCommit=false
# Use new 12.1 replay datasource
datasource=oracle.jdbc.replay.OracleDataSourceImpl
url=jdbc:oracle:thin:@(DESCRIPTION=(ADDRESS=(PROTOCOL=tcp) (HOST=
cluster01-
scan) (PORT=1521)) (CONNECT_DATA=(SERVICE_NAME=actest.
example.com)))
# 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@enode01 labs]$ diff actest_noreplay.properties
actest_replay.properties

5,6c5,6
< # Use standard 12.1 datasource no replay
```

```
< datasource=oracle.jdbc.pool.OracleDataSource
---
> # Use new 12.1 replay datasource
> datasource=oracle.jdbc.replay.OracleDataSourceImpl

[oracle@enode01 labs]$
```

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.

- Start the service that you created in step 6.

```
[oracle@enode01 labs]$ srvctl start service -db eastdb -service actest
[oracle@enode01 labs]$
```

- Examine the status of the newly created service.

Note: You may have to restart the `actest` service if it is not running.

```
[oracle@enode01 labs]$ srvctl status service -db eastdb
Service actest is running on instance(s) eastdb1
[oracle@enode01 labs]$
```

- Execute the practice application without the benefit of Application Continuity. Notice that while the application runs, a periodic status message is displayed.

```
[oracle@enode01 labs]$ ./runnoreplay.sh
#####
Connecting to
jdbc:oracle:thin:@(DESCRIPTION= (ADDRESS= (PROTOCOL=tcp) (HOST=cluster01-
scan) (PORT=1521)) (CONNECT_DATA= (SERVICE_NAME=actest.
example.com)))
# of Threads : 6
UCP pool size : 2
Thread think time : 20 ms
#####
2 active connections, avg response time from db 21 ms
2 active connections, avg response time from db 11 ms
...
```

- (Session 2) 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, crash the database instance running the `actest` service.

```
ADMIN $ ps -ef|grep smon_eastdb
oracle      5252  5219  0 07:53 pts/1      00:00:00 grep smon_eastdb
oracle     26189      1  0 Sep30 ?      00:00:05 ora_smon_eastdb1

ADMIN $ kill -9 26189
ADMIN $
```

15. (**Session 1**) Return to your primary window and you will see a series of errors caused by crashing the database instance. This is typical of applications that do not use Application Continuity. Press **Ctrl + C** to abort the application.

```
...
.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
...
[oracle@enode01 labs]$
```

16. In your practice environment, Oracle High Availability Services will restart the crashed database instance. Periodically re-examine the database status until you can confirm that both RAC database instances are up and running again.

```
[oracle@enode01 labs]$ srvctl status database -db eastdb
Instance eastdb1 is running on node enode01
Instance eastdb2 is running on node enode02

[oracle@enode01 labs]$
```

17. Examine the status of the **actest** service. The service should be automatically migrated to the **eastdb2** instance when you crashed the database instance earlier in the practice.

Note: You may need to restart the **actest** service using the **srvctl start service -db eastdb -service actest** command if it is not running for a few minutes.

```
[oracle@enode01 labs]$ srvctl status service -db eastdb
Service actest is running on instance(s) eastdb2
[oracle@enode01 labs]$
```

18. Execute the practice application with Application Continuity enabled. You should see the same status messages as before while the application is running.

```
[oracle@enode01 labs]$ ./runreplay.sh
#####
Connecting to
jdbc:oracle:thin:@(DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=cluster01-
scan)(PORT=1521))(CONNECT_DATA=(SERVICE_NAME=actest.
example.com)))
# of Threads : 6
UCP pool size : 2
Thread think time : 20 ms
#####
2 active connections, avg response time from db 304 ms
2 active connections, avg response time from db 45 ms
...
```

19. (**Session 2**) 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, crash the database instance running the `actest` service.

```
ADMIN $ srvctl status service -db eastdb
Service actest is running on instance(s) eastdb2
ADMIN $ ssh oracle@enode02
[oracle@enode02 ~]$ ps -ef|grep smon_eastdb
oracle      5252  5219  0 07:53 pts/1    00:00:00 grep smon_eastdb
oracle     26189      1  0 Sep30 ?        00:00:05 ora_smon_eastdb2
[oracle@enode02 ~]$ kill -9 26189
[oracle@enode02 ~]$ exit
logout
Connection to enode02 closed.
ADMIN $
```

20. As before, the service will automatically migrate in response to the instance crash. Confirm that the `actest` service is running on a different node compared to what you observed in the previous step.

Note: You may need to wait for a while for the service migration to complete, so periodically re-examine the service status if required.

```
ADMIN $ srvctl status service -db eastdb
Service actest is running on instance(s) eastdb1
ADMIN $
```

21. (**Session 1**) Return to the primary window and you will see that the application continues to run in spite of crashing the database instance. You should see a brief spike in the response time, which coincides with the time when the database instance crashed. 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 304 ms
2 active connections, avg response time from db 45 ms
2 active connections, avg response time from db 29 ms
2 active connections, avg response time from db 32 ms
2 active connections, avg response time from db 29 ms
2 active connections, avg response time from db 18 ms
2 active connections, avg response time from db 36 ms
2 active connections, avg response time from db 5962 ms
2 active connections, avg response time from db 98 ms
2 active connections, avg response time from db 61 ms
2 active connections, avg response time from db 72 ms
[oracle@enode01 labs]$
```

22. Determine the status of the Data Guard configuration. Make sure that your Data Guard configuration status is still fine.

Note: You may or may not see warning messages like ORA-16525, ORA-16665, ORA-16810, or ORA-16817 due to the lag and overall performance of the virtual machine environment. The warning messages will be cleared in a few minutes.

```
[oracle@enode01 labs]$ dgmgrl
DGMGRL> connect sysdg/oracle_4U@eastdb
Connected as SYSDG.

DGMGRL> show configuration

Configuration - dg_config

Protection Mode: MaxAvailability
Members:
  eastdb - Primary database
  westdb - Physical standby database

Fast-Start Failover: DISABLED

Configuration Status:
SUCCESS      (status updated 17 seconds ago)

DGMGRL>
```

23. Wait until the physical standby database is synchronized. It may take a few moments for the Lag for the standby database to clear.

Note: Do not continue with labs until the Apply Lag is cleared.

```
DGMGRL> show database westdb

Database - westdb

Role:          PHYSICAL STANDBY
Intended State: APPLY-ON
Transport Lag: 0 seconds (computed 1 second ago)
Apply Lag:   0 seconds (computed 1 second ago)
Average Apply Rate: 40.00 KByte/s
Real Time Query: OFF
Instance(s):
  westdb1 (apply instance)
  westdb2

Database Status:
SUCCESS

DGMGRL>
```

24. Exit all of your terminal sessions **except for the Observer Session**.

Practice 7-3: Using Application Continuity in Data Guard

Overview

In this practice, you will use Application Continuity in a Data Guard configuration to demonstrate how Application Continuity helps an application to seamlessly recover after the primary database failure, especially in case of fast-start failover. **This practice assumes that you have completed practice 5-5 and 7-1.**

- **Task 1:** Create the Role-Based Service on the Primary Database
- **Task 2:** Create the Role-Based Service on the Standby Database
- **Task 3:** Test Application Continuity in Data Guard
- **Task 4:** Perform Switchover to the Reinstated Database

Terminal Sessions

- **Observer Session**
- **Session 1:** ssh oracle@enode01
- **Session 2:** ssh oracle@wnode03
- **Session 3:** ssh oracle@enode01
- **Session 4:** ssh oracle@enode01

In order to avoid confusion, open all terminal windows.



Task 1: Create the Role-Based Service on the Primary Database

1. **(Session 1)** Establish a terminal session connected to enode01 as the oracle user.

```
[Classroom PC]$ ssh oracle@enode01
oracle@enode01's password: <oracle>
[oracle@enode01 ~]$ . oraenv
ORACLE_SID = [oracle] ? eastdb
The Oracle base has been set to /u01/app/oracle
[oracle@enode01 ~]$ export ORACLE_SID=eastdb1
[oracle@enode01 ~]$
```

2. Modify the actest service as the role-based service on the primary database.

```
[oracle@enode01]$ srvctl modify service -db eastdb -service actest -
role primary
[oracle@enode01 ~]$
```

3. Examine the status of the modified service. If the `actest` service is not running, start the service.

```
[oracle@enode01 ~]$ srvctl status service -db eastdb
Service actest is running on instance(s) eastdb1

### If the actest service is not running, start the service.

[oracle@enode01 ~]$ srvctl start service -db eastdb -service actest
[oracle@enode01 ~]$ srvctl status service -db eastdb
Service actest is running on instance(s) eastdb1
[oracle@enode01 ~]$
```

4. Optionally, if the `actest` service is running in the `eastdb2` instance, relocate it to the `eastdb1` instance for the purpose of this demonstration.

```
[oracle@enode01 ~]$ srvctl relocate service -db eastdb -service actest
-oldinst eastdb2 -newinst eastdb1
[oracle@enode01 ~]$ srvctl status service -db eastdb
Service actest is running on instance(s) eastdb1
[oracle@enode01 ~]$
```

Task 2: Create the Role-Based Service on the Standby Database

5. (Session 2) Establish a terminal session connected to `wnode03` as the `oracle` user and configure the terminal environment as shown below.

```
[Classroom PC]$ ssh oracle@wnode03
[oracle@wnode03 ~]$ . oraenv
ORACLE_SID = [oracle] ? westdb
The Oracle base has been set to /u01/app/oracle
[oracle@wnode03 ~]$ export ORACLE_SID=westdb1
[oracle@wnode03 ~]$
```

6. Create the `actest` service as the role-based service on the standby database to support a role transition.

```
[oracle@wnode03 ~]$ srvctl add service -db westdb \
-service actest \
-preferred westdb1 \
-available westdb2 \
-role primary \
-clbgoal SHORT \
-rlbgoal SERVICE_TIME \
-failovertype TRANSACTION \
-commit_outcome TRUE \
-failoverretry 60 \
-failoverdelay 5 \
-retention 86400 \
-replay_init_time 1800 \
-notification TRUE
[oracle@wnode03 ~]$
```

7. Examine the status of the newly created role-based service.

Note: The current database (westdb) role is PHYSICAL STANDBY. So the actest service is not running in the current role.

```
[oracle@wnode03 ~]$ srvctl status service -db westdb
Service actest is not running.
[oracle@wnode03 ~]$
```

Task 3: Test Application Continuity in Data Guard

8. (Session 3) Establish a terminal session connected to enode01 as the oracle user.

```
[Classroom PC]$ ssh oracle@enode01
oracle@enode01's password: <oracle>
[oracle@enode01 ~]$ . oraenv
ORACLE_SID = [oracle] ? eastdb
The Oracle base has been set to /u01/app/oracle
[oracle@enode01 ~]$ export ORACLE_SID=eastdb1
[oracle@enode01 ~]$
```

9. Change to the /home/oracle/labs directory to copy the runreplay.sh script for the next practice.

```
[oracle@enode01 ~]$ cd /home/oracle/labs
[oracle@enode01 labs]$ cp runreplay.sh runreplayDG.sh
[oracle@enode01 labs]$
```

10. Replace actest_replay.properties with actest_replayDG.properties in the runreplayDG.sh script.

```
[oracle@enode01 labs]$ vi runreplayDG.sh
java -classpath
./actest.jar:$ORACLE_HOME/ucp/lib/ucp.jar:$ORACLE_HOME/jdbc/lib/
ojdbc6.jar actest.ACTest actest_replayDG.properties

:wq!
[oracle@enode01 labs]$
```

11. Examine the actest_replayDG.properties properties file.

```
[oracle@enode01 labs]$ cat actest_replayDG.properties
username=scott
password=tiger
autoCommit=false

# Use new 12.1 replay datasource
datasource=oracle.jdbc.replay.OracleDataSourceImpl

url=jdbc:oracle:thin:@(DESCRIPTION_LIST=(DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=cluster01-
scan)(PORT=1521))(CONNECT_DATA=(SERVICE_NAME=actest.example.com)))(DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=cluster02-
scan)(PORT=1521))(CONNECT_DATA=(SERVICE_NAME=actest.example.com)))))

# UCP setting:
ucp_pool_size=2
ucp_validate_connection_on_borrow=true
```

```
ucp_connection_timeout=900
# 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@enode01 labs]$
```

12. Execute the practice application with Application Continuity enabled in your Data Guard environment. You should see the same status messages as before while the application is running.

```
[oracle@enode01 labs]$ ./runreplayDG.sh
#####
Connecting to
jdbc:oracle:thin:@(DESCRIPTION_LIST=(DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)
(HOST=cluster01-
scan) (PORT=1521)) (CONNECT_DATA=(SERVICE_NAME=actest.example.com))) (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=cluster02-
scan) (PORT=1521)) (CONNECT_DATA=(SERVICE_NAME=actest.example.com)))
# of Threads : 6
UCP pool size : 2
Thread think time : 20 ms
#####
2 active connections, avg response time from db 182 ms
2 active connections, avg response time from db 91 ms
2 active connections, avg response time from db 122 ms
2 active connections, avg response time from db 300 ms
...
```

13. (**Session 4**) Establish another terminal session on enode01 as the oracle user to simulate the primary database planned/unplanned outage.

```
[Classroom PC]$ ssh oracle@enode01
oracle@enode01's password: <oracle>
[oracle@enode01 ~]$ . oraenv
ORACLE_SID = [oracle] ? eastdb
The Oracle base has been set to /u01/app/oracle
[oracle@enode01 ~]$ ps -ef|grep smon_eastdb
oracle    5252  5219  0 07:53 pts/1    00:00:00 grep smon_eastdb
oracle    26189      1  0 Sep30 ?        00:00:05 ora_smon_eastdb1
[oracle@enode01 ~]$ 
[oracle@enode01 ~]$ kill -9 26189
[oracle@enode01 ~]$ 
[oracle@enode01 ~]$ ssh enode02
[oracle@enode02 ~]$ ps -ef|grep smon_eastdb
oracle   19135      1  0 23:34 ?        00:00:00 ora_smon_eastdb2
oracle   22904 13866  0 23:45 pts/4    00:00:00 grep smon_eastdb
[oracle@enode02 ~]$ kill -9 19135
[oracle@enode02 ~]$ exit
logout
Connection to enode02 closed.
[oracle@enode01 ~]$
```

14. **(Observer)** Return to the observer window. Wait until you see the "Failover succeeded, new primary is westdb" message.

Note: The Fast-Start Failover operation may take 10-12 minutes in our practice environment.

```
DGMGRL> start observer
Observer started

17:55:21.20 Friday, October 02, 2015
Initiating Fast-Start Failover to database "westdb"...
Performing failover NOW, please wait...
Failover succeeded, new primary is "westdb"
17:57:41.08 Friday, October 02, 2015
```

15. **(Session 3)** Return to the primary window and you will see that the application continues without a series of errors.

```
...
2 active connections, avg response time from db 182 ms
2 active connections, avg response time from db 91 ms
2 active connections, avg response time from db 122 ms
2 active connections, avg response time from db 300 ms
...
```

16. **(Observer)** Return to the observer window. Wait until you see the "Reinstatement of database "eastdb" succeeded" message.

Note: It may take about another 5~10 minutes to complete the final Reinstatement operation due to the performance of the virtual machine environment.

```
DGMGRL> start observer
Observer started

17:55:21.20 Friday, October 02, 2015
Initiating Fast-Start Failover to database "westdb"...
Performing failover NOW, please wait...
Failover succeeded, new primary is "westdb"
17:57:41.08 Friday, October 02, 2015

18:03:53.58 Friday, October 02, 2015
Initiating reinstatement for database "eastdb"...
Reinstating database "eastdb", please wait...
Reinstatement of database "eastdb" succeeded
18:06:11.65 Friday, October 02, 2015
```

17. **(Session 3)** Return to the primary window and you will see that the application continues to run in spite of crashing the database instance. You should see a brief spike in the response time, which coincides with the time when the database instance crashed. Now you have seen how Application Continuity masks the effect of database failure in a Data Guard environment. Press Ctrl + C to abort the application.

Note: At times, the output hangs due to the lag and overall performance of the virtual machine environment. You can safely move to the next step.

```
...
2 active connections, avg response time from db 460 ms
2 active connections, avg response time from db 13446 ms
```

```

2 active connections, avg response time from db 772 ms
2 active connections, avg response time from db 580 ms
2 active connections, avg response time from db 854 ms
2 active connections, avg response time from db 269 ms
2 active connections, avg response time from db 187 ms
2 active connections, avg response time from db 136 ms
2 active connections, avg response time from db 146 ms
2 active connections, avg response time from db 34 ms
2 active connections, avg response time from db 19 ms
2 active connections, avg response time from db 31 ms
2 active connections, avg response time from db 188 ms [oracle@enode01
labs]$
```

Task 4: Perform Switchover to the Reinstated Database

18. In the same terminal session, validate that the 'westdb' primary database is ready for switchover.

```

[oracle@enode01 labs]$ dgmgrl
DGMGRL> connect sysdg/oracle_4U@eastdb
DGMGRL> validate database westdb

Database Role: Primary database
Ready for Switchover: Yes
DGMGRL>
```

19. Validate that the 'eastdb' standby database is ready for switchover. Wait for a few minutes until the Switchover is ready. The "Apply Lag" should be cleared.

Note: You may receive warning messages like ORA-16713. It takes a few minutes to clear the messages.

```

DGMGRL> validate database eastdb

Database Role: Physical standby database
Primary Database: westdb

Ready for Switchover: No
Ready for Failover: Yes (Primary Running)

Standby Apply-Related Information:
  Apply State: Running
  Apply Lag: 11 minutes 25 seconds
  Apply Delay: 0 minutes

Transport-Related Property Settings:
  Property          westdb Value          eastdb Value
  RedoRoutes        (westdb:eastdb SYNC)    (eastdb:westdb SYNC)

  ### Waiting for a few minutes ###

DGMGRL> validate database eastdb
Database Role: Physical standby database
Primary Database: westdb

Ready for Switchover: Yes
Ready for Failover: Yes (Primary Running)
```

Transport-Related Property Settings:		
Property	westdb Value	eastdb Value
RedoRoutes	(westdb:eastdb SYNC)	(eastdb:westdb SYNC)
DGMGRL>		

20. Switch over to the 'eastdb' database.

Note: As of Oracle Database 12c Release 1, when you perform a switchover from an Oracle RAC primary database to a physical standby database, *it is no longer necessary* to shut down all but one primary database instance.

```
DGMGRL> switchover to eastdb
Performing switchover NOW, please wait...
New primary database "eastdb" is opening...
Oracle Clusterware is restarting database "westdb" ...
Switchover succeeded, new primary is "eastdb"
DGMGRL>
```

21. Display the resulting configuration.

Note: You may or may not see warning messages like ORA-16525, ORA-16665, ORA-16810, ORA-16817, or ORA-16825 due to the lag and overall performance of the virtual machine environment. The warning messages will be cleared in a few minutes. Make sure that your Data Guard configuration status is still fine.

```
DGMGRL> show configuration

Configuration - dg_config

Protection Mode: MaxAvailability
Members:
  eastdb - Primary database
  westdb - (*) Physical standby database

Fast-Start Failover: ENABLED

Configuration Status:
  SUCCESS      (status updated 57 seconds ago)

DGMGRL>
```

22. Wait until the physical standby database is synchronized. It may take a few moments for the Lag for the standby database to clear.

Note: Do not continue with labs until the Apply Lag is cleared.

```
DGMGRL> show database westdb

Database - westdb

Role:          PHYSICAL STANDBY
Intended State: APPLY-ON
Transport Lag: 0 seconds (computed 1 second ago)
Apply Lag:   0 seconds (computed 1 second ago)
Average Apply Rate: 40.00 KByte/s
Real Time Query: OFF
Instance(s):
  westdb1 (apply instance)
```

```
westdb2
```

```
Database Status:  
SUCCESS
```

```
DGMGRL>
```

23. Stop the observer process.

Note: The prompt should now be returned in the terminal window that was connected to wnode03 running the observer process.

```
DGMGRL> stop observer  
Done.  
DGMGRL>
```

24. Disable fast-start failover.

```
DGMGRL> disable fast_start failover;  
Disabled.  
DGMGRL>
```

25. Reset your environment for the next practices.

```
DGMGRL> EDIT CONFIGURATION SET PROTECTION MODE AS MaxPerformance;  
Succeeded.  
DGMGRL> EDIT DATABASE 'eastdb' SET PROPERTY 'RedoRoutes' =  
'(eastdb:westdb ASYNC)';  
Property "RedoRoutes" updated  
  
DGMGRL> EDIT DATABASE 'westdb' SET PROPERTY 'RedoRoutes' =  
'(westdb:eastdb ASYNC)';  
Property "RedoRoutes" updated  
  
DGMGRL> EDIT DATABASE westdb SET PROPERTY MaxConnections=1;  
Property "maxconnections" updated  
  
DGMGRL> EDIT DATABASE eastdb SET PROPERTY MaxConnections=1;  
Property "maxconnections" updated  
  
DGMGRL> exit  
[oracle@enode01 labs]$
```

26. Exit all of your terminal sessions.

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

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

Practices for Lesson 8: Effective Service Failover and Workload Management Using Global Data Services

Chapter 8

Practices for Lesson 8: Overview

Practices Overview

In this practice, you configure Global Service Manager against a set of replicated Oracle Databases (with replication provided by either GoldenGate or Active Data Guard). You then configure a service of Global Data Services and test it to make sure that it works as expected. You then make changes to the environment (such as shutting down nodes and/or databases) and test how Global Data Services responds to maintain service availability.

Practice Summary

- Practice 8-1: Reconfiguring the Environment for GDS
- Practice 8-2: Installing and Configuring Global Data Services
- Practice 8-3: Global Service Failover
- Practice 8-4: Role-Based Global Services
- Practice 8-5: Replication Lag-Based Routing

Practice 8-1: Reconfiguring the Environment for GDS

Overview

In this practice, you will reconfigure your practice environment in preparation for the Global Data Services practices.

Task Summary

- **Task 1:** Reconfiguring the Default Listener
- **Task 2:** Enabling Real-Time Query

Terminal Sessions

- **Session 1:** `ssh grid@enode01`
- **Session 2:** `ssh grid@wnode03`

Task 1: Reconfiguring the Default Listener

1. **(Session 1)** Open a terminal window on enode01 as the `grid` user.

```
[Classroom PC]$ ssh grid@enode01
grid@enode01's password: <oracle>
[grid@enode01 ~]$
```

2. Set your environment to manage the default listener.

```
[grid@enode01 ~]$ . oraenv
ORACLE_SID = [grid]? +ASM1
The Oracle base has been set to /u01/app/grid
[grid@enode01 ~]$
```

3. Remove the default listener from CRS Home.

```
[grid@enode01 ~]$ srvctl stop listener
[grid@enode01 ~]$ srvctl remove listener
[grid@enode01 ~]$
```

4. In the same terminal, add the default listener to the RDBMS Home as the new owner.

```
[grid@enode01 ~]$ su - oracle
Password: <oracle>
[oracle@enode01 ~]$ . oraenv
ORACLE_SID = [oracle]? eastdb
The Oracle base has been set to /u01/app/oracle
[oracle@enode01 ~]$ export ORACLE_SID=eastdb
[oracle@enode01 ~]$ srvctl add listener -l listener -oraclehome
/u01/app/oracle/product/12.1.0/dbhome_1
[oracle@enode01 ~]$ srvctl start listener
[oracle@enode01 ~]$ lsnrctl status

LSNRCTL for Linux: Version 12.1.0.2.0 - Production on 28-OCT-2015
15:09:35

Copyright (c) 1991, 2014, Oracle. All rights reserved.

Connecting to (DESCRIPTION=(ADDRESS=(PROTOCOL=IPC)(KEY=LISTENER)))
STATUS of the LISTENER
-----
```

```

Alias LISTENER
Version TNSLSNR for Linux: Version 12.1.0.2.0 -
Production
Start Date 28-OCT-2015 15:09:25
Uptime 0 days 0 hr. 0 min. 10 sec
Trace Level off
Security ON: Local OS Authentication
SNMP OFF
Listener Parameter File
/u01/app/oracle/product/12.1.0/dbhome_1/network/admin/listener.ora
Listener Log File
/u01/app/oracle/diag/tnslsnr/enode01/listener/alert/log.xml
Listening Endpoints Summary...
  (DESCRIPTION=(ADDRESS=(PROTOCOL=ipc)(KEY=LISTENER)))
  (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=192.0.2.111)(PORT=1521)))
  (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=192.0.2.161)(PORT=1521)))

  (DESCRIPTION=(ADDRESS=(PROTOCOL=tcps)(HOST=enode01.example.com)(PORT=55
00))(Security=(my_wallet_directory=/u01/app/oracle/product/12.1.0/dbhom
e_1/admin/eastdb/xdb_wallet))(Presentation=HTTP)(Session=RAW))
Services Summary...
Service "actest.example.com" has 1 instance(s).
  Instance "eastdb1", status READY, has 1 handler(s) for this
service...
Service "eastdb.example.com" has 1 instance(s).
  Instance "eastdb1", status READY, has 1 handler(s) for this
service...
Service "eastdbXDB.example.com" has 1 instance(s).
  Instance "eastdb1", status READY, has 1 handler(s) for this
service...
Service "eastdb_DGB.example.com" has 1 instance(s).
  Instance "eastdb1", status READY, has 1 handler(s) for this
service...
Service "eastdb_DGMGRL.example.com" has 1 instance(s).
  Instance "eastdb1", status UNKNOWN, has 1 handler(s) for this
service...
Service "gdscat.example.com" has 1 instance(s).
  Instance "gdscat", status READY, has 1 handler(s) for this service...
Service "gdscatXDB.example.com" has 1 instance(s).
  Instance "gdscat", status READY, has 1 handler(s) for this service...
The command completed successfully

[oracle@enode01 ~]$
```

5. **(Session 2)** Open another terminal window on wnode03 as the grid user.

```
[Classroom PC]$ ssh grid@wnode03
grid@wnode03's password: <oracle>
[grid@wnode03 ~]$
```

6. Set your environment to manage the default listener.

```
[grid@wnode03 ~]$ . oraenv
ORACLE_SID = [grid]? +ASM1
The Oracle base has been set to /u01/app/grid
[grid@wnode03 ~]$
```

7. Remove the default listener from the CRS Home.

```
[grid@wnode03 ~]$ srvctl stop listener  
[grid@wnode03 ~]$ srvctl remove listener  
[grid@wnode03 ~]$
```

8. In the same terminal, add the default listener to the RDBMS Home as the new owner.

```
[grid@wnode03 ~]$ su - oracle  
Password: <oracle>  
[oracle@wnode03 ~]$ . oraenv  
ORACLE_SID = [oracle]? westdb  
The Oracle base has been set to /u01/app/oracle  
[oracle@wnode03 ~]$ export ORACLE_SID=westdb1  
[oracle@wnode03 ~]$ srvctl add listener -l listener -oraclehome  
/u01/app/oracle/product/12.1.0/dbhome_1  
[oracle@wnode03 ~]$ srvctl start listener  
[oracle@wnode03 ~]$ lsnrctl status  
  
LSNRCTL for Linux: Version 12.1.0.2.0 - Production on 28-OCT-2015  
15:13:29  
  
Copyright (c) 1991, 2014, Oracle. All rights reserved.  
  
Connecting to (DESCRIPTION=(ADDRESS=(PROTOCOL=IPC)(KEY=LISTENER)))  
STATUS of the LISTENER  
-----  
Alias LISTENER  
Version TNSLSNR for Linux: Version 12.1.0.2.0 -  
Production  
Start Date 28-OCT-2015 15:13:09  
Uptime 0 days 0 hr. 0 min. 20 sec  
Trace Level off  
Security ON: Local OS Authentication  
SNMP OFF  
Listener Parameter File /u01/app/oracle/product/12.1.0/dbhome_1/network/admin/listener.ora  
Listener Log File /u01/app/oracle/diag/tnslsnr/wnode03/listener/alert/log.xml  
Listening Endpoints Summary...  
  (DESCRIPTION=(ADDRESS=(PROTOCOL=ipc)(KEY=LISTENER)))  
  (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=192.0.2.121)(PORT=1521)))  
  (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=192.0.2.171)(PORT=1521)))  
Services Summary...  
Service "+ASM" has 1 instance(s).  
  Instance "+ASM1", status READY, has 1 handler(s) for this service...  
Service "westdb.example.com" has 1 instance(s).  
  Instance "westdb1", status READY, has 1 handler(s) for this service...  
Service "westdb_DGB.example.com" has 1 instance(s).  
  Instance "westdb1", status READY, has 1 handler(s) for this service...  
Service "westdb_DGMGRL.example.com" has 1 instance(s).  
  Instance "westdb1", status UNKNOWN, has 1 handler(s) for this service...  
The command completed successfully  
[oracle@wnode03 ~]$
```

Task 2: Enabling Real-Time Query

9. (Session 2) In the same terminal, restart the Physical Standby Database to enable the Real-Time Query feature.

Note: This step is required for the GDS practices.

```
[oracle@wnode03 ~]$ srvctl stop database -db westdb
[oracle@wnode03 ~]$ srvctl start database -db westdb -startoption open
[oracle@wnode03 ~]$ srvctl status database -db westdb -verbose
Instance westdb1 is running on node wnode03. Instance status:
Open,Readonly.
Instance westdb2 is running on node wnode04. Instance status:
Open,Readonly.
[oracle@wnode03 ~]$
```

10. Verify the OPEN MODE database. It must be READ ONLY WITH APPLY because the Data Guard broker enabled the Real-Time Query feature automatically.

Note: This step is required for the GDS practices.

```
[oracle@wnode03 ~]$ sqlplus / as sysdba

SQL> select open_mode from v$database;

OPEN_MODE
-----
READ ONLY WITH APPLY

SQL> exit
[oracle@wnode03 ~]$
```

11. Using DGMGRL, determine the status of the Real-Time Query feature.

```
[oracle@wnode03 ~]$ dgmgrl
DGMGRL> connect sysdg/oracle_4U@westdb
Connected as SYSDG.
DGMGRL> show database westdb

Database - westdb

Role: PHYSICAL STANDBY
Intended State: APPLY-ON
Transport Lag: 0 seconds (computed 0 seconds ago)
Apply Lag: 0 seconds (computed 1 second ago)
Average Apply Rate: 2.00 KByte/s
Real Time Query: ON
Instance(s):
westdb1 (apply instance)
westdb2

Database Status:
SUCCESS

DGMGRL> exit
[oracle@wnode03 ~]$
```

12. Exit from all your terminal sessions.

Practice 8-2: Installing and Configuring Global Data Services

Overview

This practice will demonstrate the creation of GDS configuration that offers the global services failover/load balancing capabilities.

You will create a GDS configuration that contains the following components:

- GDS catalog hosted in GDSCAT
- Two GDS Regions: EAST region and WEST region
- Two Global Services Managers (GSMs): one GSM per region (GSMEAST and GSMWEST)
- One GDS pool called SALES
- Two GDS pool RAC databases in an Active Data Guard configuration, with one database in each of the datacenters. The `eastdb` database is in the EAST region and the `westdb` database is in the WEST region.

In this practice, the GSMEAST Global Service Manager, GDSCAT, and the `eastdb` database are hosted on CLUSTER01. The GSMWEST Global Service Manager and the `westdb` database are hosted on CLUSTER02.

- **Task 1:** Install Global Service Managers
- **Task 2:** GDS setup and configuration
- **Task 3:** Set up GDS Administrator Accounts & Privileges
- **Task 4:** Configure GDS
- **Task 5:** Create the GDS regions
- **Task 6:** Create the GDS pools
- **Task 7:** Add Data Guard broker configuration to the GDS pool
- **Task 8:** Create and start Global Service in a pool
- **Task 9:** Add TNS entries for clients

Task 1: Install Global Service Managers

1. Install and deploy GDS in your environment. Establish an `ssh` connection to host `enode01` by using the `-X` option as the `oracle` user.

```
[Classroom PC]$ ssh -X oracle@enode01
oracle@enode01's password: <oracle>
[oracle@enode01 ~]$
```

2. Change directory to `/stage/GSM/` and start the installer.

Note: Depending on your environment, you will need to wait to 1-2 minutes.

```
[oracle@enode01 ~]$ cd /stage/12.1/gsm
[oracle@enode01 Disk1]$ ./runInstaller
Starting Oracle Universal Installer...
Checking Temp space: must be greater than 500 MB. Actual 7934
MB Passed
```

```

Checking swap space: must be greater than 150 MB. Actual 8632
MB Passed
Checking monitor: must be configured to display at least 256
colors. Actual 16777216 Passed
Preparing to launch Oracle Universal Installer from
/tmp/OraInstall2013-05-22_03-35-08PM. Please wait ...

```

3. On the Specify Installation Location screen, specify /u01/app/oracle as the Oracle base value and **/u01/app/oracle/product/12.1.0/gsmhome_1** as the Software location. Click Next.
4. On the Summary page, click Install.
5. When the Execute Configuration scripts dialog box appears, open a terminal window to enode01 as the root user and execute the root.sh script.

```

[root@enode01 ~]# /u01/app/oracle/product/12.1.0/gsmhome_1/root.sh
The following environment variables are set as:
ORACLE_OWNER= oracle
ORACLE_HOME= /u01/app/oracle/product/12.1.0/gsmhome_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.
You have new mail in /var/spool/mail/root
[root@enode01 ~]# exit

```

6. When the script has finished running, close the dialog box by clicking OK.
7. To install GDS, repeat steps 1–6 on the wnode03 host.

Task 2: GDS Setup and Configuration

8. **(enode01)** On enode01, there is a precreated single instance database called GDSCAT to host the GDS catalog. Review the following TNS entries for the GDS catalog and pool databases in /u01/app/oracle/product/12.1.0/gsmhome_1/network/admin. Set the environment and use tnsping to test the new connect string.

Note: Perform this task on the GSM hosts, enode01 and wnode03.

```

[oracle@enode01 ~]$ cp /home/oracle/labs/tnsnames_GDS.ora
/u01/app/oracle/product/12.1.0/gsmhome_1/network/admin/tnsnames.ora
[oracle@enode01 ~]$ cat
/u01/app/oracle/product/12.1.0/gsmhome_1/network/admin/tnsnames.ora

GDSCAT =
(DESCRIPTION =
(ADDRESS = (PROTOCOL = TCP)(HOST = enode01.example.com)(PORT = 1521))
(CONNECT_DATA =
(SERVER = DEDICATED)
(SERVICE_NAME = gdscat.example.com)
)
)

```

```

EASTDB,EASTDB.EXAMPLE.COM =
(DESCRIPTION =
  (ADDRESS = (PROTOCOL = TCP)(HOST = cluster01-scan)(PORT = 1521))
  (CONNECT_DATA =
    (SERVER = DEDICATED)
    (SERVICE_NAME = eastdb.example.com)
  )
)

WESTDB,WESTDB.EXAMPLE.COM =
(DESCRIPTION =
  (ADDRESS = (PROTOCOL = TCP)(HOST = cluster02-scan)(PORT = 1521))
  (CONNECT_DATA =
    (SERVER = DEDICATED)
    (SERVICE_NAME = westdb.example.com)
  )
)

[oracle@enode01 ~]$ export
ORACLE_HOME=/u01/app/oracle/product/12.1.0/gsmhome_1
[oracle@enode01 ~]$ export ORACLE_BASE=/u01/app/oracle
[oracle@enode01 ~]$ export PATH=$ORACLE_HOME/bin:$PATH
[oracle@enode01 ~]$ tnsping gdscat
TNS Ping Utility for Linux: Version 12.1.0.2.0 - Production on
22-AUG-2014 16:05:26
Copyright (c) 1997, 2014, Oracle. All rights reserved.
Used parameter files:
Used TNSNAMES adapter to resolve the alias
Attempting to contact (DESCRIPTION = (ADDRESS = (PROTOCOL =
TCP)(HOST = enode01.example.com)(PORT = 1521)) (CONNECT_DATA =
(SERVER = DEDICATED) (SERVICE_NAME = gdscat.example.com)))
OK (90 msec)
[oracle@enode01 ~]$
[oracle@enode01 ~]$ ssh wnode03
oracle@wnode03's password: <oracle>
[oracle@wnode03 ~]$ cp /home/oracle/labs/tnsnames_GDS.ora
/u01/app/oracle/product/12.1.0/gsmhome_1/network/admin/tnsnames.ora
[oracle@wnode03 ~]$ cat
/u01/app/oracle/product/12.1.0/gsmhome_1/network/admin/tnsnames.ora

GDSCAT =
(DESCRIPTION =
  (ADDRESS = (PROTOCOL = TCP)(HOST = enode01.example.com)(PORT = 1521))
  (CONNECT_DATA =
    (SERVER = DEDICATED)
    (SERVICE_NAME = gdscat.example.com)
  )
)

EASTDB,EASTDB.EXAMPLE.COM =
(DESCRIPTION =
  (ADDRESS = (PROTOCOL = TCP)(HOST = cluster01-scan)(PORT = 1521))
  (CONNECT_DATA =
    (SERVER = DEDICATED)
    (SERVICE_NAME = eastdb.example.com)
  )
)

```

```

WESTDB,WESTDB.EXAMPLE.COM =
(DESCRIPTION =
  (ADDRESS = (PROTOCOL = TCP)(HOST = cluster02-scan)(PORT = 1521))
  (CONNECT_DATA =
    (SERVER = DEDICATED)
    (SERVICE_NAME = westdb.example.com)
  )
)

[oracle@wnode03 ~]$ export
ORACLE_HOME=/u01/app/oracle/product/12.1.0/gsmhome_1
[oracle@wnode03 ~]$ export ORACLE_BASE=/u01/app/oracle
[oracle@wnode03 ~]$ export PATH=$ORACLE_HOME/bin:$PATH
[oracle@wnode03 ~]$ tnsping gdscat
TNS Ping Utility for Linux: Version 12.1.0.2.0 - Production on 11-OCT-
2015 19:22:20

Copyright (c) 1997, 2014, Oracle. All rights reserved.

Used parameter files:
Used TNSNAMES adapter to resolve the alias
Attempting to contact (DESCRIPTION = (ADDRESS = (PROTOCOL = TCP)(HOST =
enode01.example.com)(PORT = 1521)) (CONNECT_DATA = (SERVER = DEDICATED)
(SERVICE_NAME = gdscat.example.com)))
OK (0 msec)
[oracle@wnode03 ~]$ exit
[oracle@enode01 ~]$

```

Task 3: Set Up GDS Administrator Accounts and Privileges

9. **(enode01)** Make sure that the listeners and databases westdb, eastdb, and GDSCAT are up. Connect to the pool database eastdb (current Primary) and unlock the GSM user.

```

[oracle@enode01 ~]$ . oraenv
ORACLE_SID = [oracle] ? eastdb
The Oracle base has been set to /u01/app/oracle
[oracle@enode01 ~]$ sqlplus system/oracle_4U@eastdb
SQL> show parameter db_unique

NAME          TYPE        VALUE
-----
db_unique_name      string      eastdb

SQL>
SQL> alter user gsmuser account unlock;
User altered.

SQL> alter user gsmuser identified by oracle_4U;
User altered.
SQL>

```

10. **(enode01)** In this practice, you are hosting the GDS catalog in the GDSCAT database. So, connect to GDSCAT and unlock gsmcatuser and grant GSMADMIN_ROLE to an existing user or a newly created user.

```

SQL> connect system/oracle_4U@gdscat
Connected.

SQL> show parameter db_unique

NAME          TYPE        VALUE
-----
db_unique_name    string      gdscat

SQL> alter user gsmcatuser account unlock;
User altered.

SQL> alter user gsmcatuser identified by oracle_4U;
User altered.

SQL> create user mygdsadmin identified by oracle_4U;
User created.

SQL> grant gsmadmin_role to mygdsadmin;
Grant succeeded.

SQL> exit
[oracle@enode01 ~]$

```

Task 4: Configure GDS

11. (enode01) From the terminal window of the EAST GSM node, enode01, set the following environment variables. Using the GDSCTL interface, the user with GSMADMIN_ROLE creates the GDS catalog.

```

[oracle@enode01 ~]$ export
ORACLE_HOME=/u01/app/oracle/product/12.1.0/gsmhome_1
[oracle@enode01 ~]$ export ORACLE_BASE=/u01/app/oracle
[oracle@enode01 ~]$ export PATH=$ORACLE_HOME/bin:$PATH
[oracle@enode01 ~]$ gdsctl
GDSCTL: Version 12.1.0.2.0

Copyright (c) 2011, 2012, Oracle. All rights reserved.

Welcome to GDSCTL, type "help" for information.

Current GSM is set to GSMORA
GDSCTL> create catalog -database enode01:1521:gdscat -user
mygdsadmin/oracle_4U
Catalog is created
GDSCTL>

```

12. (enode01) Add the East Global Service Manager (GSM).

```

GDSCTL> add gsm -gsm gsmeast -listener 1571 -catalog
enode01:1521:gdscat
"gsmcatuser" password: <oracle_4U>
Create credential oracle.security.client.connect_string1
GSM successfully added
GDSCTL>

```

Because the `-pwd` parameter has not been provided, GDSCTL asks for the `gsmcatuser` password. Port 1571 is specified for the GSM listener of GSMEAST.

13. (**enode01**) Start the East Global Service Manager (gsmeast) on enode01.

```
GDSCTL> start gsm -gsm gsmeast
GSM is started successfully
GDSCTL> config gsm -gsm gsmeast
Name: gsmeast
Endpoint 1:
(ADDRESS=(HOST=enode01.example.com) (PORT=1571) (PROTOCOL=tcp))
Local ONS port: 6123
Remote ONS port: 6234
ORACLE_HOME path: /u01/app/oracle/product/12.1.0/gsmhome_1
GSM Host name: enode01.example.com
Region: regionora

Buddy
-----
GDSCTL> status gsm -gsm gsmeast
Alias           GSMEAST
Version         12.1.0.2.0
Start Date      06-OCT-2015 22:06:34
Trace Level     off
Listener Log File
/u01/app/oracle/diag/gsm/enode01/gsmeast/alert/log.xml
Listener Trace File
/u01/app/oracle/diag/gsm/enode01/gsmeast/trace/ora_6259_139672824591808
.trc
Endpoint summary
(ADDRESS=(HOST=enode01.example.com) (PORT=1571) (PROTOCOL=tcp))
GSMOCI Version   0.1.11
Mastership       Y
Connected to GDS catalog Y
Process Id       6381
Number of reconnections 0
Pending tasks. Total 0
Tasks in process. Total 0
Regional Mastership TRUE
Total messages published 0
Time Zone        +00:00
Orphaned Buddy Regions:
    None
GDS region       regionora

GDSCTL>
```

14. (**wnode03**) Open a terminal window on wnode03 as the `oracle` user. Then set the GSM environment, and create the West Global Service Manager (GSM) on wnode03.

```
[Classroom PC]$ ssh oracle@wnode03
oracle@wnode03's password: <oracle>
[oracle@wnode03 ~]$ export
ORACLE_HOME=/u01/app/oracle/product/12.1.0/gsmhome_1
[oracle@wnode03 ~]$ export ORACLE_BASE=/u01/app/oracle
[oracle@wnode03 ~]$ export PATH=$ORACLE_HOME/bin:$PATH
[oracle@wnode03 ~]$ gdsctl
```

```
GDSCTL: Version 12.1.0.2.0 - Production on Fri Aug 22 18:12:45
UTC 2014
```

```
Copyright (c) 2011, 2014, Oracle. All rights reserved.
```

```
Welcome to GDSCTL, type "help" for information.
```

```
Current GSM is set to GSMORA
GDSCTL> add gsm -gsm gsmwest -listener 1572 -catalog
enode01:1521:gdsctl
"gsmcatuser" password: <oracle_4U>
Create credential oracle.security.client.connect_string1
GSM successfully added
GDSCTL>
```

Because the `-pwd` parameter has not been provided, GDSCTL asks for the `gsmcatuser` password. Port 1572 is specified for the GSM listener of `GSMWEST`.

15. (wnode03) Start the newly created GSM. Check the GSM configuration and status after starting `GSMWEST`.

```
GDSCTL> start gsm -gsm gsmwest
GSM is started successfully
GDSCTL> config gsm -gsm gsmwest
Name: gsmwest
Endpoint 1:
(ADDRESS=(HOST=wnode03.example.com)(PORT=1572)(PROTOCOL=tcp))
Local ONS port: 6123
Remote ONS port: 6234
ORACLE_HOME path: /u01/app/oracle/product/12.1.0/gsmhome_1
GSM Host name: wnode03.example.com
Region: regionora

Buddy
-----
GDSCTL> status gsm -gsm gsmwest
Alias           GSMWEST
Version        12.1.0.2.0
Start Date     06-OCT-2015 22:14:16
Trace Level    off
Listener Log File
/u01/app/oracle/diag/gsm/wnode03/gsmwest/alert/log.xml
Listener Trace File
/u01/app/oracle/diag/gsm/wnode03/gsmwest/trace/ora_25363_14055190760390
4.trc
Endpoint summary
(ADDRESS=(HOST=wnode03.example.com)(PORT=1572)(PROTOCOL=tcp))
GSMOCI Version  0.1.11
Mastership      N
Connected to GDS catalog Y
Process Id      25377
Number of reconnections 0
Pending tasks. Total 0
Tasks in process. Total 0
Regional Mastership FALSE
Total messages published 0
```

```

Time Zone           +00:00
Orphaned Buddy Regions:
    None
GDS region          regionora

GDSCTL>

```

Task 5: Create the GDS Regions

16. **(enode01)** In the enode01 terminal window, create two regions, EAST and WEST.

```

GDSCTL> add region -region west, east
GDSCTL>

```

17. **(enode01)** Assign the east region to the east GSM.

```

GDSCTL> modify gsm -gsm gsmeast -region east
GSM modified
GDSCTL>

```

18. **(wnode03)** From the West GSM terminal (wnode03), assign the west region to the west GSM.

```

GDSCTL> connect mygdsadmin/oracle_4U@enode01:1521/gdscat.example.com
Catalog connection is established
GDSCTL> modify gsm -gsm gsmwest -region west
GSM modified
GDSCTL> exit
[oracle@wnode03 ~]$

```

Task 6: Create the GDS Pools

19. **(enode01)** Return to the terminal session on enode01. Because you need just one gdspool, you can use either the predefined dbpoolora or create a new one called SALES.

```

GDSCTL> add gdspool -gdspool sales
GDSCTL> exit
[oracle@enode01 ~]$

```

Task 7: Add Data Guard Broker Configuration to the GDS Pool

20. **(enode01)** In the same terminal session, add the Data Guard broker configuration to the SALES gdspool. Using DGMGRL, first check how the Data Guard configuration looks like from one of the database nodes. Use DGMGRL to check the databases to confirm that redo transport and apply are active.

```

[oracle@enode01 ~]$ . oraenv
ORACLE_SID = [eastdb] ? eastdb
The Oracle base remains unchanged with value /u01/app/oracle
[oracle@enode01 ~]$ export ORACLE_SID=eastdb1
[oracle@enode01 ~]$ dgmgrl
DGMGRL> connect sysdg/oracle_4U@eastdb
Connected as SYSDG.
DGMGRL> show configuration

Configuration - dg_config

```

```

Protection Mode: MaxPerformance
Members:
  eastdb - Primary database
  westdb - Physical standby database

Fast-Start Failover: DISABLED

Configuration Status:
SUCCESS      (status updated 0 seconds ago)

DGMGRL>
DGMGRL> show database eastdb

Database - eastdb

Role:           PRIMARY
Intended State: TRANSPORT-ON
Instance(s):
  eastdb1
  eastdb2

Database Status:
SUCCESS

DGMGRL> show database westdb

Database - westdb

Role:           PHYSICAL STANDBY
Intended State: APPLY-ON
Transport Lag:  0 seconds (computed 0 seconds ago)
Apply Lag:     0 seconds (computed 0 seconds ago)
Average Apply Rate: 2.00 KByte/s
Real Time Query: ON
Instance(s):
  westdb1 (apply instance)
  westdb2

Database Status:
SUCCESS

DGMGRL> exit
[oracle@enode01 ~]$

```

21. **(enode01)** Check the open status of the primary database instances to avoid ORA-12154 in step 23.

Note: The status of the primary instances must be OPEN. If the instance is not OPEN, but MOUNT, open it manually.

```
[oracle@enode01 ~]$ srvctl status database -db eastdb -verbose
Instance eastdb1 is running on node enode01. Instance status: Open.
Instance eastdb2 is running on node enode02. Instance status: Open.
[oracle@enode01 ~]$
```

22. (**wnode03**) Return to the terminal on wnode03 to check the OPEN status of the standby database instances to avoid ORA-12154 in step 23.

Note: The status of the standby instances must be OPEN. If the instance is not OPEN, but MOUNT, open it manually.

```
[oracle@wnode03 ~]$ srvctl status database -db westdb -verbose
Instance westdb1 is running on node wnode03. Instance status:
Open,Readonly.
Instance westdb2 is running on node wnode04. Instance status:
Open,Readonly.
[oracle@wnode03 ~]$
```

23. (**enode01**) From the enode01 terminal, set your environment for GSM and add the broker configuration by using GDSCTL. Always add the broker configuration connecting to the current PRIMARY database. In this case, eastdb is the PRIMARY database.

```
[oracle@enode01 ~]$ export
ORACLE_HOME=/u01/app/oracle/product/12.1.0/gsmhome_1
[oracle@enode01 ~]$ export ORACLE_BASE=/u01/app/oracle
[oracle@enode01 ~]$ export PATH=$ORACLE_HOME/bin:$PATH
[oracle@enode01 ~]$ gdsctl
GDSCTL: Version 12.1.0.2.0 - Production on Fri Aug 22 18:39:20
UTC 2014

Copyright (c) 2011, 2014, Oracle. All rights reserved.

Welcome to GDSCTL, type "help" for information.

Current GSM is set to GSMEAST
GDSCTL> add brokerconfig -connect enode01:1521:eastdb1 -region east -
gdspool sales
"gsmuser" password: <oracle_4U>
DB Unique Name: eastdb

GDSCTL>
```

Note: If the add brokerconfig command fails with ORA-12154, it could be because of the tnsnames.ora settings in step 8. Go back to step 8 to review the tnsnames.ora settings in enode01 and wnode03. For more information, review the support note (Doc ID 1641409.1) – “In order to use the “add brokerconfig”, the same tnsnames alias has to be configured in the GSM home too, under \$ORACLE_HOME/network/admin, or where the \$TNS_ADMIN pointed.”

24. (**enode01**) From the GDSCTL prompt on enode01, assign the westdb database to the WEST region.

```
GDSCTL> modify database -database westdb -region west -gdspool sales -
connect wnode03:1521/westdb.example.com
"gsmuser" password: <oracle_4U>
GDSCTL>
```

25. (**enode01**) From the GDSCTL prompt on enode01, check whether the databases are registered to the GSM.

```
GDSCTL> databases
Database: "eastdb" Registered: Y State: Ok ONS: Y. Role: PRIMARY
Instances: 2 Region: east
```

```

Registered instances:
  sales%1
  sales%2
Database: "westdb" Registered: Y State: Ok ONS: Y. Role: PH_STNDBY
Instances: 2 Region: west
  Registered instances:
    sales%11
    sales%12
GDSCTL>

```

Task 8: Create and Start Global Service in a Pool

26. (enode01) From the GDSCTL prompt on enode01, create a service called SALES_REPORTING_SRVC, which should always run on the standby database. In the event the standby database is not available, the service can run on the primary database. When the service has been created, use the modify service command to add instances to the respective databases (because they are administrator managed). Start the new service and check the status.

```

GDSCTL> add service -service sales_reporting_srvc -gdspool sales -
preferred_all -role PHYSICAL_STANDBY -failover_primary

GDSCTL> modify service -gdspool sales -service sales_reporting_srvc -
database westdb -add_instances -preferred westdb1,westdb2

GDSCTL> modify service -gdspool sales -service sales_reporting_srvc -
database eastdb -add_instances -preferred eastdb1,eastdb2

GDSCTL> start service -service sales_reporting_srvc -gdspool sales

GDSCTL> status service -service sales_reporting_srvc
Service "sales_reporting_srvc.sales.oradbcloud" has 2 instance(s).
Affinity: ANYWHERE
  Instance "sales%11", name: "westdb1", db: "westdb", region: "west",
status: ready.
  Instance "sales%12", name: "westdb2", db: "westdb", region: "west",
status: ready.
GDSCTL>

```

27. (enode01) View the configuration information of the newly created service.

```

GDSCTL> config service -service sales_reporting_srvc
Name: sales_reporting_srvc
Network name: sales_reporting_srvc.sales.oradbcloud
Pool: sales
Started: Yes
Preferred all: Yes
Locality: ANYWHERE
Region Failover: No
Role: PHYSICAL_STANDBY
Primary Failover: Yes
Lag: ANY
Runtime Balance: SERVICE_TIME
Connection Balance: LONG
Notification: Yes
TAF Policy: NONE

```

```

Policy: AUTOMATIC
DTP: No
Failover Method: NONE
Failover Type: NONE
Failover Retries:
Failover Delay:
Edition:
PDB:
Commit Outcome:
Retention Timeout:
Replay Initiation Timeout:
Session State Consistency:
SQL Translation Profile:

Databases
-----
Database          Preferred Status
-----
eastdb           Yes      Enabled
westdb           Yes      Enabled

GDSCTL>

```

28. **(enode01)** Display the configuration information of all the components that are part of your GDS configuration.

```

GDSCTL> config
Regions
-----
Name          Buddy
-----
regionora
west
east

GSMs
-----
gsmeast
gsmwest

GDS pools
-----
dbpoolora
sales

Databases
-----
eastdb
westdb

Services
-----
sales_reporting_srvc

GDSCTL pending requests
-----
Command          Object          Status

```

```
-----
Global properties
-----
Name: oradbcloud
Master GSM: gsmeast

GDSCTL> exit
[oracle@enode01 ~]$
```

29. **(wnode03)** Return to the terminal session on wnode03. Now that the global service has been created and started, you can query DBA_SERVICES and V\$ACTIVE_SERVICES to learn more about global services. So, connect to the standby database where the global service is currently running:

```
[oracle@wnode03 ~]$ . oraenv
ORACLE_SID = [oracle] ? westdb
The Oracle base remains unchanged with value /u01/app/oracle
[oracle@wnode03 ~]$ sqlplus sys/oracle_4U@westdb as sysdba
SQL> column name format a30
SQL> column network_name format a40
SQL> column global format a10
SQL> set linesize 120
SQL> select name, network_name, global_service from dba_services;

NAME          NETWORK_NAME          GLO
-----
SYS$BACKGROUND                               NO
SYS$USERS                                    NO
eastdb_DGB         eastdb_DGB          NO
westdb_DGB        westdb_DGB          NO
eastdbXDB        eastdbXDB          NO
eastdb.example.com  eastdb.example.com NO
westdbXDB        westdbXDB          NO
westdb.example.com  westdb.example.com NO
actest           actest            NO
sales_reporting_srvc      sales_reporting_srvc.sales.oradbcloud
YES

10 rows selected.
```

Note: For sales_reporting_srvc, the value of the column GLOBAL_SERVICE is “Yes”, denoting that it is a global service.

```
SQL> select name, network_name, global from v$active_services;
```

NAME	NETWORK_NAME	GLOBAL
sales_reporting_srvc	sales_reporting_srvc.sales.oradbcloud	YES
westdb_DGB	westdb_DGB	NO
westdbXDB	westdbXDB	NO
westdb.example.com	westdb.example.com	NO
SYS\$BACKGROUND		NO
SYS\$USERS		NO

```
6 rows selected.
```

Note: The query above shows that currently the `sales_reporting_srvc` global service is listed in the `v$active_services` because it has been started.

```
SQL> exit
[oracle@wnode03 ~]$
```

Task 9: Add TNS Entries for Clients

30. **(enode01)** Return to the terminal session on enode01. Add the TNS entries (based on GSM listeners) to the client's `tnsnames.ora`. Here is an example TNS entry for the "sales_reporting_srvc" application clients, where `SERVICE_NAME` is the name of the global service and `REGION` is the region that the client is coming from. This entry should appear in the `tnsnames.ora` file on any host from which a client connection to the service is required. However, in this example, you will update `tnsnames.ora` on enode01 only for the sake of simplicity and originate your connections from there.

Note: The connect descriptor in `tnsnames.ora` in a GDS configuration will use the GSM listener end points and not the RAC SCAN listeners or local listeners. When you copy and paste the following TNS entry, you may encounter an error later due to spacing issue. So, try to format the TNS entry as shown in the following, leaving spaces in front of each line.

```
[oracle@enode01 ~]$ vi
/u01/app/oracle/product/12.1.0/dbhome_1/network/admin/tnsnames.ora

***** Add entry below *****

sales_reporting_srvc =
(DESCRIPTION =
(ADDRESS_LIST =
(ADDRESS = (PROTOCOL = TCP)(HOST = enode01)(PORT = 1571))
(ADDRESS_LIST =
(ADDRESS = (PROTOCOL = TCP)(HOST = wnode03)(PORT = 1572))
(CONNECT_DATA =
(SERVICE_NAME = sales_reporting_srvc.sales.oradbcloud)
(REGION=WEST)
)
)
)

Note: 1572 is the GSM listener port on wnode03 and 1571 is the GSM
listener port on enode01

wq!

[oracle@enode01 ~]$
```

31. (**enode01**) Clients can connect to the pool databases either via the Easy Connect Naming or with TNSNAMES. In a terminal window on enode01, set the environment and connect to the service as shown in the following:

```
[oracle@enode01 ~]$ . oraenv
ORACLE_SID = [oracle] ? eastdb
The Oracle base has been set to /u01/app/oracle
[oracle@enode01 ~]$
[oracle@enode01 ~]$ ### Via Easy Connect Naming Method:
[oracle@enode01 ~]$
[oracle@enode01 ~]$ sqlplus
system/oracle_4U@//enode01:1571/sales_reporting_srvc.sales.oradbcloud

SQL> select instance_name from v$instance;

INSTANCE_NAME
-----
westdb1

SQL> -- Via TNSNAMES Connect Descriptor:
SQL>
SQL> connect system/oracle_4U@sales_reporting_srvc
Connected.
SQL> select instance_name from v$instance;

INSTANCE_NAME
-----
westdb2

SQL> exit
[oracle@enode01 ~]$
```

32. Exit from all your terminal sessions.

Practice 8-3: Global Service Failover

Overview

To see how the failover of global services happens, you will shut down one of the databases and observe that the services are automatically started in the other pool database.

Terminal Sessions

- **Session 1:** ssh oracle@enode01
- **Session 2:** ssh oracle@wnode03



Tasks

1. **(Session 1)** Open a terminal window on enode01 as the oracle user.

```
[Classroom PC]$ ssh oracle@enode01
oracle@enode01's password: <oracle>
[oracle@enode01 ~]$
```

2. Using GDSCTL, observe that the sales_reporting_srvc global service is currently running on the standby database, westdb, per the service attributes that you have defined.

```
[oracle@enode01 ~]$ export
ORACLE_HOME=/u01/app/oracle/product/12.1.0/gsmhome_1
[oracle@enode01 ~]$ export ORACLE_BASE=/u01/app/oracle
[oracle@enode01 ~]$ export PATH=$ORACLE_HOME/bin:$PATH
[oracle@enode01 ~]$ gdsctl
GDSCTL: Version 12.1.0.2.0 - Production on Wed Oct 07 17:58:25 UTC 2015

Copyright (c) 2011, 2014, Oracle. All rights reserved.

Welcome to GDSCTL, type "help" for information.

Current GSM is set to GSMEAST
GDSCTL> services
Service "sales_reporting_srvc.sales.oradbcloud" has 2 instance(s).
Affinity: ANYWHERE
  Instance "sales%11", name: "westdb1", db: "westdb", region: "west",
  status: ready.
  Instance "sales%12", name: "westdb2", db: "westdb", region: "west",
  status: ready.
GDSCTL>
GDSCTL> databases
Database: "eastdb" Registered: Y State: Ok ONS: Y. Role: PRIMARY
Instances: 2 Region: east
```

```

Service: "sales_reporting_srvc" Globally started: Y Started: N
          Scan: N Enabled: Y Preferred: Y
Registered instances:
  sales%1
  sales%2
Database: "westedb" Registered: Y State: Ok ONS: Y. Role: PH_STNDBY
Instances: 2 Region: west
  Service: "sales_reporting_srvc" Globally started: Y Started: Y
          Scan: N Enabled: Y Preferred: Y
  Registered instances:
    sales%11
    sales%12
GDSCTL>

```

3. **(Session 2)** From the wnode03 terminal, stop the pool database, westedb.

```

[Classroom PC]$ ssh oracle@wnode03
oracle@wnode03's password: <oracle>
[oracle@wnode03 ~]$ . oraenv
ORACLE_SID = [oracle] ? westedb
The Oracle base has been set to /u01/app/oracle
[oracle@wnode03 ~]$ srvctl stop database -db westedb
[oracle@wnode03 ~]$

```

4. **(Session 1)** Return to the terminal session on enode01. Observe that the sales_reporting_srvc global service is automatically failed over to the eastedb database as per the Service attributes that you have defined.

```

On enode01:

GDSCTL> services
Service "sales_reporting_srvc.sales.oradbcloud" has 2 instance(s).
Affinity: ANYWHERE
  Instance "sales%1", name: "eastedb1", db: "eastedb", region: "east",
  status: ready.
  Instance "sales%2", name: "eastedb2", db: "eastedb", region: "east",
  status: ready.
GDSCTL>
GDSCTL> databases
Database: "eastedb" Registered: Y State: Ok ONS: Y. Role: PRIMARY
Instances: 2 Region: east
  Service: "sales_reporting_srvc" Globally started: Y Started: Y
          Scan: Y Enabled: Y Preferred: Y
  Registered instances:
    sales%1
    sales%2
Database: "westedb" Registered: Y State: Warnings ONS: Y. Role: N/A
Instances: 0 Region: west
  Service: "sales_reporting_srvc" Globally started: Y Started: N
          Scan: N Enabled: Y Preferred: Y
GDSCTL>

```

5. **(Session 1)** In the same terminal session on enode01, stop and remove the service that was created for this test case.

Note: You may or may not receive warning messages, such as ORA-12514 or ORA-44894, when removing the global service. You can safely ignore it.

On enode01:

```
GDSCTL> stop service -service sales_reporting_srvc -gdspool sales -  
database eastdb  
Catalog connection is established  
  
GDSCTL> remove service -gdspool sales -service sales_reporting_srvc  
  
GDSCTL> services  
  
GDSCTL> exit  
[oracle@enode01 ~]$
```

6. **(Session 2)** Return to the terminal session on wnode03. Bring the westdb database up.

```
[oracle@wnode03 ~]$ srvctl start database -d westdb -startoption open  
[oracle@wnode03 ~]$ srvctl status database -db westdb -verbose  
Instance westdb1 is running on node wnode03. Instance status:  
Open, Readonly.  
Instance westdb2 is running on node wnode04. Instance status:  
Open, Readonly.  
[oracle@wnode03 ~]$
```

7. Do NOT close the current terminal sessions for the next practice.

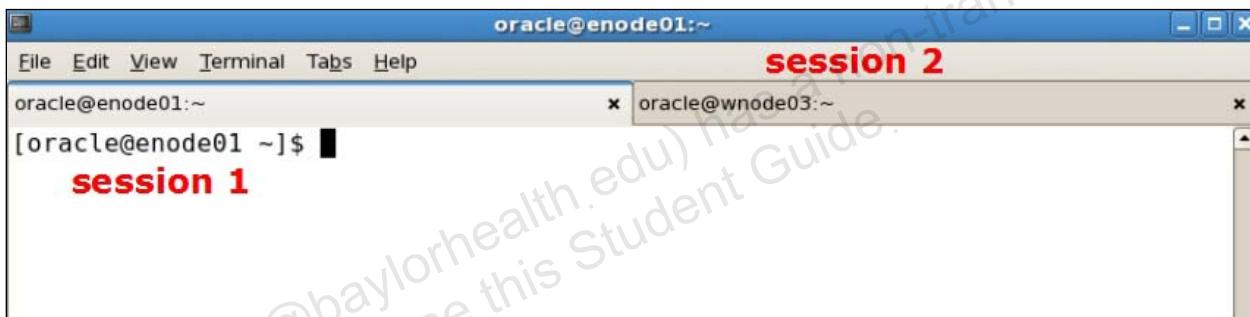
Practice 8-4: Role-Based Global Services

Overview

When a Data Guard role transition is performed either manually or via Fast-Start Failover, GDS automatically relocates the global services based on the role of the databases. GDS does this without Oracle Clusterware. To comprehend how the role-based global services function, create two global services: `sales_entry_srvc` targeted to run on the Primary (`eastdb`) and `sales_adhoc_srvc` targeted to run on the standby (`westdb`). Then execute the Data Guard switchover operation and observe that after the role change, the `sales_entry_srvc` global service automatically gets relocated to the new primary (`westdb`) and `sales_adhoc_srvc` to the new Standby (`eastdb`).

Terminal Sessions

- **Session 1:** `ssh oracle@enode01`
- **Session 2:** `ssh oracle@wnode03`



Tasks

1. **(Session 1)** Return to the terminal window on `enode01` as `oracle` and configure your environment.

```
[oracle@enode01 ~]$ . oraenv
ORACLE_SID = [oracle] ? eastdb
The Oracle base has been set to /u01/app/oracle
[oracle@enode01 ~]$
```

2. In preparation for the practice, verify the status of the Data Guard configuration. Wait until the Physical Standby Database is synchronized. It may take a few moments for Lag for the standby database to clear.

Note: Do not continue with the practices until the Apply Lag is cleared.

```
[oracle@enode01 ~]$ dgmgrl
DGMGRL> connect sysdg/oracle_4U@eastdb
Connected as SYSDG.
DGMGRL> show database westdb

Database - westdb

Role:          PHYSICAL STANDBY
Intended State:  APPLY-ON
```

```

Transport Lag:          0 seconds (computed 0 seconds ago)
Apply Lag:          0 seconds (computed 0 seconds ago)
Average Apply Rate: 14.00 KByte/s
Real Time Query:      ON
Instance(s):
  westdb1 (apply instance)
  westdb2

Database Status:
SUCCESS
DGMGRL> exit
[oracle@enode01 ~]$
```

3. Perform the connection testing to eastdb1, eastdb2, westdb1, and westdb2. If a connection to any instance hangs, restart the hung instance. Otherwise, step 4 will fail due to the connectivity issue.

```

[oracle@enode01 ~]$ sqlplus sys/oracle_4U@eastdb1 as sysdba
SQL> connect sys/oracle_4U@eastdb2 as sysdba
Connected.
SQL> connect sys/oracle_4U@westdb1 as sysdba
Connected.
SQL> connect sys/oracle_4U@westdb2 as sysdba
Connected.
SQL> exit
[oracle@enode01 ~]$
```

4. **(Session 1)** In the same terminal, create the global services sales_entry_srvc and sales_adhoc_srvc using GDSCTL. Add instances as shown in the following:

```

[oracle@enode01 ~]$ export
ORACLE_HOME=/u01/app/oracle/product/12.1.0/gsmhome_1
[oracle@enode01 ~]$ export ORACLE_BASE=/u01/app/oracle
[oracle@enode01 ~]$ export PATH=$ORACLE_HOME/bin:$PATH
[oracle@enode01 ~]$ gdsctl
GDSCTL: Version 12.1.0.2.0 - Production on Wed Oct 07 18:26:30 UTC 2015
Copyright (c) 2011, 2014, Oracle. All rights reserved.

Welcome to GDSCTL, type "help" for information.

Current GSM is set to GSMEAEST
GDSCTL> add service -service sales_entry_srvc -gdspool sales -preferred_all -role PRIMARY
Catalog connection is established

GDSCTL> modify service -gdspool sales -service sales_entry_srvc -database eastdb -add_instances -preferred eastdb1 -available eastdb2
GDSCTL> modify service -gdspool sales -service sales_entry_srvc -database westdb -add_instances -preferred westdb1 -available westdb2
GDSCTL>
GDSCTL> start service -service sales_entry_srvc -gdspool sales
GDSCTL>
GDSCTL> services
Service "sales_entry_srvc.sales.oradbcloud" has 1 instance(s).
Affinity: ANYWHERE
  Instance "sales%1", name: "eastdb1", db: "eastdb", region: "east",
status: ready.
```

```

GDSCTL>
GDSCTL> add service -service sales_adhoc_srvc -gdspool sales -
preferred_all -role PHYSICAL_STANDBY
GDSCTL>
GDSCTL> modify service -gdspool sales -service sales_adhoc_srvc -
database westdb -add_instances -preferred westdb2 -available westdb1
GDSCTL>
GDSCTL> modify service -gdspool sales -service sales_adhoc_srvc -
database eastdb -add_instances -preferred eastdb2 -available eastdb1
GDSCTL>
GDSCTL> start service -service sales_adhoc_srvc -gdspool sales

GDSCTL> services
Service "sales_adhoc_srvc.sales.oradbcloud" has 1 instance(s).
Affinity: ANYWHERE
Instance "sales%12", name: "westdb2", db: "westdb", region:
"west", status: ready.
Service "sales_entry_srvc.sales.oradbcloud" has 1 instance(s).
Affinity: ANYWHERE
Instance "sales%1", name: "eastdb1", db: "eastdb", region:
"east", status: ready.
GDSCTL>
GDSCTL> databases
Database: "eastdb" Registered: Y State: Ok ONS: Y. Role: PRIMARY
Instances: 2 Region: east
Service: "sales_adhoc_srvc" Globally started: Y Started: N
Scan: Y Enabled: Y Preferred: Y
Service: "sales_entry_srvc" Globally started: Y Started: Y
Scan: N Enabled: Y Preferred: Y
Registered instances:
sales%1
sales%2
Database: "westdb" Registered: Y State: Ok ONS: Y. Role:
PH_STNDBY Instances: 2 Region: west
Service: "sales_adhoc_srvc" Globally started: Y Started: Y
Scan: Y Enabled: Y Preferred: Y
Service: "sales_entry_srvc" Globally started: Y Started: N
Scan: N Enabled: Y Preferred: Y
Registered instances:
sales%11
sales%12
GDSCTL>
```

5. **(Session 2)** Return to the terminal session on wnode03. With DGMGRL, perform the Data Guard switchover. In this example, the switchover is performed from wnode03.

```

[oracle@wnode03 ~]$ dgmgrl
DGMGRL> connect sysdg/oracle_4U@westdb
Connected as SYSDG.

DGMGRL> validate database eastdb

Database Role: Primary database

Ready for Switchover: Yes

DGMGRL> validate database westdb
```

```

Database Role: Physical standby database
Primary Database: eastdb

Ready for Switchover: Yes
Ready for Failover: Yes (Primary Running)

Transport-Related Property Settings:
Property      eastdb Value          westdb Value
RedoRoutes    (eastdb:westdb ASYNC) (westdb:eastdb ASYNC)

DGMGRL> switchover to westdb
Performing switchover NOW, please wait...
Operation requires a connection to instance "westdb2" on
database "westdb"
Connecting to instance "westdb2"...
Connected as SYSDBA.
New primary database "westdb" is opening...
Oracle Clusterware is restarting database "eastdb" ...
Switchover succeeded, new primary is "westdb"
DGMGRL>

```

6. **(Session 2)** Determine the status of the Data Guard configuration.

Note: You may or may not see warning messages, such as ORA-16525, ORA-16665, ORA-16778, ORA-16810, or ORA-16817, due to the lag and overall performance of the virtual machine environment. The warning messages will be cleared in a few minutes. Make sure that your Data Guard configuration status is still fine.

```

DGMGRL> show configuration

Configuration - dg_config

Protection Mode: MaxPerformance
Members:
westdb - Primary database
eastdb - Physical standby database

Fast-Start Failover: DISABLED

Configuration Status:
SUCCESS (status updated 6 seconds ago)

DGMGRL>

```

7. **(Session 2)** Wait until the Physical Standby Database is synchronized. It may take a few moments for the Lag for the standby database to clear.

Note: Do not continue with the practices until the Apply Lag is cleared.

```

DGMGRL> show database eastdb

Database - eastdb

Role: PHYSICAL STANDBY
Intended State: APPLY-ON
Transport Lag: 0 seconds (computed 1 second ago)
Apply Lag: 0 seconds (computed 2 seconds ago)
Average Apply Rate: 17.00 KByte/s
Real Time Query: ON

```

```

Instance(s):
  eastdb1 (apply instance)
  eastdb2

Database Status:
SUCCESS

DGMGRL> exit
[oracle@wnode03 ~]$
```

8. **(Session 1)** Return to the terminal session on enode01. Observe that the sales_entry_srvc global service is automatically relocated to the new Primary (westdb) and the sales_adhoc_srvc global service to the new standby (eastdb).

```

GDSCTL> services
Service "sales_adhoc_srvc.sales.oradbcloud" has 1 instance(s).
Affinity: ANYWHERE
Instance "sales%2", name: "eastdb2", db: "eastdb", region:
"east", status: ready.
Service "sales_entry_srvc.sales.oradbcloud" has 1 instance(s).
Affinity: ANYWHERE
Instance "sales%11", name: "westdb1", db: "westdb", region:
"west", status: ready.
GDSCTL>
GDSCTL> databases
Database: "eastdb" Registered: Y State: Ok ONS: Y. Role:
PH_STNDBY Instances: 2 Region: east
Service: "sales_adhoc_srvc" Globally started: Y Started: Y
Scan: N Enabled: Y Preferred: Y
Service: "sales_entry_srvc" Globally started: Y Started: N
Scan: N Enabled: Y Preferred: Y
Registered instances:
sales%1
sales%2
Database: "westdb" Registered: Y State: Ok ONS: Y. Role: PRIMARY
Instances: 2 Region: west
Service: "sales_adhoc_srvc" Globally started: Y Started: N
Scan: N Enabled: Y Preferred: Y
Service: "sales_entry_srvc" Globally started: Y Started: Y
Scan: N Enabled: Y Preferred: Y
Registered instances:
sales%11
sales%12
GDSCTL>
```

9. **(Session 1)** At this point, you may stop and remove the services that you have created for this exercise. Note that the `sales_entry_srvc` service was running in the `westdb` database and the `sales_adhoc_srvc` service was running in the `eastdb` database.

```
GDSCTL> stop service -service sales_entry_srvc -gdspool sales -database westdb
GDSCTL> remove service -gdspool sales -service sales_entry_srvc
GDSCTL> stop service -service sales_adhoc_srvc -gdspool sales -database eastdb
GDSCTL> remove service -gdspool sales -service sales_adhoc_srvc
GDSCTL> exit
[oracle@enode01 ~]$
```

10. Do NOT close the current terminal sessions for the next practice.

Practice 8-5: Replication Lag-Based Routing

Overview

Sometimes the Data Guard standby databases may lag behind the primary database due to various reasons. If the replication lag exceeds the lag limit, the global service is relocated to another available database that lags below the threshold. With the `-failover_primary` clause, you can even relocate the service to the Primary database.

Terminal Sessions

- **Session 1:** `ssh oracle@enode01`
- **Session 2:** `ssh oracle@wnode03`



Tasks

1. **(Session 1)** Return to the terminal session on `enode01`. To understand Lag tolerance-based routing, create a global service `sales_reader_lag15_srvc` and set the `-lag` attribute to 15 seconds. Add instances using the `modify service` command as shown below. You will artificially create the lag by turning off the Apply Process. You then will observe that once the lag exceeds the 15 seconds threshold, GDS automatically relocates the global service to the Primary (because you used the `-failover_primary` clause).

```
[oracle@enode01 ~]$ export
ORACLE_HOME=/u01/app/oracle/product/12.1.0/gsmhome_1
[oracle@enode01 ~]$ export ORACLE_BASE=/u01/app/oracle
[oracle@enode01 ~]$ export PATH=$ORACLE_HOME/bin:$PATH
[oracle@enode01 ~]$ gdsctl
GDSCTL: Version 12.1.0.2.0 - Production on Fri Aug 22 18:39:20
UTC 2014

Copyright (c) 2011, 2014, Oracle. All rights reserved.

Welcome to GDSCTL, type "help" for information.

Current GSM is set to GSMEAEST
GDSCTL> add service -service sales_reader_lag15_srvc -gdspool sales -
preferred_all -role PHYSICAL_STANDBY -lag 15 -failover_primary
GDSCTL>
GDSCTL> modify service -gdspool sales -service sales_reader_lag15_srvc
-database westdb -add_instances -preferred westdb1,westdb2
GDSCTL>
```

```

GDSCTL> modify service -gdspool sales -service sales_reader_lag15_srvc
-database eastdb -add_instances -preferred eastdb1,eastdb2
GDSCTL>
GDSCTL> start service -service sales_reader_lag15_srvc -gdspool sales
GDSCTL>
GDSCTL> services
Service "sales_reader_lag15_srvc.sales.oradbcloud" has 2 instance(s).
Affinity: ANYWHERE
  Instance "sales%11", name: "westdb1", db: "westdb", region: "west",
status: ready.
  Instance "sales%12", name: "westdb2", db: "westdb", region: "west",
status: ready.
GDSCTL>
GDSCTL> databases
Database: "eastdb" Registered: Y State: Ok ONS: Y. Role: PRIMARY
Instances: 2 Region: east
  Service: "sales_reader_lag15_srvc" Globally started: Y Started: N
    Scan: N Enabled: Y Preferred: Y
  Registered instances:
    sales%1
    sales%2
Database: "westdb" Registered: Y State: Ok ONS: Y. Role: PH_STNDBY
Instances: 2 Region: west
  Service: "sales_reader_lag15_srvc" Globally started: Y Started: Y
    Scan: N Enabled: Y Preferred: Y
  Registered instances:
    sales%11
    sales%12

GDSCTL> config service -service sales_reader_lag15_srvc
Name: sales_reader_lag15_srvc
Network name: sales_reader_lag15_srvc.sales.oradbcloud
Pool: sales
Started: Yes
Preferred all: Yes
Locality: ANYWHERE
Region Failover: No
Role: PHYSICAL_STANDBY
Primary Failover: Yes
Lag: 15
Runtime Balance: SERVICE_TIME
Connection Balance: LONG
Notification: Yes
TAF Policy: NONE
Policy: AUTOMATIC
DTP: No
Failover Method: NONE
Failover Type: NONE
Failover Retries:
Failover Delay:
Edition:
PDB:
Commit Outcome:
Retention Timeout:
Replay Initiation Timeout:
Session State Consistency:
SQL Translation Profile:

```

```
Databases
-----
Database          Preferred Status
-----
eastdb            Yes      Enabled
westdb            Yes      Enabled

GDSCTL>
```

2. **(Session 2)** Return to the terminal session on wnode03. Start DGMGRL and stop the Apply Process on enode01.

```
[oracle@wnode03 ~]$ dgmgrl
DGMGRL> connect sysdg/oracle_4U@westdb
Connected as SYSDG.
DGMGRL> show configuration

Configuration - dg_config

Protection Mode: MaxPerformance
Members:
westdb - Primary database
eastdb - Physical standby database

Fast-Start Failover: DISABLED

Configuration Status:
SUCCESS (status updated 58 seconds ago)

DGMGRL> show database westdb

Database - westdb

Role:           PRIMARY
Intended State: TRANSPORT-ON
Instance(s):
westdb1
westdb2

Database Status:
SUCCESS

DGMGRL> show database eastdb

Database - eastdb

Role:           PHYSICAL STANDBY
Intended State: APPLY-ON
Transport Lag:   0 seconds (computed 0 seconds ago)
Apply Lag:      0 seconds (computed 1 second ago)
Average Apply Rate: 4.00 KByte/s
Real Time Query: ON
Instance(s):
eastdb1
eastdb2 (apply instance)

Database Status:
```

```
SUCCESS

DGMGRL> edit database eastdb set state='APPLY-OFF';
Succeeded.
DGMGRL> show database eastdb

Database - eastdb

Role: PHYSICAL STANDBY
Intended State: APPLY-OFF
Transport Lag: 0 seconds
Apply Lag: 4 minutes 10 seconds
Average Apply Rate: (unknown)
Real Time Query: OFF
Instance(s):
  eastdb1
  eastdb2 (apply instance)

Database Status:
SUCCESS

DGMGRL>
```

3. **(Session 1)** Return to the terminal session on enode01. Wait for 15 seconds and run the services command from GDSCTL and observe that the sales_reader_lag15_srvc global service has failed over to primary.

```
GDSCTL> services
Service "sales_reader_lag15_srvc.sales.oradbcloud" has 2 instance(s).
Affinity: ANYWHERE
  Instance "sales%11", name: "westdb1", db: "westdb", region: "west",
  status: ready.
  Instance "sales%12", name: "westdb2", db: "westdb", region: "west",
  status: ready.
GDSCTL>
GDSCTL> databases
Database: "eastdb" Registered: Y State: Ok ONS: Y. Role: PH_STNDBY
Instances: 2 Region: east
  Service: "sales_reader_lag15_srvc" Globally started: Y Started: N
Scan: Y Enabled: Y Preferred: Y
  Registered instances:
    sales%1
    sales%2
Database: "westdb" Registered: Y State: Ok ONS: Y. Role: PRIMARY
Instances: 2 Region: west
  Service: "sales_reader_lag15_srvc" Globally started: Y Started: Y
Scan: Y Enabled: Y Preferred: Y
  Registered instances:
    sales%11
    sales%12
GDSCTL>
```

4. **(Session 2)** On wnode03, make sure that the Apply Process is started.

```
DGMGRL> edit database eastdb set state='APPLY-ON';
Succeeded.
DGMGRL>
```

5. **(Session 2)** Wait until the Physical Standby Database is synchronized. It may take a few moments for the Lag for the standby database to clear.

Note: Do not continue with labs until the Apply Lag is cleared.

```
DGMGRL> show database eastdb

Database - eastdb

Role: PHYSICAL STANDBY
Intended State: APPLY-ON
Transport Lag: 0 seconds
Apply Lag: 0 seconds
Average Apply Rate: (unknown)
Real Time Query: ON
Instance(s):
  eastdb1
  eastdb2 (apply instance)

Database Status:
SUCCESS

DGMGRL>
```

6. **(Session 1)** Return to the GDSCTL session on enode01 to remove the global service. Note that the sales_reader_lag15_srvc service was running in the westdb database.

Note: You may or may not receive the GSM warning message NET-40073 when removing the global service. It could be because the global service was already failed back to the Physical Standby Database (eastdb) after restarting the apply service. Determine the current location of the global service and stop it.

```
GDSCTL> stop service -service sales_reader_lag15_srvc -gdspool sales -
database westdb
GDSCTL>
GDSCTL> remove service -gdspool sales -service sales_reader_lag15_srvc
GDSCTL>

# If the "remove service" command fails, do the following tasks.

GDSCTL> remove service -gdspool sales -service sales_reader_lag15_srvc
GSM Errors:
NET-40073: Service is running and cannot be removed
GDSCTL>
GDSCTL> services
Service "sales_reader_lag15_srvc.sales.oradbcloud" has 2 instance(s).
Affinity: ANYWHERE
  Instance "sales%1", name: "eastdb1", db: "eastdb", region: "east",
status: ready.
  Instance "sales%2", name: "eastdb2", db: "eastdb", region: "east",
status: ready.
GDSCTL>
GDSCTL> stop service -service sales_reader_lag15_srvc -gdspool sales -
database eastdb
GDSCTL>
GDSCTL> remove service -gdspool sales -service sales_reader_lag15_srvc
GDSCTL>
GDSCTL> exit
```

7. Exit all of your terminal sessions.

This practice illustrated replication lag tolerance-based routing for Active Data Guard configuration, which allows applications achieve better data quality. Instead of accessing data in the standby database that is lagging behind, you can automatically relocate the service to a database that is not lagging more than the defined threshold.

Practices for Lesson 9: Performing Database Recovery in an Oracle Data Guard Environment

Chapter 9

Practices for Lesson 9: Overview

Practice Overview

In this practice, you recover the Primary Database from media and logical failures and synchronize the Physical Standby Database using various approaches.

Practice Summary

- Practice 9-1: Reconfiguring the Environment
- Practice 9-2: Performing Complete Recovery on the Primary database *Over the Network* in a Data Guard environment
- Practice 9-3: Recovering Standby Control Files (Media Failure)
- Practice 9-4: Performing Steps for Rolling Forward a Physical Standby Database *Using RMAN Incremental Backup*
- Practice 9-5: Performing Incomplete Recovery *Using Flashback Database* (Logical Failure)

Practice 9-1: Reconfiguring the Environment

Overview

You have moved the default listener to RDBMS Home for the GDS practices. In this practice, you move the default listener back to CRS Home to use the existing configuration settings. You also perform the switchover operation to the original state in preparation for the Recovery practices.

Task Summary

- **Task 1:** Reconfiguring the Default Listener
- **Task 2:** Switch Over to the Original Configuration

Terminal Sessions

- **Session 1:** ssh oracle@enode01
- **Session 2:** ssh oracle@wnode03

Task 1: Reconfiguring the Default Listener

1. **(Session 1)** Open a terminal window on enode01 as the current owner of the default listener, oracle user.

```
[Classroom PC]$ ssh oracle@enode01
oracle@enode01's password: <oracle>
[oracle@enode01 ~]$
```

2. Set your environment to manage the default listener.

```
[oracle@enode01 ~]$ . oraenv
ORACLE_SID = [oracle]? eastdb
The Oracle base has been set to /u01/app/oracle
[oracle@enode01 ~]$
```

3. Remove the default listener from RDBMS Home.

```
[oracle@enode01 ~]$ srvctl stop listener
[oracle@enode01 ~]$ srvctl remove listener
[oracle@enode01 ~]$
```

4. As the new owner of the default listener, add the default listener to CRS Home.

```
[oracle@enode01 ~]$ su - grid
Password: <oracle>
[grid@enode01 ~]$ . oraenv
ORACLE_SID = [grid]? +ASM1
The Oracle base has been set to /u01/app/grid
[grid@enode01 ~]$ srvctl add listener -l listener -oraclehome
/u01/app/12.1.0/grid
[grid@enode01 ~]$ srvctl start listener
[grid@enode01 ~]$
```

5. **(Session 2)** Open another terminal window on wnode03 as the oracle user.

```
[Classroom PC]$ ssh oracle@wnode03
oracle@wnode03's password: <oracle>
[oracle@wnode03 ~]$
```

6. Set your environment to manage the default listener.

```
[oracle@wnode03 ~]$ . oraenv
ORACLE_SID = [oracle]? westdb
The Oracle base has been set to /u01/app/oracle
[oracle@wnode03 ~]$
```

7. Remove the default listener from RDBMS Home.

```
[oracle@wnode03 ~]$ srvctl stop listener
[oracle@wnode03 ~]$ srvctl remove listener
[oracle@wnode03 ~]$
```

8. As the new owner of the default listener, add the default listener to CRS Home.

```
[oracle@wnode03 ~]$ su - grid
Password: <oracle>
[grid@wnode03 ~]$ . oraenv
ORACLE_SID = [grid]? +ASM1
The Oracle base has been set to /u01/app/grid
[grid@wnode03 ~]$ srvctl add listener -l listener -oraclehome
/u01/app/12.1.0/grid
[grid@wnode03 ~]$ srvctl start listener
[grid@wnode03 ~]$ exit
logout
[oracle@wnode03 ~]$
```

Task 2: Switch Over to the Original Configuration

9. In the same terminal, switch over to the eastdb database.

```
[oracle@wnode03 ~]$ dgmgrl sysdg/oracle_4U@westdb
Connected as SYSDG.
DGMGRL> validate database westdb

Database Role: Primary database

Ready for Switchover: Yes

DGMGRL> validate database eastdb

Database Role: Physical standby database
Primary Database: westdb

Ready for Switchover: Yes
Ready for Failover: Yes (Primary Running)

Transport-Related Property Settings:
  Property      westdb Value          eastdb Value
  RedoRoutes    (westdb:eastdb SYNC)  (eastdb:westdb SYNC)

DGMGRL>
DGMGRL> switchover to eastdb;
Performing switchover NOW, please wait...
Operation requires a connection to instance "eastdb2" on database
"eastdb"
Connecting to instance "eastdb2"...
```

```
Connected as SYSDG.  
New primary database "eastdb" is opening...  
Oracle Clusterware is restarting database "westdb" ...  
Switchover succeeded, new primary is "eastdb"  
DGMGRL>
```

10. Determine the status of the Data Guard configuration.

Note: You may or may not see warning messages, such as ORA-16525, ORA-16665, ORA-16810, or ORA-16817, due to the lag and overall performance of the virtual machine environment. The warning messages will be cleared in a few minutes. Make sure that your Data Guard configuration status is still fine.

```
DGMGRL> show configuration  
DGMGRL>  
  
Configuration - dg_config  
  
Protection Mode: MaxPerformance  
Members:  
  eastdb - Primary database  
  westdb - Physical standby database  
  
Fast-Start Failover: DISABLED  
  
Configuration Status:  
SUCCESS      (status updated 17 seconds ago)  
  
DGMGRL>
```

11. Wait until the physical standby database is synchronized. It may take a few moments for the Lag for the standby database to clear.

Note: Do not continue with the practices until the Apply Lag is cleared.

```
DGMGRL> show database westdb  
  
Database - westdb  
  
Role:          PHYSICAL STANDBY  
Intended State: APPLY-ON  
Transport Lag: 0 seconds (computed 1 second ago)  
Apply Lag:    0 seconds (computed 1 second ago)  
Average Apply Rate: 40.00 KByte/s  
Real Time Query: ON  
Instance(s):  
  westdb1 (apply instance)  
  westdb2  
  
Database Status:  
SUCCESS  
  
DGMGRL>
```

12. Exit from all your terminal sessions.

Practice 9-2: Performing Complete Recovery on the Primary database Over the Network in a Data Guard environment

Overview

In this practice, you recover a lost data file on the primary database. As of Oracle Database 12c Release 1 (12.1), you can restore and recover files over the network by connecting to a physical standby database that contains the required files. This can be useful when you want to restore lost data files, control files, or tablespaces on a primary database by using the corresponding files on the physical standby database. You can also use the same process to restore files on a physical standby database by using the primary database.

Task Summary

- **Task 1:** Identifying the issue on the primary database
- **Task 2:** Recovering the missing data file over the network

Task 1: Identifying the Issue on the Primary Database

1. Open a terminal window to connect to enode01 as the oracle user.

```
[Classroom PC] $ ssh oracle@enode01
oracle@enode01's password: <oracle>
[oracle@enode01]$
```

2. Make sure that you set up your environment variables correctly for the oracle user.

```
[oracle@enode01 ~]$ . oraenv
ORACLE_SID = [oracle]? eastdb
The Oracle base has been set to /u01/app/oracle
[oracle@enode01 ~]$ export ORACLE_SID=eastdb1
[oracle@enode01 ~]$
```

3. Navigate to the /home/oracle/labs directory. Then execute the setup_09_01.sh script to set up the environment for this practice.

Note: Ignore ORA-01918 and ORA-00595.

```
[oracle@enode01 ~]$ cd /home/oracle/labs
[oracle@enode01 labs]$ ./setup_09_01.sh

Begin of setup_09_01.sh

SQL*Plus: Release 12.1.0.2.0 Production on Mon Oct 12 20:17:49 2015

Copyright (c) 1982, 2014, Oracle. All rights reserved.

SQL> Connected.
SQL> SQL> DROP USER bar CASCADE
      *
ERROR at line 1:
ORA-01918: user 'BAR' does not exist

SQL> DROP TABLESPACE bartbs INCLUDING CONTENTS AND DATAFILES
      *
ERROR at line 1:
```

```
ORA-00959: tablespace 'BARTBS' does not exist

SQL> SQL> SQL>    2      3
Tablespace created.

SQL> SQL> SQL>    2      3
User created.

SQL> SQL>
Grant succeeded.

SQL> SQL> SQL> SQL>    2      3
Table created.

SQL> SQL>    2
107 rows created.

SQL> SQL>    2
214 rows created.

SQL> 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

Recovery Manager: Release 12.1.0.2.0 - Production on Mon Oct 12
20:18:05 2015

Copyright (c) 1982, 2014, Oracle and/or its affiliates. All rights
reserved.

connected to target database: EASTDB (DBID=814850587)

RMAN> 2> 3> 4>
using target database control file instead of recovery catalog
allocated channel: ch1
channel ch1: SID=90 instance=eastdb1 device type=DISK

Starting backup at 12-OCT-15
channel ch1: starting datafile copy
input datafile file number=00007 name=+DATA/eastdb/bartbs.dbf
output file name=+FRA/EASTDB/DATAFILE/bartbs.488.892930697 tag=BARTEST
RECID=5 STAMP=892930698
channel ch1: datafile copy complete, elapsed time: 00:00:03
Finished backup at 12-OCT-15
released channel: ch1

RMAN>

Recovery Manager complete.

SQL*Plus: Release 12.1.0.2.0 Production on Mon Oct 12 20:18:32 2015

Copyright (c) 1982, 2014, Oracle. All rights reserved.

SQL> Connected.
```

```

SQL>
428 rows updated.

SQL>
Commit complete.

SQL>
System altered.

SQL> 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
End of setup_09_01.sh
[oracle@enode01 labs] $

```

- Cause a failure in the primary database by removing data files in the USERS tablespace.

```

[oracle@enode01 labs] $ srvctl stop database -db eastdb -stopoption
abort
[oracle@enode01 labs] $ srvctl status database -db eastdb
Instance eastdb1 is not running on node enode01
Instance eastdb2 is not running on node enode02
[oracle@enode01 labs] $ su - grid
Password: <oracle>
[grid@enode01 ~]$ . oraenv
ORACLE_SID = [grid] ? +ASM1
The Oracle base has been set to /u01/app/grid
[grid@enode01 ~]$ asmcmd rm -f +data/eastdb/datafile/users*
[grid@enode01 ~]$ exit
logout
[oracle@enode01 labs] $

```

- Attempt to start the eastdb database.

Note: The error message indicates that a data file in the USERS tablespace is missing.

```

[oracle@enode01 labs] $ srvctl start database -db eastdb
PRCR-1079 : Failed to start resource ora.eastdb.db
CRS-5017: The resource action "ora.eastdb.db start" encountered the
following error:
ORA-01157: cannot identify/lock data file 6 - see DBWR trace file
ORA-01110: data file 6: '+DATA/EASTDB/DATAFILE/users.289.892718177'
. For details refer to "(:CLSN00107:)" in
"/u01/app/grid/diag/crs/enode02/crs/trace/crsd_oraagent_oracle.trc".

CRS-2674: Start of 'ora.eastdb.db' on 'enode02' failed
CRS-2632: There are no more servers to try to place resource
'ora.eastdb.db' on that would satisfy its placement policy
CRS-5017: The resource action "ora.eastdb.db start" encountered the
following error:
ORA-01157: cannot identify/lock data file 6 - see DBWR trace file
ORA-01110: data file 6: '+DATA/EASTDB/DATAFILE/users.289.892718177'
. For details refer to "(:CLSN00107:)" in
"/u01/app/grid/diag/crs/enode01/crs/trace/crsd_oraagent_oracle.trc".
CRS-2674: Start of 'ora.eastdb.db' on 'enode01' failed
[oracle@enode01 labs] $

```

Task 2: Recovering the Missing Data File Over the Network

- Using the RMAN command line, connect to the eastdb1 instance.

```
[oracle@enode01 labs]$ export ORACLE_SID=eastdb1
[oracle@enode01 labs]$ rman target /
Recovery Manager: Release 12.1.0.2.0 - Production on Mon Oct 12
19:48:21 2015
Copyright (c) 1982, 2014, Oracle and/or its affiliates. All rights
reserved.

connected to target database (not started)

RMAN>
```

- Mount the database instance to restore the lost data file.

```
RMAN> startup mount
Oracle instance started
database mounted

Total System Global Area    1048576000 bytes
Fixed Size                  2932336 bytes
Variable Size                771752336 bytes
Database Buffers             268435456 bytes
Redo Buffers                 5455872 bytes

RMAN>
```

- Restore the lost data file from the physical standby database (westdb) over the network.

Note: You can recover a missing data file using different methods. However, you focus on the network restore by leveraging the physical standby database.

```
RMAN> restore tablespace users from service 'westdb';

Starting restore at 12-OCT-15
using target database control file instead of recovery catalog
allocated channel: ORA_DISK_1
channel ORA_DISK_1: SID=41 instance=eastdb1 device type=DISK

channel ORA_DISK_1: starting datafile backup set restore
channel ORA_DISK_1: using network backup set from service westdb
channel ORA_DISK_1: specifying datafile(s) to restore from backup set
channel ORA_DISK_1: restoring datafile 00006 to
+DATA/EASTDB/DATAFILE/users.289.892718177
channel ORA_DISK_1: restore complete, elapsed time: 00:00:05
Finished restore at 12-OCT-15

RMAN>
```

9. Recover the restored tablespace using archivelogs available on the primary database.

```
RMAN> recover tablespace users;

Starting recover at 12-OCT-15
using channel ORA_DISK_1

starting media recovery
media recovery complete, elapsed time: 00:00:01

Finished recover at 12-OCT-15

RMAN> exit
[oracle@enode01 labs]$
```

10. Start the primary database to bring the users tablespace online.

```
[oracle@enode01 labs]$ srvctl stop database -db eastdb
[oracle@enode01 labs]$ srvctl start database -db eastdb
[oracle@enode01 labs]$
```

11. Verify the recovered tablespace.

```
[oracle@enode01 labs]$ sqlplus / as sysdba
SQL> col name format a50
SQL> select name, status from v$datafile;
NAME                               STATUS
-----
+DATA/EASTDB/DATAFILE/system.290.892718111      SYSTEM
+DATA/EASTDB/DATAFILE/undotbs2.285.892721243    ONLINE
+DATA/EASTDB/DATAFILE/sysaux.291.892718045     ONLINE
+DATA/EASTDB/DATAFILE/undotbs1.276.892718179    ONLINE
+DATA/EASTDB/DATAFILE/example.279.892718299    ONLINE
+DATA/EASTDB/DATAFILE/users.289.892929237      ONLINE
+DATA/eastdb/bartbs.dbf                  ONLINE
+FRA/eastdb/online_tbs01.dbf            ONLINE

8 rows selected.

SQL> exit
[oracle@enode01 labs]$
```

12. Determine the status of the Data Guard configuration.

Note: You may or may not see warning messages, such as ORA-16525, ORA-16665, ORA-16810, or ORA-16817, due to the lag and overall performance of the virtual machine environment. The warning messages will be cleared in a few minutes. Make sure that your Data Guard configuration status is still fine.

```
[oracle@enode01 labs]$ dgmgrl
DGMGRL> connect sysdg/oracle_4U@eastdb
Connected as SYSDG.
DGMGRL> show configuration

Configuration - dg_config

Protection Mode: MaxPerformance
Members:
eastdb - Primary database
```

```
westdb - Physical standby database  
Fast-Start Failover: DISABLED  
  
Configuration Status:  
SUCCESS      (status updated 54 seconds ago)  
  
DGMGRL>
```

13. Wait until the physical standby database is synchronized. It may take a few moments for the Lag for the standby database to clear.

Note: Do not continue with the practices until the Apply Lag is cleared.

```
DGMGRL> show database westdb  
  
Database - westdb  
  
Role:          PHYSICAL STANDBY  
Intended State:  APPLY-ON  
Transport Lag:   0 seconds (computed 1 second ago)  
Apply Lag:    0 seconds (computed 1 second ago)  
Average Apply Rate: 40.00 KByte/s  
Real Time Query: OFF  
Instance(s):  
    westdb1 (apply instance)  
    westdb2  
  
Database Status:  
SUCCESS  
  
DGMGRL>
```

14. Exit all of your terminal sessions.

Practice 9-3: Recovering Standby Control Files (Media Failure)

Overview

In this practice, you create an issue by removing all control files in the standby database. After creating the issue, you re-create the standby control file. The data files are on ASM and using Oracle Managed Files (OMF).

Task Summary

- **Task 1:** Identifying the issue on the standby database
- **Task 2:** Restoring the standby control file over the network
- **Task 3:** Recovering the standby control file
- **Task 4:** Starting Redo Transport and Verify Operation

Terminal Sessions

- **Session 1:** `ssh oracle@wnode03`
- **Session 2:** `ssh oracle@enode01`

Task 1: Identifying the Issue on the Standby Database

1. **(Session 1)** Open a terminal window on wnode03 as the `oracle` user.

```
[Classroom PC] $ ssh oracle@wnode03
oracle@wnode03's password: <oracle>
[oracle@wnode03] $
```

2. Make sure that you set up your environment variables correctly for the `oracle` user.

```
[oracle@wnode03 ~] $ . oraenv
ORACLE_SID = [oracle] ? westdb
The Oracle base has been set to /u01/app/oracle
[oracle@wnode03 ~] $ export ORACLE_SID=westdb1
[oracle@wnode03 ~] $
```

3. Cause a failure in the database by deleting all existing standby control files from the standby database.

```
[oracle@wnode03 ~] $ srvctl stop database -db westdb -stopoption abort
[oracle@wnode03 ~] $ srvctl status database -db westdb
Instance westdb1 is not running on node wnode03
Instance westdb2 is not running on node wnode04
[oracle@wnode03 ~] $ su - grid
Password: <oracle>
[grid@wnode03 ~] $ . oraenv
ORACLE_SID = [grid] ? +ASM1
The Oracle base has been set to /u01/app/grid
[grid@wnode03 ~] $ asmcmd rm -f +fra/westdb/controlfile/*
[grid@wnode03 ~] $ asmcmd rm -f +data/westdb/controlfile/*
[grid@wnode03 ~] $ exit
logout
[oracle@wnode03 ~] $
```

4. Attempt to start the standby database. Notice the error message.

```
[oracle@wnode03 labs]$ srvctl start database -db westdb
PRCR-1079 : Failed to start resource ora.westdb.db
CRS-5017: The resource action "ora.westdb.db start" encountered the
following error:
ORA-00205: error in identifying control file, check alert log for more
info
. For details refer to "(:CLSN00107:)" in
"/u01/app/grid/diag/crs/wnode04/crs/trace/crsd_oraagent_oracle.trc".

CRS-2674: Start of 'ora.westdb.db' on 'wnode04' failed
CRS-2632: There are no more servers to try to place resource
'ora.westdb.db' on that would satisfy its placement policy
CRS-5017: The resource action "ora.westdb.db start" encountered the
following error:
ORA-00205: error in identifying control file, check alert log for more
info
. For details refer to "(:CLSN00107:)" in
"/u01/app/grid/diag/crs/wnode03/crs/trace/crsd_oraagent_oracle.trc".

CRS-2674: Start of 'ora.westdb.db' on 'wnode03' failed
[oracle@wnode03 labs]$
```

Task 2: Restoring the Standby Control File Over the Network

5. Use a new Oracle 12c RMAN functionality to restore the standby control file.

Note: If the instance is already in the shutdown state, the shutdown abort command will fail. You can safely move on.

```
[oracle@wnode03 labs]$ export ORACLE_SID=westdb1
[oracle@wnode03 labs]$ rman target /

Recovery Manager: Release 12.1.0.2.0 - Production on Tue Oct 13
18:01:46 2015

Copyright (c) 1982, 2014, Oracle and/or its affiliates. All rights
reserved.

connected to target database (not started)

RMAN> shutdown abort;
RMAN> startup nomount;

Oracle instance started

Total System Global Area      1048576000 bytes

Fixed Size                  2932336 bytes
Variable Size                876609936 bytes
Database Buffers             163577856 bytes
Redo Buffers                 5455872 bytes

RMAN> restore standby controlfile from service eastdb;

Starting restore at 13-OCT-15
using target database control file instead of recovery catalog
```

```

allocated channel: ORA_DISK_1
channel ORA_DISK_1: SID=40 instance=westdb1 device type=DISK

channel ORA_DISK_1: starting datafile backup set restore
channel ORA_DISK_1: using network backup set from service eastdb
channel ORA_DISK_1: restoring control file
channel ORA_DISK_1: restore complete, elapsed time: 00:00:11
output file name=+DATA/WESTDB/CONTROLFILE/current.293.893009367
output file name=+FRA/WESTDB/CONTROLFILE/current.327.893009369
Finished restore at 13-OCT-15

RMAN>

```

- Because the primary control file has been restored to the standby database, attempt to mount the standby database.

```

RMAN> alter database mount;

Statement processed

RMAN>

```

Task 3: Recovering the Standby Control File

At this point, the standby control file is referring to the primary database files. So, you need to update the standby control file to point to the standby database files.

- Catalog the data files of standby database. The catalog command will give you a list of files and ask if they should all be catalog. Review the list and say YES if all the datafiles are properly listed.

```

RMAN> catalog start with '+DATA/WESTDB/DATAFILE/';

Starting implicit crosscheck backup at 13-OCT-15
allocated channel: ORA_DISK_1
Crosschecked 8 objects
Finished implicit crosscheck backup at 13-OCT-15

Starting implicit crosscheck copy at 13-OCT-15
using channel ORA_DISK_1
Crosschecked 3 objects
Finished implicit crosscheck copy at 13-OCT-15

searching for all files in the recovery area
cataloging files...
cataloging done

List of Cataloged Files
=====
File Name: +FRA/WESTDB/online_tbs.dbf
File Name:
+FRA/WESTDB/ARCHIVELOG/2015_10_13/thread_1_seq_40.533.892944433
File Name:
+FRA/WESTDB/ARCHIVELOG/2015_10_13/thread_2_seq_43.534.892954883
File Name:

```

```
+FRA/WESTDB/ARCHIVELOG/2015_10_13/thread_1_seq_41.535.892971173
File Name:

      --- The output truncated ---

File Name:
+FRA/WESTDB/ARCHIVELOG/2015_10_10/thread_1_seq_29.282.892735315
File Name:
File Name:
+FRA/WESTDB/ARCHIVELOG/2015_10_10/thread_2_seq_16.472.892763159
File Name:
+FRA/WESTDB/ARCHIVELOG/2015_10_10/thread_1_seq_14.473.892763165

searching for all files that match the pattern +DATA/WESTDB/DATAFILE/

List of Files Unknown to the Database
=====
File Name: +DATA/WESTDB/DATAFILE/example.281.892725093
File Name: +DATA/WESTDB/DATAFILE/system.284.892725097
File Name: +DATA/WESTDB/DATAFILE/sysaux.287.892725103
File Name: +DATA/WESTDB/DATAFILE/undotbs2.286.892725109
File Name: +DATA/WESTDB/DATAFILE/undotbs1.276.892725221
File Name: +DATA/WESTDB/DATAFILE/users.294.892725321
File Name: +DATA/WESTDB/DATAFILE/bartbs.303.892930675

Do you really want to catalog the above files (enter YES or NO)? yes
cataloging files...
cataloging done

List of Cataloged Files
=====
File Name: +DATA/WESTDB/DATAFILE/example.281.892725093
File Name: +DATA/WESTDB/DATAFILE/system.284.892725097
File Name: +DATA/WESTDB/DATAFILE/sysaux.287.892725103
File Name: +DATA/WESTDB/DATAFILE/undotbs2.286.892725109
File Name: +DATA/WESTDB/DATAFILE/undotbs1.276.892725221
File Name: +DATA/WESTDB/DATAFILE/users.294.892725321
File Name: +DATA/WESTDB/DATAFILE/bartbs.303.892930675

RMAN>
```

Note: This will work only if you are using OMF. If you are using ASM without OMF, you have to catalog all non-OMF data files as Datafile Copies manually using RMAN> catalog datafilecopy '<File-Specification>'. If you have data files on different diskgroups, you have to catalog from all diskgroups.

8. Using the SWITCH command, commit the changes to the standby control file.

```
RMAN> switch database to copy;

datafile 1 switched to datafile copy
"+DATA/WESTDB/DATAFILE/system.284.892725097"
datafile 2 switched to datafile copy
"+DATA/WESTDB/DATAFILE/undotbs2.286.892725109"
datafile 3 switched to datafile copy
"+DATA/WESTDB/DATAFILE/sysaux.287.892725103"
datafile 4 switched to datafile copy
"+DATA/WESTDB/DATAFILE/undotbs1.276.892725221"
```

```

datafile 5 switched to datafile copy
"+DATA/WESTDB/DATAFILE/example.281.892725093"
datafile 6 switched to datafile copy
"+DATA/WESTDB/DATAFILE/users.294.892725321"
datafile 7 switched to datafile copy
"+DATA/WESTDB/DATAFILE/bartbs.303.892930675"
datafile 8 switched to datafile copy "+FRA/WESTDB/online_tbs.dbf"

RMAN>

```

9. Re-enable flashback on the standby database.

Note: If more standby database recovery is needed, you can run the recover database command instead of waiting.

```

RMAN> alter database flashback off;

Statement processed

RMAN> alter database flashback on;

RMAN-00571: =====
RMAN-00569: ===== ERROR MESSAGE STACK FOLLOWS =====
RMAN-00571: =====
RMAN-03002: failure of sql statement command at 10/13/2015 18:25:05
ORA-38706: Cannot turn on FLASHBACK DATABASE logging.
ORA-38788: More standby database recovery is needed

RMAN> recover database;

Starting recover at 13-OCT-15
using channel ORA_DISK_1

starting media recovery

archived log for thread 1 with sequence 42 is already on disk as file
+FRA/WESTDB/ARCHIVELOG/2015_10_13/thread_1_seq_42.537.892998467
archived log for thread 1 with sequence 43 is already on disk as file
+FRA/WESTDB/ARCHIVELOG/2015_10_13/thread_1_seq_43.538.893009547
archived log for thread 2 with sequence 45 is already on disk as file
+FRA/WESTDB/ARCHIVELOG/2015_10_13/thread_2_seq_45.328.893009545
archived log file
name=+FRA/WESTDB/ARCHIVELOG/2015_10_13/thread_2_seq_45.328.893009545
thread=2 sequence=45
archived log file
name=+FRA/WESTDB/ARCHIVELOG/2015_10_13/thread_1_seq_42.537.892998467
thread=1 sequence=42
archived log file
name=+FRA/WESTDB/ARCHIVELOG/2015_10_13/thread_1_seq_43.538.893009547
thread=1 sequence=43
media recovery complete, elapsed time: 00:00:42
Finished recover at 13-OCT-15

RMAN> alter database flashback on;

RMAN-00571: =====
RMAN-00569: ===== ERROR MESSAGE STACK FOLLOWS =====
RMAN-00571: =====
RMAN-03002: failure of sql statement command at 10/13/2015 18:30:02
ORA-01153: an incompatible media recovery is active

```

```
RMAN>

### If MRP is active, stop MRP first ###

RMAN> alter database recover managed standby database cancel;
Database altered.

RMAN> alter database flashback on;
Database altered.

RMAN> alter database recover managed standby database disconnect;
Database altered.

RMAN> exit
[oracle@wnode03 ~]$ 
[oracle@wnode03 ~]$
```

10. Query v\$log and clear all online redo log groups.

```
[oracle@wnode03 ~]$ sqlplus / as sysdba
SQL> SELECT THREAD#, GROUP# FROM V$LOG order by 1,2;

  THREAD#      GROUP#
----- -----
        1          1
        1          2
        1          3
        2          4
        2          11
        2          12

SQL> alter database clear logfile group 1;
Database altered.

SQL> alter database clear logfile group 2;
Database altered.

SQL> alter database clear logfile group 3;
Database altered.

SQL> alter database clear logfile group 4;
Database altered.

SQL> alter database clear logfile group 11;
Database altered.

SQL> alter database clear logfile group 12;
Database altered.

SQL> SELECT THREAD#, GROUP#, SEQUENCE#,BYTES/1024/1024 "SIZE MB",
ARCHIVED,STATUS FROM V$LOG order by 1,2;

  THREAD#      GROUP#      SEQUENCE#    SIZE MB ARC STATUS
----- -----
        1          1            0         50 YES UNUSED
        1          2            0         50 YES UNUSED
        1          3            0         50 YES UNUSED
        2          4            0         50 YES UNUSED
```

2	11	0	50 YES UNUSED
2	12	0	50 YES UNUSED
SQL>			

11. Query v\$standby_log and clear all standby redo logs.

SELECT THREAD#, GROUP# FROM V\$STANDBY_LOG order by 1,2;					
THREAD#	GROUP#	SEQUENCE#	SIZE MB	ARC	STATUS
1	5	0	50	YES	UNASSIGNED
1	6	0	50	YES	UNASSIGNED
1	7	0	50	YES	UNASSIGNED
1	8	0	50	YES	UNASSIGNED
2	9	0	50	YES	UNASSIGNED
2	10	0	50	YES	UNASSIGNED
2	13	0	50	YES	UNASSIGNED
2	14	0	50	YES	UNASSIGNED

SQL> alter database clear logfile group 5;					
Database altered.					
SQL> alter database clear logfile group 6;					
Database altered.					
SQL> alter database clear logfile group 7;					
Database altered.					
SQL> alter database clear logfile group 8;					
Database altered.					
SQL> alter database clear logfile group 9;					
Database altered.					
SQL> alter database clear logfile group 10;					
Database altered.					
SQL> alter database clear logfile group 13;					
Database altered.					
SQL> alter database clear logfile group 14;					
Database altered.					
SQL> SELECT THREAD#, GROUP#, SEQUENCE#,BYTES/1024/1024 "SIZE MB", ARCHIVED, STATUS FROM V\$STANDBY_LOG order by 1,2;					
THREAD#	GROUP#	SEQUENCE#	SIZE MB	ARC	STATUS
1	5	0	50	YES	UNASSIGNED
1	6	0	50	YES	UNASSIGNED
1	7	0	50	YES	UNASSIGNED
1	8	0	50	YES	UNASSIGNED
2	9	0	50	YES	UNASSIGNED
2	10	0	50	YES	UNASSIGNED
2	13	0	50	YES	UNASSIGNED
2	14	0	50	YES	UNASSIGNED

```
SQL>
SQL>
```

Note: You need to re-create the standby redo logs on the standby database if standby redo logs are not present on the primary.

- Start managed recovery process on the standby instance (westdb1).

```
SQL> alter database recover managed standby database disconnect;
Database altered.
SQL>
```

Task 4: Starting Redo Transport and Verify Operation

- (Session 2) Open a terminal window on enode01 as the oracle user.

```
[Classroom PC]$ ssh oracle@enode01
oracle@enode01's password: <oracle>
[oracle@enode01]$
```

- Make sure that you set up your environment variables correctly for the oracle user.

```
[oracle@enode01 ~]$ . oraenv
ORACLE_SID = [oracle]? eastdb
The Oracle base has been set to /u01/app/oracle
[oracle@enode01 ~]$ export ORACLE_SID=eastdb1
[oracle@enode01 ~]$
```

- Determine the last number archived on the primary database. Your number may be different in the lab.

```
[oracle@enode01 ~]$ sqlplus / as sysdba
SQL> SELECT THREAD#, MAX(SEQUENCE#) FROM GV$ARCHIVED_LOG WHERE
RESETLOGS_CHANGE# = (SELECT MAX(RESETLOGS_CHANGE#) FROM
GV$ARCHIVED_LOG) GROUP BY THREAD#;

THREAD# MAX (SEQUENCE#)
-----
1          46
2          45

SQL>
```

- (Session 1) Return to the terminal session on wnode03. Determine the last sequence number of the physical standby database instance.

```
SQL> SELECT THREAD#, MAX(SEQUENCE#) FROM GV$ARCHIVED_LOG WHERE
RESETLOGS_CHANGE# = (SELECT MAX(RESETLOGS_CHANGE#) FROM
GV$ARCHIVED_LOG) GROUP BY THREAD#;

THREAD# MAX (SEQUENCE#)
-----
1          46
2          45

SQL>
```

17. (**Session 2**) Return to the terminal window on `enode01`, and force a log switch to advance the online redo log sequence number. Verify that the sequence number has increased.

```
SQL> alter system archive log current;
System altered.

SQL> SELECT THREAD#, MAX(SEQUENCE#) FROM GV$ARCHIVED_LOG WHERE
RESETLOGS_CHANGE# = (SELECT MAX(RESETLOGS_CHANGE#) FROM
GV$ARCHIVED_LOG) GROUP BY THREAD#;

THREAD# MAX (SEQUENCE#)
-----
1          47
2          46

SQL>
```

18. (**Session 1**) Return to the terminal window on `wnode03`, and verify that the physical standby instance is receiving redo from the primary database instance.

```
SQL> SELECT THREAD#, MAX(SEQUENCE#) FROM GV$ARCHIVED_LOG WHERE
RESETLOGS_CHANGE# = (SELECT MAX(RESETLOGS_CHANGE#) FROM
GV$ARCHIVED_LOG) GROUP BY THREAD#;

THREAD# MAX (SEQUENCE#)
-----
1          47
2          46

SQL> exit
[oracle@wnode03 ~]$
```

19. Determine the status of the Data Guard configuration.

Note: You may or may not see warning messages, such as ORA-16525, ORA-16665, ORA-16810, or ORA-16817, due to the lag and overall performance of the virtual machine environment. The warning messages will be cleared in a few minutes.

```
[oracle@wnode03 ~]$ dgmgrl
DGMGRL> connect sysdg/oracle_4U@westdb
Connected as SYSDG.
DGMGRL> show configuration

Configuration - dg_config

Protection Mode: MaxPerformance
Members:
eastdb - Primary database
westdb - Physical standby database

Fast-Start Failover: DISABLED

Configuration Status:
SUCCESS      (status updated 54 seconds ago)

DGMGRL>
```

20. Wait until the physical standby database is synchronized. It may take a few moments for the Lag for the standby database to clear.

Note: Do not continue with the practices until the Apply Lag is cleared.

```
DGMGRL> show database westdb

Database - westdb

Role: PHYSICAL STANDBY
Intended State: APPLY-ON
Transport Lag: 0 seconds (computed 1 second ago)
Apply Lag: 0 seconds (computed 1 second ago)
Average Apply Rate: 40.00 KByte/s
Real Time Query: OFF
Instance(s):
    westdb1 (apply instance)
    westdb2

Database Status:
SUCCESS

DGMGRL>
```

21. Exit all of your terminal sessions.

Practice 9-4: Performing the Steps for Rolling Forward a Physical Standby Database Using RMAN Incremental Backup

Overview

In this practice, you will perform the steps that can be used to resolve problems such as missing or corrupted archive log file, an unrecoverable archive gap, or the need to roll standby forward in time without applying a large number of archivelog files.

Task Summary

- **Task 1:** Identifying the issue in the Data Guard environment
- **Task 2:** Synchronizing the physical standby database

Terminal Sessions

- **Session 1:** ssh oracle@enode01
- **Session 2:** ssh oracle@wnode03
- **Session 3:** ssh grid@enode01

Task 1: Identifying the Issue in the Data Guard Environment

1. **(Session 1)** Open a terminal window on enode01 as the oracle user.

```
[Classroom PC]$ ssh oracle@enode01
oracle@enode01's password: <oracle>
[oracle@enode01 ~]$
```

2. Make sure that you set up your environment variables correctly for the oracle user.

```
[oracle@enode01 ~]$ . oraenv
ORACLE_SID = [oracle]? eastdb
The Oracle base has been set to /u01/app/oracle
[oracle@enode01 ~]$ export ORACLE_SID=eastdb1
[oracle@enode01 ~]$
```

3. Disable the redo transport service in preparation of the practice.

```
[oracle@enode01 ~]$ dgmgrl
DGMGRL for Linux: Version 12.1.0.2.0 - 64bit Production

Copyright (c) 2000, 2013, Oracle. All rights reserved.

Welcome to DGMGRL, type "help" for information.
DGMGRL> connect sysdg/oracle_4U@eastdb
Connected as SYSDG.
DGMGRL> edit database eastdb set state='TRANSPORT-OFF';
Succeeded.
DGMGRL> exit
[oracle@enode01 ~]$
```

4. **(Session 2)** Open a terminal window on wnode03 as the oracle user.

```
[Classroom PC]$ ssh oracle@wnode03
oracle@wnode03's password: <oracle>
[oracle@wnode03 ~]$
```

5. Make sure that you set up your environment variables correctly for the oracle user.

```
[oracle@enode01 ~]$ . oraenv
ORACLE_SID = [oracle]? westdb
The Oracle base has been set to /u01/app/oracle
[oracle@wnode03 ~]$ export ORACLE_SID=westdb1
[oracle@wnode03 ~]$
```

6. Stop the physical standby database.

```
[oracle@wnode03 ~]$ srvctl stop database -db westdb
[oracle@wnode03 ~]$
```

7. **(Session 1)** Return to the terminal session on enode01. Then make a note of the current log sequence number. Record the sequence number of the online redo log file in thread 1. (In your case, the sequence# is 68.)

```
[oracle@enode01 ~]$ sqlplus / as sysdba

SQL> select thread#, sequence# from v$log where status='CURRENT' order
by 1;

  THREAD#    SEQUENCE#
-----
      1          68
      2          67

SQL>
```

8. Create a simple table named hr.test09 and insert a few rows.

```
SQL> create table hr.test09 (col number);

Table created.

SQL> insert into hr.test09 values (1);
1 row created.

SQL> /
1 row created.

SQL> /
1 row created.

SQL> commit;
Commit complete.

SQL> select count(*) from hr.test09;

  COUNT(*)
-----
      3

SQL>
```

9. Switch the current log file by issuing ALTER SYSTEM ARCHIVE LOG CURRENT to advance the online redo log sequence number.

```
SQL> alter system archive log current;
System altered.

SQL> SELECT THREAD#, MAX(SEQUENCE#) FROM GV$ARCHIVED_LOG WHERE
RESETLOGS_CHANGE# = (SELECT MAX(RESETLOGS_CHANGE#) FROM
GV$ARCHIVED_LOG) GROUP BY THREAD#;

THREAD# MAX (SEQUENCE#)
-----
1          68
2          67

SQL>
```

10. Identify the most current archived log files from thread 1 by using the number identified in step 9.

Note: Your sequence number may be different.

```
SQL> col name format a65
SQL> select thread#, name from v$archived_log where thread#=1 and
sequence#=68 and name like '+FRA/EASTDB/ARCHIVELOG%';

THREAD# NAME
-----
1 +FRA/EASTDB/ARCHIVELOG/2015_10_29/thread_1_seq_68.404.894392781

SQL> exit
[oracle@enode01 ~]$
```

11. (**Session 3**) Open a terminal window on enode01 as the grid user.

```
[Classroom PC] $ ssh grid@enode01
grid@enode01's password: <oracle>
[grid@enode01 ~]$
```

12. Make sure that you set up your environment variables correctly for the grid user.

```
[grid@enode01 ~]$ . oraenv
ORACLE_SID = [grid]? +ASM1
The Oracle base has been set to /u01/app/grid
[grid@enode01 ~]$
```

13. Using asmcmd, remove the identified archived log file in step 10 to simulate the archived log file gap issue.

```
[grid@enode01 ~]$ asmcmd rm -f
+FRA/EASTDB/ARCHIVELOG/2015_10_29/thread_1_seq_68.404.894392781
[grid@enode01 ~]$
```

14. (**Session 1**) Return to the terminal session on enode01 connected as the oracle user to start the redo transport service to the physical standby database (westdb).

```
[oracle@enode01 ~]$ dgmgrl
DGMRGL for Linux: Version 12.1.0.2.0 - 64bit Production

Copyright (c) 2000, 2013, Oracle. All rights reserved.
```

```
Welcome to DGMGRL, type "help" for information.
DGMGRL> connect sysdg/oracle_4U@eastdb
Connected as SYSDG.
DGMGRL> edit database eastdb set state='TRANSPORT-ON';
Succeeded.
DGMGRL> exit
[oracle@enode01 ~]$
```

15. **(Session 2)** Return to the terminal session on wnode03 to start the physical standby database.

```
[oracle@wnode03 ~]$ srvctl start database -db westdb -startoption open
[oracle@wnode03 ~]$ srvctl status database -db westdb -verbose
Instance westdb1 is running on node wnode03. Instance status:
Open, Readonly.
Instance westdb2 is running on node wnode04. Instance status:
Open, Readonly.
[oracle@wnode03 ~]$
```

16. Verify that the physical standby applies the change made to the primary database.

```
[oracle@wnode03 ~]$ sqlplus / as sysdba

SQL> select count(*) from hr.test09;
select count(*) from hr.test09
*
ERROR at line 1:
ORA-00942: table or view does not exist

SQL>
```

Note: The changes made to the primary database have not been applied due to the missing archived log files in the primary database. Remember that you have removed the most current archived log files to simulate the unrecoverable archived log file gap issue.

17. Identify the missing archive log file.

```
SQL> select * from v$archive_gap;

  THREAD# LOW_SEQUENCE# HIGH_SEQUENCE#      CON_ID
----- ----- ----- -----
          1           68           68          1

SQL>
```

Task 2: Synchronizing the Physical Standby Database

In the following task, you will synchronize the physical standby database by using RMAN incremental backups.

18. **(Session 1)** Return to the terminal window on enode01 to stop the Data Guard broker in the primary database. Then exit SQL*Plus.

```
[oracle@enode01 ~]$ sqlplus / as sysdba
SQL> alter system set dg_broker_start=false;
System altered.
```

```
SQL> exit
[oracle@enode01 ~] $
```

19. **(Session 2)** Return to the terminal session on wnode03 to stop the Data Guard broker in the standby database.

```
SQL> alter system set dg_broker_start=false;
System altered.
SQL>
```

20. Stop the managed recovery process (MRP) on the STANDBY database.

```
SQL> ALTER DATABASE RECOVER MANAGED STANDBY DATABASE CANCEL;
Database altered.
SQL> exit
[oracle@wnode03 ~] $
```

21. Stop and mount the standby database in a MOUNT state in the remaining steps.

```
[oracle@wnode03 ~] $ srvctl stop database -db westdb
[oracle@wnode03 ~] $ srvctl start database -db westdb -startoption mount
[oracle@wnode03 ~] $
```

22. Determine the checkpoint_change# of the standby (westdb) database. In the standby database, find the SCN, which will be used for the incremental backup at the primary database. You need to use the lowest SCN from the queries below. Then exit SQL*Plus.

```
[oracle@wnode03 ~] $ sqlplus / as sysdba

SQL> select min(checkpoint_change#) from v$datafile_header;
MIN(CHECKPOINT_CHANGE#)
-----
3802036

SQL> select min(f.fhscn) from x$kcvh f, v$datafile d
where f.hxfil =d.file# and d.enabled != 'READ ONLY';

MIN(CHECKPOINT_CHANGE#)
-----
3802036

SQL> exit
[oracle@wnode03 ~] $
```

You need to use the “lowest SCN” from the queries. In this example, SCN is 3802036. Therefore, from the above, you need to back up from SCN 3802036.

23. **(Session 1)** Return to the terminal session on enode01 to take an incremental backup of the primary database. In RMAN, connect to the primary database and create an incremental backup from the SCN derived in the previous step.

```
[oracle@enode01 ~] $ rman
RMAN> connect target /
connected to target database: EASTDB (DBID=814727097)

RMAN> BACKUP INCREMENTAL FROM SCN 3802036 DATABASE FORMAT
```

```
'/tmp/ForStandby_%U' tag 'FORSTANDBY';

Starting backup at 25-OCT-15
using target database control file instead of recovery catalog
allocated channel: ORA_DISK_1
channel ORA_DISK_1: SID=1 instance=eastdb1 device type=DISK
channel ORA_DISK_1: starting full datafile backup set
channel ORA_DISK_1: specifying datafile(s) in backup set
input datafile file number=00003
name=+DATA/EASTDB/DATAFILE/sysaux.291.892718045
input datafile file number=00005
name=+DATA/EASTDB/DATAFILE/example.279.89271829
9
input datafile file number=00001
name=+DATA/EASTDB/DATAFILE/system.290.892718111
input datafile file number=00002
name=+DATA/EASTDB/DATAFILE/undotbs2.285.8927212
43
input datafile file number=00004
name=+DATA/EASTDB/DATAFILE/undotbs1.276.8927181
79
input datafile file number=00007 name=+DATA/eastdb/bartbs.dbf
input datafile file number=00008 name=+FRA/eastdb/online_tbs01.dbf
input datafile file number=00006
name=+DATA/EASTDB/DATAFILE/users.289.892929237
channel ORA_DISK_1: starting piece 1 at 25-OCT-15
channel ORA_DISK_1: finished piece 1 at 25-OCT-15
piece handle=/tmp/ForStandby_0vqkjvgj_1_1 tag=FORSTANDBY comment=NONE
channel ORA_DISK_1: backup set complete, elapsed time: 00:04:17
channel ORA_DISK_1: starting full datafile backup set
channel ORA_DISK_1: specifying datafile(s) in backup set
including current control file in backup set
channel ORA_DISK_1: starting piece 1 at 25-OCT-15
channel ORA_DISK_1: finished piece 1 at 25-OCT-15
piece handle=/tmp/ForStandby_10qkjvol_1_1 tag=FORSTANDBY comment=NONE
channel ORA_DISK_1: backup set complete, elapsed time: 00:00:01
Finished backup at 25-OCT-15

RMAN> exit
[oracle@enode01 ~] $
```

24. Transfer all backup sets to wnode03. All backups created in the previous step must be transferred from the primary to the standby system.
Note: If the backup is written to NFS device in your production environment, this step can be skipped.

```
[oracle@enode01 ~]$ scp /tmp/ForStandby_* wnode03:/tmp
oracle@wnode03's password: <oracle>
ForStandby_0vqkjvgj_1_1          100%   151MB  30.2MB/s  00:05
ForStandby_10qkjvol_1_1          100%    13MB  13.0MB/s  00:01
[oracle@enode01 ~]$
```

25. (**Session 2**) Return to the terminal session on wnode03 to catalog the backups in the standby control file. For the standby database to know about the backups, catalog the backup pieces on the standby database. Then exit RMAN.

```
[oracle@wnode03 ~]$ rman
RMAN> connect target /
```

Copyright © 2016, Oracle and/or its affiliates. All rights reserved.

```

connected to target database: EASTDB (DBID=814727097, not open)

RMAN> CATALOG START WITH '/tmp/ForStandby';

using target database control file instead of recovery catalog
searching for all files that match the pattern /tmp/ForStandby

List of Files Unknown to the Database
=====
File Name: /tmp/ForStandby_0vqkjvgj_1_1
File Name: /tmp/ForStandby_10qkjvol_1_1

Do you really want to catalog the above files (enter YES or NO)? yes
cataloging files...
cataloging done

List of Cataloged Files
=====
File Name: /tmp/ForStandby_0vqkjvgj_1_1
File Name: /tmp/ForStandby_10qkjvol_1_1

RMAN>

```

26. Use the RMAN RECOVER command with the NOREDO option to apply the incremental backup to the standby database. All changed blocks captured in the incremental backup are updated at the standby database, bringing it up-to-date with the primary database.

```

RMAN> RECOVER DATABASE NOREDO;

Starting recover at 25-OCT-15
allocated channel: ORA_DISK_1
channel ORA_DISK_1: SID=53 instance=westdb1 device type=DISK
channel ORA_DISK_1: starting incremental datafile backup set restore
channel ORA_DISK_1: specifying datafile(s) to restore from backup set
destination for restore of datafile 00001:
+DATA/westdb/DATAFILE/system.284.892725097
destination for restore of datafile 00002:
+DATA/westdb/DATAFILE/undotbs2.286.892725109
destination for restore of datafile 00003:
+DATA/westdb/DATAFILE/sysaux.287.892725103
destination for restore of datafile 00004:
+DATA/westdb/DATAFILE/undotbs1.276.892725221
destination for restore of datafile 00005:
+DATA/westdb/DATAFILE/example.281.892725093
destination for restore of datafile 00006:
+DATA/westdb/DATAFILE/users.294.892725321
destination for restore of datafile 00007:
+DATA/westdb/DATAFILE/bartbs.303.892930675
destination for restore of datafile 00008: +FRA/WESTDB/online_tbs.dbf
channel ORA_DISK_1: reading from backup piece
/tmp/ForStandby_1cqkk6oi_1_1
channel ORA_DISK_1: piece handle=/tmp/ForStandby_1cqkk6oi_1_1
tag=FORSTANDBY
channel ORA_DISK_1: restored backup piece 1
channel ORA_DISK_1: restore complete, elapsed time: 00:00:45

```

```
Finished recover at 25-OCT-15
RMAN> exit
[oracle@wnode03 ~]$
```

27. **(Session 1)** Return to the terminal session on enode01. In RMAN, connect to the primary database and create a standby control file backup. Then exit RMAN.

```
[oracle@enode01 ~]$ rman
RMAN> connect target /
connected to target database: EASTDB (DBID=814727097)

RMAN> BACKUP CURRENT CONTROLFILE FOR STANDBY FORMAT
' /tmp/ForStandbyCTRL.bck';

Starting backup at 25-OCT-15
using target database control file instead of recovery catalog
allocated channel: ORA_DISK_1
channel ORA_DISK_1: SID=104 instance=eastdb1 device type=DISK
channel ORA_DISK_1: starting full datafile backup set
channel ORA_DISK_1: specifying datafile(s) in backup set
including standby control file in backup set
channel ORA_DISK_1: starting piece 1 at 25-OCT-15
channel ORA_DISK_1: finished piece 1 at 25-OCT-15
piece handle=/tmp/ForStandbyCTRL.bck tag=TAG20151025T170317
comment=NONE
channel ORA_DISK_1: backup set complete, elapsed time: 00:00:03
Finished backup at 25-OCT-15

RMAN> exit
[oracle@enode01 ~]$
```

28. Copy the standby control file backup to wnode03.

```
[oracle@enode01 ~]$ scp /tmp/ForStandbyCTRL.bck wnode03:/tmp
oracle@wnode03's password: <oracle>
ForStandbyCTRL.bck                                100%    10MB   9.9MB/s   00:00
[oracle@enode01 ~]$
```

29. **(Session 2)** Return to the terminal session on wnode03. Capture datafile information in the standby database (westdb). You now need to refresh the standby control file from primary control file (for standby) backup. However, because the datafile names are likely to be different from the primary, save the name of the datafiles on the standby first, which you can refer after restoring the control file from the primary backup to check if there is any discrepancy. Run the following query from standby and save the results for further use. Then exit SQL*Plus.

Note: The output may be different.

```
[oracle@wnode03 ~]$ sqlplus / as sysdba
SQL> spool datafile_names_step29.txt
SQL> set lines 200
SQL> col name format a60
SQL> select file#, name from v$datafile order by file#;

FILE# NAME
-----
```

```

1 +DATA/WESTDB/DATAFILE/system.284.892725097
2 +DATA/WESTDB/DATAFILE/undotbs2.286.892725109
3 +DATA/WESTDB/DATAFILE/sysaux.287.892725103
4 +DATA/WESTDB/DATAFILE/undotbs1.276.892725221
5 +DATA/WESTDB/DATAFILE/example.281.892725093
6 +DATA/WESTDB/DATAFILE/users.294.892725321
7 +DATA/WESTDB/DATAFILE/bartbs.303.892930675
8 +FRA/WESTDB/online_tbs.dbf

8 rows selected.

SQL> spool off
SQL> exit
[oracle@wnode03 ~] $

```

30. Start the standby database (westdb) in NOMOUNT state to restore the standby control file using RMAN.

```

[oracle@wnode03 ~] $ srvctl stop database -db westdb
[oracle@wnode03 ~] $ srvctl start database -db westdb -startoption
nomount
[oracle@wnode03 ~] $ rman
RMAN> connect target /

connected to target database: EASTDB (not mounted)

RMAN> RESTORE STANDBY CONTROLFILE FROM '/tmp/ForStandbyCTRL.bck';

Starting restore at 25-OCT-15
using target database control file instead of recovery catalog
allocated channel: ORA_DISK_1
channel ORA_DISK_1: SID=41 instance=westdb1 device type=DISK

channel ORA_DISK_1: restoring control file
channel ORA_DISK_1: restore complete, elapsed time: 00:00:03
output file name=+DATA/WESTDB/CONTROLFILE/current.293.893009367
output file name=+FRA/WESTDB/CONTROLFILE/current.327.893009369
Finished restore at 25-OCT-15

RMAN> exit
[oracle@wnode03 ~] $

```

31. Shut down the standby database (westdb) and start up the database in MOUNT state:

```

[oracle@wnode03 ~] $ srvctl stop database -db westdb
[oracle@wnode03 ~] $ srvctl start database -db westdb -startoption mount
[oracle@wnode03 ~] $

```

32. Catalog datafiles in STANDBY because the primary and standby databases have different directory structure and datafile names. If the primary and standby databases have identical directory structure and datafile names, this step can be skipped. Perform the following step in STANDBY for each diskgroup where the datafile directory structure between primary and standby are different.

```

[oracle@wnode03 ~] $ rman
RMAN> connect target /

```

```

connected to target database: EASTDB (DBID=814727097, not open)

RMAN> CATALOG START WITH '+DATA/westdb/datafile/';

Starting implicit crosscheck backup at 25-OCT-15
using target database control file instead of recovery catalog
allocated channel: ORA_DISK_1
Crosschecked 7 objects
Finished implicit crosscheck backup at 25-OCT-15

Starting implicit crosscheck copy at 25-OCT-15
using channel ORA_DISK_1
Crosschecked 2 objects
Finished implicit crosscheck copy at 25-OCT-15

searching for all files in the recovery area
cataloging files...
cataloging done

List of Cataloged Files
=====
File Name: +FRA/WESTDB/online_tbs.dbf
File Name:
+FRA/WESTDB/ARCHIVELOG/2015_10_25/thread_1_seq_74.596.893985301
File Name:
+FRA/WESTDB/ARCHIVELOG/2015_10_25/thread_1_seq_75.473.894040281
File Name:
+FRA/WESTDB/ARCHIVELOG/2015_10_25/thread_2_seq_75.550.893997699
File Name:
+FRA/WESTDB/ARCHIVELOG/2015_10_25/thread_2_seq_79.600.894042015
File Name:
+FRA/WESTDB/ARCHIVELOG/2015_10_25/thread_2_seq_80.599.894040899
File Name:
+FRA/WESTDB/ARCHIVELOG/2015_10_25/thread_1_seq_77.555.894042355
File Name:
+FRA/WESTDB/ARCHIVELOG/2015_10_24/thread_2_seq_72.601.893939325
File Name:
+FRA/WESTDB/ARCHIVELOG/2015_10_24/thread_1_seq_71.584.893938645
File Name:
+FRA/WESTDB/ARCHIVELOG/2015_10_24/thread_1_seq_72.586.893940017
File Name:
+FRA/WESTDB/ARCHIVELOG/2015_10_24/thread_2_seq_73.543.893940547
File Name:
+FRA/WESTDB/ARCHIVELOG/2015_10_24/thread_1_seq_73.579.893966087

searching for all files that match the pattern +DATA/westdb/datafile/

List of Files Unknown to the Database
=====
File Name: +DATA/westdb/DATAFILE/example.281.892725093
File Name: +DATA/westdb/DATAFILE/system.284.892725097
File Name: +DATA/westdb/DATAFILE/sysaux.287.892725103
File Name: +DATA/westdb/DATAFILE/undotbs2.286.892725109
File Name: +DATA/westdb/DATAFILE/undotbs1.276.892725221
File Name: +DATA/westdb/DATAFILE/users.294.892725321
File Name: +DATA/westdb/DATAFILE/bartbs.303.892930675

```

```
Do you really want to catalog the above files (enter YES or NO)? yes
cataloging files...
cataloging done

List of Cataloged Files
=====
File Name: +DATA/westdb/DATAFILE/example.281.892725093
File Name: +DATA/westdb/DATAFILE/system.284.892725097
File Name: +DATA/westdb/DATAFILE/sysaux.287.892725103
File Name: +DATA/westdb/DATAFILE/undotbs2.286.892725109
File Name: +DATA/westdb/DATAFILE/undotbs1.276.892725221
File Name: +DATA/westdb/DATAFILE/users.294.892725321
File Name: +DATA/westdb/DATAFILE/bartbs.303.892930675

RMAN>
```

33. (**Session 1**) Return to the terminal session on enode01 to determine if any files have been added to the primary since the standby current scn. If any datafiles have been added to primary after SCN 3802036, they will also have to be restored to the standby host (see Note 1531031.1 Steps to perform for rolling forward a standby database using RMAN incremental backup when datafile is added to primary) and cataloged as shown above before doing the switch.

```
[oracle@enode01 ~]$ sqlplus / as sysdba
SQL> SELECT FILE#, NAME FROM V$DATAFILE WHERE CREATION_CHANGE# >
3802036;

no rows selected

SQL>
```

34. (**Session 2**) Return to the terminal session on wnode03. If the above query returns with zero rows, you can switch the datafiles. This will rename the datafiles to its correct name at the standby site:

```
RMAN> SWITCH DATABASE TO COPY;

datafile 1 switched to datafile copy
"+DATA/westdb/DATAFILE/system.284.892725097"
datafile 2 switched to datafile copy
"+DATA/westdb/DATAFILE/undotbs2.286.892725109"
datafile 3 switched to datafile copy
"+DATA/westdb/DATAFILE/sysaux.287.892725103"
datafile 4 switched to datafile copy
"+DATA/westdb/DATAFILE/undotbs1.276.892725221"
datafile 5 switched to datafile copy
"+DATA/westdb/DATAFILE/example.281.892725093"
datafile 6 switched to datafile copy
"+DATA/westdb/DATAFILE/users.294.892725321"
datafile 7 switched to datafile copy
"+DATA/westdb/DATAFILE/bartbs.303.892930675"
datafile 8 switched to datafile copy "+FRA/WESTDB/online_tbs.dbf"

RMAN>
```

At this point, you can compare the query output from step 29 for any discrepancy (other than newly added datafiles) to ensure that you have all the datafiles added in the standby.

35. On the standby database (westdb), clear all standby redo log groups:

```
RMAN> select group# from v$standby_log order by 1;  
  
GROUP#  
-----  
      5  
      6  
      7  
      8  
      9  
     10  
     13  
     14  
  
8 rows selected  
  
RMAN> ALTER DATABASE CLEAR LOGFILE GROUP 5;  
Statement processed  
  
RMAN> ALTER DATABASE CLEAR LOGFILE GROUP 6;  
Statement processed  
  
RMAN> ALTER DATABASE CLEAR LOGFILE GROUP 7;  
Statement processed  
  
RMAN> ALTER DATABASE CLEAR LOGFILE GROUP 8;  
Statement processed  
  
RMAN> ALTER DATABASE CLEAR LOGFILE GROUP 9;  
Statement processed  
  
RMAN> ALTER DATABASE CLEAR LOGFILE GROUP 10;  
Statement processed  
  
RMAN> ALTER DATABASE CLEAR LOGFILE GROUP 13;  
Statement processed  
  
RMAN> ALTER DATABASE CLEAR LOGFILE GROUP 14;  
Statement processed  
  
RMAN>
```

36. Open the standby database in READ ONLY mode.

```
RMAN> alter database open read only;  
Statement processed  
  
RMAN>
```

37. On the standby database, start the MRP.

```
RMAN> ALTER DATABASE RECOVER MANAGED STANDBY DATABASE DISCONNECT;
Statement processed

RMAN>
```

To resolve NOLOGGING operations only, see the online documentation, and Note 958181.1

38. Verify that the changes made to the primary database have been successfully applied.

```
RMAN> select count(*) from hr.test09;

COUNT(*)
-----
3

RMAN>
```

39. Restart the Data Guard broker.

```
RMAN> alter system set dg_broker_start=true;
Statement processed

RMAN> exit
[oracle@wnode03 ~] $
```

40. Restart the physical standby database.

```
[oracle@wnode03 ~] $ srvctl stop database -db westdb
[oracle@wnode03 ~] $ srvctl start database -db westdb
[oracle@wnode03 ~] $
```

41. (Session 1) Return to the terminal session on enode01 to restart the Data Guard broker.

```
SQL> alter system set dg_broker_start=true;

System altered.

SQL> exit
[oracle@enode01 ~] $
```

42. Using DGMGRL, display the status of the Data Guard configuration.

Note: You may or may not see warning messages, such as ORA-16525, ORA-16665, ORA-16810, or ORA-16817 due to the lag and overall performance of the virtual machine environment. The warning messages will be cleared in a few minutes.

```
[oracle@enode01 ~] $ dgmgrl
DGMGRL> connect sysdg/oracle_4U@eastdb
Connected as SYSDG.
DGMGRL> show configuration

Configuration - dg_config

Protection Mode: MaxPerformance
Members:
  eastdb - Primary database
  westdb - Physical standby database
```

```
Fast-Start Failover: DISABLED  
Configuration Status:  
SUCCESS      (status updated 20 seconds ago)  
DGMGRL>
```

43. Wait until the physical standby database is synchronized. It may take a few moments for the Lag for the standby database to clear.

Note: Do not continue with labs until the Apply Lag is cleared.

```
DGMGRL> show database westdb  
  
Database - westdb  
  
Role:          PHYSICAL STANDBY  
Intended State:  APPLY-ON  
Transport Lag:   0 seconds (computed 1 second ago)  
Apply Lag:    0 seconds (computed 1 second ago)  
Average Apply Rate: 40.00 KByte/s  
Real Time Query: OFF  
Instance(s):  
    westdb1 (apply instance)  
    westdb2  
  
Database Status:  
SUCCESS  
DGMGRL>
```

44. Exit from all of your terminal sessions.

Practice 9-5: Performing Incomplete Recovery Using Flashback Database (Logical Failure)

Overview

In this practice, you will flash back your primary database after some incorrect updates to the database. After the primary database is recovered, you will resynchronize the physical standby database by using the Flashback Database feature.

This practice is for learning purposes. If you have a scenario similar to this in a production environment, you would probably choose a different solution to limit your flashback to the affected objects, rather than choose a flashback of the entire database.

Background

Incomplete recovery of the primary database is normally done in cases such as when the database is logically corrupted (by a user or an application) or when a tablespace or data file was accidentally dropped from database. Depending on the current database checkpoint SCN on the standby database instances, you can use one of the following procedures to perform incomplete recovery of the primary database in case of a logical failure.

Using Flashback Database is the recommended procedure when the Flashback Database feature is enabled on the primary database, none of the database files are lost, and the point-in-time recovery is greater than oldest flashback SCN or the oldest flashback time.

Task Summary

- **Task 1:** Enabling Flashback Logging
- **Task 2:** Performing Flashback Database on the primary database
- **Task 3:** Synchronizing the physical standby database

Terminal Sessions

- **Session 1:** ssh oracle@enode01
- **Session 2:** ssh oracle@wnode03

Task 1: Enabling Flashback Logging

1. **(Session 1)** Open a terminal window on enode01 as the oracle user.

```
[Classroom PC]$ ssh oracle@enode01
oracle@enode01's password: <oracle>
[oracle@enode01 ~]$
```

2. Make sure that you set up your environment variables correctly for the oracle user.

```
[oracle@enode01 ~]$ . oraenv
ORACLE_SID = [oracle]? eastdb
The Oracle base has been set to /u01/app/oracle
[oracle@enode01 ~]$ export ORACLE_SID=eastdb1
[oracle@enode01 ~]$
```

3. Log in to SQL*Plus to connect to the eastdb database and determine whether flashback logging is enabled.

```
[oracle@enode01 ~]$ sqlplus / as sysdba
SQL> SELECT flashback_on FROM v$database;

FLASHBACK_ON
-----
NO

SQL>
```

4. Enable flashback logging with a tool of your choice if it is disabled.

```
SQL> ALTER DATABASE FLASHBACK ON;
SQL>
```

5. Verify that flashback logging has been enabled.

```
SQL> SELECT FLASHBACK_ON FROM V$DATABASE;

FLASHBACK_ON
-----
YES

SQL>
```

6. (Session 2) Open a terminal window to connect to wnode03 as the oracle user.

```
[Classroom PC]$ ssh oracle@wnode03
oracle@wnode03's password: <oracle>
[oracle@wnode03 ~]$
```

7. Make sure that you set up your environment variables correctly for the oracle user.

```
[oracle@wnode03 ~]$ . oraenv
ORACLE_SID = [oracle]? westdb1
The Oracle base has been set to /u01/app/oracle
[oracle@wnode03 ~]$
```

8. Log in to SQL*Plus to connect to the westdb database and determine whether flashback logging is enabled.

```
[oracle@wnode03 ~]$ sqlplus / as sysdba
SQL> SELECT flashback_on FROM v$database;

FLASHBACK_ON
-----
NO

SQL>
SQL>
```

9. Stop MRP to enable flashback logging if it is disabled.

```
SQL> alter database recover managed standby database cancel;
Database altered.
SQL> ALTER DATABASE FLASHBACK ON;
Database altered.
SQL>
```

10. Verify that flashback logging has been enabled and start MRP again.

```
SQL> SELECT FLASHBACK_ON FROM V$DATABASE;

FLASHBACK_ON
-----
YES

SQL> alter database recover managed standby database disconnect;
Database altered.

SQL>
```

Task 2: Performing Flashback Database on the Primary Database

There are several ways in which you can perform a flashback database operation. You can use a guaranteed restore point, SCN, time value, threads, and so on. This example uses the SCN, but you could also use the restore point.

11. (Session 1) Return to the first terminal window to determine your current SCN. You will need to use it in a later practice step.

Note: Your SCN number may be different.

```
SQL> SELECT current_scn FROM v$database;

CURRENT_SCN
-----
15990866

SQL>
```

12. View HR data. You will use this information for comparison during this practice.

- a. Determine the sum of the SALARY column in the HR.EMPLOYEES table.

```
SQL> SELECT sum(salary) FROM hr.employees;

SUM(SALARY)
-----
691416

SQL>
```

- b. Determine the total number of employees in department 90.

```
SQL> SELECT count(*) FROM hr.employees where department_id=90;

COUNT(*)
-----
3

SQL>
```

13. Execute the following UPDATE commands to update tables in the HR schema. It creates issues from which you will “recover” by flashing back the database in this practice.

```
SQL> update hr.employees set department_id = 90 where job_id =
'IT_PROG';
5 rows updated.

SQL> update hr.employees e set salary = least(e.salary,(select
(min_salary + max_salary)/2 * 1.10 from hr.jobs j where j.job_id =
e.job_id)) where job_id not like 'AD_%';
```

```
103 rows updated.
```

```
SQL>
```

14. Commit your data and determine the current SCN.

Note: Your SCN number may be different.

```
SQL> COMMIT;
Commit complete.

SQL> SELECT current_scn FROM v$database;

CURRENT_SCN
-----
15991580

SQL>
```

15. Query the data in the HR schema again and compare the results to the values you received in the queries in step 12.

- a. Determine the sum of the SALARY column in the HR.EMPLOYEES table.

```
SQL> SELECT sum(salary) FROM hr.employees;

SUM(SALARY)
-----
679092.4

SQL>
```

- b. Determine the total number of employees in department 90.

```
SQL> SELECT count(*) FROM hr.employees where department_id=90;

COUNT(*)
-----
8

SQL>
```

16. Switch the current log file by issuing ALTER SYSTEM ARCHIVE LOG CURRENT to advance the online redo log sequence number.

Note: This step is to avoid an unknown issue, such as ORA-38729: Not enough flashback database log data to do FLASHBACK in step 21.

```
SQL> alter system archive log current;
System altered.
SQL> exit
[oracle@enode01 ~] $
```

17. You need to restore the database so that the data is as it was when you started this practice. For training purposes, use Flashback Database for this operation.

- a. Shut down the database instance and start it in MOUNT mode.

```
[oracle@enode01 ~] $ srvctl stop database -db eastdb
[oracle@enode01 ~] $ srvctl start database -db eastdb -startoption mount
[oracle@enode01 ~] $
```

- b. Log in to RMAN and use the FLASHBACK DATABASE command to flash back the database to the SCN you noted in step 11. Exit RMAN.

Note: Enter your SCN number.

```
[oracle@enode01 ~]$ rman target ''/ as sysbackup''  
  
Recovery Manager: Release 12.1.0.2.0 - Production on Thu Oct 29  
20:05:30 2015  
  
Copyright (c) 1982, 2014, Oracle and/or its affiliates. All rights  
reserved.  
  
connected to target database: EASTDB (DBID=816306320, not open)  
using target database control file instead of recovery catalog  
  
RMAN> flashback database to scn=15990866;  
  
Starting flashback at 26-OCT-15  
allocated channel: ORA_DISK_1  
channel ORA_DISK_1: SID=54 instance=eastdb1 device type=DISK  
  
starting media recovery  
media recovery complete, elapsed time: 00:00:07  
...  
  
Finished flashback at 26-OCT-15  
  
RMAN> exit  
[oracle@enode01 ~]$
```

- c. Verify that the database was flashed back correctly by querying the HR.EMPLOYEES table again. The values should match the values you obtained in the queries in step 12.

```
[oracle@enode01 ~]$ sqlplus / as sysdba  
  
SQL> alter database open read only;  
  
Database altered.  
  
SQL> SELECT sum(salary) FROM hr.employees;  
  
SUM(SALARY)  
-----  
691416  
  
SQL> SELECT count(*) FROM hr.employees where department_id=90;  
  
COUNT(*)  
-----  
3  
  
SQL> exit  
[oracle@enode01 ~]$
```

- d. After you verified that you flashed back to the desired state, open the database with RESETLOGS by using one of the primary database instances.

```
[oracle@enode01 ~]$ srvctl stop database -db eastdb
[oracle@enode01 ~]$ sqlplus / as sysdba
SQL> startup mount
ORACLE instance started.

Total System Global Area  524288000 bytes
Fixed Size                  2926320 bytes
Variable Size                436209936 bytes
Database Buffers              79691776 bytes
Redo Buffers                   5459968 bytes
Database mounted.
SQL> alter database open resetlogs;
Database altered.
SQL> exit
[oracle@enode01 ~]$
```

18. Restart the primary database.

```
[oracle@enode01 ~]$ srvctl stop database -db eastdb
[oracle@enode01 ~]$ srvctl start database -db eastdb
[oracle@enode01 ~]$
```

Task 3: Synchronizing the Physical Standby Database

19. On the primary database, determine an SCN that is at least two SCNs prior to the SCN where the OPEN RESETLOGS command was issued. This is necessary to enable the standby to recover properly through OPEN RESETLOGS. Use the following query to find the “BEFORE RESETLOGS” SCN.

Note: Your SCN number may be different.

```
[oracle@enode01 ~]$ sqlplus / as sysdba
SQL> SELECT to_char(resetlogs_change# - 2) FROM v$database;
TO_CHAR(RESETLOGS_CHANGE#-2)
-----
15990866

SQL>
```

20. (**Session 2**) Return to the terminal session on wnode03 to obtain the current SCN by using the following query.

Note: Your SCN number may be different.

```
SQL> SELECT to_char(current_scn) FROM v$database;
TO_CHAR(CURRENT_SCN)
-----
15993228

SQL>
```

21. Flash back the standby database to the “before RESETLOGS” SCN that you queried in step 19.

Note: Use your SCN number.

```
SQL> FLASHBACK STANDBY DATABASE TO SCN 15990866;  
Flashback complete.  
SQL>
```

22. Restart managed recovery on the standby database. The standby database will be ready to receive and apply logs from the primary database.

```
SQL> ALTER DATABASE RECOVER MANAGED STANDBY DATABASE DISCONNECT;  
Database altered.  
SQL>
```

23. **(Session 1)** Return to the terminal session on enode01 to start the redo transport service.

Note: This step is to speed up the synchronization and optional.

```
SQL> alter system set log_archive_dest_state_2=enable;  
System altered.  
SQL> exit  
[oracle@enode01 ~] $
```

24. Determine the status of the Data Guard configuration.

Note: You may or may not see warning messages, such as ORA-16525, ORA-16665, ORA-16810, or ORA-16817 due to the lag and overall performance of the virtual machine environment. The warning messages will be cleared in a few minutes.

```
[oracle@enode01 ~]$ dgmgrl  
DGMGRL> connect sysdg/oracle_4U@eastdb  
Connected as SYSDG.  
DGMGRL> show configuration  
Configuration - dg_config  
  
Protection Mode: MaxPerformance  
Members:  
    eastdb - Primary database  
    westdb - Physical standby database  
  
Fast-Start Failover: DISABLED  
  
Configuration Status:  
SUCCESS      (status updated 33 seconds ago)  
  
DGMGRL>
```

25. Wait until the physical standby database is synchronized. It may take a few moments for the Lag for the standby database to clear.

Note: Do not continue with the practices until the Apply Lag is cleared.

```
DGMGRL> show database westdb

Database - westdb

Role: PHYSICAL STANDBY
Intended State: APPLY-ON
Transport Lag: 0 seconds (computed 1 second ago)
Apply Lag: 0 seconds (computed 1 second ago)
Average Apply Rate: 40.00 KByte/s
Real Time Query: OFF
Instance(s):
    westdb1 (apply instance)
    westdb2

Database Status:
SUCCESS

DGMGRL>
```

26. Exit all of your terminal sessions.

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

Practices for Lesson 10: Performing Data Guard Standby-First Patch Apply

Chapter 10

Practices for Lesson 10: Overview

Practice Overview

In this practice, you will apply patch 20108098 to the Oracle RAC database homes in a Data Guard configuration by using the **Standby-First Rolling patch apply technique**. This patch fixes a problem with the GV\$SEQUENCES global view. This technique can be used for any certified patches, such as Patch Set Update (PSU), Critical Patch Update (CPU), and so on. You will patch the RAC homes in the physical standby database hosts first in a rolling fashion and then in the primary hosts in the same way.

Practice Summary

- Practice 10-1: Patching the RAC Database Homes in the *Original* Standby Hosts
- Practice 10-2: Performing Switchover
- Practice 10-3: Patching the RAC Database Homes in the *New* Standby Hosts
- Practice 10-4: Performing Switchover to the Original Configuration

Practice 10-1: Patching the RAC Database Homes in the Standby Hosts

Overview

In this practice, you will apply patch 20108098 to the Oracle RAC database homes in the standby hosts by using the Standby-First Rolling patch apply technique.

Terminal Sessions

- **Session 1:** `ssh oracle@wnode03`
- **Session 2:** `ssh oracle@wnode03`

Tasks

1. **(Session 1)** Establish a terminal window as the `oracle` user to connect to the first standby host `wnode03`. Use `oraenv` to set the Oracle environment.

```
[Classsroom_PC ~]$ ssh oracle@wnode03
oracle@wnode03's password:
[oracle@wnode03 ~]$ . oraenv
ORACLE_SID = [oracle] ? westdb1
The Oracle base has been set to /u01/app/oracle
[oracle@wnode03 ~]$
```

2. The patch you will install is 20108098. It is located in `/home/oracle/patch/20108098`. It fixes an issue with the `GV$SEQUENCES` view. Navigate to the directory and take a few moments to view the `README` file, particularly Section I (Prerequisites) and Section II (Installation).

```
[oracle@wnode03 ~]$ cd /home/oracle/patch/20108098
[oracle@wnode03 20108098]$ ls -al
total 24
drwxrwxr-x 4 root root 4096 Jan 20 20:05 .
drwxr-xr-x 3 oracle oinstall 4096 Jan 22 18:31 ..
drwxr-xr-x 3 root root 4096 Jan 20 20:05 etc
drwxr-xr-x 3 root root 4096 Jan 20 20:05 files
-rw-rw-r-- 1 root root 5855 Jan 20 20:05 README.txt

[oracle@wnode03 20108098]$ cat README.txt

Oracle Database 12c Release 12.1.0.2.0
ORACLE DATABASE Patch for Bug# 20108098 for Linux-x86-64 Platforms

This patch is RAC Rolling Installable - Please read My Oracle
Support Document 244241.1
https://support.us.oracle.com/oip/faces/secure/km/DocumentDisplay.jspx?id=244241.1 Rolling Patch - OPatch Support for RAC.

This patch is Data Guard Standby-First Installable - Please read My
Oracle Support Note 1265700.1
https://support.us.oracle.com/oip/faces/secure/km/DocumentDisplay.jspx?id=1265700.1

Oracle Patch Assurance - Data Guard Standby-First Patch Apply for
details on how to remove risk and reduce downtime when applying
this patch. Released: Tue Jan 20 11:47:25 2015
```

This document describes how you can install the ORACLE DATABASE interim patch for bug# 20108098 on your Oracle Database 12c Release 12.1.0.2.0

(I) Prerequisites

Before you install or deinstall the patch, ensure that you meet the following requirements:

Note: In case of an Oracle RAC environment, meet these prerequisites on each of the nodes.

1. Ensure that the Oracle home on which you are installing the patch or from which you are rolling back the patch is Oracle Database 12c Release 12.1.0.2.0.
2. Oracle recommends that you use the latest version available for 12c Release 12.1.0.2.0. If you do not have OPatch 12c Release 12.1.0.1.4 or the latest version available for 12c Release 12.1.0.2.0, then download it from patch# 6880880 for 12.1.0.2.0 release.

For information about OPatch documentation, including any known issues, see My Oracle Support Document 293369.1 OPatch documentation list:

<https://support.oracle.com/CSP/main/article?cmd=show&type=NOT&id=224346.1>

3. Ensure that you set (as the home user) the ORACLE_HOME environment variable to the Oracle home.

4. Ensure that the \$PATH definition has the following executables: make, ar, ld and nm. The location of these executables depends on your operating system. On many operating systems, they are located in /usr/ccs/bin.

5. Ensure that you verify the Oracle Inventory because Opatch accesses it to install the patches. To verify the inventory, run the following command.

\$ opatch lsinventory

Note:

- If this command succeeds, it will list the Top-Level Oracle Products and one-off patches if any that are installed in the Oracle Home.
- Save the output so you have the status prior to the patch apply.
- If the command displays some errors, then contact Oracle Support and resolve the issue first before proceeding further.

6. (Only for Installation) Maintain a location for storing the contents of the patch ZIP file. In the rest of the document, this location (absolute path) is referred to as <PATCH_TOP_DIR>. Extract the contents of the patch ZIP file to the location (PATCH_TOP_DIR) you have created above. To do so, run the following command:

\$ unzip -d <PATCH_TOP_DIR> p20108098_121020_Linux-x86-64.zip

7. (Only for Installation) Determine whether any currently installed interim patches conflict with this patch 20108098 as shown below:

```
$ cd <PATCH_TOP_DIR>/20108098
$ opatch prereq CheckConflictAgainstTOHWithDetail -ph .
The report will indicate the patches that conflict with
this patch and the patches for which the current 20108098 is a
superset.
```

Note:

When OPatch starts, it validates the patch and ensures that there are no conflicts with the software already installed in the ORACLE_HOME. OPatch categorizes conflicts into the following types:

- Conflicts with a patch already applied to the ORACLE_HOME that is a subset of the patch you are trying to apply
- In this case, continue with the patch installation because the new patch contains all the fixes from the existing patch in the ORACLE_HOME. The subset patch will automatically be rolled back prior to the installation of the new patch.
- Conflicts with a patch already applied to the ORACLE_HOME - In this case, stop the patch installation and contact Oracle Support Services.

8. Ensure that you shut down all the services running from the Oracle home.

Note:

- For a Non-RAC environment, shut down all the services running from the Oracle home.
- For a RAC environment, shut down all the services (database, ASM, listeners, nodeapps, and CRS daemons) running from the Oracle home of the node you want to patch. After you patch this node, start the services on this node. Repeat this process for each of the other nodes of the Oracle RAC system. OPatch is used on only one node at a time.

(II) Installation

To install the patch, follow these steps:

1. Set your current directory to the directory where the patch is located and then run the OPatch utility by entering the following commands:

```
$ cd <PATCH_TOP_DIR>/20108098
$ opatch apply
```

2. Verify whether the patch has been successfully installed by running the following command:

```
$ opatch lsinventory
```

3. Start the services from the Oracle home.

(III) Deinstallation

Ensure to follow the Prerequisites (Section I). To deinstall the patch, follow these steps:

1. Deinstall the patch by running the following command:

```
$ opatch rollback -id 20108098
```

2. Start the services from the Oracle home.
3. Ensure that you verify the Oracle Inventory and compare the output with the one run before the patch installation and re-apply any patches that were rolled back as part of this patch apply. To verify the inventory, run the following command:

```
$ opatch lsinventory

(IV) Bugs Fixed by This Patch
-----
The following are the bugs fixed by this patch:
20108098: GV$SEQUENCES BROKEN
-----
Copyright 2015, Oracle and/or its affiliates. All rights reserved.
-----
[oracle@wnode03 20108098]$
```

3. Make sure that your PATH definition points to the make, ar, ld, and nm executables. In your practice environment, they are located in /usr/bin.

```
[oracle@wnode03 20108098]$ which make ar ld nm
/usr/bin/make
/usr/bin/ar
/usr/bin/ld
/usr/bin/nm
[oracle@wnode03 ~]$ echo $PATH
/usr/lib64/qt-
3.3/bin:/usr/local/bin:/bin:/usr/bin:/home/oracle/bin:/u01/app/oracle/product/12.1.0/dbhome_1/bin
[oracle@wnode03 20108098]$
```

4. Put the OPatch directory in your path and verify that OPatch has access to the Oracle inventory with the opatch lsinventory command.

```
[oracle@wnode03 20108098]$ export
PATH=$PATH:/u01/app/oracle/product/12.1.0/dbhome_1/OPatch
[oracle@wnode03 ~]$ opatch lsinventory
Oracle Interim Patch Installer version 12.1.0.1.3
Copyright (c) 2015, Oracle Corporation. All rights reserved.

Oracle Home      : /u01/app/oracle/product/12.1.0/dbhome_1
Central Inventory : /u01/app/oraInventory
                  from          : /u01/app/oracle/product/12.1.0/dbhome_1/oraInst.loc
OPatch version   : 12.1.0.1.3
OUI version     : 12.1.0.2.0
Log file location :
/u01/app/oracle/product/12.1.0/dbhome_1/cfgtoollogs/opatch/opatch2015-10-11_20-35-49PM_1.log

Lsinventory Output file location :
/u01/app/oracle/product/12.1.0/dbhome_1/cfgtoollogs/opatch/lsinv/lsinventory2015-10-11_20-35-49PM.txt

-----
Installed Top-level Products (1) :
```

```
Oracle Database 12c
12.1.0.2.0
There are 1 products installed in this Oracle Home.
```

There are no Interim patches installed in this Oracle Home.

```
Rac system comprising of multiple nodes
Local node = wnode03
Remote node = wnode04
```

```
OPatch succeeded.
[oracle@wnode03 20108098] $
```

- Determine whether any currently installed interim patches conflict with patch 20108098 by running the `opatch prereq` command as shown in the following:

```
[oracle@wnode03 20108098] $ opatch prereq
CheckConflictAgainstOHWithDetail -ph ./
Oracle Interim Patch Installer version 12.1.0.1.3
Copyright (c) 2015, Oracle Corporation. All rights reserved.

PREREQ session

Oracle Home      : /u01/app/oracle/product/12.1.0/dbhome_1
Central Inventory : /u01/app/oraInventory
                  from       : /u01/app/oracle/product/12.1.0/dbhome_1/oraInst.loc
OPatch version   : 12.1.0.1.3
OUI version      : 12.1.0.2.0
Log file location :
/u01/app/oracle/product/12.1.0/dbhome_1/cfgtoollogs/opatch/opatch2015-
10-11_20-37-36PM_1.log

Invoking prereq "checkconflictagainstohwithdetail"

Prereq "checkConflictAgainstOHWithDetail" passed.

OPatch succeeded.
[oracle@wnode03 20108098] $
```

- (Session 2) Open a second terminal window as the `oracle` user. Set the environment and use the `DGMGRl` command to determine the apply instance.

Note: You may or may not see Transport Lag caused by idle time. As long as the physical standby database is synchronized (Apply Lag is zero), you can safely move to the next step.

```
[Classroom_PC ~] $ ssh oracle@wnode03
oracle@wnode03's password: <oracle>
[oracle@wnode03 ~] $
[oracle@wnode03 ~] $ . oraenv
ORACLE_SID = [oracle] ? westdb1
The Oracle base has been set to /u01/app/oracle
[oracle@wnode03 ~] $
[oracle@wnode03 ~] $ dgmgrl
```

```
DGMGRL> connect sysdg/oracle_4U@westdb
Connected as SYSDG.
DGMGRL> show database westdb

Database - westdb

Role: PHYSICAL STANDBY
Intended State: APPLY-ON
Transport Lag: 0 seconds (computed 1 second ago)
Apply Lag: 0 seconds (computed 1 second ago)
Average Apply Rate: 4.00 KByte/s
Real Time Query: OFF
Instance(s):
  westdb1 (apply instance)
  westdb2

Database Status:
SUCCESS

DGMGRL>
```

7. You will patch the first standby host (wnode03). If the current apply instance is running on wnode03, switch the apply instance to the second standby host (wnode04).

Note: You may or may not see Transport Lag caused by the idle time. As long as the physical standby database is synchronized (Apply Lag is zero), you can safely move to the next step.

```
DGMGRL> edit database westdb set state ='APPLY-ON' with apply
instance = 'westdb2';
Succeeded.
DGMGRL> show database westdb

Database - westdb

Role: PHYSICAL STANDBY
Intended State: APPLY-ON
Transport Lag: 0 seconds
Apply Lag: 0 seconds
Real Time Query: OFF
Instance(s):
  westdb1
  westdb2 (apply instance)

Database Status:
SUCCESS

DGMGRL> exit
[oracle@wnode03 ~] $
```

8. Shut down the instance running on the local node, wnode03.

```
[oracle@wnode03 ~] $ srvctl stop instance -db westdb -instance westdb1
[oracle@wnode03 ~] $
```

9. **(Session 1)** Return to the first terminal and start the patch installation by executing `opatch apply`. When OPatch asks “Is the local system ready for patching?”, answer yes by typing `Y` and pressing Enter.

```
[oracle@wnode03 20108098]$ opatch apply

Oracle Interim Patch Installer version 12.1.0.1.3
Copyright (c) 2015, Oracle Corporation. All rights reserved.

Oracle Home      : /u01/app/oracle/product/12.1.0/dbhome_1
Central Inventory : /u01/app/oraInventory
                  from       : /u01/app/oracle/product/12.1.0/dbhome_1/oraInst.loc
OPatch version   : 12.1.0.1.3
OUI version     : 12.1.0.2.0
Log file location :
/u01/app/oracle/product/12.1.0/dbhome_1/cfgtoollogs/opatch/2
0108098_Oct_11_2015_20_53_03/apply2015-10-11_20-53-03PM_1.log

Applying interim patch '20108098' to OH
'/u01/app/oracle/product/12.1.0/dbhome_1'
'

Verifying environment and performing prerequisite checks...
All checks passed.

This node is part of an Oracle Real Application Cluster.
Remote nodes: 'wnode04'
Local node: 'wnode03'
Please shutdown Oracle instances running out of this ORACLE_HOME on the
local sy
stem.
(Oracle Home = '/u01/app/oracle/product/12.1.0/dbhome_1')

Is the local system ready for patching? [y|n]
Y
User Responded with: Y
Backing up files...

Patching component oracle.rdbms, 12.1.0.2.0...

Verifying the update...

The local system has been patched. You can restart Oracle instances on
it.

Patching in rolling mode.

The node 'wnode04' will be patched next.

Please shutdown Oracle instances running out of this ORACLE_HOME on
'wnode04'.
(Oracle Home = '/u01/app/oracle/product/12.1.0/dbhome_1')

[oracle@wnode03 20108098]$
```

When patching is finished on wnode03, the OPatch dialog box will inform you that the instance can be restarted on wnode03 and will prompt you for the name of the next node to patch, wnode04. Before continuing, go to the next step.

10. **(Session 2)** Go to the second terminal window and restart the instance on wnode03.

```
[oracle@wnode03 ~]$ srvctl start instance -db westdb -instance westdb1
[oracle@wnode03 ~]$
```

11. From the second terminal window, switch the apply instance to the *patched* standby host (wnode03) using DGMGRL command.

Note: You may or may not see Transport Lag caused by the idle time. As long as the physical standby database is synchronized (Apply Lag is zero), you can safely move to the next step.

```
[oracle@wnode03 ~]$ dgmgrl
DGMGRl> connect sysdg/oracle_4U@westdb
Connected as SYSDG.
DGMGRl> edit database westdb set state ='APPLY-ON' with apply
instance = 'westdb1';
Succeeded.
DGMGRl> show database westdb
Database - westdb

Role: PHYSICAL STANDBY
Intended State: APPLY-ON
Transport Lag: 0 seconds
Apply Lag: 0 seconds
Real Time Query: OFF
Instance(s):
  westdb1 (apply instance)
  westdb2

Database Status:
SUCCESS
DGMGRl> exit
[oracle@wnode03 ~]$
```

12. Stop the instance on wnode04.

```
[oracle@wnode03 ~]$ srvctl stop instance -db westdb -instance westdb2
[oracle@wnode03 ~]$
```

13. **(Session 1)** Return to the first terminal window, type wnode04, and press Enter. When asked "Is the node ready for patching? [y|n]", type Y and press Enter.

```
Is the node ready for patching? [y|n]
Y
User Responded with: Y
Updating nodes 'wnode04'
  Apply-related files are:
    FP =
    "/u01/app/oracle/product/12.1.0/dbhome_1/.patch_storage/20108098_Jan_2
0_2015_12_04_56/rac/copy_files.txt"
    DP =
    "/u01/app/oracle/product/12.1.0/dbhome_1/.patch_storage/20108098_Jan_2
0_2015_12_04_56/rac/copy_dirs.txt"
    MP =
    "/u01/app/oracle/product/12.1.0/dbhome_1/.patch_storage/20108098_Jan_2
```

```

0_2015_12_04_56/rac/make_cmds.txt"
RC =
"/u01/app/oracle/product/12.1.0/dbhome_1/.patch_storage/20108098_Jan_2
0_2015_12_04_56/rac/remote_cmds.txt"

Instantiating the file
"/u01/app/oracle/product/12.1.0/dbhome_1/.patch_storage/2
0108098_Jan_20_2015_12_04_56/rac/copy_files.txt.instantiated" by
replacing $ORAC
LE_HOME in
"/u01/app/oracle/product/12.1.0/dbhome_1/.patch_storage/20108098_Jan_
20_2015_12_04_56/rac/copy_files.txt" with actual path.
Propagating files to remote nodes...
Instantiating the file
"/u01/app/oracle/product/12.1.0/dbhome_1/.patch_storage/2
0108098_Jan_20_2015_12_04_56/rac/copy_dirs.txt.instantiated" by
replacing $ORACL
E_HOME in
"/u01/app/oracle/product/12.1.0/dbhome_1/.patch_storage/20108098_Jan_2
0_2015_12_04_56/rac/copy_dirs.txt" with actual path.
Propagating directories to remote nodes...
Instantiating the file
"/u01/app/oracle/product/12.1.0/dbhome_1/.patch_storage/2
0108098_Jan_20_2015_12_04_56/rac/make_cmds.txt.instantiated" by
replacing $ORACL
E_HOME in
"/u01/app/oracle/product/12.1.0/dbhome_1/.patch_storage/20108098_Jan_2
0_2015_12_04_56/rac/make_cmds.txt" with actual path.
Running command on remote node 'wnode04':
cd /u01/app/oracle/product/12.1.0/dbhome_1/rdbms/lib; /usr/bin/make -f
ins_rdbms
          .mk ioracle
ORACLE_HOME=/u01/app/oracle/product/12.1.0/dbhome_1 || echo REMOTE_M
AKE FAILED:::>&2

The node 'wnode04' has been patched. You can restart Oracle instances
on it.

There were relinks on remote nodes. Remember to check the binary size
and times                                tamp on the
nodes 'wnode04' .
The following make commands were invoked on remote nodes:
'cd /u01/app/oracle/product/12.1.0/dbhome_1/rdbms/lib; /usr/bin/make -f
ins_rdbm
          s.mk ioracle
ORACLE_HOME=/u01/app/oracle/product/12.1.0/dbhome_1
'

Patch 20108098 successfully applied
Log file location:
/u01/app/oracle/product/12.1.0/dbhome_1/cfgtoollogs/patch/20
108098_Oct_11_2015_20_53_03/apply2015-10-11_20-53-03PM_1.log

OPatch succeeded.
[oracle@wnode03 20108098] $

```

14. **(Session 2)** After OPatch has finished patching, go to the second terminal window and restart the instance on wnode04.

```
[oracle@wnode03 ~]$ srvctl start instance -db westdb -instance westdb2  
[oracle@wnode03 ~]$
```

15. **(Session 1)** Return to the first terminal session and run the opatch lsinventory command to ensure that the Oracle homes on two nodes were patched successfully.

```
[oracle@wnode03 20108098]$ opatch lsinventory  
Oracle Interim Patch Installer version 12.1.0.1.3  
Copyright (c) 2015, Oracle Corporation. All rights reserved.  
  
Oracle Home      : /u01/app/oracle/product/12.1.0/dbhome_1  
Central Inventory : /u01/app/oraInventory  
    from          : /u01/app/oracle/product/12.1.0/dbhome_1/oraInst.loc  
OPatch version   : 12.1.0.1.3  
OUI version     : 12.1.0.2.0  
Log file location :  
/u01/app/oracle/product/12.1.0/dbhome_1/cfgtoollogs/opatch/o  
patch2015-10-11_21-18-12PM_1.log  
  
Lsinventory Output file location :  
/u01/app/oracle/product/12.1.0/dbhome_1/cfgto  
ollogs/opatch/lsinv/lsinventory2015-10-11_21-18-12PM.txt  
  
-----  
Installed Top-level Products (1) :  
  
Oracle Database 12c  
12.1.0.2.0  
There are 1 products installed in this Oracle Home.  
  
Interim patches (1) :  
  
Patch 20108098      : applied on Sun Oct 11 20:59:01 UTC 2015  
Unique Patch ID: 18461252  
Created on 20 Jan 2015, 12:04:56 hrs PST8PDT  
Bugs fixed:  
20108098  
  
Rac system comprising of multiple nodes  
Local node = wnode03  
Remote node = wnode04  
  
-----  
OPatch succeeded.  
[oracle@wnode03 20108098]$
```

16. Close all terminal windows opened for practice. At this stage, all Oracle RAC Homes in the standby hosts have been patched successfully.

Note: In a real scenario, you will not perform SQL installation (for example, run catbundle.sql or datapatch) at this time for the patch that requires SQL installation. SQL installation is performed after the primary database and all standby databases have their database home binaries patched to the same level if required.

Practice 10-2: Performing Switchover

Overview

In this practice, you will perform switchover to complete the patch installation in the new physical standby database on enode01 and enode02.

Tasks

- Establish an ssh connection as the `oracle` user to connect to the first standby host enode01. Use `oraenv` to set the Oracle environment.

```
[Classroom Pc ~]$ ssh oracle@enode01
oracle@enode01's password: <oracle>
[oracle@enode01 ~]$ . oraenv
ORACLE_SID = [oracle] ? eastdb
The Oracle base has been set to /u01/app/oracle
[oracle@enode01 ~]$ export ORACLE_SID=eastdb1
[oracle@enode01 ~]$
```

- Using DGMGRL, perform the switchover operation.

```
[oracle@enode01 ~]$ dgmgrl
DGMGRL> connect sysdg/oracle_4U@eastdb
Connected as SYSDG.
DGMGRL> validate database eastdb

Database Role: Primary database

Ready for Switchover: Yes

DGMGRL> validate database westdb

Database Role: Physical standby database
Primary Database: eastdb

Ready for Switchover: Yes
Ready for Failover: Yes (Primary Running)

Transport-Related Property Settings:
Property          eastdb Value           westdb Value
RedoRoutes        (eastdb:westdb ASYNC)  (westdb:eastdb ASYNC)

DGMGRL>
DGMGRL> switchover to westdb
Performing switchover NOW, please wait...
New primary database "westdb" is opening...
Oracle Clusterware is restarting database "eastdb" ...
Switchover succeeded, new primary is "westdb"
DGMGRL>
```

- Verify the current database role in the Data Guard environment.

Note: You may or may not see warning messages, such as ORA-16525, ORA-16665, ORA-16810, or ORA-16817 due to the lag and overall performance of the virtual machine environment. The warning messages will be cleared in a few minutes.

```
DGMGRL> show configuration
Configuration - dg_config
```

```
Protection Mode: MaxPerformance
Members:
westdb - Primary database
eastdb - Physical standby database

Fast-Start Failover: DISABLED

Configuration Status:
SUCCESS      (status updated 282 seconds ago)

DGMGRL>
```

4. Wait until the physical standby database is synchronized. It may take a few moments for the Lag for the standby database to clear.

Note: Do not continue with the practices until the Apply Lag is cleared.

```
DGMGRL> show database eastdb

Database - eastdb

Role:          PHYSICAL STANDBY
Intended State: APPLY-ON
Transport Lag: 0 seconds
Apply Lag:    0 seconds
Real Time Query: OFF
Instance(s):
  eastdb1 (apply instance)
  eastdb2

Database Status:
SUCCESS

DGMGLR> exit
[oracle@enode01 ~]$
```

5. Close all terminal windows opened for practice.

Practice 10-3: Patching the RAC Database Homes in the New Standby Hosts

Overview

In this practice, you will apply patch 20108098 to the Oracle RAC database homes in the *new* standby hosts by using the Standby-First Rolling patch apply technique.

Terminal Sessions

- **Session 1:** `ssh oracle@enode01`
- **Session 2:** `ssh oracle@enode01`

Tasks

1. **(Session1)** Establish a terminal window as the `oracle` user to connect to the first standby host `enode01`. Use `oraenv` to set the Oracle environment.

```
[Classroom PC ~]$ ssh oracle@enode01
oracle@enode01's password: <oracle>
[oracle@enode01 ~]$ . oraenv
ORACLE_SID = [oracle] ? eastdb
The Oracle base has been set to /u01/app/oracle
[oracle@enode01 ~]$ export ORACLE_SID=eastdb1
[oracle@enode01 ~]$
```

2. The patch you will install is 20108098. It is located in `/home/oracle/patch/20108098`.

```
[oracle@enode01 ~]$ cd /home/oracle/patch/20108098
[oracle@enode01 20108098]$ ls -al
total 24
drwxrwxr-x 4 root root 4096 Jan 20 20:05 .
drwxr-xr-x 3 oracle oinstall 4096 Jan 22 18:31 ..
drwxr-xr-x 3 root root 4096 Jan 20 20:05 etc
drwxr-xr-x 3 root root 4096 Jan 20 20:05 files
-rw-rw-r-- 1 root root 5855 Jan 20 20:05 README.txt
[oracle@enode01 20108098]$
```

3. Make sure that your PATH definition points to the `make`, `ar`, `ld`, and `nm` executables. In your practice environment, they are located in `/usr/bin`.

```
[oracle@enode01 20108098]$ which make ar ld nm
/usr/bin/make
/usr/bin/ar
/usr/bin/ld
/usr/bin/nm
[oracle@enode01 20108098]$ echo $PATH
/usr/lib64/qt-
3.3/bin:/usr/local/bin:/bin:/usr/bin:/home/oracle/bin:/u01/app/oracle/product/12.1.0/dbhome_1/bin
[oracle@enode01 ~]$
```

4. Put the OPatch directory in your path and verify that OPatch has access to the Oracle inventory with the `opatch lsinventory` command.

```
[oracle@enode01 ~]$ export
PATH=$PATH:/u01/app/oracle/product/12.1.0/dbhome_1/OPatch
[oracle@enode01 ~]$ opatch lsinventory
```

```
Oracle Interim Patch Installer version 12.1.0.1.3
Copyright (c) 2015, Oracle Corporation. All rights reserved.
```

```
Oracle Home      : /u01/app/oracle/product/12.1.0/dbhome_1
Central Inventory : /u01/app/oraInventory
      from       : /u01/app/oracle/product/12.1.0/dbhome_1/oraInst.loc
OPatch version   : 12.1.0.1.3
OUI version     : 12.1.0.2.0
Log file location :
/u01/app/oracle/product/12.1.0/dbhome_1/cfgtoollogs/patch/patch2015-10-11_21-43-35PM_1.log

Lsinventory Output file location :
/u01/app/oracle/product/12.1.0/dbhome_1/cfgto
ollogs/patch/lsinv/lsinventory2015-10-11_21-43-35PM.txt

-----
Installed Top-level Products (1) :

Oracle Database 12c
12.1.0.2.0
There are 1 products installed in this Oracle Home.

There are no Interim patches installed in this Oracle Home.

Rac system comprising of multiple nodes
  Local node = enode01
  Remote node = enode02

-----
OPatch succeeded.
[oracle@enode01 ~]$
```

5. Determine whether any currently installed interim patches conflict with patch 20108098 by running the `opatch prereq` command as shown in the following:

```
[oracle@enode01 20108098]$ opatch prereq
CheckConflictAgainstOHWithDetail -ph ./
Oracle Interim Patch Installer version 12.1.0.1.3
Copyright (c) 2015, Oracle Corporation. All rights reserved.

PREREQ session

Oracle Home      : /u01/app/oracle/product/12.1.0/dbhome_1
Central Inventory : /u01/app/oraInventory
      from       : /u01/app/oracle/product/12.1.0/dbhome_1/oraInst.loc
OPatch version   : 12.1.0.1.3
OUI version     : 12.1.0.2.0
Log file location :
/u01/app/oracle/product/12.1.0/dbhome_1/cfgtoollogs/patch/patch2015-
10-11_21-44-49PM_1.log

Invoking prereq "checkconflictagainstohwithdetail"
```

```
Prereq "checkConflictAgainstOHWithDetail" passed.

OPatch succeeded.
[oracle@enode01 20108098] $
```

6. **(Session 2)** Open another terminal window as the `oracle` user. Set the environment and use the `DGMGRL` command to determine the apply instance.

Note: You may or may not see Transport Lag caused by the idle time. As long as the physical standby database is synchronized (Apply Lag is zero), you can safely move to the next step.

```
[Classsroom_PC ~] $ ssh oracle@enode01
oracle@enode01's password: <oracle>
[oracle@enode01 ~] $
[oracle@enode01 ~] $ . oraenv
ORACLE_SID = [oracle] ? eastdb
The Oracle base has been set to /u01/app/oracle
[oracle@enode01 ~] $
[oracle@enode01 ~] $ dgmgrl
DGMGRL> connect sysdg/oracle_4U@eastdb
Connected as SYSDG.
DGMGRL> show database eastdb

Database - eastdb

Role: PHYSICAL STANDBY
Intended State: APPLY-ON
Transport Lag: 0 seconds (computed 1 second ago)
Apply Lag: 0 seconds (computed 1 second ago)
Average Apply Rate: 4.00 KByte/s
Real Time Query: OFF
Instance(s):
  eastdb1 (apply instance)
eastdb2

Database Status:
SUCCESS

DGMGRL>
```

7. You will patch the first standby host (`enode01`). If the current apply instance is running on `enode01`, switch the apply instance to the second standby host (`enode02`). Otherwise, skip to step 8.

Note: You may or may not see Transport Lag caused by the idle time. As long as the physical standby database is synchronized (Apply Lag is zero), you can continue.

```
DGMGRL> edit database eastdb set state ='APPLY-ON' with apply
instance = 'eastdb2';
Succeeded.
DGMGRL> show database eastdb

Database - eastdb

Role: PHYSICAL STANDBY
Intended State: APPLY-ON
```

```

Transport Lag:      0 seconds (computed 0 seconds ago)
Apply Lag:        0 seconds (computed 1 second ago)
Average Apply Rate: 188.00 KByte/s
Real Time Query:    OFF
Instance(s):
  eastdb1
eastdb2 (apply instance)

Database Status:
SUCCESS

DGMGRL> exit
[oracle@enode01 ~] $
```

- Shut down the instance running on the local node, enode01.

```
[oracle@enode01 ~] $ srvctl stop instance -db eastdb -instance eastdb1
[oracle@enode01 ~] $
```

- (Session 1) Return to the first terminal and start the patch installation by executing **opatch apply**. When OPatch asks “Is the local system ready for patching?”, answer yes by typing **y** and pressing Enter.

```

[oracle@enode01 20108098] $ opatch apply
Oracle Interim Patch Installer version 12.1.0.1.3
Copyright (c) 2015, Oracle Corporation. All rights reserved.

Oracle Home       : /u01/app/oracle/product/12.1.0/dbhome_1
Central Inventory : /u01/app/oraInventory
                   from          : /u01/app/oracle/product/12.1.0/dbhome_1/oraInst.loc
OPatch version    : 12.1.0.1.3
OUI version      : 12.1.0.2.0
Log file location :
/u01/app/oracle/product/12.1.0/dbhome_1/cfgtoollogs/opatch/20108098_Oct
_11_2015_21_53_17/apply2015-10-11_21-53-17PM_1.log

Applying interim patch '20108098' to OH
'/u01/app/oracle/product/12.1.0/dbhome_1'
Verifying environment and performing prerequisite checks...
All checks passed.

This node is part of an Oracle Real Application Cluster.
Remote nodes: 'enode02'
Local node: 'enode01'
Please shutdown Oracle instances running out of this ORACLE_HOME on the
local system.
(Oracle Home = '/u01/app/oracle/product/12.1.0/dbhome_1')

Is the local system ready for patching? [y|n]
Y
User Responded with: Y
Backing up files...

Patching component oracle.rdbms, 12.1.0.2.0...
```

```
Verifying the update...
```

```
The local system has been patched. You can restart Oracle instances on it.
```

Patching in rolling mode.

The node 'enode02' will be patched next.

```
Please shutdown Oracle instances running out of this ORACLE_HOME on 'enode02'.
```

```
(Oracle Home = '/u01/app/oracle/product/12.1.0/dbhome_1')
```

```
[oracle@enode01 20108098] $
```

When patching is finished on enode01, the OPatch dialog box will inform you that the instance can be restarted on enode01 and will prompt you for the name of the next node to patch, enode02. Before continuing, go to the next step.

10. **(Session 2)** Go to the second terminal window and restart the instance on enode01.

```
[oracle@enode01 ~] $ srvctl start instance -db eastdb -instance eastdb1
[oracle@enode01 ~] $
```

11. From the second terminal window, switch the apply instance to the *patched* standby host (enode01) using the DGMGRL command.

```
[oracle@enode01 ~] $ dgmgrl
DGMGRL> connect sysdg/oracle_4U@eastdb
Connected as SYSDG.
DGMGRL> edit database eastdb set state ='APPLY-ON' with apply
instance = 'eastdb1';
Succeeded.

DGMGRL> exit
[oracle@enode01 ~] $
```

12. Stop the instance on enode02.

```
[oracle@enode01 ~] $ srvctl stop instance -db eastdb -instance eastdb2
[oracle@enode01 ~] $
```

13. **(Session 1)** Return to the first terminal window, type enode02, and press Enter. When asked "Is the node ready for patching? [y|n]", answer Y and press Enter.

```
Is the node ready for patching? [y|n]
```

Y

User Responded with: Y

Updating nodes 'enode02'

Apply-related files are:

FP =

"/u01/app/oracle/product/12.1.0/dbhome_1/.patch_storage/20108098_Jan_20_2015_12_04_56/rac/copy_files.txt"

DP =

"/u01/app/oracle/product/12.1.0/dbhome_1/.patch_storage/20108098_Jan_20_2015_12_04_56/rac/copy_dirs.txt"

MP =

```

"/u01/app/oracle/product/12.1.0/dbhome_1/.patch_storage/20108098_Jan_20
_2015_12_04_56/rac/make_cmds.txt"
    RC =
"/u01/app/oracle/product/12.1.0/dbhome_1/.patch_storage/20108098_Jan_20
_2015_12_04_56/rac/remote_cmds.txt"

Instantiating the file
"/u01/app/oracle/product/12.1.0/dbhome_1/.patch_storage/20108098_Jan_20
_2015_12_04_56/rac/copy_files.txt.instantiated" by replacing
$ORACLE_HOME in
"/u01/app/oracle/product/12.1.0/dbhome_1/.patch_storage/20108098_Jan_20
_2015_12_04_56/rac/copy_files.txt" with actual path.
Propagating files to remote nodes...
Instantiating the file
"/u01/app/oracle/product/12.1.0/dbhome_1/.patch_storage/20108098_Jan_20
_2015_12_04_56/rac/copy_dirs.txt.instantiated" by replacing
$ORACLE_HOME in
"/u01/app/oracle/product/12.1.0/dbhome_1/.patch_storage/20108098_Jan_20
_2015_12_04_56/rac/copy_dirs.txt" with actual path.
Propagating directories to remote nodes...
Instantiating the file
"/u01/app/oracle/product/12.1.0/dbhome_1/.patch_storage/20108098_Jan_20
_2015_12_04_56/rac/make_cmds.txt.instantiated" by replacing
$ORACLE_HOME in
"/u01/app/oracle/product/12.1.0/dbhome_1/.patch_storage/20108098_Jan_20
_2015_12_04_56/rac/make_cmds.txt" with actual path.
Running command on remote node 'enode02':
cd /u01/app/oracle/product/12.1.0/dbhome_1/rdbms/lib; /usr/bin/make -f
ins_rdbms.mk ioracle
ORACLE_HOME=/u01/app/oracle/product/12.1.0/dbhome_1 || echo
REMOTE_MAKE_FAILED:::>2

The node 'enode02' has been patched. You can restart Oracle instances
on it.

There were relinks on remote nodes. Remember to check the binary size
and timestamp on the nodes 'enode02'.
The following make commands were invoked on remote nodes:
'cd /u01/app/oracle/product/12.1.0/dbhome_1/rdbms/lib; /usr/bin/make -f
ins_rdbms.mk ioracle
ORACLE_HOME=/u01/app/oracle/product/12.1.0/dbhome_1
'

Patch 20108098 successfully applied
Log file location:
/u01/app/oracle/product/12.1.0/dbhome_1/cfgtoollogs/patch/20108098_Oct
_11_2015_21_53_17/apply2015-10-11_21-53-17PM_1.log

OPatch succeeded.
[oracle@enode01 20108098] $

```

14. **(Session 2)** After OPatch has finished patching, go to the second terminal window and restart the instance on enode02.

[oracle@enode01 ~]\$ srvctl start instance -db eastdb -instance eastdb2 [oracle@enode01 ~]\$
--

15. **(Session 1)** Return to the first terminal session and run the opatch lsinventory command to ensure that the Oracle homes on two nodes were patched successfully.

```
[oracle@enode01 20108098]$ opatch lsinventory
Oracle Interim Patch Installer version 12.1.0.1.3
Copyright (c) 2015, Oracle Corporation. All rights reserved.

Oracle Home      : /u01/app/oracle/product/12.1.0/dbhome_1
Central Inventory : /u01/app/oraInventory
                  from       : /u01/app/oracle/product/12.1.0/dbhome_1/oraInst.loc
OPatch version   : 12.1.0.1.3
OUI version     : 12.1.0.2.0
Log file location :
/u01/app/oracle/product/12.1.0/dbhome_1/cfgtoollogs/opatch/opatch2015-
10-11_22-22-48PM_1.log

Lsinventory Output file location :
/u01/app/oracle/product/12.1.0/dbhome_1/cfgtoollogs/opatch/lsinv/lsinve-
ntory2015-10-11_22-22-48PM.txt

-----
Installed Top-level Products (1) :

Oracle Database 12c
12.1.0.2.0
There are 1 products installed in this Oracle Home.

Interim patches (1) :

Patch 20108098    : applied on Sun Oct 11 22:05:20 UTC 2015
Unique Patch ID: 18461252
  Created on 20 Jan 2015, 12:04:56 hrs PST8PDT
  Bugs fixed:
  20108098

Rac system comprising of multiple nodes
  Local node = enode01
  Remote node = enode02

-----
OPatch succeeded.
[oracle@enode01 20108098]$
```

16. Close all terminal windows opened for practice. At this stage, all Oracle RAC Homes in the new standby hosts have been patched successfully.

Note: Perform SQL installation on the new primary (that is, old standby) at this time if required in a real-life environment. Changes made to the new primary database during SQL installation will propagate to the new standby via redo.

Practice 10-4: Performing Switchover to the Original Configuration

Overview

In this practice, you will perform switchover to get back to the original Data Guard configuration.

Tasks

- Establish an ssh connection as the oracle user to connect to the first standby host enode01. Use oraenv to set the Oracle environment.

```
[Classroom PC ~]$ ssh oracle@enode01
oracle@enode01's password: <oracle>
[oracle@enode01 ~]$ . oraenv
ORACLE_SID = [oracle] ? eastdb
The Oracle base has been set to /u01/app/oracle
[oracle@enode01 ~]$
```

- Using DGMGRL, perform the switchover operation.

```
[oracle@enode01 ~]$ dgmgrl
DGMGRL> connect sysdg/oracle_4U@eastdb
Connected as SYSDG.
DGMGRL> validate database westdb

Database Role: Primary database

Ready for Switchover: Yes

DGMGRL> validate database eastdb

Database Role: Physical standby database
Primary Database: westdb

Ready for Switchover: Yes
Ready for Failover: Yes (Primary Running)

Transport-Related Property Settings:
Property      westdb Value          eastdb Value
RedoRoutes    (westdb:eastdb SYNC)  (eastdb:westdb SYNC)

DGMGRL> switchover to eastdb
Performing switchover NOW, please wait...
New primary database "eastdb" is opening...
Oracle Clusterware is restarting database "westdb" ...
Switchover succeeded, new primary is "eastdb"
DGMGRL>
```

- Verify the current database role in the Data Guard environment.

Note: You may or may not see warning messages, such as ORA-16525, ORA-16665, ORA-16810, or ORA-16817 due to the lag and overall performance of the virtual machine environment. The warning messages will be cleared in a few minutes.

```
DGMGRL> show configuration

Configuration - dg_config

Protection Mode: MaxPerformance
```

```
Members:  
eastdb - Primary database  
westdb - Physical standby database  
  
Fast-Start Failover: DISABLED  
  
Configuration Status:  
SUCCESS (status updated 26 seconds ago)  
  
DGMGLR>
```

4. Wait until the physical standby database is synchronized. It may take a few moments for the Lag for the standby database to clear.

Note: Do not continue with the practices until the Apply Lag is cleared.

```
DGMGRL> show database westdb  
  
Database - westdb  
  
Role: PHYSICAL STANDBY  
Intended State: APPLY-ON  
Transport Lag: 0 seconds  
Apply Lag: 0 seconds  
Real Time Query: ON  
Instance(s):  
    westdb1 (apply instance)  
    westdb2  
  
Database Status:  
SUCCESS  
  
DGMGLR> exit  
[oracle@enode01 ~] $
```

5. Close all terminal windows opened for practice.

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

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

Practices for Lesson 11: Disassociating a Snapshot Standby Database from a Data Guard Configuration

Chapter 11

Practices for Lesson 11: Overview

Practice Overview

In this practice, you convert a Snapshot Standby Database into a New Independent Database by using the DBNEWID (NID) utility.

Practice Summary

- Practice 11-1a: Converting a Snapshot Standby Database into a New Independent Database by Using the DBNEWID (NID) utility (12.1.0.2)
- Practice 11-1b: Converting a Snapshot Standby Database into a New Independent Database by Using the DBNEWID (NID) utility (Optional, 12.1.0.1)
- Practice 11-2: Reconfiguring the Environment for Rolling Upgrade

Practice Guidelines

You can complete the practice activities in the following way and order.

- **(Optional)** If you have enough time, you can perform any interesting troubleshooting cases in the appendix before starting practice 11.
 - Practice 13-4: Re-creating Data Guard Broker Configuration (Case 4)
 - Practice 13-5: Resynchronizing a Physical Standby Database using RMAN Incremental Backup (Case 5)
 - Practice 13-6: Recovering the Primary Database Using Archival Backup (Case 6)
 - Practice 13-7: Re-creating the Physical Standby Database (Case 7)
- Depending on when you want to complete practice 11-1, you can choose either practice 11-1a or 11-1b.
 - Practice 11-1a: It can be done in the current practice environment.
 - Practice 11-1b: It is the same as practice 11-1a. It can be done later after practice 12-5 is complete.
- Practice 11-2 is required for practice 12.

Practice 11-1a: Converting a Snapshot Standby Database into a New Independent Database by Using the DBNEWID (NID) utility

Overview

In this practice, you will disassociate the existing snapshot standby database from the Data Guard environment as a new independent database. Note that this practice does not include the additional steps that can be performed in the remaining Data Guard configuration.

Tasks

1. (Session 1) Establish a terminal session connected to enode01 as the oracle user and configure the terminal environment as shown in the following:

```
[Classroom PC] $ ssh oracle@enode01
oracle@enode01's password: <oracle>
[oracle@enode01 ~]$ . oraenv
ORACLE_SID = [oracle] ? eastdb
The Oracle base has been set to /u01/app/oracle
[oracle@enode01 ~]$ export ORACLE_SID=eastdb1
[oracle@enode01 ~]$
```

2. Using DGMGRL, verify the current Data Guard configuration.

```
[oracle@enode01 ~]$ dgmgrl
DGMGRL> connect sysdg/oracle_4U@westdb
Connected as SYSDG.
DGMGRL> show configuration

Configuration - dg_config

Protection Mode: MaxPerformance
Members
Databases:
eastdb - Primary database
    westdb - Physical standby database

Fast-Start Failover: DISABLED

Configuration Status:
SUCCESS

DGMGRL>
```

3. In the same terminal, attempt to convert the physical standby database into a snapshot standby database.

```
DGMGRL> convert database westdb to snapshot standby;
Converting database "westdb" to a Snapshot Standby database, please
wait...
Error: ORA-16692: operation disallowed for a database or far sync
instance that sends redo data

Failed.
Failed to convert database "westdb"
DGMGRL> exit
```

4. Review the ORA-16692 error message.

```
[oracle@enode01 ~]$ oerr ora 16692
16692, 0000, "operation disallowed for a database or far sync instance
that sends redo data"
// *Cause: An attempt was made to delete, disable, or convert a //
database that sends redo data to another database or far sync //
instance.
// *Action: Reset the value of the RedoRoutes property to the
// empty string of the database or far sync instance to be
// deleted, disabled, or converted (physical standby database).
[oracle@enode01 ~]$
```

5. Reset the value of the RedoRoutes property to empty string.

```
[oracle@enode01 ~]$ dgmgrl
DGMGRL> connect sysdg/oracle_4U@eastdb
Connected as SYSDG.
DGMGRL> show database westdb RedoRoutes
RedoRoutes = '(westdb:eastdb ASYNC)'
DGMGRL> show database eastdb RedoRoutes
RedoRoutes = '(eastdb:westdb ASYNC)'
DGMGRL> edit database westdb set property RedoRoutes = '';
Property "redoroutes" updated
DGMGRL> edit database eastdb set property RedoRoutes = '';
Property "redoroutes" updated
DGMGRL> show database westdb RedoRoutes
RedoRoutes = ''
DGMGRL> show database eastdb RedoRoutes
RedoRoutes = ''
DGMGRL>
```

6. Convert a physical standby database into a snapshot standby database.

```
DGMGRL> convert database westdb to snapshot standby;
Converting database "westdb" to a Snapshot Standby database, please
wait...
Database "westdb" converted successfully
DGMGRL>
```

7. Verify the new role of the westdb database.

```
DGMGRL> show configuration

Configuration - dg_config

Protection Mode: MaxPerformance
Databases:
eastdb - Primary database
westdb - Snapshot standby database

Fast-Start Failover: DISABLED

Configuration Status:
SUCCESS

DGMGRL>
```

8. **(Session 2)** Open a new terminal window, connected to wnode03 as the oracle user, and configure the terminal environment.

```
[Classroom PC]$ ssh oracle@wnode03
oracle@wnode03's password: <oracle>
[oracle@wnode03 ~]$ . oraenv
ORACLE_SID = [oracle] ? westdb
The Oracle base has been set to /u01/app/oracle
[oracle@wnode03 ~]$ export ORACLE_SID=westdb1
[oracle@wnode03 ~]$
```

9. Using SQL*Plus, view the control file type of the snapshot standby database.

```
[oracle@wnode03 ~]$ sqlplus / as sysdba

SQL> SELECT database_role, controlfile_type FROM v$database;
DATABASE_ROLE      CONTROL
-----
SNAPSHOT STANDBY CURRENT

SQL>
```

10. The guaranteed restore point has been created implicitly to convert back to the physical standby database. This must be dropped to make the snapshot standby database as a new independent database.

Note: Because the guaranteed restore point is used internally by the Data Guard, it will be dropped automatically later.

```
SQL> col name format a70
SQL> select name, guarantee_flashback_database from v$restore_point;

NAME                               GUA
-----
SNAPSHOT_STANDBY_REQUIRED_10/30/2015 08:45:20      YES

SQL> drop restore point "SNAPSHOT_STANDBY_REQUIRED_10/30/2015
08:45:20";
drop restore point "SNAPSHOT_STANDBY_REQUIRED_10/30/2015 08:45:20"
*
ERROR at line 1:
ORA-38799: Cannot drop guaranteed restore point internally created for
snapshot standby

SQL> exit
[oracle@wnode03 ~]$
```

11. **(Session 1)** Return to the first terminal window to remove the snapshot standby database from the Data Guard broker configuration, and exit the DGMGRL prompt.

```
DGMGRL> remove database westdb;
Removed database "westdb" from the configuration

DGMGRL> show configuration

Configuration - dg_config

Protection Mode: MaxPerformance
Members:
```

```

eastdb - Primary database

Fast-Start Failover: DISABLED

Configuration Status:
SUCCESS      (status updated 11 seconds ago)

DGMGRL> exit
[oracle@enode01 ~] $

```

12. **(Session 2)** Return to the second terminal. You will use the DBNEWID utility to change DBID and DBNAME to make the snapshot standby database as a new independent database.

Note: To use the DBNEWID utility for RAC, you must start only one instance in the MOUNT state and set CLUSTER_DATABASE to FALSE.

```

[oracle@wnode03 ~]$ srvctl stop database -db westdb
[oracle@wnode03 ~]$ srvctl start instance -db westdb -instance westdb1
-startoption mount
[oracle@wnode03 ~]$ sqlplus / as sysdba
SQL> alter system set cluster_database=false scope=spfile;
System altered.

SQL> shutdown abort
ORACLE instance shut down.

SQL> startup mount
ORACLE instance started.

Total System Global Area 1503199232 bytes
Fixed Size                  2288584 bytes
Variable Size                469763128 bytes
Database Buffers            1023410176 bytes
Redo Buffers                 7737344 bytes
Database mounted.
SQL> exit
[oracle@wnode03 ~] $

```

13. Change DBID and DBNAME of the snapshot standby database by using the DBNEWID utility. Enter y to continue.

```

[oracle@wnode03 ~]$ nid target=sys dbname=westdb

DBNEWID: Release 12.1.0.2.0 - Production on Tue Sep 8 15:34:57 2015

Copyright (c) 1982, 2013, Oracle and/or its affiliates. All rights reserved.

Password: <oracle_4U>

Connected to database EASTDB (DBID=900774776)

Connected to server version 12.1.0

Control Files in database:
+DATA/westdb/controlfile/current.257.856487161

```

```

Change database ID and database name EASTDB to WESTDB? (Y/[N]) => Y

Proceeding with operation
Changing database ID from 900774776 to 1633252133
Changing database name from EASTDB to WESTDB
  Control File +DATA/westdb/controlfile/current.257.856487161 -
modified
  Datafile +DATA/westdb/datafile/system.258.85648720 - dbid changed,
wrote new name
  Datafile +DATA/westdb/datafile/sysaux.259.85648720 - dbid changed,
wrote new name
  Datafile +DATA/westdb/datafile/undotbs1.261.85648720 - dbid
changed, wrote new name
  Datafile +DATA/westdb/datafile/example.260.85648720 - dbid changed,
wrote new name
  Datafile +DATA/westdb/datafile/undotbs2.262.85648740 - dbid
changed, wrote new name
  Datafile +DATA/westdb/datafile/users.263.85648743 - dbid changed,
wrote new name
  Datafile +DATA/westdb/tempfile/temp.277.86083701 - dbid changed,
wrote new name
  Control File +DATA/westdb/controlfile/current.257.856487161 - dbid
changed, wrote new name
  Instance shut down

Database name changed to WESTDB.
Modify parameter file and generate a new password file before
restarting.
Database ID for database WESTDB changed to 1633252133.
All previous backups and archived redo logs for this database are
unusable.
Database is not aware of previous backups and archived logs in Recovery
Area.
Database has been shutdown, open database with RESETLOGS option.
Successfully changed database name and ID.
DBNEWID - Completed successfully.

[oracle@wnode03 ~] $

```

14. Create pfile from spfile for the new database.

```

[oracle@wnode03 dbs]$ sqlplus / as sysdba

SQL*Plus: Release 12.1.0.2.0 Production on Tue Sep 8 15:46:52 2015

Copyright (c) 1982, 2013, Oracle. All rights reserved.

Connected to an idle instance.

SQL> create pfile='/tmp/init.ora' from
spfile='+DATA/westdb/spfilewestdb.ora';

File created.

SQL> exit
[oracle@wnode03 ~] $

```

15. Modify pfile to remove the Data Guard parameters.

Note: You may have to perform the additional tasks in your remaining Data Guard environment.

```
[oracle@wnode03 ~]$ vi /tmp/init.ora
westdb2.__data_transfer_cache_size=0
westdb1.__data_transfer_cache_size=0
westdb2.__db_cache_size=1040187392
westdb1.__db_cache_size=1023410176
westdb2.__java_pool_size=16777216
westdb1.__java_pool_size=16777216
westdb2.__large_pool_size=16777216
westdb1.__large_pool_size=16777216
westdb1.__oracle_base='/u01/app/oracle'#ORACLE_BASE set from
environment
westdb2.__oracle_base='/u01/app/oracle'#ORACLE_BASE set from
environment
westdb2.__pga_aggregate_target=503316480
westdb1.__pga_aggregate_target=503316480
westdb2.__sga_target=1509949440
westdb1.__sga_target=1509949440
westdb2.__shared_io_pool_size=0
westdb1.__shared_io_pool_size=0
westdb2.__shared_pool_size=419430400
westdb1.__shared_pool_size=436207616
westdb2.__streams_pool_size=0
westdb1.__streams_pool_size=0
*.archive_lag_target=0
*.audit_file_dest='/u01/app/oracle/admin/westdb/adump'
*.audit_trail='NONE'
*.cluster_database=TRUE
*.compatible='12.1.0.1.0'
*.control_files='+DATA/westdb/controlfile/current.257.856487161'
*.db_block_size=8192
*.db_create_file_dest='+DATA'
*.db_create_online_log_dest_1='+DATA'
*.db_domain='example.com'
*.db_file_multiblock_read_count=8
*.db_name='westdb'
*.db_recovery_file_dest='+FRA'
*.db_recovery_file_dest_size=9437184000
*.db_unique_name='westdb'
##.dg_broker_config_file1='+DATA/westdb/dr1.dat'
##.dg_broker_config_file2='+DATA/westdb/dr2.dat'
##.dg_broker_start=TRUE
*.diagnostic_dest='/u01/app/oracle'
*.event=''
##.fal_server='eastdb'
westdb1.instance_number=1
westdb2.instance_number=2
*.log_archive_config='nodg_config'
*.log_archive_dest_1='location=USE_DB_RECOVERY_FILE_DEST', 'valid_for=(A
LL_LOGFILES, ALL_ROLES)'
##.log_archive_dest_2=''
##.log_archive_dest_state_2='ENABLE'
*.log_archive_format='%t_%s_%r.dbf'
westdb2.log_archive_format='%t_%s_%r.dbf'
westdb1.log_archive_format='%t %s %r.dbf'
```

```
*.log_archive_max_processes=4
*.log_archive_min_succeed_dest=1
*.log_archive_trace=0
westdb2.log_archive_trace=0
westdb1.log_archive_trace=0
*.open_cursors=300
*.pga_aggregate_target=503316480
*.processes=300
*.remote_listener='cluster02-scan.example.com:1521'
*.remote_login_passwordfile='EXCLUSIVE'
*.sga_target=1509949440
##.standby_file_management='AUTO'
westdb1.thread=1
westdb2.thread=2
westdb1.undo_tablespace='UNDOTBS1'
westdb2.undo_tablespace='UNDOTBS2'
:wq!
[oracle@wnode03 ~]$
```

16. Create spfile from the modified pfile.

```
[oracle@wnode03 dbs]$ sqlplus / as sysdba

SQL*Plus: Release 12.1.0.2.0 Production on Tue Sep 8 15:59:48 2015

Copyright (c) 1982, 2013, Oracle. All rights reserved.

Connected to an idle instance.

SQL> create spfile='+DATA/westdb/spfilewestdb.ora' from
pfile='/tmp/init.ora';

File created.

SQL> exit
[oracle@wnode03 ~]$
```

17. Create a shared password file for the new RAC database.

```
[oracle@wnode03 ~]$ orapwd file=/tmp/orapwwestdb password=oracle_4U
force=y
[oracle@wnode03 ~]$ su - grid
Password: <oracle>
[grid@wnode03 ~]$ . oraenv
ORACLE_SID = [grid] ? +ASM1
The Oracle base has been set to /u01/app/grid
[grid@wnode03 ~]$ 
[grid@wnode03 ~]$ asmcmd pwcOPY /tmp/orapwwestdb
+data/westdb/orapwwestdb
copying /tmp/orapwwestdb -> +data/westdb/orapwwestdb
[grid@wnode03 ~]$ exit
[oracle@wnode03 ~]$
```

18. Start the database with RESETLOGS.

```
[oracle@wnode03 ~]$ sqlplus / as sysdba
SQL> startup mount;
ORACLE instance started.
```

```
Total System Global Area 1503199232 bytes
Fixed Size                2288584 bytes
Variable Size             469763128 bytes
Database Buffers          1023410176 bytes
Redo Buffers              7737344 bytes
Database mounted.
SQL> alter database open resetlogs;
Database altered.
SQL>
```

19. Change the global database name for the new database.

```
SQL> alter database rename global_name to westdb.example.com;
Database altered.

SQL>
```

20. Shut down and restart the database.

```
SQL> shutdown immediate
Database closed.
Database dismounted.
ORACLE instance shut down.
SQL> exit
[oracle@wnode03 ~]$
```

21. Update the clusterware entry to match with the DB_NAME, role, and password file location.

```
[oracle@wnode03 ~]$ srvctl modify database -db westdb -dbname westdb -
role PRIMARY -pfile +data/westdb/orapwwestdb
[oracle@wnode03 ~]$
```

22. Display the database configuration.

```
[oracle@wnode03 ~]$ srvctl config database -db westdb
Database unique name: westdb
Database name: westdb
Oracle home: /u01/app/oracle/product/12.1.0/dbhome_1
Oracle user: oracle
Spfile: +DATA/westdb/spfilewestdb.ora
Password file: +data/westdb/orapwwestdb
Domain: example.com
Start options: open
Stop options: immediate
Database role: PRIMARY
Management policy: AUTOMATIC
Server pools:
Disk Groups: DATA, FRA
Mount point paths:
Services: actest
Type: RAC
Start concurrency:
Stop concurrency:
OSDBA group: dba
OSOPER group: oper
Database instances: westdb1,westdb2
Configured nodes: wnode03,wnode04
Database is administrator managed
[oracle@wnode03 ~]$
```

23. Using `srvctl`, restart the database.

```
[oracle@wnode03 ~]$ srvctl start database -db westdb  
[oracle@wnode03 ~]$
```

24. Verify the new database role of the `westdb` database.

```
[oracle@wnode03 ~]$ sqlplus / as sysdba  
SQL> SELECT database_role, controlfile_type FROM v$database;  
DATABASE_ROLE      CONTROLFILE_TYPE  
-----  
PRIMARY           CURRENT  
[oracle@wnode03 ~]$
```

25. Close all of your terminal sessions.

Practice 11-1b: Converting a Snapshot Standby Database into a New Independent Database by Using the DBNEWID (NID) Utility (Optional)

Overview

In this practice, you will disassociate the existing snapshot standby database from the Data Guard environment as a new independent database. Note that this practice does not include the additional steps that can be performed in the remaining Data Guard configuration.

Tasks

1. **(Session 1)** Establish a terminal session connected to c01n01 as the oracle user and configure the terminal environment as shown in the following:

```
[Classroom PC] $ ssh oracle@c01n01
oracle@c01n01's password: <oracle>
[oracle@c01n01 ~]$ . oraenv
ORACLE_SID = [oracle] ? c01orcl
The Oracle base has been set to /u01/app/oracle
[oracle@c01n01 ~]$ export ORACLE_SID=c01orcl1
[oracle@c01n01 ~]$
```

2. Using DGMGRL, verify the current Data Guard configuration.

```
[oracle@c01n01 ~]$ dgmgrl
DGMGRL> connect sys/oracle_4U
Connected as SYSDG.
DGMGRL> show configuration

Configuration - dg_config

Protection Mode: MaxPerformance
Members
Databases:
c01orcl - Primary database
c02orcl - Physical standby database

Fast-Start Failover: DISABLED

Configuration Status:
SUCCESS

DGMGRL>
```

3. Convert a physical standby database into a snapshot standby database.

```
DGMGRL> convert database c02orcl to snapshot standby;
Converting database "c02orcl" to a Snapshot Standby database, please
wait...
Database "c02orcl" converted successfully
DGMGRL>
```

4. Verify the new role of the c02orcl database.

```
DGMGRL> show configuration

Configuration - dg_config

Protection Mode: MaxPerformance
```

```
Databases:
c01orcl - Primary database
c02orcl - Snapshot standby database

Fast-Start Failover: DISABLED

Configuration Status:
SUCCESS

DGMGRL>
```

5. **(Session 2)** Open a new terminal window, connected to c02n01 as the oracle user, and configure the terminal environment.

```
[Classroom PC]$ ssh oracle@c02n01
oracle@c02n01's password: <oracle>
[oracle@c02n01 ~]$ . oraenv
ORACLE_SID = [oracle] ? c02orcl
The Oracle base has been set to /u01/app/oracle
[oracle@c02n01 ~]$ export ORACLE_SID=c02orcl1
[oracle@c02n01 ~]$
```

6. Using SQL*Plus, view the control file type of the snapshot standby database.

```
[oracle@c02n01 ~]$ sqlplus / as sysdba

SQL> SELECT database_role, controlfile_type FROM v$database;
DATABASE_ROLE      CONTROL
-----
SNAPSHOT STANDBY CURRENT

SQL>
```

7. The guaranteed restore point has been created implicitly to convert back to the physical standby database. This must be dropped to make the snapshot standby database as a new independent database.

Note: Because the guaranteed restore point is used internally by Data Guard, it will be dropped automatically later.

```
SQL> col name format a70
SQL> select name, guarantee_flashback_database from v$restore_point;

NAME                               GUA
-----
SNAPSHOT_STANDBY_REQUIRED_10/30/2015 08:45:20      YES

SQL> drop restore point "SNAPSHOT_STANDBY_REQUIRED_10/30/2015
08:45:20";
drop restore point "SNAPSHOT_STANDBY_REQUIRED_10/30/2015 08:45:20"
*
ERROR at line 1:
ORA-38799: Cannot drop guaranteed restore point internally created for
snapshot standby

SQL> exit
[oracle@c02n01 ~]$
```

8. **(Session 1)** Return to the first terminal window to remove the snapshot standby database from the Data Guard broker configuration. Then exit the DGMGRL prompt.

```
DGMGRL> remove database c02orcl;
Removed database "c02orcl" from the configuration

DGMGRL> show configuration

Configuration - dg_config

  Protection Mode: MaxPerformance
  Members:
    c01orcl - Primary database

  Fast-Start Failover: DISABLED

  Configuration Status:
    SUCCESS      (status updated 11 seconds ago)

DGMGRL> exit
[oracle@c01n01 ~]$
```

9. **(Session 2)** Return to the second terminal. You will use the DBNEWID utility to change DBID and DBNAME to make the snapshot standby database as a new independent database.

Note: To use the DBNEWID utility for RAC, you must start only one instance in the MOUNT state and set CLUSTER_DATABASE to FALSE.

```
[oracle@c02n01 ~]$ srvctl stop database -db c02orcl
[oracle@c02n01 ~]$ srvctl start instance -db c02orcl -instance c02orcl1
-o mount
[oracle@c02n01 ~]$ sqlplus / as sysdba
SQL> alter system set cluster_database=false scope=spfile;
System altered.

SQL> shutdown abort
ORACLE instance shut down.

SQL> startup mount
ORACLE instance started.

Total System Global Area 1503199232 bytes
Fixed Size                  2288584 bytes
Variable Size                469763128 bytes
Database Buffers            1023410176 bytes
Redo Buffers                 7737344 bytes
Database mounted.
SQL> exit
[oracle@c02n01 ~]$
```

10. Change DBID and DBNAME of the snapshot standby database by using the DBNEWID utility. Enter y to continue.

```
[oracle@c02n01 ~]$ nid target=sys dbname=c02orcl

DBNEWID: Release 12.1.0.1.0 - Production on Tue Sep 8 15:34:57 2015

Copyright (c) 1982, 2013, Oracle and/or its affiliates. All rights
```

```

reserved.

Password: <oracle_4U>

Connected to database C01ORCL (DBID=900774776)

Connected to server version 12.1.0

Control Files in database:
  +DATA/c02orcl/controlfile/current.257.856487161

Change database ID and database name C01ORCL to C02ORCL? (Y/[N]) => Y

Proceeding with operation
Changing database ID from 900774776 to 1633252133
Changing database name from C01ORCL to C02ORCL
  Control File +DATA/c02orcl/controlfile/current.257.856487161 -
modified
  Datafile +DATA/c02orcl/datafile/system.258.85648720 - dbid changed,
wrote new name
  Datafile +DATA/c02orcl/datafile/sysaux.259.85648720 - dbid changed,
wrote new name
  Datafile +DATA/c02orcl/datafile/undotbs1.261.85648720 - dbid
changed, wrote new name
  Datafile +DATA/c02orcl/datafile/example.260.85648720 - dbid
changed, wrote new name
  Datafile +DATA/c02orcl/datafile/undotbs2.262.85648740 - dbid
changed, wrote new name
  Datafile +DATA/c02orcl/datafile/users.263.85648743 - dbid changed,
wrote new name
  Datafile +DATA/c02orcl/tempfile/temp.277.86083701 - dbid changed,
wrote new name
  Control File +DATA/c02orcl/controlfile/current.257.856487161 - dbid
changed, wrote new name
  Instance shut down

Database name changed to C02ORCL.
Modify parameter file and generate a new password file before
restarting.
Database ID for database C02ORCL changed to 1633252133.
All previous backups and archived redo logs for this database are
unusable.
Database is not aware of previous backups and archived logs in Recovery
Area.
Database has been shutdown, open database with RESETLOGS option.
Successfully changed database name and ID.
DBNEWID - Completed successfully.

[oracle@c02n01 ~]$
```

11. Create pfile from spfile for the new database.

```

[oracle@c02n01 dbs]$ sqlplus / as sysdba

SQL*Plus: Release 12.1.0.1.0 Production on Tue Sep 8 15:46:52 2015

Copyright (c) 1982, 2013, Oracle. All rights reserved.
```

```
Connected to an idle instance.

SQL> create pfile='/tmp/init.ora' from
spfile='+DATA/c02orcl/spfilec02orcl.ora';

File created.

SQL> exit
[oracle@c02n01 ~]$
```

12. Modify pfile to remove the Data Guard parameters.

Note: You may have to perform the additional tasks in your remaining Data Guard environment.

```
[oracle@c02n01 ~]$ vi /tmp/init.ora
c02orcl2.__data_transfer_cache_size=0
c02orcl1.__data_transfer_cache_size=0
c02orcl2.__db_cache_size=1040187392
c02orcl1.__db_cache_size=1023410176
c02orcl2.__java_pool_size=16777216
c02orcl1.__java_pool_size=16777216
c02orcl2.__large_pool_size=16777216
c02orcl1.__large_pool_size=16777216
c02orcl1.__oracle_base='/u01/app/oracle'#ORACLE_BASE set from
environment
c02orcl2.__oracle_base='/u01/app/oracle'#ORACLE_BASE set from
environment
c02orcl2.__pga_aggregate_target=503316480
c02orcl1.__pga_aggregate_target=503316480
c02orcl2.__sga_target=1509949440
c02orcl1.__sga_target=1509949440
c02orcl2.__shared_io_pool_size=0
c02orcl1.__shared_io_pool_size=0
c02orcl2.__shared_pool_size=419430400
c02orcl1.__shared_pool_size=436207616
c02orcl2.__streams_pool_size=0
c02orcl1.__streams_pool_size=0
*.archive_lag_target=0
*.audit_file_dest='/u01/app/oracle/admin/c02orcl/adump'
*.audit_trail='NONE'
*.cluster_database=TRUE
*.compatible='12.1.0.1.0'
*.control_files='+DATA/c02orcl/controlfile/current.257.856487161'
*.db_block_size=8192
*.db_create_file_dest='+DATA'
*.db_create_online_log_dest_1='+DATA'
*.db_domain='example.com'
*.db_file_multiblock_read_count=8
*.db_name='c02orcl'
*.db_recovery_file_dest='+FRA'
*.db_recovery_file_dest_size=9437184000
*.db_unique_name='c02orcl'
##*.dg_broker_config_file1='+DATA/c02orcl/dr1.dat'
##*.dg_broker_config_file2='+DATA/c02orcl/dr2.dat'
##*.dg_broker_start=TRUE
*.diagnostic_dest='/u01/app/oracle'
*.event=''
##*.fal_server='c01orcl'
```

```
c02orcl1.instance_number=1
c02orcl2.instance_number=2
*.log_archive_config='nodg_config'
*.log_archive_dest_1='location=USE_DB_RECOVERY_FILE_DEST', 'valid_for=(A
LL_LOGFILES,ALL_ROLES)'
#*.log_archive_dest_2=''
#*.log_archive_dest_state_2='ENABLE'
*.log_archive_format='%t_%s_%r.dbf'
c02orcl2.log_archive_format='%t_%s_%r.dbf'
c02orcl1.log_archive_format='%t_%s_%r.dbf'
*.log_archive_max_processes=4
*.log_archive_min_succeed_dest=1
*.log_archive_trace=0
c02orcl2.log_archive_trace=0
c02orcl1.log_archive_trace=0
*.open_cursors=300
*.pga_aggregate_target=503316480
*.processes=300
*.remote_listener='cluster02-scan.example.com:1521'
*.remote_login_passwordfile='EXCLUSIVE'
*.sga_target=1509949440
#*.standby_file_management='AUTO'
c02orcl1.thread=1
c02orcl2.thread=2
c02orcl1.undo_tablespace='UNDOTBS1'
c02orcl2.undo_tablespace='UNDOTBS2'
:wq!
[oracle@c02n01 ~]$
```

13. Create spfile from the modified pfile.

```
[oracle@c02n01 dbs]$ sqlplus / as sysdba

SQL*Plus: Release 12.1.0.1.0 Production on Tue Sep 8 15:59:48 2015

Copyright (c) 1982, 2013, Oracle. All rights reserved.

Connected to an idle instance.

SQL> create spfile='+DATA/c02orcl/spfilec02orcl.ora' from
pfile='/tmp/init.ora';

File created.

SQL> exit
[oracle@c02n01 ~]$
```

14. Attempt to create a shared password file for the new RAC database.

```
[oracle@c02n01 ~]$ orapwd file=/tmp/orapwc02orcl password=oracle_4U
force=y
[oracle@c02n01 ~]$ su - grid
Password: <oracle>
[grid@c02n01 ~]$ . oraenv
ORACLE_SID = [grid] ? +ASM1
The Oracle base has been set to /u01/app/grid
[grid@c02n01 ~]$
[grid@c02n01 ~]$ asmcmd pwcOPY /tmp/orapwc02orcl
```

```
+data/c02orcl/orapwc02orcl
copying /tmp/orapwc02orcl -> +data/c02orcl/orapwc02orcl
ASMCMD-8016: copy source '/tmp/orapwc02orcl' and target
'+data/c02orcl/orapwc02orcl' failed
ORA-19505: failed to identify file "+DATA/c02orcl/orapwc02orcl"
ORA-17502: ksfdcre:4 Failed to create file +data/c02orcl/orapwc02orcl
ORA-15056: additional error message
ORA-15221: ASM operation requires compatible.asm of 12.1.0.0.0 or
higher
ORA-06512: at line 4
ORA-06512: at "SYS.X$DBMS_DISKGROUP", line 467
ORA-06512: at line 3 (DBD ERROR: OCIStmtExecute)
[grid@c02n01 ~]$
```

15. Display attributes of the +DATA disk group.

```
[grid@c02n01 ~]$ asmcmd lsattr -l -G DATA
Name           Value
access_control.enabled FALSE
access_control.umask 066
au_size        1048576
cell.smart_scan_capable FALSE
compatible.asm 11.2.0.0.0
compatible.rdbms 10.1.0.0.0
disk_repair_time 3.6h
sector_size    512
[grid@c02n01 ~]$
```

16. Set the compatible.asm and compatible.rdbms to 12.1.0.0.0 to create the password file in the +DATA disk group.

```
[grid@c02n01 ~]$ asmcmd setattr -G DATA compatible.asm 12.1.0.0.0
[grid@c02n01 ~]$ asmcmd setattr -G DATA compatible.rdbms 12.1.0.0.0
[grid@c02n01 ~]$ asmcmd pwcopy /tmp/orapwc02orcl
+data/c02orcl/orapwc02orcl
copying /tmp/orapwc02orcl -> +data/c02orcl/orapwc02orcl
[grid@c02n01 ~]$ exit
[oracle@c02n01 ~]$
```

17. Start the database with RESETLOGS.

```
[oracle@c02n01 ~]$ sqlplus / as sysdba
SQL> startup mount;
ORACLE instance started.

Total System Global Area 1503199232 bytes
Fixed Size                  2288584 bytes
Variable Size                469763128 bytes
Database Buffers             1023410176 bytes
Redo Buffers                 7737344 bytes
Database mounted.
SQL> alter database open resetlogs;

Database altered.

SQL>
```

18. Change the global database name for the new database.

```
SQL> alter database rename global_name to c02orcl.example.com;  
Database altered.  
SQL>
```

19. Shut down and restart the database.

```
SQL> shutdown immediate  
Database closed.  
Database dismounted.  
ORACLE instance shut down.  
SQL> exit  
[oracle@c02n01 ~]$
```

20. Update the clusterware entry to match with the DB_NAME, role, and password file location.

```
[oracle@c02n01 ~]$ srvctl modify database -db c02orcl -dbname c02orcl -  
role PRIMARY -pwfile +data/c02orcl/orapwc02orcl  
[oracle@c02n01 ~]$
```

21. Display the database configuration.

```
[oracle@c02n01 ~]$ srvctl config database -db c02orcl  
Database unique name: c02orcl  
Database name: c02orcl  
Oracle home: /u01/app/oracle/product/12.1.0/dbhome_1  
Oracle user: oracle  
Spfile: +DATA/c02orcl/spfilec02orcl.ora  
Password file: +data/c02orcl/orapwc02orcl  
Domain: example.com  
Start options: open  
Stop options: immediate  
Database role: PRIMARY  
Management policy: AUTOMATIC  
Server pools:  
Disk Groups: DATA, FRA  
Mount point paths:  
Services: actest  
Type: RAC  
Start concurrency:  
Stop concurrency:  
OSDBA group: dba  
OSOPER group: oper  
Database instances: c02orcl1,c02orcl2  
Configured nodes: c02n01,c02n02  
Database is administrator managed  
[oracle@c02n01 ~]$
```

22. Using srvctl, restart the database.

```
[oracle@c02n01 ~]$ srvctl start database -db c02orcl  
[oracle@c02n01 ~]$
```

23. Verify the new database role of the c02orcl database.

```
[oracle@c02n01 ~]$ sqlplus / as sysdba
SQL> SELECT database_role, controlfile_type FROM v$database;
DATABASE_ROLE      CONTROLFILE_TYPE
-----          -----
PRIMARY           CURRENT
[oracle@c02n01 ~]$
```

24. Close all of your terminal sessions.

Practice 11-2: Reconfiguring the Environment for Rolling Upgrade

Overview

In this practice, you will reconfigure your practice environment in preparation for the next practice.

Estimated Time: 130 min (**Step 1:** 100min, **Step 2~4:** 30min)

Tasks

- Establish a terminal session on your practice environment and execute the following command to perform a series of reconfiguration steps in preparation for the next set of practices. Ensure that you allow the reconfiguration process to complete and check to make sure that the VM status output includes entries for c01n01, c01n02, c02n01, and c02n02.

```
[Classroom PC] $ sudo /ovs/seed_pool/setup_Upgrade.sh
Removing current VMs...
Extracting Upgrade VMs...
Starting VMs...
VM status...
Name ID Mem VCPUs State
Time(s)
Domain-0 0 1024 8 r----- 12228.3
c01n01 1 3500 1 -b---- 20.1
c01n02 2 3500 1 -b---- 6.3
c02n01 3 3500 1 -b---- 5.1
c02n02 4 3500 1 -b---- 4.5

[Classroom PC] $
```

Note: Even though VMs are up and running, you may have to wait for 15-20 minutes until the clusterware, database, Data Guard broker, and other components are ready.

- Determine the status of the Data Guard configuration.

Note: You may or may not see warning messages, such as ORA-00604, ORA-01219, ORA-12514, ORA-16525, ORA-16665, ORA-16810, ORA-16782, or ORA-16817, due to the lag and overall performance of the virtual machine environment. The warning messages will be cleared in a few minutes. Make sure that your Data Guard configuration status is fine.

```
[Classroom PC ~]$ ssh oracle@c01n01
oracle@c01n01's password: <oracle>
[oracle@c01n01 ~]$ . oraenv
ORACLE_SID = [oracle]? c01orcl
The Oracle base has been set to /u01/app/oracle
[oracle@c01n01 ~]$ export ORACLE_SID=c01orcl1
[oracle@c01n01 ~]$ dgmgrl
DGMGRL> connect sys/oracle_4U
Connected.
DGMGRL> show configuration

Configuration - dg_config
```

```
Protection Mode: MaxPerformance
Members:
c01orcl - Primary database
c02orcl - Physical standby database

Fast-Start Failover: DISABLED

Configuration Status:
SUCCESS      (status updated 17 seconds ago)
DGMGRL>
```

3. Wait until the physical standby database is synchronized. It may take a few moments for the Lag for the standby database to clear.

Note: Do not continue with the practices until the Apply Lag is cleared.

```
DGMGRL> show database c02orcl
Database - c02orcl

Role:          PHYSICAL STANDBY
Intended State: APPLY-ON
Transport Lag: 0 seconds (computed 1 second ago)
Apply Lag: 0 seconds (computed 1 second ago)
Average Apply Rate: 40.00 KByte/s
Real Time Query: ON
Instance(s):
  c02orcl1 (apply instance)
  c02orcl2

Database Status:
SUCCESS

DGMGRL>
```

4. Close all of your terminal sessions.

Practices for Lesson 12: Rolling Database Upgrade Using Transient Logical Standby

Chapter 12

Practices for Lesson 12: Overview

Practice Overview

In this practice, you will upgrade a two-node Oracle RAC database and an Oracle Data Guard physical standby database running on another two-node cluster. The initial Oracle Database software version on both clusters is 11.2.0.4, and the target software version for the upgraded environment is 12.1.0.1.

The type of upgrade you will perform is known as a rolling database upgrade using a transient logical standby database. The aim of this upgrade method is to maximize database availability during the upgrade process. To achieve this aim, the process requires many operations and role changes to be performed in a precise order.

To simplify the process and reduce the likelihood of administrator error, Oracle has created a script-based utility, which is available for download from My Oracle Support. In this practice, you will use the script to automate various parts of the process and provide advice about what to do when manual intervention is required. The script will also perform many checks along the way to maximize the probability of success.

Practice Summary

- Practice 12-1: Laboratory Introduction
- Practice 12-2: Grid Infrastructure Upgrade
- Practice 12-3: Database Preparation for Rolling Upgrade
- Practice 12-4: Database Rolling Upgrade
- Practice 12-5: Post-Upgrade Steps

Practice 12-1: Laboratory Introduction

Overview

In the first part of this practice, you will be introduced to the technical environment that you will use to perform a rolling upgrade.

Terminal Sessions

- **Session 1:** `ssh oracle@c01n01`
- **Session 2:** `ssh oracle@c02n01`

Tasks

1. **(Session 1)** Establish a terminal session connected to `c01n01` as the `oracle` user and configure the terminal environment as shown in the following:

```
[Classroom PC]$ ssh oracle@c01n01
oracle@c01n01's password: <oracle>
[oracle@c01n01 ~]$ . oraenv
ORACLE_SID = [oracle] ? c01orcl
The Oracle base has been set to /u01/app/oracle
[oracle@c01n01 ~]$ export ORACLE_SID=c01orcl1
[oracle@c01n01 ~]$
```

2. `c01n01` is part of a two-node cluster named `cluster01`. The other server in `cluster01` is `c01n02`. The cluster contains an Oracle RAC database named `c01orcl`. Take a moment to explore the configuration and status of `c01orcl`.

```
[oracle@c01n01 ~]$ srvctl config database -d c01orcl -v
Database unique name: c01orcl
Database name: c01orcl
Oracle home: /u01/app/oracle/product/11.2.0/dbhome_1
Oracle user: oracle
Spfile: +DATA/c01orcl/spfilec01orcl.ora
Domain: example.com
Start options: open
Stop options: immediate
Database role: PRIMARY
Management policy: AUTOMATIC
Server pools: c01orcl
Database instances: c01orcl1,c01orcl2
Disk Groups: DATA,FRA
Mount point paths:
Services:
Type: RAC
Database is administrator managed
[oracle@c01n01 ~]$ srvctl status database -d c01orcl -v
Instance c01orcl1 is running on node c01n01. Instance status:
Open.
Instance c01orcl2 is running on node c01n02. Instance status:
Open.
[oracle@c01n01 ~]$
```

3. **(Session 2)** Leave your other terminal session open and establish another terminal session connected to c02n01 as the oracle user. Configure the terminal environment as shown in the following:

```
[Classroom PC]$ ssh oracle@c02n01
oracle@c02n01's password: <oracle>
[oracle@c02n01 ~]$ . oraenv
ORACLE_SID = [oracle] ? c02orcl
The Oracle base has been set to /u01/app/oracle
[oracle@c02n01 ~]$ export ORACLE_SID=c02orcl
[oracle@c02n01 ~]$
```

4. c02n01 is part of a two-node cluster named cluster02. The other server in cluster02 is c02n02. The cluster contains an Oracle RAC database named c02orcl. Take a few minutes to explore the configuration and status of c02orcl.

```
[oracle@c02n01 ~]$ srvctl config database -d c02orcl -v
Database unique name: c02orcl
Database name:
Oracle home: /u01/app/oracle/product/11.2.0/dbhome_1
Oracle user: oracle
Spfile:
Domain: example.com
Start options: open
Stop options: immediate
Database role: PHYSICAL_STANDBY
Management policy: AUTOMATIC
Server pools: c02orcl
Database instances: c02orcl1,c02orcl2
Disk Groups: DATA
Mount point paths:
Services:
Type: RAC
Database is administrator managed
[oracle@c02n01 ~]$ srvctl status database -d c02orcl -v
Instance c02orcl1 is running on node c02n01. Instance status:
Open.
Instance c02orcl2 is running on node c02n02. Instance status:
Open.
[oracle@c02n01 ~]$
```

Note: From your investigation of c01orcl and c02orcl, you may have seen clues indicating that both databases are part of an Oracle Data Guard configuration. In this configuration, c01orcl is nominally the primary database and c02orcl is a physical standby database; however, these roles will switch back and forth during different phases of the rolling upgrade process. Oracle Data Guard Broker is also enabled.

5. **(Session 1)** Return to the terminal session on c01n01. Then use the Data Guard Broker management utility (dgmgrl) to explore the configuration and current status of the Oracle Data Guard configuration.

```
[oracle@c01n01 ~]$ dgmgrl sys/oracle_4U

DGMGRL> show configuration

Configuration - dg_config
```

```
Protection Mode: MaxPerformance
Databases:
  c01orcl - Primary database
  c02orcl - Physical standby database

Fast-Start Failover: DISABLED

Configuration Status:
SUCCESS

DGMGRL> show database c01orcl

Database - c01orcl

Role:           PRIMARY
Intended State: TRANSPORT-ON
Instance(s):
  c01orcl1
  c01orcl2

Database Status:
SUCCESS

DGMGRL> show database c02orcl

Database - c02orcl

Role:           PHYSICAL STANDBY
Intended State: APPLY-ON
Transport Lag:  0 seconds (computed 0 seconds ago)
Apply Lag:     0 seconds (computed 1 second ago)
Apply Rate:    349.00 KByte/s
Real Time Query: ON
Instance(s):
  c02orcl1 (apply instance)
  c02orcl2

Database Status:
SUCCESS

DGMGRL> exit
[oracle@c01n01 ~]$
```

6. Close all of your terminal sessions.

Practice 12-2: Grid Infrastructure Upgrade

Overview

In any installation, the Oracle Database software version cannot be newer than the Oracle Grid Infrastructure software that supports it. Because of this, the Oracle Grid Infrastructure software must be upgraded before the database upgrade can be performed. In the next part of this practice, you will upgrade the Grid Infrastructure installations across your practice environment. In line with best practice recommendations, you will upgrade each cluster while it is configured as the standby cluster. This minimizes the down time for the primary cluster and the databases it supports.

Here is an outline of the tasks you will perform:

- Upgrade Grid Infrastructure on `cluster02` while it is the standby cluster.
- Perform a Data Guard switchover making `cluster02` the new primary and `cluster01` the standby.
- Upgrade Grid Infrastructure on `cluster01` while it is the standby cluster.
- Perform another switchover to return to the original Data Guard configuration. Note that this switchover is optional, which means that it is not required to complete the upgrade process. However, you will perform the switchover so that `cluster01` can remain the primary cluster and `cluster02` can remain the standby cluster as much as possible.

Terminal Sessions

- **Session 1:** `ssh -X grid@c02n01`
- **Session 2:** `ssh oracle@c01n01`
- **Session 3:** `ssh -X grid@c01n01`

Tasks

1. **(Session 1)** Open a terminal window on `c02n01` to connect as the `grid` user. Configure the terminal environment as shown in the following.

Note: Make sure that you use `ssh` with the `-X` option. The `-X` option is required to ensure that your terminal environment is correctly configured to run GUI applications and utilities.

```
[Classroom PC]$ ssh -X grid@c02n01
grid@c02n01's password: <oracle>
[grid@c02n01 ~]$ . oraenv
ORACLE_SID = [grid] ? +ASM1
The Oracle base has been set to /u01/app/grid
[grid@c02n01 ~]$
```

2. Confirm the current active version of the Oracle Grid Infrastructure software on `cluster02`.

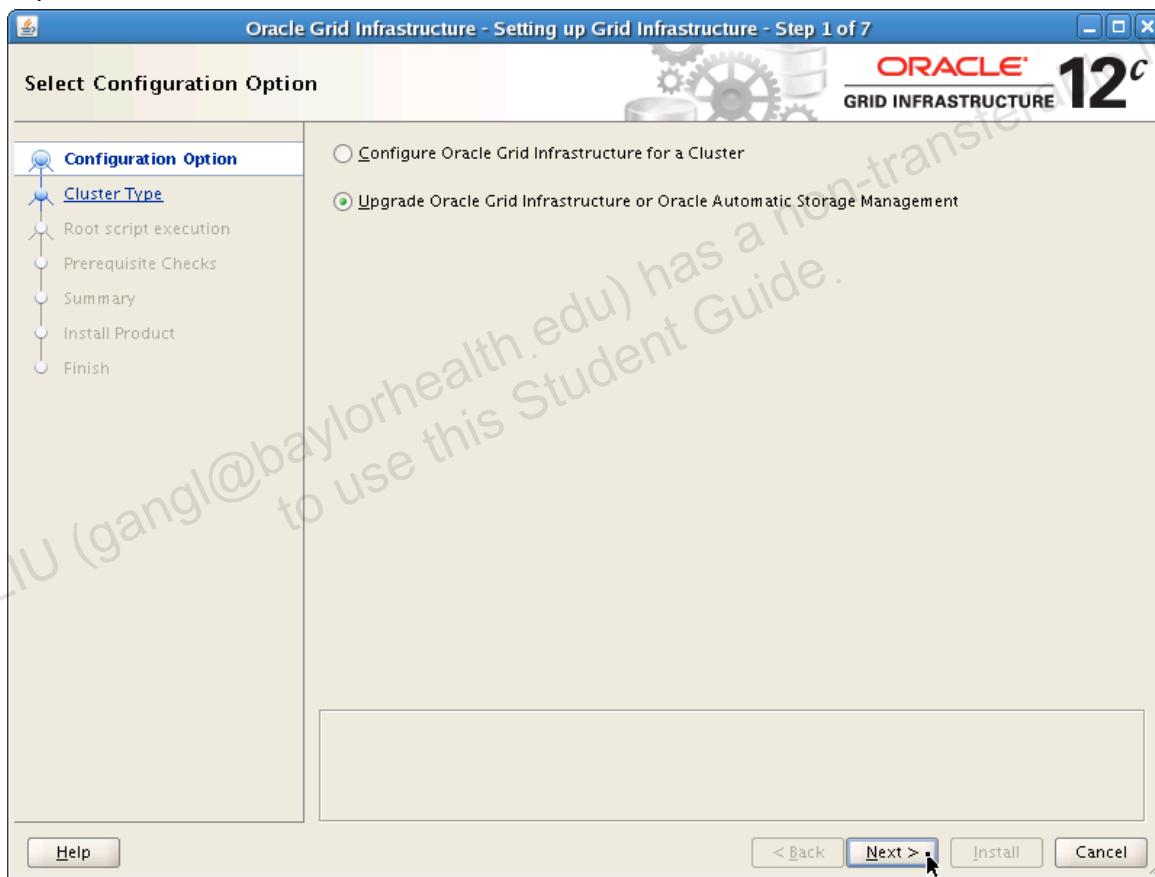
```
[grid@c02n01 ~]$ crsctl query crs activeversion
Oracle Clusterware active version on the cluster is [11.2.0.4.0]
[grid@c02n01 ~]$
```

3. Launch the Oracle Grid Infrastructure 12c configuration utility.

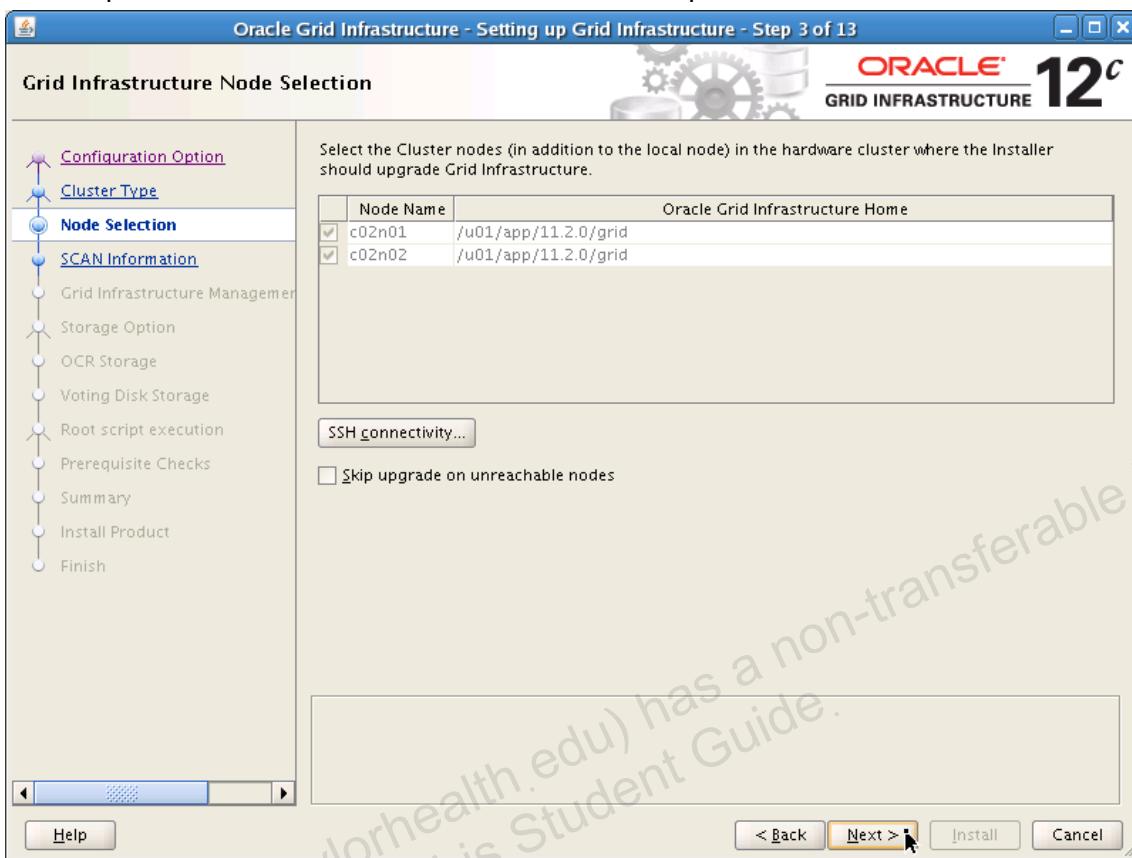
```
[grid@c02n01 ~]$ /u01/app/12.1.0/grid/crs/config/config.sh &
```

Note: In this practice, you will use the Oracle Grid Infrastructure configuration utility to perform the Grid Infrastructure upgrade on each cluster. The configuration utility is launched from a preinstalled copy of the Oracle Grid Infrastructure 12c binaries. Your practice environment was built this way to speed up the upgrade process. Alternatively, the Oracle Universal Installer can be used to install the upgraded software and upgrade the cluster in one operation. Note that if you use Oracle Universal Installer, you will see essentially the same configuration screens as the ones provided by the Oracle Grid Infrastructure configuration utility.

4. On the Select Configuration Option page, ensure that the “Upgrade Oracle Grid Infrastructure or Oracle Automatic Storage Management” option is selected and click Next to proceed.

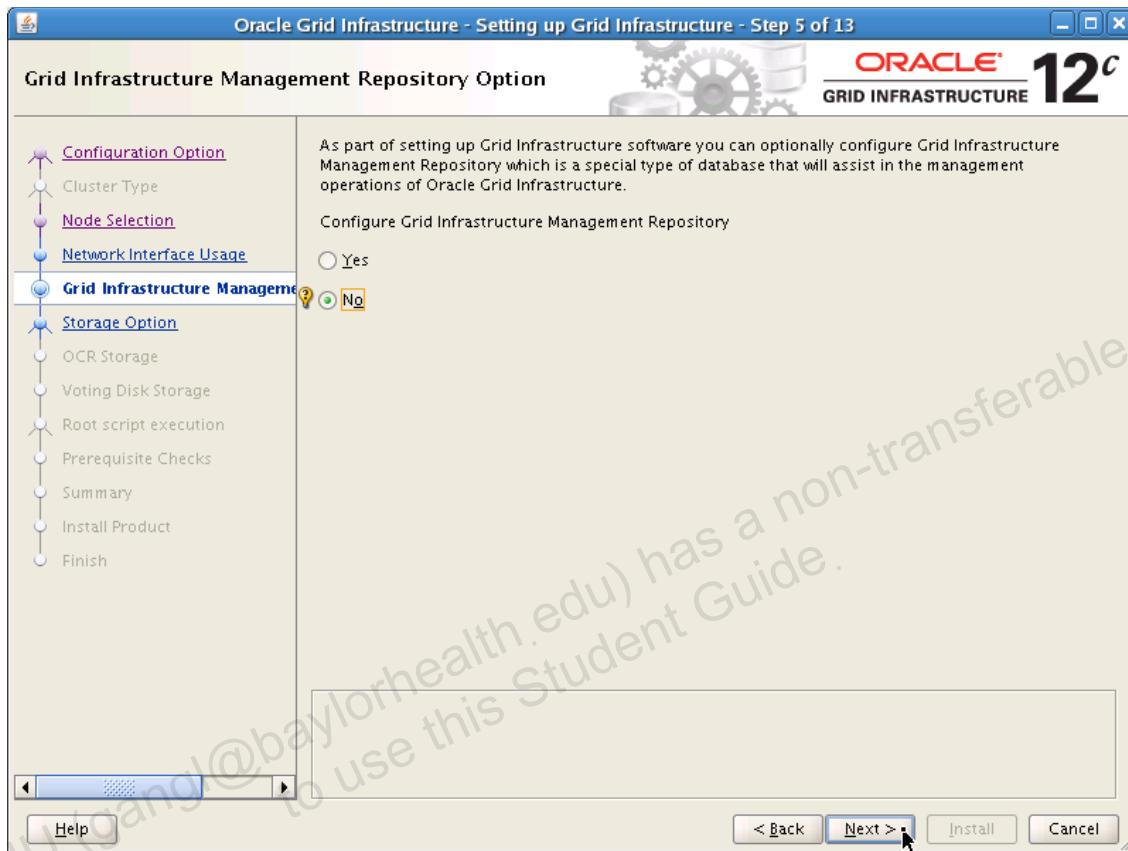


5. On the Grid Infrastructure Node Selection page, ensure that the details on the page match the composition of the cluster, and then click Next to proceed.



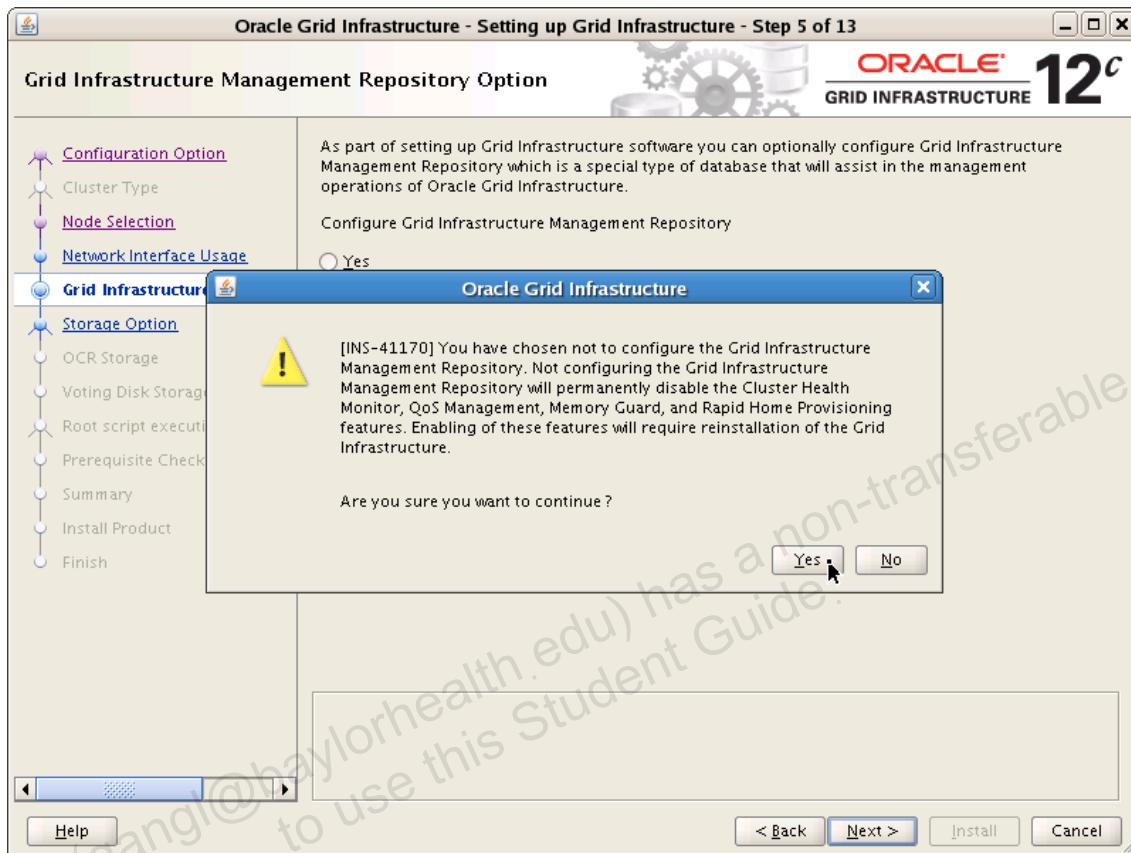
6. On the Grid Infrastructure Management Repository Option page, select No to not configure the Grid Infrastructure Management Repository and click Next to proceed.

Note: It is recommended that you do not configure the Grid Infrastructure Management Repository in your practice environment because it is not used in this practice; it takes a significant amount of time to create the repository database and it consumes a significant amount of machine resources (relative to the modest capacity of the practice environment).

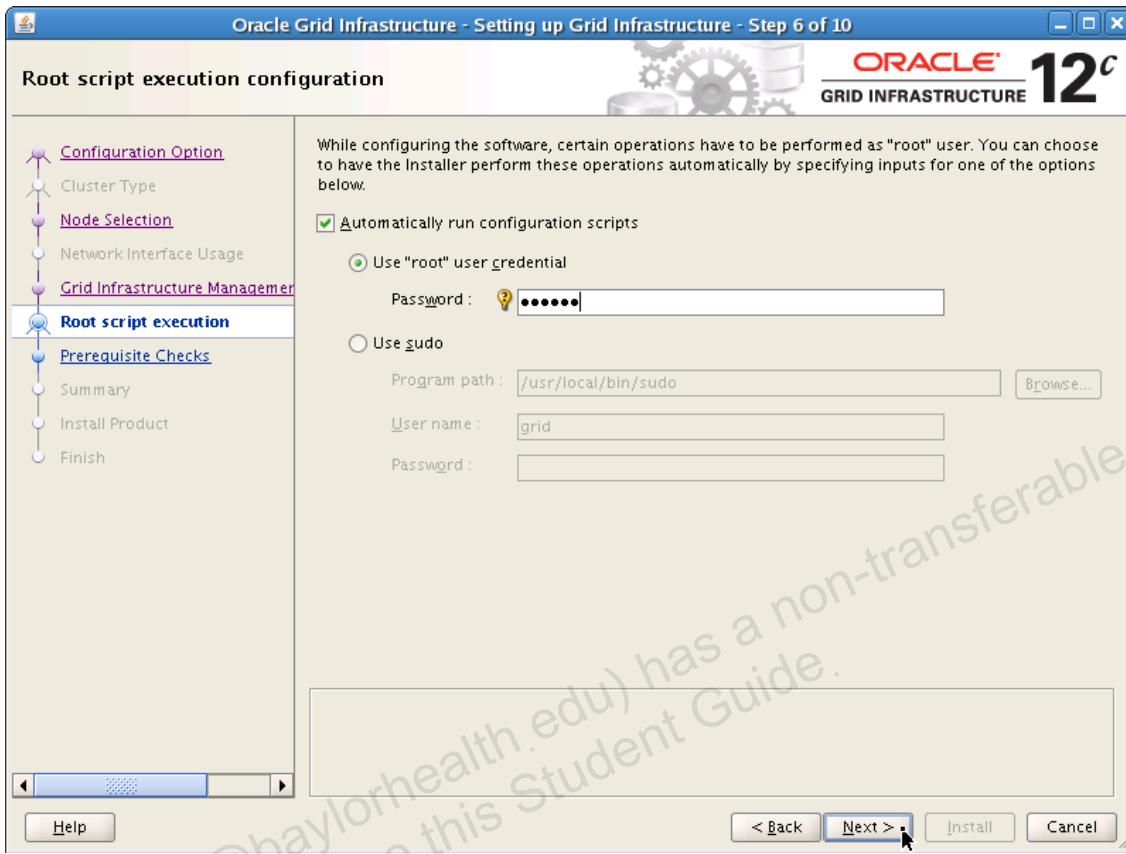


7. Click Yes to acknowledge the additional warning about the Grid Infrastructure Management Repository.

Note: Because the Grid Infrastructure Management Repository cannot be added without reinstalling Oracle Grid Infrastructure, it is generally recommended that you should configure the repository.



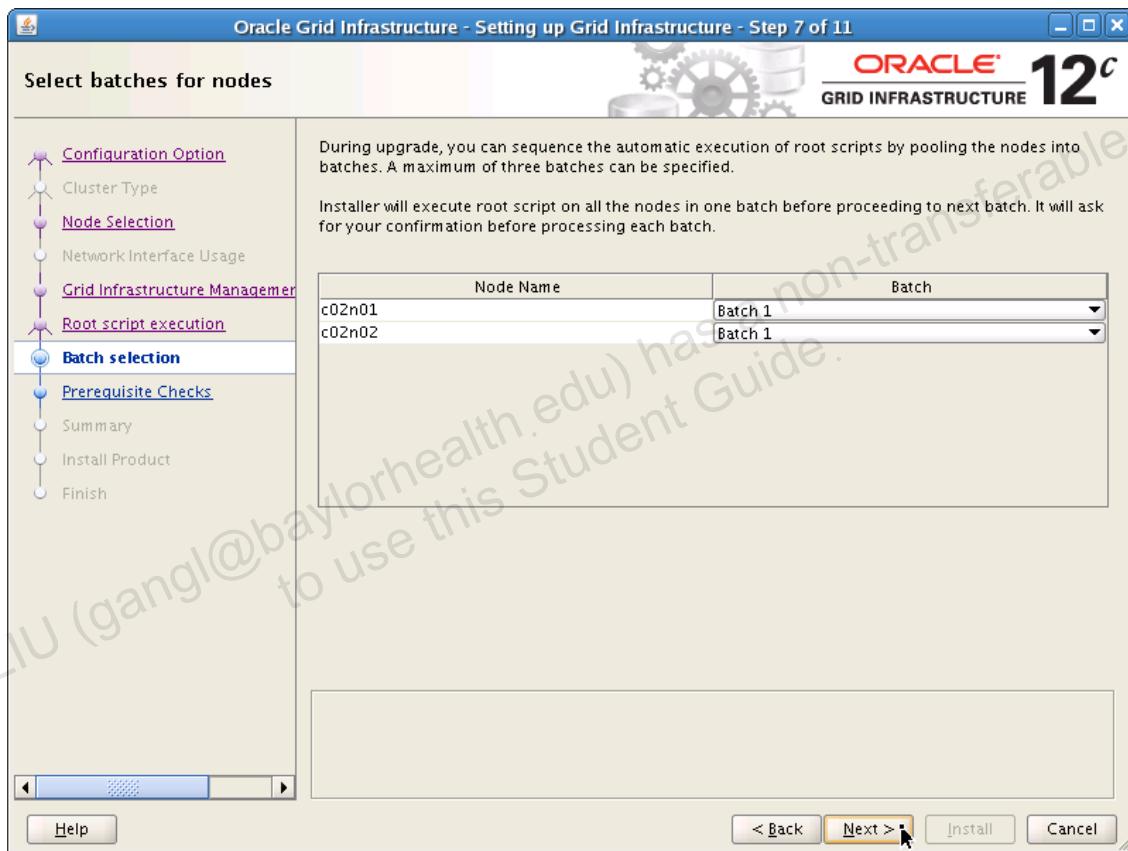
- On the “Root script execution configuration” page, select the option to automatically run the configuration scripts by using the `root` user credential. Enter `oracle` as the `root` user password and click Next to proceed.



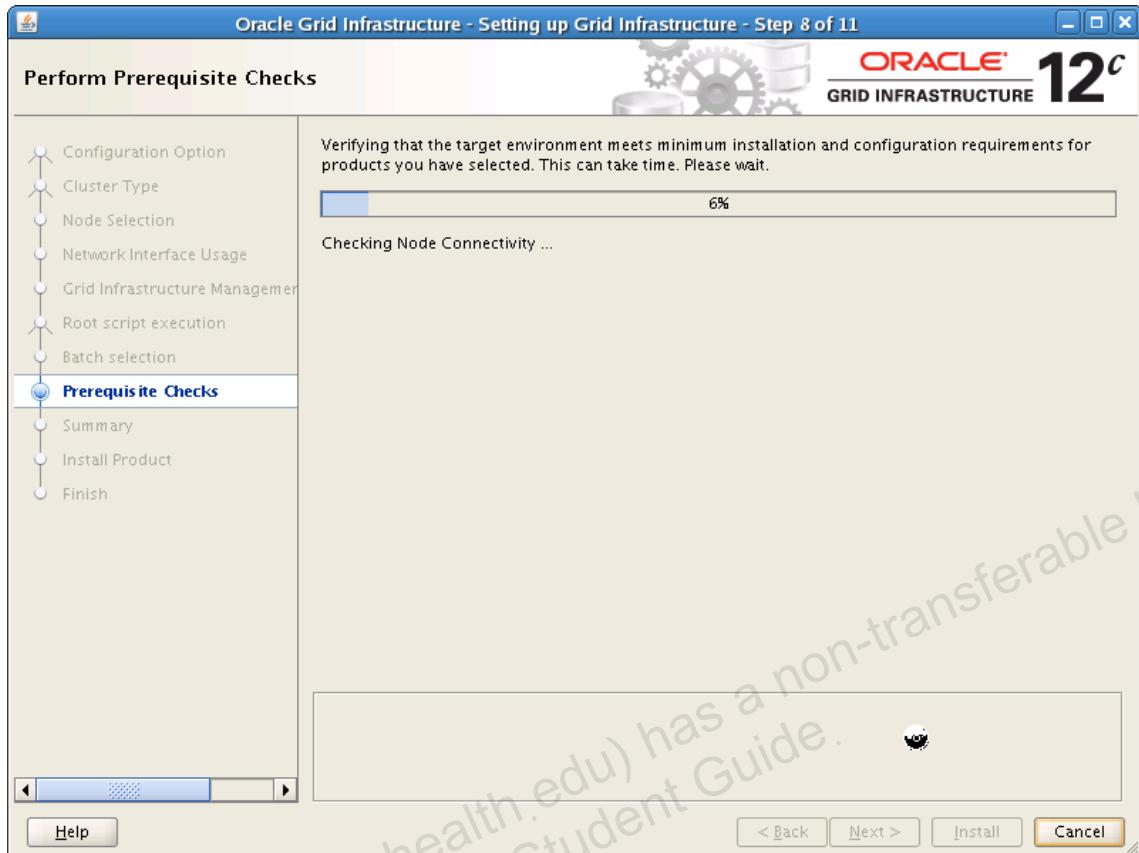
9. On the “Select batches for nodes” page, leave the default configuration and click Next to proceed.

Note: The upgrade process provides an option to configure groups of nodes in batches. This facility enables administrators to move services around the cluster during the upgrade to maintain the availability of vital services throughout the upgrade process. For example, you could define two batches and ensure that your databases continue to run on one batch while the other batch is being upgraded. Then, after the first batch is upgraded, you could move your database to the upgraded batch of servers while the remaining servers are upgraded.

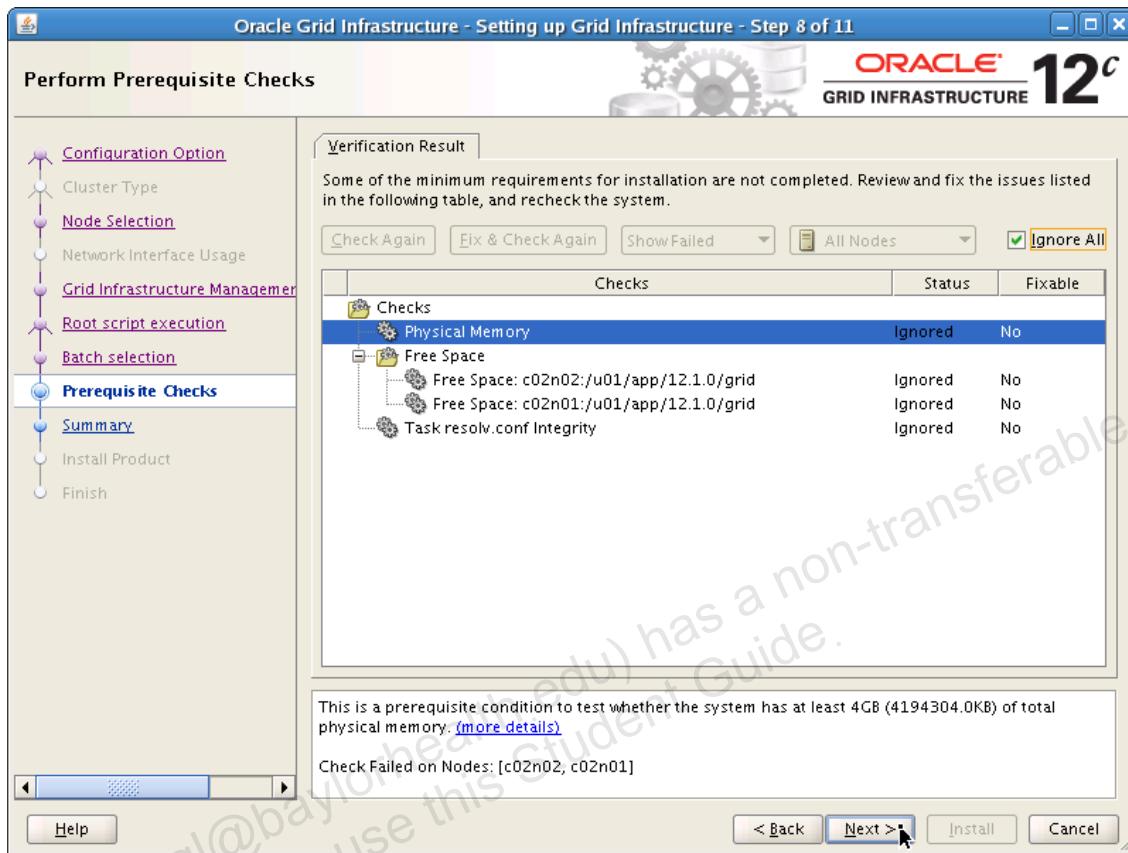
You will not use this feature during this practice; however, take note of its existence because you may find it useful for managing more complex upgrades involving larger clusters or clusters that support multiple databases.



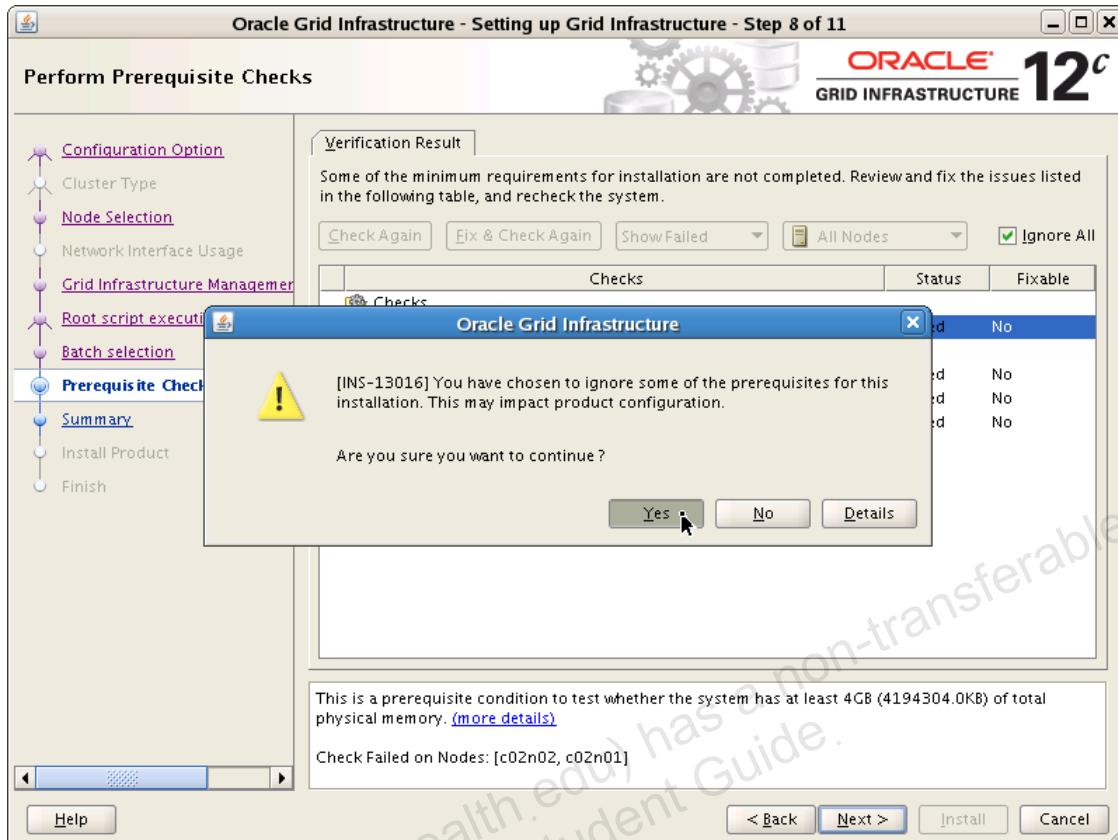
10. Wait while the configuration utility performs a series of checks.



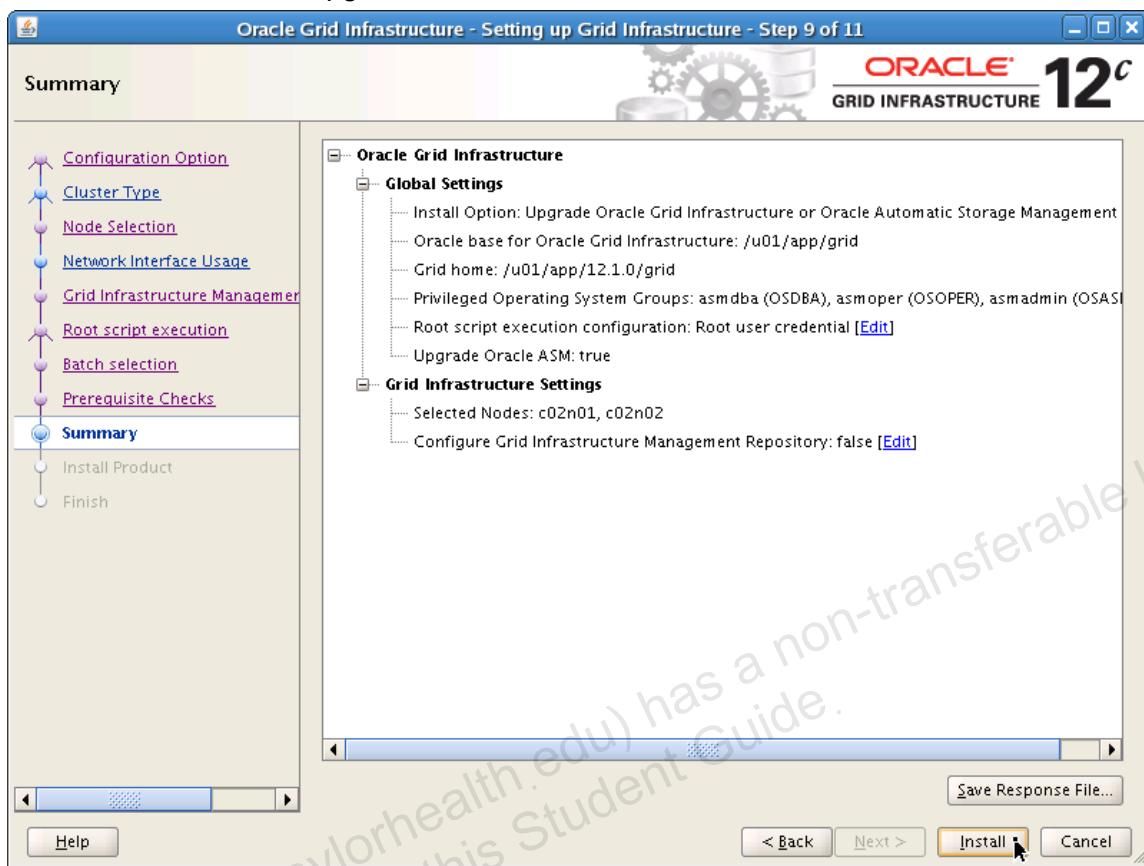
11. Because of constraints associated with your practice environment, some of the prerequisite checks will fail. However, none of these are critical issues and you can ignore the warnings associated with them. To proceed, select the option to ignore all the warnings and click Next.



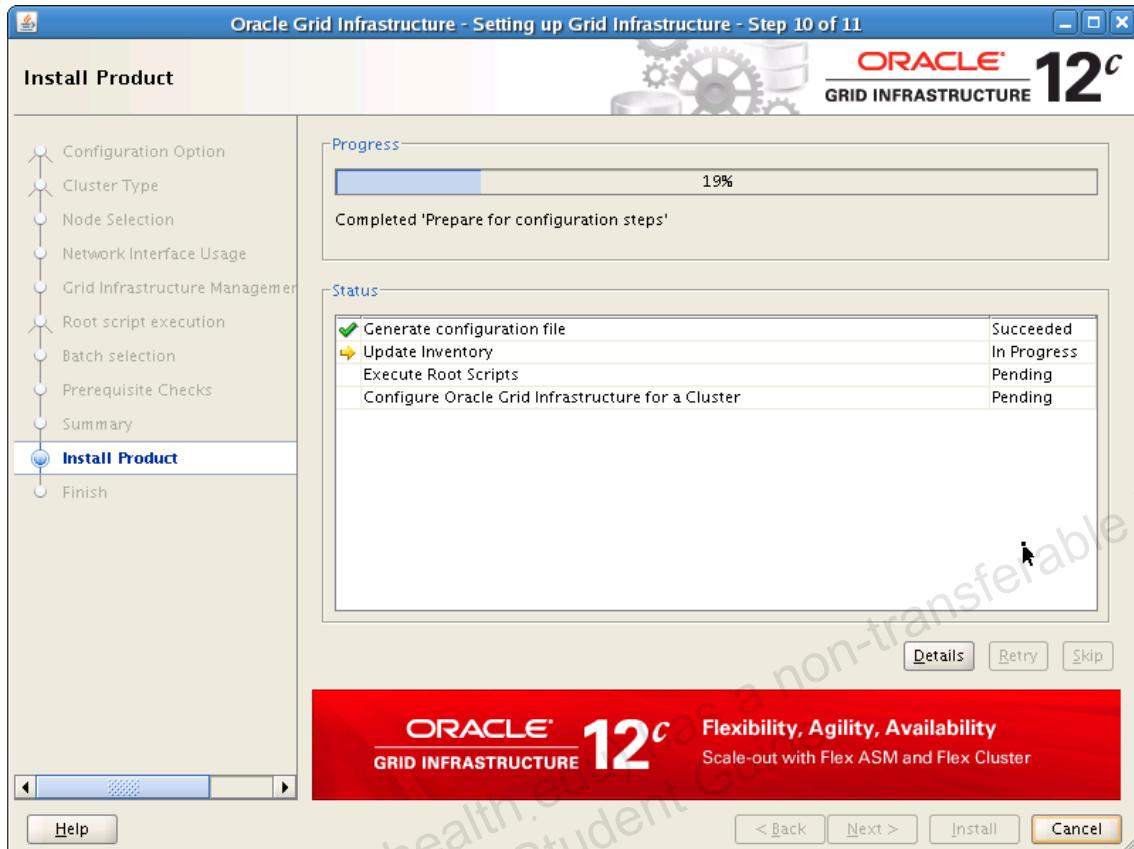
12. Click Yes to acknowledge the additional warning about the installation prerequisites.



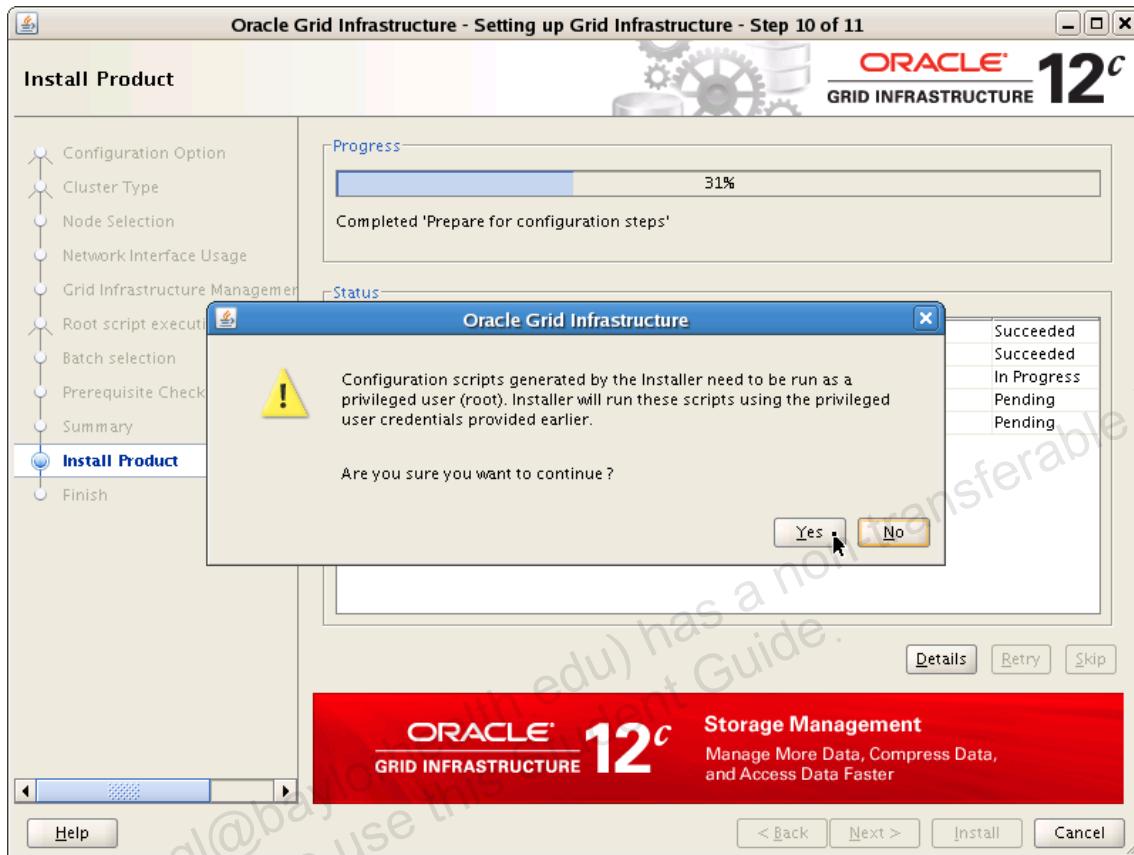
13. On the Summary page, confirm the configuration options that are displayed. When you are satisfied, click Install to upgrade Grid Infrastructure on cluster02.



14. You can monitor the configuration process by using the Install Product page.

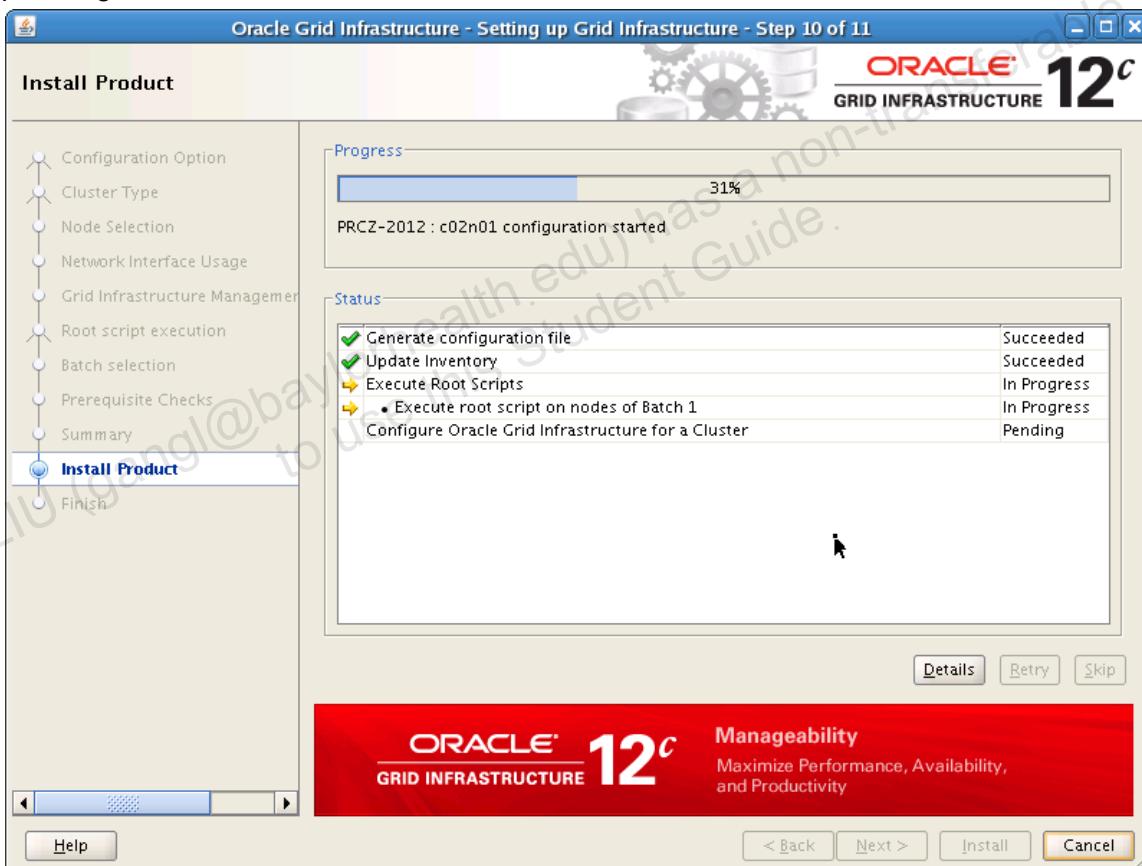


15. Before the root configuration scripts are executed, the configuration process will pause and wait until you acknowledge the dialog box. This is your last opportunity to safely exit the configuration process without upgrading the cluster. Click Yes to proceed with the cluster upgrade process.

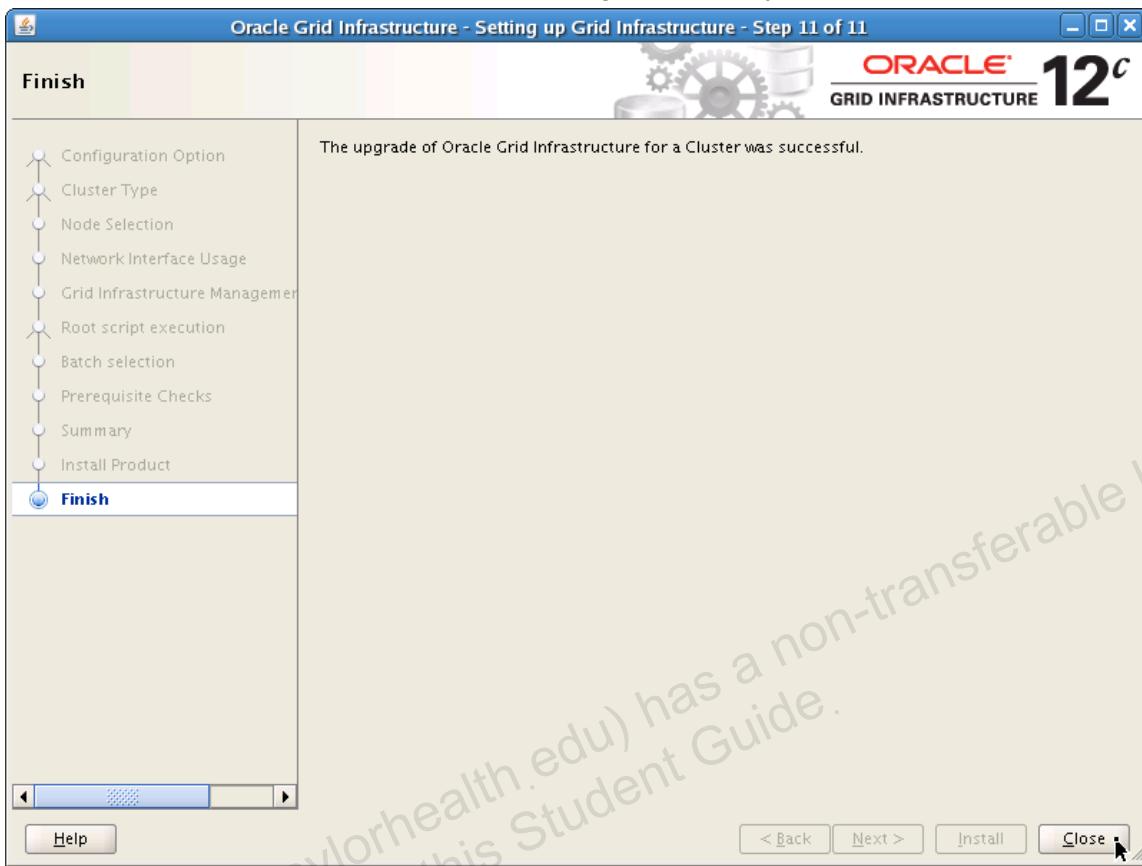


16. Use the Install Product page to monitor the Grid Infrastructure upgrade process. In addition, you can view the log files in /u01/app/oraInventory/logs on c02n01 and also in /u01/app/12.1.0/grid/cfgtoollogs/crsconfig on c02n01 and c02n02. You can also monitor each database instance by viewing its alert log file to determine when they are stopped and restarted. In your practice environment, the cluster upgrade process can take approximately 30 minutes to complete.

Note: During the cluster upgrade process, the Redo Apply process on the standby database is unavailable for a short period of time when Oracle Clusterware is shut down on the node running the apply instance. When this happens, redo apply starts on the other standby database instance. During this changeover period, a failure of the primary database could result in a longer outage than would normally be expected for a Data Guard environment. Depending on the nature of the failure, some data loss could also be experienced. To mitigate these risks, various approaches can be taken, including the use of an archived redo log repository or a bystander standby database. These options are not explored during this practice; however, be aware of them during your own upgrade planning.



17. When the Grid Infrastructure upgrade process completes, you will see the following page. Click Close to exit the Grid Infrastructure configuration utility.



18. **(Session 1)** In the same terminal window, refresh the environment settings in your grid@c02n01 terminal session to ensure that you use the upgraded Grid Infrastructure software from now on.

```
[grid@c02n01 ~]$ . oraenv
ORACLE_SID = [+ASM1] ? +ASM1
The Oracle base remains unchanged with value /u01/app/grid
[grid@c02n01 ~]$
```

19. Confirm that Oracle Grid Infrastructure has been upgraded on cluster02.

```
[grid@c02n01 ~]$ crsctl query crs activeversion
Oracle Clusterware active version on the cluster is [12.1.0.1.0]
```

20. **(Session 2)** Open a terminal window on c01n01 to connect as the oracle user.

```
[Classroom PC]$ ssh oracle@c01n01
oracle@c01n01's password: <oracle>
[oracle@c01n01 ~]$ . oraenv
ORACLE_SID = [oracle] ? c01orcl
The Oracle base has been set to /u01/app/oracle
[oracle@c01n01 ~]$ export ORACLE_SID=c01orcl1
[oracle@c01n01 ~]$
```

21. Launch the Data Guard Broker management utility.

```
[oracle@c01n01 ~]$ dgmgrl sys/oracle_4U
DGMGRL for Linux: Version 11.2.0.4.0 - 64bit Production

Copyright (c) 2000, 2009, Oracle. All rights reserved.

Welcome to DGMGRL, type "help" for information.
Connected.
DGMGRL>
```

22. Show the current configuration. Confirm that the configuration status is SUCCESS.

```
DGMGRL> show configuration

Configuration - dg_config

Protection Mode: MaxPerformance
Databases:
  c01orcl - Primary database
  c02orcl - Physical standby database

Fast-Start Failover: DISABLED

Configuration Status:
SUCCESS

DGMGRL>
```

23. Show the status of the physical standby database (c02orcl) and ensure that both the transport lag and the apply lag are at or near zero.

```
DGMGRL> show database c02orcl

Database - c02orcl

Role: PHYSICAL STANDBY
Intended State: APPLY-ON
Transport Lag: 0 seconds (computed 1 second ago)
Apply Lag: 0 seconds (computed 1 second ago)
Apply Rate: 40.00 KByte/s
Real Time Query: ON
Instance(s):
  c02orcl1 (apply instance)
  c02orcl2

Database Status:
SUCCESS

DGMGRL>
```

24. Perform a switchover to make c02orcl the primary database and c01orcl the standby database.

```
DGMGRL> switchover to c02orcl
Performing switchover NOW, please wait...
Operation requires a connection to instance "c02orcl2" on
database "c02orcl"
```

```
Connecting to instance "c02orcl2"...
Connected.
New primary database "c02orcl" is opening...
Operation requires startup of instance "c01orcl2" on database
"c01orcl"
Starting instance "c01orcl2"...
ORACLE instance started.
Database mounted.
Database opened.
Switchover succeeded, new primary is "c02orcl"
DGMGRL>
```

25. Confirm that the switchover has succeeded. If the configuration status reports anything other than SUCCESS, wait a moment and retry the show configuration command.

```
DGMGRL> show configuration

Configuration - dg_config

Protection Mode: MaxPerformance
Databases:
  c02orcl - Primary database
    c01orcl - Physical standby database

Fast-Start Failover: DISABLED

Configuration Status:
SUCCESS

DGMGRL>
```

26. Show the status of the new physical standby database (c01orcl) and ensure that both the transport lag and the apply lag are at or near zero. If a lag exceeding 5 seconds is reported, wait a moment and retry the show database command.

```
DGMGRL> show database c01orcl

Database - c01orcl

Role:          PHYSICAL STANDBY
Intended State: APPLY-ON
Transport Lag: 0 seconds (computed 1 second ago)
Apply Lag:     0 seconds (computed 1 second ago)
Apply Rate:    0 Byte/s
Real Time Query: ON
Instance(s):
  c01orcl1 (apply instance)
  c01orcl2

Database Status:
SUCCESS

DGMGRL>
```

27. Exit the Data Guard Broker management utility.

```
DGMGRL> exit
[oracle@c01n01 ~]$
```

28. **(Session 3)** Open another terminal session on c01n01 to connect as the grid user. Configure the terminal environment as shown in the following.

Note: Make sure that you use ssh with the -X option. The -X option is required to ensure that your terminal environment is correctly configured to run GUI applications and utilities.

```
[Classroom PC]$ ssh -X grid@c01n01
grid@c01n01's password: <oracle>
[grid@c01n01 ~]$ . oraenv
ORACLE_SID = [grid] ? +ASM1
The Oracle base has been set to /u01/app/grid
[grid@c01n01 ~]$
```

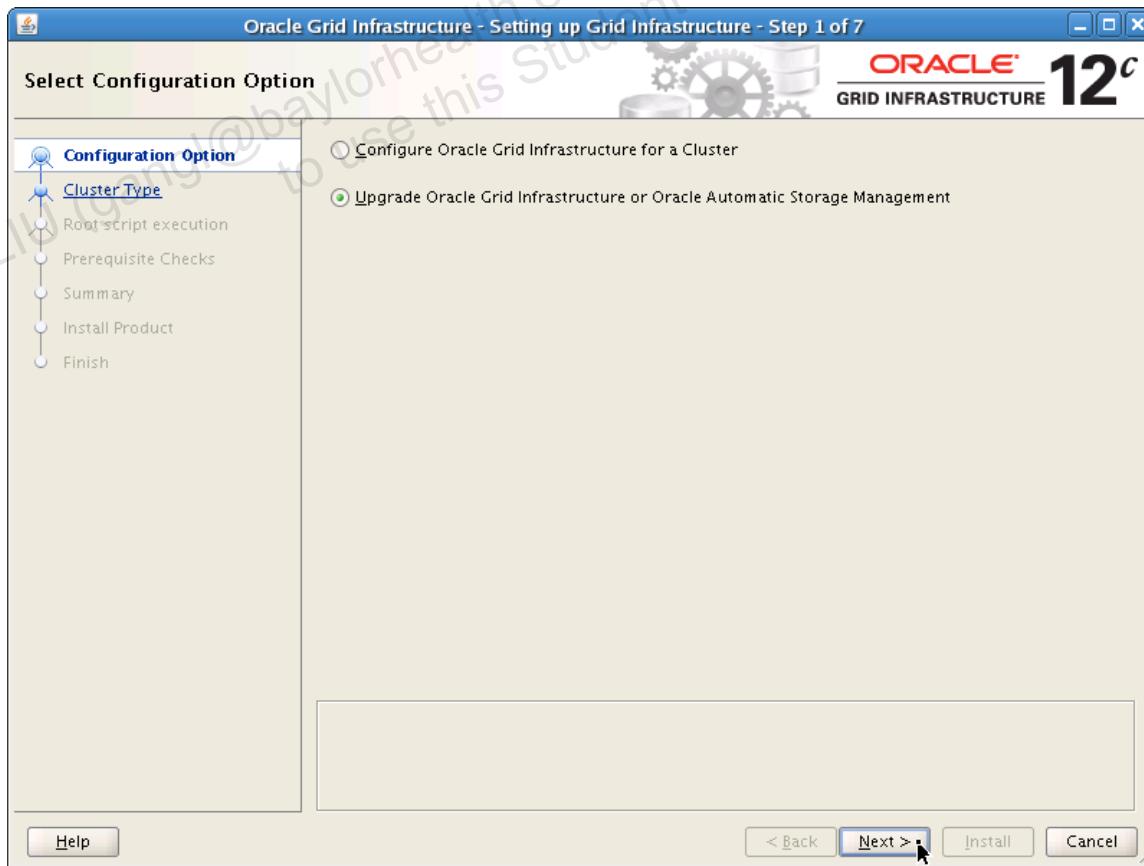
29. Confirm the current active version of Oracle Grid Infrastructure software on cluster01.

```
[grid@c01n01 ~]$ crsctl query crs activeversion
Oracle Clusterware active version on the cluster is [11.2.0.4.0]
[grid@c01n01 ~]$
```

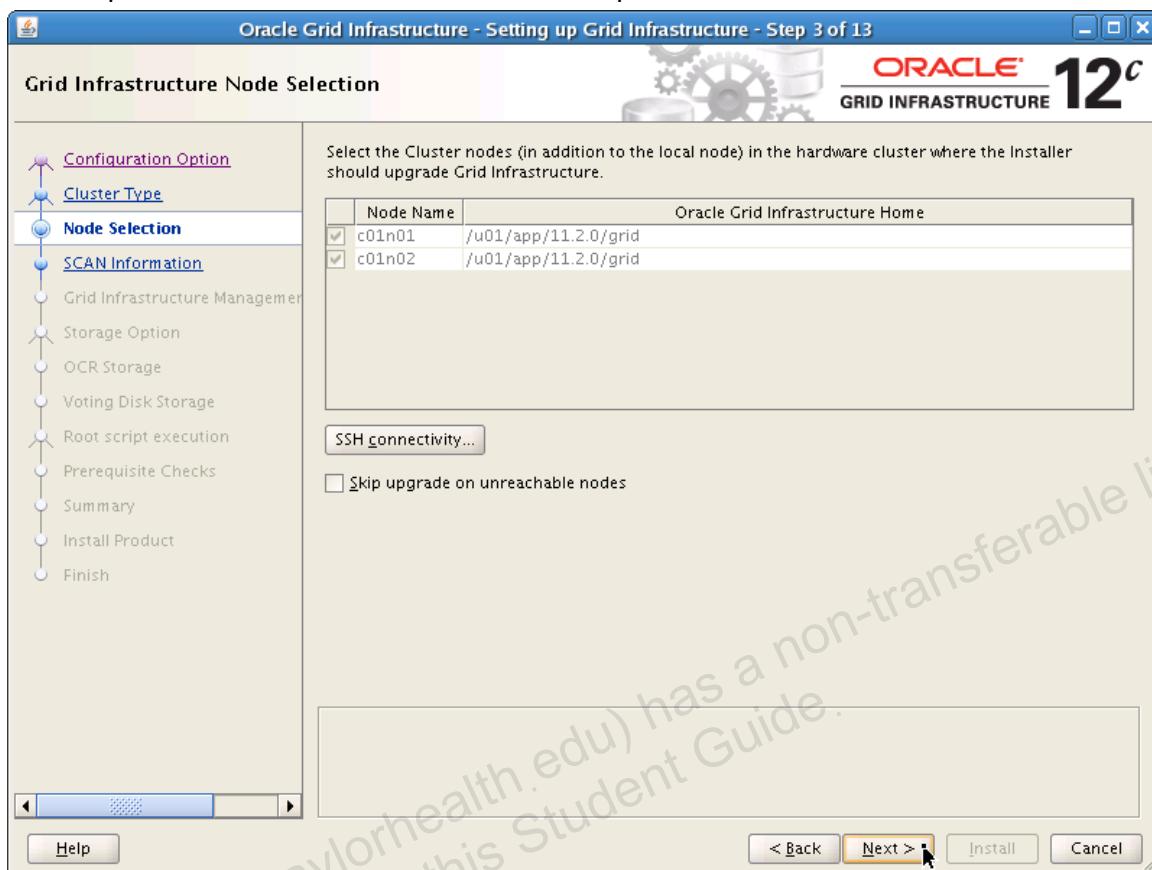
30. Launch the Oracle Grid Infrastructure configuration utility.

```
[grid@c01n01 ~]$ /u01/app/12.1.0/grid/crs/config/config.sh &
```

31. On the Select Configuration Option page, ensure that the “Upgrade Oracle Grid Infrastructure or Oracle Automatic Storage Management” option is selected and click Next to proceed.

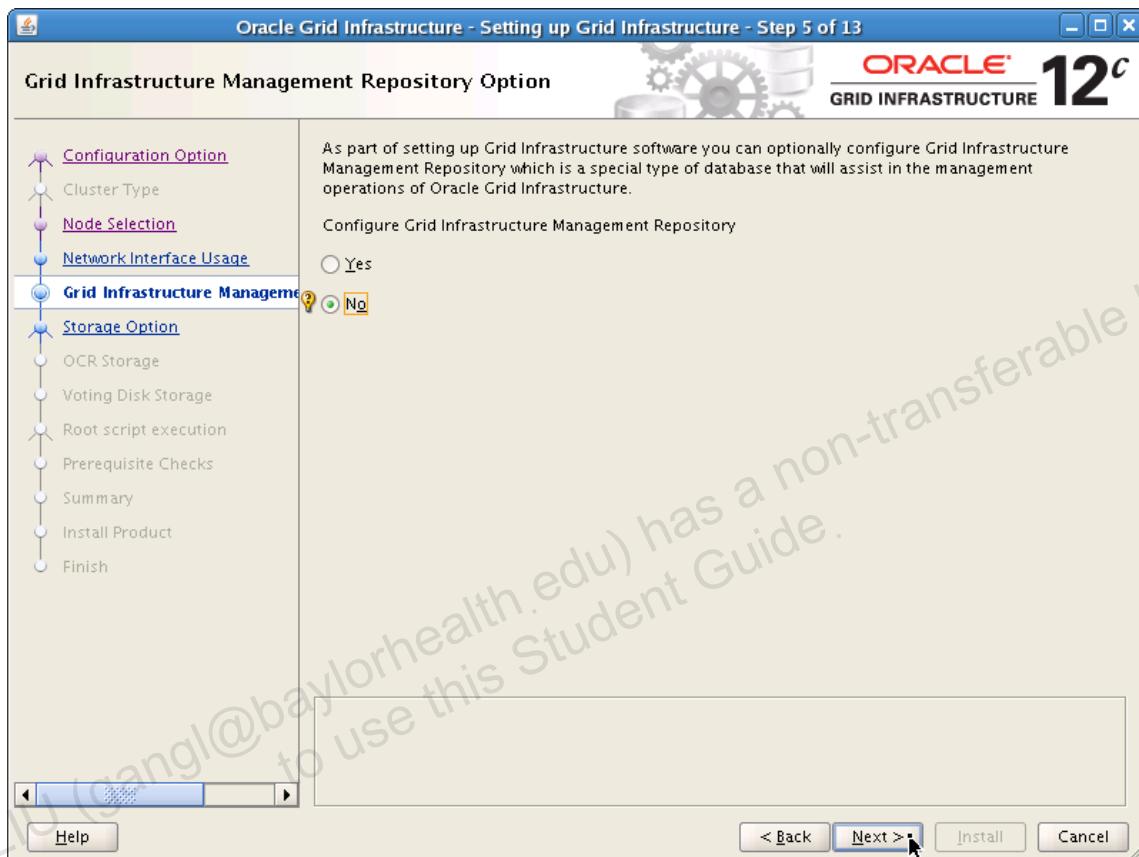


32. On the Grid Infrastructure Node Selection page, ensure that the details on the page match the composition of the cluster and click Next to proceed.



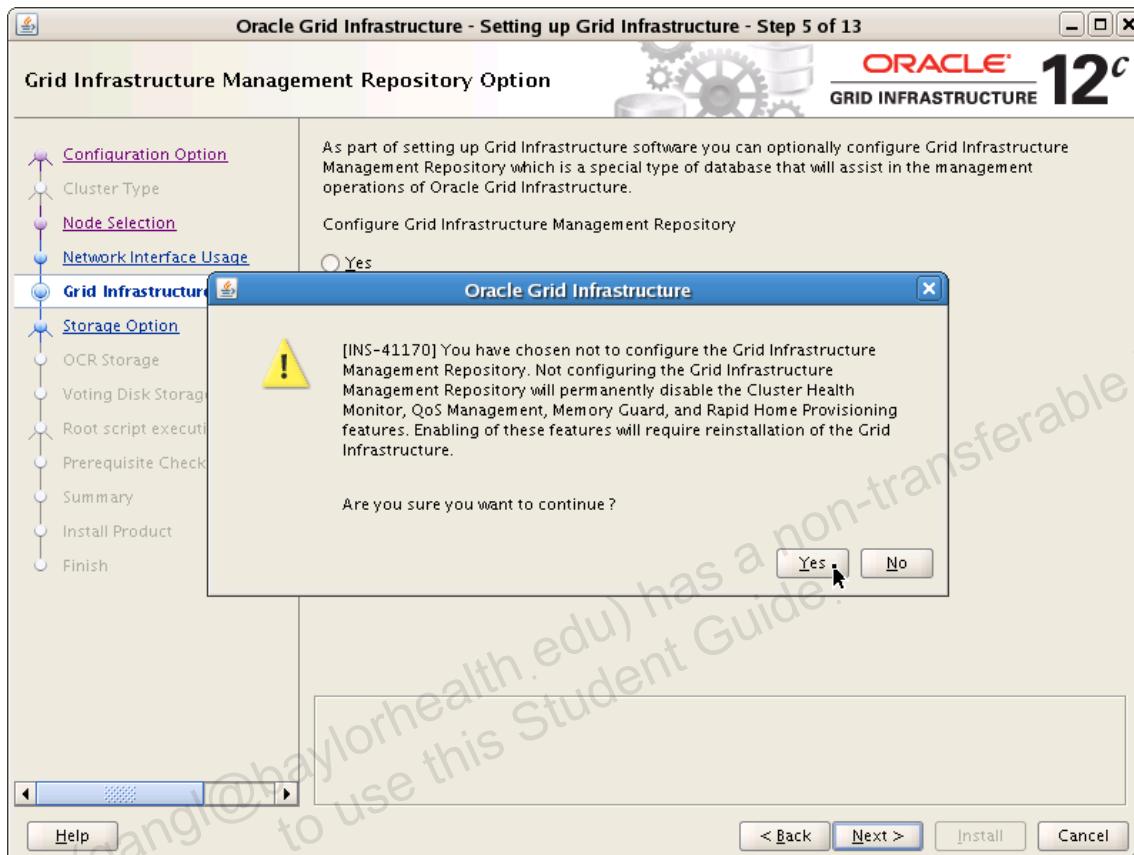
33. On the Grid Infrastructure Management Repository Option page, select the No option to not configure the Grid Infrastructure Management Repository and click Next to proceed.

Note: It is recommended that you do not configure the Grid Infrastructure Management Repository in your practice environment because it is not used in this practice. Moreover, it takes a significant amount of time to create the repository database and it consumes a significant amount of machine resources (relative to the modest capacity of the practice environment).

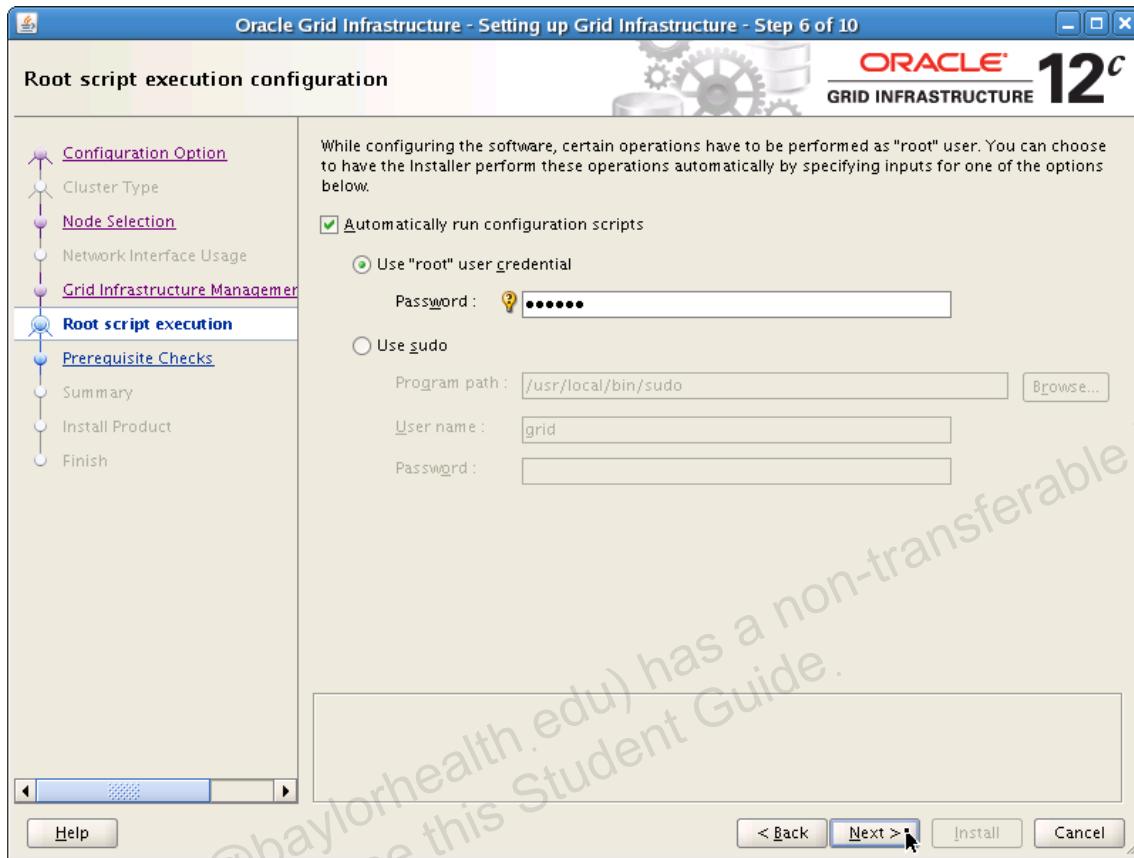


34. Click Yes to acknowledge the additional warning about the Grid Infrastructure Management Repository.

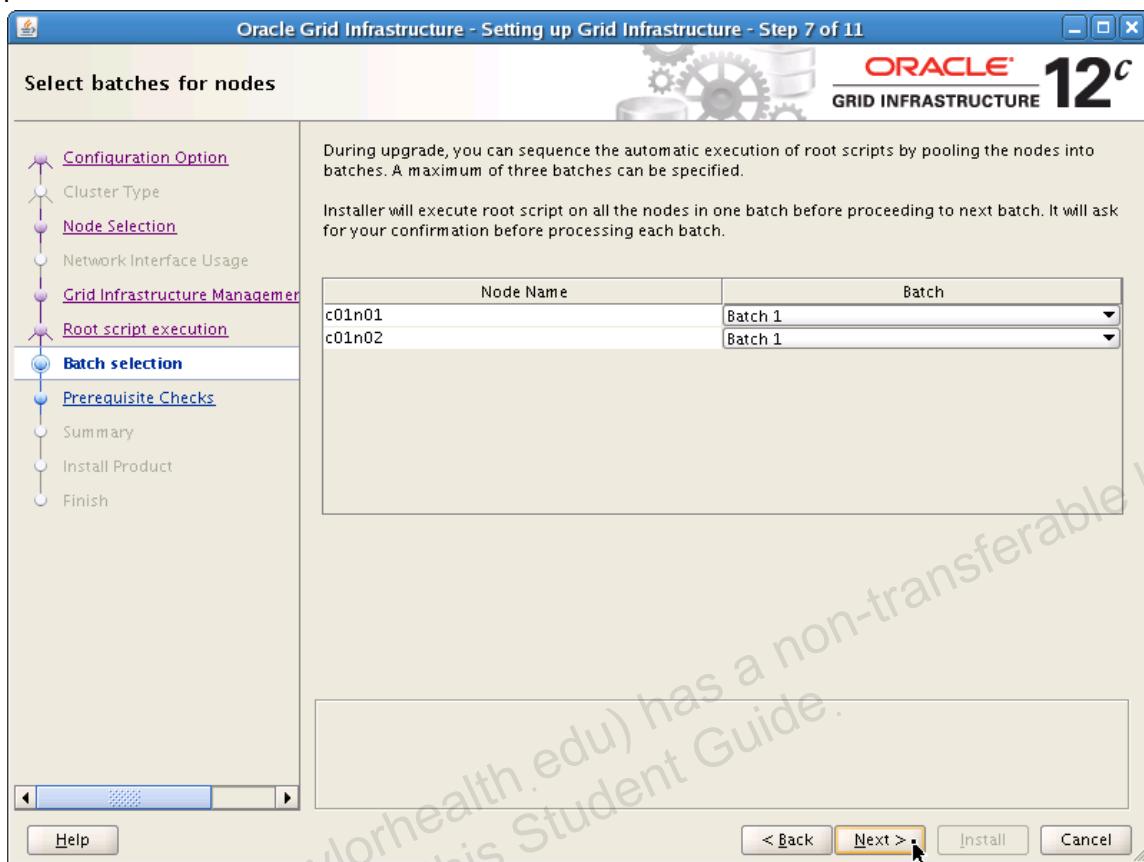
Note: Because the Grid Infrastructure Management Repository cannot be added without reinstalling Oracle Grid Infrastructure, it is generally recommended that you should configure the repository.



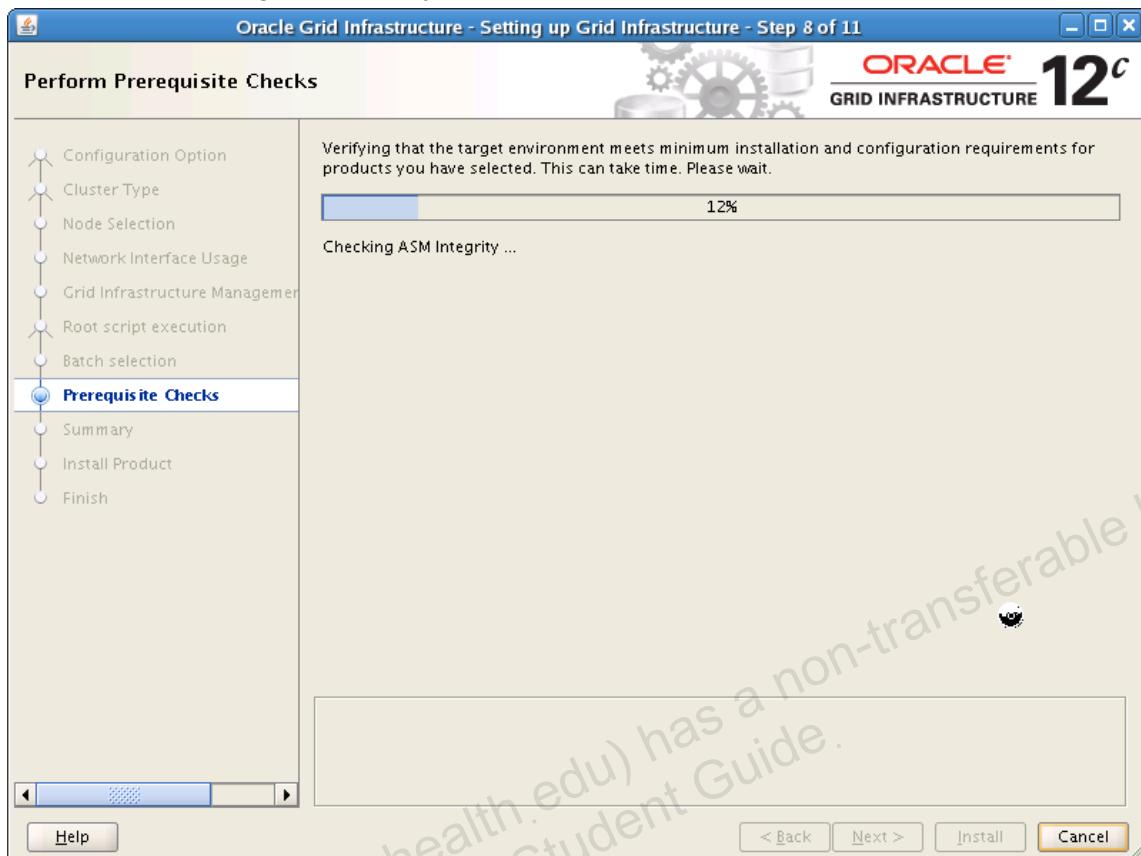
35. On the “Root script execution configuration” page, select the option to automatically run the configuration scripts by using the `root` user credential. Enter `oracle` as the `root` user password and click Next to proceed.



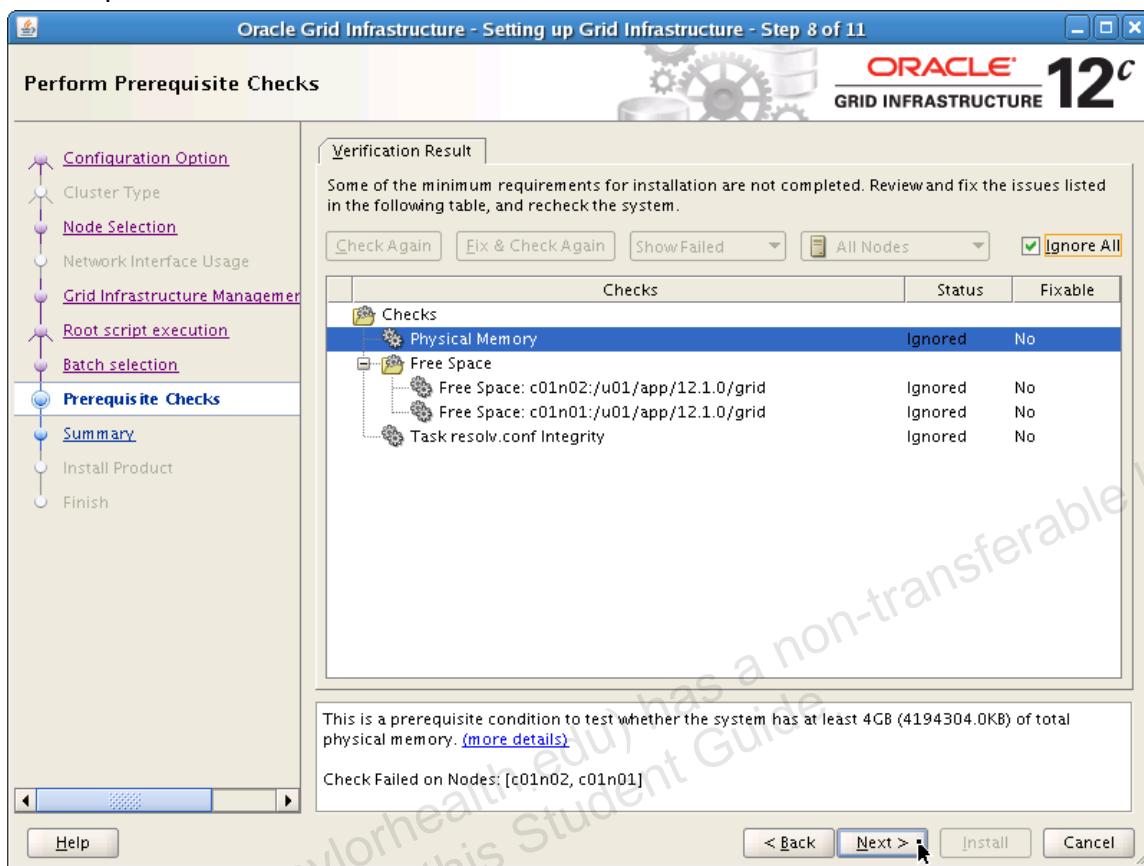
36. On the “Select batches for nodes” page, leave the default configuration and click Next to proceed.



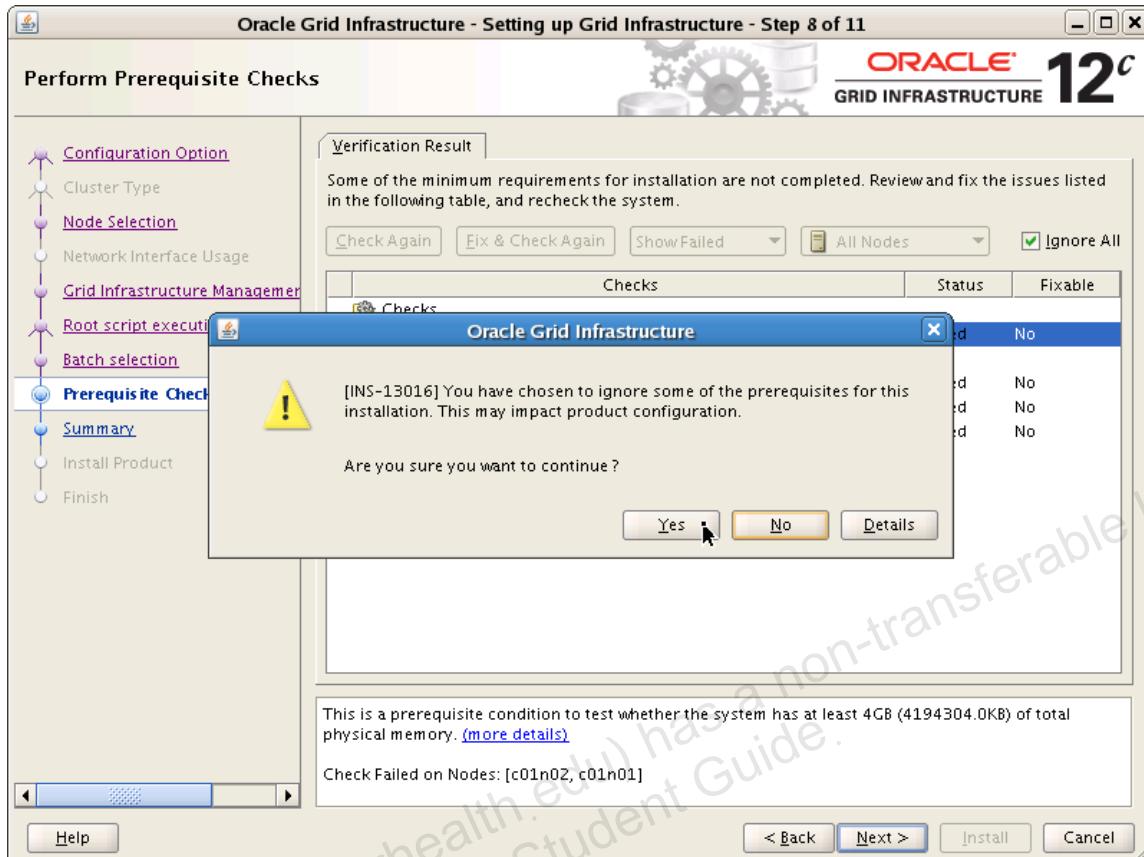
37. Wait while the configuration utility performs a series of checks.



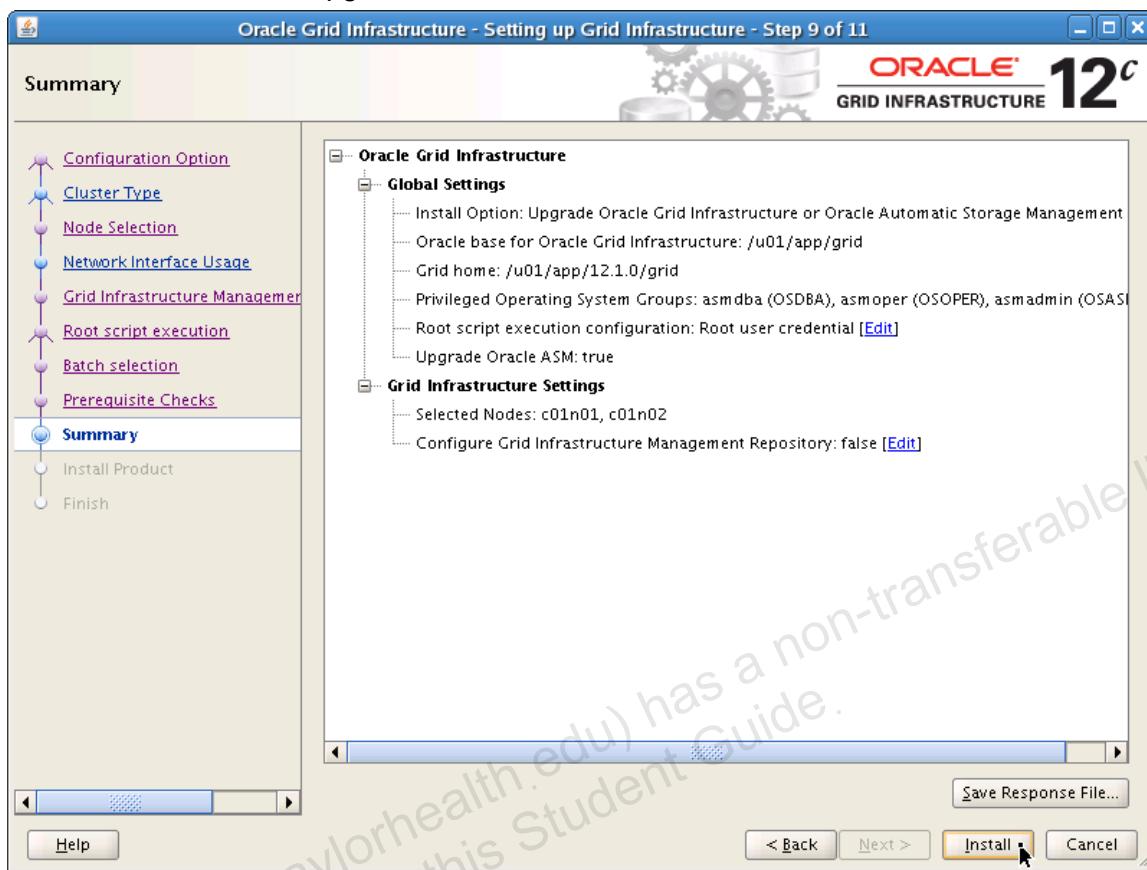
38. When the verification results appear, select the option to ignore all the warnings and click Next to proceed.



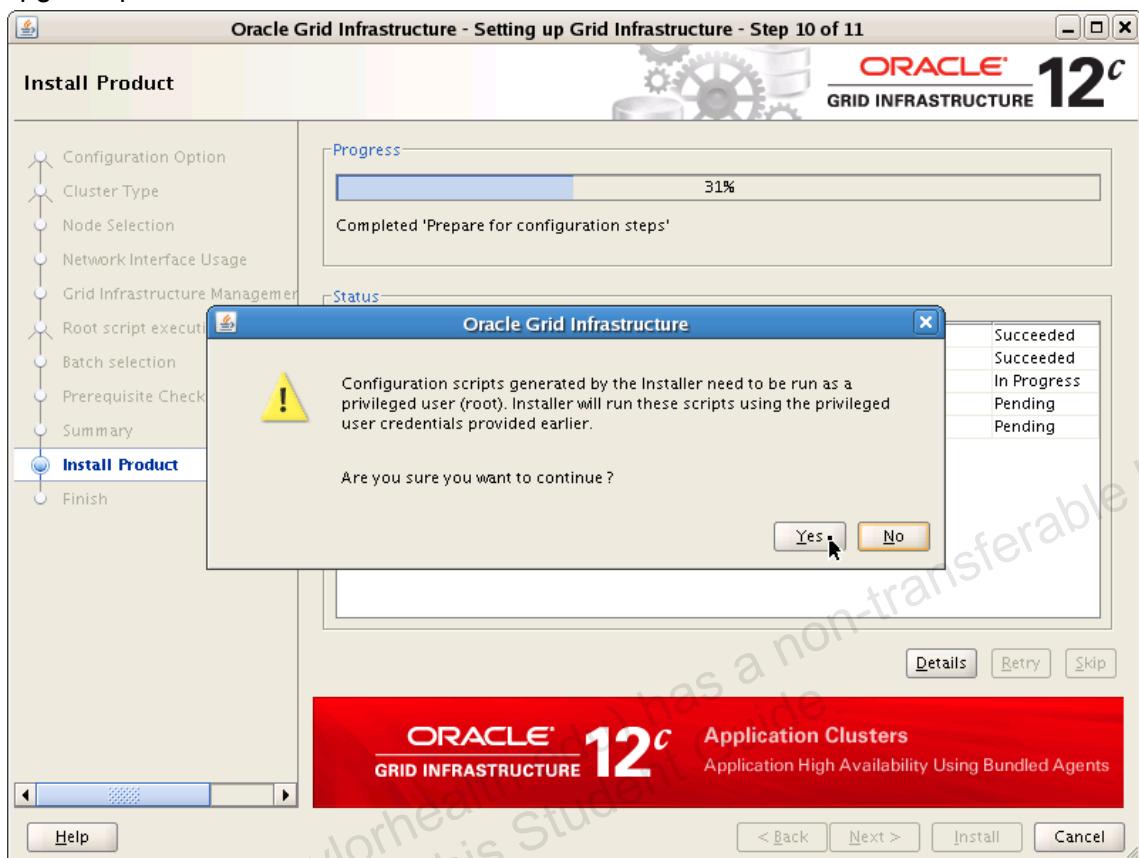
39. Click Yes to acknowledge the additional warning about the installation prerequisites.



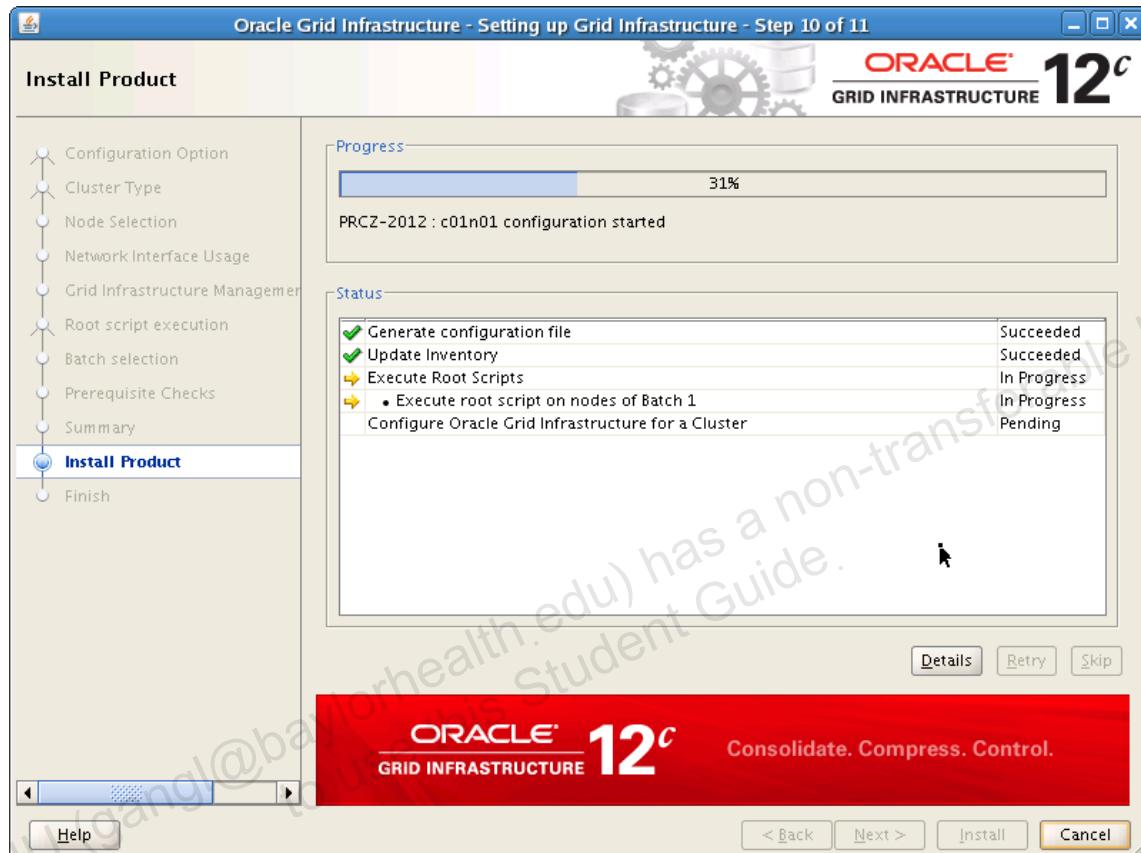
40. On the Summary page, confirm the configuration options that are displayed. When you are satisfied, click Install to upgrade Grid Infrastructure on cluster01.



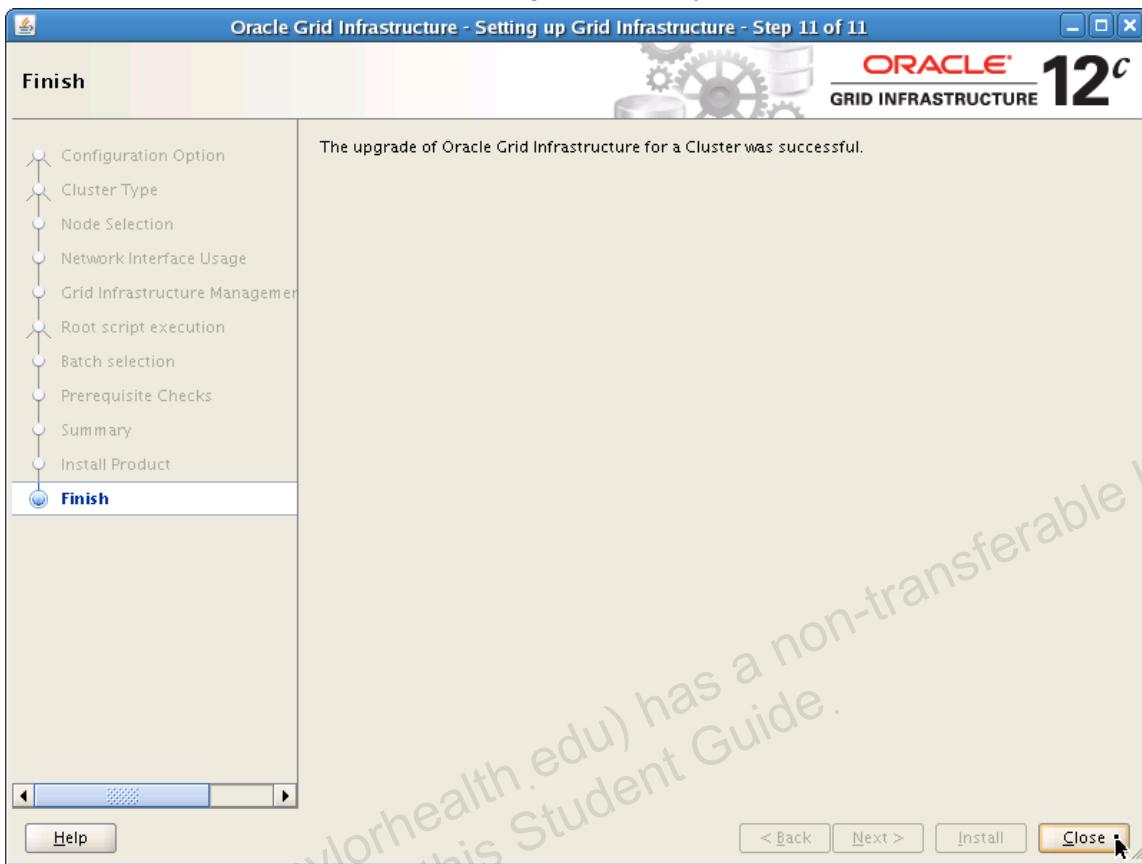
41. When the root script confirmation dialog box appears, click Yes to proceed with the cluster upgrade process.



42. Use the Install Product page to monitor the Grid Infrastructure upgrade process. In addition, you can view the log files in /u01/app/oraInventory/logs on c01n01 and also in /u01/app/12.1.0/grid/cfgtoollogs/crsconfig on c01n01 and c01n02. You can also monitor each database instance by viewing its alert log file to determine when they are stopped and restarted. Like cluster02, the upgrade process for cluster01 can take approximately 30 minutes to complete.



43. When the Grid Infrastructure upgrade process completes, you see the following page. Click Close to exit the Grid Infrastructure configuration utility.



44. **(Session 3)** In the same terminal session, refresh the environment settings in your grid@c01n01 terminal session to ensure that you use the upgraded Grid Infrastructure software from now on.

```
[grid@c01n01 ~]$ . oraenv
ORACLE_SID = [+ASM1] ? +ASM1
The Oracle base remains unchanged with value /u01/app/grid
[grid@c01n01 ~]$
```

45. Confirm that Oracle Grid Infrastructure has been upgraded on cluster01.

```
[grid@c01n01 ~]$ crsctl query crs activeversion
Oracle Clusterware active version on the cluster is [12.1.0.1.0]
[grid@c01n01 ~]$
```

46. **(Session 2)** Return to your oracle@c01n01 terminal session, and launch the Data Guard Broker management utility.

```
[oracle@c01n01 ~]$ dgmgrl sys/oracle_4U
DGMGRl for Linux: Version 11.2.0.4.0 - 64bit Production

Copyright (c) 2000, 2009, Oracle. All rights reserved.

Welcome to DGMGRl, type "help" for information.
Connected.
DGMGRl>
```

47. Show the current configuration. Confirm that the configuration status is SUCCESS.

```
DGMGRL> show configuration

Configuration - dg_config

Protection Mode: MaxPerformance
Databases:
  c02orcl - Primary database
  c01orcl - Physical standby database

Fast-Start Failover: DISABLED

Configuration Status:
SUCCESS

DGMGRL>
```

48. Show the status of the physical standby database (c01orcl) and ensure that both the transport lag and the apply lag are at or near zero.

```
DGMGRL> show database c01orcl

Database - c01orcl

Role: PHYSICAL STANDBY
Intended State: APPLY-ON
Transport Lag: 0 seconds (computed 1 second ago)
Apply Lag: 0 seconds (computed 2 seconds ago)
Apply Rate: 19.00 KByte/s
Real Time Query: ON
Instance(s):
  c01orcl1 (apply instance)
  c01orcl2

Database Status:
SUCCESS

DGMGRL>
```

49. Perform a switchover to once again make c01orcl the primary database and c02orcl the standby database.

```
DGMGRL> switchover to c01orcl
Performing switchover NOW, please wait...
New primary database "c01orcl" is opening...
Operation requires startup of instance "c02orcl2" on database
"c02orcl"
Starting instance "c02orcl2"...
ORACLE instance started.
Database mounted.
Database opened.
Switchover succeeded, new primary is "c01orcl"
DGMGRL>
```

50. Confirm that the switchover has succeeded. If the configuration status reports anything other than SUCCESS, wait a moment and retry the show configuration command.

```
DGMGRL> show configuration

Configuration - dg_config

Protection Mode: MaxPerformance
Databases:
  c01orcl - Primary database
  c02orcl - Physical standby database

Fast-Start Failover: DISABLED

Configuration Status:
SUCCESS

DGMGRL>
```

51. Show the status of the new physical standby database (c02orcl) and ensure that both the transport lag and the apply lag are at or near zero. If a lag exceeding 5 seconds is reported, wait a moment and retry the show database command.

```
DGMGRL> show database c02orcl

Database - c02orcl

Role:          PHYSICAL STANDBY
Intended State: APPLY-ON
Transport Lag: 0 seconds (computed 1 second ago)
Apply Lag:    0 seconds (computed 1 second ago)
Apply Rate:   0 Byte/s
Real Time Query: ON
Instance(s):
  c02orcl1 (apply instance)
  c02orcl2

Database Status:
SUCCESS

DGMGRL>
```

52. Close all of your terminal sessions.

Practice 12-3: Database Preparation for Rolling Upgrade

Overview

At this point, you have upgraded both clusters in your Oracle RAC and Data Guard environment. Next, you will prepare the databases on both clusters so that they can participate in the database rolling upgrade process.

Here is an outline of the tasks that you will perform in this section:

- Disable the Data Guard Broker.
- Ensure that the recovery area for each database is sized appropriately for the upgrade process.
- Ensure that the Flashback Database feature is enabled for each database.
- Ensure that managed recovery is enabled on the standby database.

Terminal Sessions

- **Session 1:** `ssh oracle@c01n01`
- **Session 2:** `ssh oracle@c02n01`

Tasks

1. **(Session 1)** Establish a terminal session connected to `c01n01` as the `oracle` user and configure the terminal environment as shown in the following:

```
[Classroom PC] $ ssh oracle@c01n01
oracle@c01n01's password: <oracle>
[oracle@c01n01 ~]$ . oraenv
ORACLE_SID = [oracle] ? c01orcl
The Oracle base has been set to /u01/app/oracle
[oracle@c01n01 ~]$ export ORACLE_SID=c01orcl1
[oracle@c01n01 ~]$
```

2. Disable the Data Guard Broker configuration and exit the Data Guard Broker management utility.

```
[oracle@c01n01 ~]$ dgmgrl sys/oracle_4u
DGMGRL> disable configuration
Disabled.
DGMGRL> exit
[oracle@c01n01 ~]$
```

3. Using SQL*Plus as `oracle` on `c01n01`, connect to your primary database (`c01orcl`) and disable Data Guard Broker.

```
[oracle@c01n01 ~]$ sqlplus / as sysdba
SQL*Plus: Release 11.2.0.4.0 Production ...
SQL> alter system set dg_broker_start=false scope=both;
System altered.
SQL>
```

- Set the size of the database recovery area so that it is large enough to accommodate the archived redo logs generated during the rolling upgrade process.

Notes

- The amount of space required here depends on the composition of your database, including which database options are installed, along with the transaction volume generated during the rolling upgrade. Although the size specified here (9000M) is sufficient for this environment, it is likely to be insufficient in many customer situations.
- Setting the `db_recovery_file_dest_size` parameter sets the limit for the amount of recovery area space used by that database. It does not ensure that this amount of space is actually available. You should verify that the file system or ASM disk group hosting your recovery area has sufficient space available.

```
SQL> alter system set db_recovery_file_dest_size=9000M
scope=both;
System altered.
SQL>
```

- Enable the Flashback Database feature on your primary database and exit SQL*Plus.

```
SQL> alter database flashback on;
Database altered.
SQL> exit
Disconnected from Oracle Database 11g Enterprise Edition ...
[oracle@c01n01 ~]$
```

- (Session 2) Open another terminal session connected to c02n01 as the oracle user. Configure the terminal environment as shown in the following:

```
[Classroom PC]$ ssh oracle@c02n01
oracle@c02n01's password: <oracle>
[oracle@c02n01 ~]$ . oraenv
ORACLE_SID = [oracle] ? c02orcl
The Oracle base has been set to /u01/app/oracle
[oracle@c02n01 ~]$ export ORACLE_SID=c02orcl1
[oracle@c02n01 ~]$
```

- Using SQL*Plus as oracle on c02n01, connect to your standby database (c02orcl) and disable the Data Guard Broker.

```
[oracle@c02n01 ~]$ sqlplus / as sysdba
SQL*Plus: Release 11.2.0.4.0 Production ...
SQL> alter system set dg_broker_start=false scope=both;
System altered.
SQL>
```

- Set the size of the recovery area in line with the size that you previously set on the primary database.

```
SQL> alter system set db_recovery_file_dest_size=9000M
scope=both;
System altered.
SQL>
```

9. Stop the managed recovery process on the standby database.

```
SQL> alter database recover managed standby database cancel;  
Database altered.  
SQL>
```

10. Enable the Flashback Database feature on your standby database.

```
SQL> alter database flashback on;  
Database altered.  
SQL>
```

11. Restart the managed recovery process on the standby database and exit SQL*Plus.

```
SQL> alter database recover managed standby database disconnect;  
Database altered.  
SQL> exit  
Disconnected from Oracle Database 11g Enterprise Edition ...  
[oracle@c02n01 ~]$
```

12. Close all your terminal sessions.

Practice 12-4: Database Rolling Upgrade

Overview

At this point, your environment should be ready to perform a database rolling upgrade. The database rolling upgrade process requires many operations and role changes to be performed in a precise order. To simplify the process and reduce the likelihood of administrator error, the `physru.sh` script is available through My Oracle Support bulletin 949322.1. This script automates much of the process and provides guidance regarding the required manual steps. The script also performs many checks along the way to maximize the probability of success. In your practice environment, the script is preloaded in the `oracle` user's home directory on `c01n01`.

Terminal Sessions

- **Session 1:** `ssh oracle@c01n01 (physru.sh session)`
- **Session 2:** `ssh -X oracle@c02n01`
- **Session 3:** `ssh oracle@c01n01`
- **Session 4:** `ssh grid@c02n01`
- **Session 5:** `ssh grid@c01n01`

- To avoid confusion, you may want to start all five sessions in the beginning.



Tasks

1. **(Session 1)** Establish a terminal session connected to `c01n01` as the `oracle` user and configure the terminal environment as shown in the following:

```
[Classroom PC] $ ssh oracle@c01n01
oracle@c01n01's password: <oracle>
[oracle@c01n01 ~]$ . oraenv
ORACLE_SID = [oracle] ? c01orcl
The Oracle base has been set to /u01/app/oracle
[oracle@c01n01 ~]$ export ORACLE_SID=c01orcl
[oracle@c01n01 ~]$
```

2. Launch `physru.sh` without any parameters and examine the output.

```
[oracle@c01n01 ~]$ ./physru.sh
Usage: physru <username> <primary_tns> <standby_tns>
          <primary_name> <standby_name> <upgrade_version>
Purpose:
```

Perform a rolling upgrade between a primary and physical standby database. This script simplifies a physical standby rolling upgrade. While numerous steps have been automated, this script must be called at least three times in order to complete a rolling upgrade. When this script reaches a point where user intervention is required, it outputs a message indicating what is expected of the user. Once the user action is complete, this script can be called to resume the rolling upgrade. In the event of an error, a user can take corrective action, and simply call this script again to resume the rolling upgrade. In the event one wishes to abandon the rolling upgrade, and revert the configuration back to its pre-upgrade state, this script creates guaranteed flashback database restore points on both the primary and standby databases, and backs up each databases' associated control file. The names of the restore points and backup control files are output to the console and logfile when they are initially created.

When this script is called, it assumes all databases to be either mounted or open. It requires flashback database to be enabled on both the primary and standby instances. RAC configurations are permitted but there is limited automation provided by the script. At specific points it may become necessary to manually shutdown/startup instances and change init.ora parameter values. When appropriate, the script will output when these requirements are expected of the user. RAC configurations are also required to define static tns services since this script expects a given tns service name to contact the same instance on successive calls.

Arguments:

```
<username> = dba username
<primary_tns> = tns service name to primary
<standby_tns> = tns service name to physical standby
<primary_name> = db_unique_name of primary
<standby_name> = db_unique_name of standby
<upgrade_version> = target rdbms version
```

Example:

```
physru sys hq_tnspri hq_tnsstb hq_primary hq_standby 11.2.0.2.0
```

NOTE: This script performs role transitions, and it is not necessary to adjust the tns and db name arguments to their respective database roles on successive calls. That is, the arguments must remain the same from first-invocation to completion.

ERROR: 0 of the 6 required parameters have been specified

```
[oracle@c01n01 ~]$
```

3. Launch physru.sh as shown below to commence the database rolling upgrade process. Enter oracle_4U when you are prompted for the sysdba password.

Notes

- Pay careful attention to the output from physru.sh. If your output differs significantly, check with your instructor before proceeding to the next step. To help you better understand what is happening, additional notes are included throughout the example output. In addition, particularly interesting output will be highlighted along the way.

- While physru.sh is executing, you may find it interesting to monitor the alert log for your primary (c01orcl) and standby (c02orcl) databases because the contents of these logs will provide additional insight about the operations being performed on both databases.
- In this practice, Oracle Net connection strings using easy connect naming are used to connect to the primary and standby databases. These connection strings leverage the static service registration that is already configured to support Data Guard Broker. Alternatively, administrators can use Oracle Net service names. However, note that such Oracle Net service names must always resolve to the same database instance in RAC configurations, which means that no Oracle Net connection load balancing mechanisms can be used.
- During the rolling upgrade process, the standby database is periodically unable to accept updates from the primary. While the standby database is unavailable, a failure of the primary database could result in a longer outage than would normally be expected for a Data Guard environment. Depending on the nature of the failure, some data loss could also be experienced. To mitigate these risks, various approaches can be taken, including the use of an archived redo log repository or a bystander standby database. These options are not explored during this practice; however, be aware of them during your own upgrade planning.

```
[oracle@c01n01 ~]$ ./physru.sh sys
c01n01:1521/c01orcl_DGMGRL.example.com
c02n01:1521/c02orcl_DGMGRL.example.com c01orcl c02orcl 12.1.0.1.0

Please enter the sysdba password:
<oracle_4U>

### Initialize script to either start over or resume execution
Aug 25 06:16:02 2014 [0-1] Identifying dbms software version
Aug 25 06:16:03 2014 [0-1] database c01orcl is at version 11.2.0.4.0
Aug 25 06:16:03 2014 [0-1] database c02orcl is at version 11.2.0.4.0
Aug 25 06:16:03 2014 [0-1] verifying flashback database is enabled at c01orcl
and c02orcl
Aug 25 06:16:04 2014 [0-1] verifying available flashback restore points
Aug 25 06:16:04 2014 [0-1] verifying DG Broker is disabled
Aug 25 06:16:04 2014 [0-1] looking up prior execution history
Aug 25 06:16:04 2014 [0-1] purging script execution state from database
c01orcl
Aug 25 06:16:04 2014 [0-1] purging script execution state from database
c02orcl
Aug 25 06:16:05 2014 [0-1] starting new execution of script
```

Note: Every time you run physru.sh, it will first perform an initialization sequence to ensure that the main elements of your environment are in the expected state. If any issues are detected, the script will produce an error message and exit. After you have corrected the problem, you can safely restart the script.

```
### Stage 1: Backup user environment in case rolling upgrade is aborted

Aug 25 06:16:05 2014 [1-1] stopping media recovery on c02orcl
Aug 25 06:16:07 2014 [1-1] creating restore point PRU_0000_0001 on database
c02orcl
Aug 25 06:16:08 2014 [1-1] backing up current control file on c02orcl
Aug 25 06:16:13 2014 [1-1] created backup control file
```

```
/u01/app/oracle/product/11.2.0/dbhome_1/dbs/PRU_0001_c02orcl_f.f
Aug 25 06:16:13 2014 [1-1] creating restore point PRU_0000_0001 on database
c01orcl
Aug 25 06:16:15 2014 [1-1] backing up current control file on c01orcl
Aug 25 06:16:19 2014 [1-1] created backup control file
/u01/app/oracle/product/11.2.0/dbhome_1/dbs/PRU_0001_c01orcl_f.f
NOTE: Restore point PRU_0000_0001 and backup control file
PRU_0001_c02orcl_f.f
can be used to restore c02orcl back to its original state as a
physical standby, in case the rolling upgrade operation needs to be
aborted prior to the first switchover done in Stage 4.
```

Note: Stage 1 uses the Flashback Database feature to create restore points in both the primary and standby databases. These can be used to return your environment to the preupgrade state, if required, any time before the first switchover that occurs later in stage 4. Note that flashing back to these restore points will not only undo any upgrade operations, but will also undo valid user transactions, which will continue to flow into the primary database throughout the upgrade process.

Stage 2: Create transient logical standby from existing physical standby

```
Aug 25 06:16:20 2014 [2-1] verifying RAC is disabled at c02orcl
```

WARN: c02orcl is a RAC database. Before this script can continue, you must manually reduce the RAC to a single instance, disable the RAC, and restart instance c02orcl1 in mounted mode. This can be accomplished with the following steps:

- 1) Shutdown all instances other than instance c02orcl1.
eg: srvctl stop instance -d c02orcl -i c02orcl2 -o abort
- 2) On instance c02orcl1, set the cluster_database parameter to FALSE.
eg: SQL> alter system set cluster_database=false scope=spfile;
- 3) Shutdown instance c02orcl1.
eg: SQL> shutdown abort;
- 4) Startup instance c02orcl1 in mounted mode.
eg: SQL> startup mount;

Once these steps have been performed, enter 'y' to continue the script. If desired, you may enter 'n' to exit the script to perform the required steps, and recall the script to resume from this point.

Are you ready to continue? (y/n) :

Note: Stage 2 converts the existing physical standby database into a transient logical standby database. The transient logical standby database is a key element of the rolling upgrade process because it allows user transactions to flow into it from the primary database at the same time as it is being upgraded. However, before the physical-to-logical conversion can take place, Oracle RAC must be disabled on the existing physical standby database.

4. **(Session 2)** Open another terminal session connected to c02n01 as the oracle user. Configure the terminal environment as shown in the following.
Note: Make sure that you use ssh with the -X option. The -X option is required to ensure that your terminal environment is correctly configured to run GUI applications and utilities.

```
[Classroom PC]$ ssh -X oracle@c02n01
oracle@c02n01's password: <oracle>
[oracle@c02n01 ~]$ . oraenv
ORACLE_SID = [oracle] ? c02orcl
The Oracle base has been set to /u01/app/oracle
[oracle@c02n01 ~]$ export ORACLE_SID=c02orcl1
[oracle@c02n01 ~]$
```

5. Follow the instructions supplied in the latest physru.sh output to disable Oracle RAC and restart a single instance of the physical standby database.

```
[oracle@c02n01 ~]$ srvctl stop instance -d c02orcl -i c02orcl2 -o abort
[oracle@c02n01 ~]$ sqlplus / as sysdba

SQL*Plus: Release 11.2.0.4.0 Production ...

SQL> alter system set cluster_database=false scope=spfile;
System altered.

SQL> shutdown abort
ORACLE instance shut down.
SQL> startup mount
ORACLE instance started.

Total System Global Area 1503199232 bytes
Fixed Size                  2253424 bytes
Variable Size                452988304 bytes
Database Buffers            1040187392 bytes
Redo Buffers                 7770112 bytes
Database mounted.
SQL> exit
Disconnected from Oracle Database 11g Enterprise Edition ...
[oracle@c02n01 ~]$
```

6. **(Session 1)** After reconfiguring your standby database, return to your physru.sh session and enter y to continue.

```
Are you ready to continue? (y/n): y

Aug 25 06:20:49 2014 [2-1] continuing
Aug 25 06:20:49 2014 [2-1] verifying RAC is disabled at c02orcl
Aug 25 06:20:49 2014 [2-1] verifying database roles
Aug 25 06:20:49 2014 [2-1] verifying physical standby is mounted
Aug 25 06:20:49 2014 [2-1] verifying database protection mode
Aug 25 06:20:49 2014 [2-1] verifying transient logical standby datatype support

WARN: Objects have been identified on the primary database which will not be replicated on the transient logical standby. The complete list of objects and their associated unsupported
```

datatypes can be found in the dba_logstdby_unsupported view. For convenience, this script has written the contents of this view to a file - physru_unsupported.log.

Various options exist to deal with these objects such as:

- disabling applications that modify these objects
- manually resolving these objects after the upgrade
- extending support to these objects (see metalink note: 559353.1)

If you need time to review these options, you should enter 'n' to exit the script. Otherwise, you should enter 'y' to continue with the rolling upgrade.

Are you ready to proceed with the rolling upgrade? (y/n) :

Note: Logical standby databases have limitations that prevent the replication of some primary database objects. If your databases include objects that are affected by these limitations, physru.sh pauses during stage 2 to alert you and the script produces a file that lists the affected objects. You can use this information to determine the best course of action for dealing with this situation. It is highly recommended that you understand how the limitations associated with logical standby databases affect your database as one of the key planning tasks before attempting any rolling upgrade. To do this, you can also connect to your database and examine the contents of the dba_logstdby_unsupported view.

7. **(Session 3)** Open another terminal session connected to c01n01 as the oracle user and configure the terminal environment as shown in the following:

```
[Classroom PC] $ ssh oracle@c01n01
oracle@c01n01's password: <oracle>
[oracle@c01n01 ~]$ . oraenv
ORACLE_SID = [oracle] ? c01orcl
The Oracle base has been set to /u01/app/oracle
[oracle@c01n01 ~]$ export ORACLE_SID=c01orcl1
[oracle@c01n01 ~]$
```

8. Examine the contents of the physru_unsupported.log file to list the objects in your database that are not supported by the logical standby database.

```
[oracle@c01n01 ~]$ cat physru_unsupported.log
OWNER TABLE_NAME
-----
COLUMN_NAME ATTRIBUTES
-----
DATA_TYPE
-----
OE PURCHASEORDER
SYS_NC_ROWINFO$ Unsupported XML
OPAQUE
OE CATEGORIES_TAB
CATEGORY_NAME Object Table
VARCHAR2
OE CATEGORIES_TAB
CATEGORY_DESCRIPTION Object Table
VARCHAR2
OE CATEGORIES_TAB
CATEGORY_ID Object Table
```

```

NUMBER
OE_CATEGORIES_TAB
PARENT_CATEGORY_ID Object Table
NUMBER
PM_PRINT_MEDIA
AD_TEXTDOCS_NTAB Unsupported Virtual Column
NESTED TABLE
PM_PRINT_MEDIA
AD_GRAPHIC
BFILE
...
IX_AQ$_STREAMS_QUEUE_TABLE_C
MSGCNT AQ queue table
NUMBER
SH_DIMENSION_EXCEPTIONS
BAD_ROWID
ROWID
[oracle@c01n01 ~]$
```

Note: In this practice, you will deal with the logical standby database limitations by ensuring that none of the affected objects are updated during the rolling upgrade process.

9. (Session 1) Return to your physru.sh session and enter y to continue.

```

Are you ready to proceed with the rolling upgrade? (y/n) : y

Aug 25 06:22:45 2014 [2-1] continuing
Aug 25 06:22:46 2014 [2-2] starting media recovery on c02orcl
Aug 25 06:22:52 2014 [2-2] confirming media recovery is running
Aug 25 06:22:53 2014 [2-2] waiting for apply lag to fall under 30 seconds
Aug 25 06:26:42 2014 [2-2] apply lag measured at 229 seconds
Aug 25 06:26:46 2014 [2-2] apply lag measured at 4 seconds
Aug 25 06:26:46 2014 [2-2] stopping media recovery on c02orcl
Aug 25 06:26:47 2014 [2-2] executing dbms_logstdby.build on database
c01orcl
Aug 25 06:27:33 2014 [2-2] converting physical standby into transient
logical
standby
Aug 25 06:27:43 2014 [2-3] opening database c02orcl
Aug 25 06:27:57 2014 [2-4] configuring transient logical standby parameters
for rolling upgrade
Aug 25 06:27:59 2014 [2-4] starting logical standby on database c02orcl
Aug 25 06:28:08 2014 [2-4] waiting until logminer dictionary has fully
loaded
Aug 25 06:31:03 2014 [2-4] dictionary load 22% complete
Aug 25 06:31:13 2014 [2-4] dictionary load 41% complete
Aug 25 06:31:23 2014 [2-4] dictionary load 62% complete
Aug 25 06:31:33 2014 [2-4] dictionary load 75% complete
Aug 25 06:31:54 2014 [2-4] dictionary load 99% complete
Aug 25 06:32:04 2014 [2-4] dictionary load is complete
Aug 25 06:32:07 2014 [2-4] waiting for apply lag to fall under 30 seconds
Aug 25 06:32:22 2014 [2-4] apply lag measured at 5 seconds
```

NOTE: Database c02orcl is now ready to be upgraded. This script has left the

database open in case you want to perform any further tasks before upgrading the database. Once the upgrade is complete, the database must be opened in READ WRITE mode before this script can be called to resume the rolling upgrade.

NOTE: Database c02orcl may be reverted back to a RAC database upon completion of the dbms upgrade. This can be accomplished by performing the following steps:

- 1) On instance c02orcl1, set the cluster_database parameter to TRUE.
eg: SQL> alter system set cluster_database=true scope=spfile;
- 2) Shutdown instance c02orcl1.
eg: SQL> shutdown abort;
- 3) Startup and open all instances for database c02orcl.
eg: srvctl start database -d c02orcl

```
[oracle@c01n01 ~]$
```

Note: During the final part of stage 2, the standby database (c02orcl) is converted into a transient logical standby. After this operation completes, the standby database is ready to be upgraded. In the next part of this practice, you will use the Database Upgrade Assistant (DBUA) to upgrade c02orcl. However, before DBUA is started, the following preparations are required.

10. **(Session 2)** Return to the terminal session on c02n01 as the oracle user. When DBUA is used on an Oracle RAC database, it requires all the database instances to be open at the beginning of the DBUA session. Therefore, using the following commands, re-enable Oracle RAC for c02orcl and restart all the database instances.

```
[oracle@c02n01 ~]$ sqlplus / as sysdba
SQL*Plus: Release 11.2.0.4.0 Production
...
SQL> alter system set cluster_database=true scope=spfile;
System altered.

SQL> shutdown abort
ORACLE instance shut down.

SQL> exit
Disconnected from Oracle Database 11g Enterprise Edition
...
[oracle@c02n01 ~]$ srvctl start database -d c02orcl
[oracle@c02n01 ~]$
```

Note: In your practice environment, you will use a preinstalled set of Oracle Database 12c binaries, which were previously installed by using Oracle Universal Installer to perform a software-only installation. This approach was taken so that you would not have to wait for the binaries to be installed. However, to use the pre-installed binaries in conjunction with DBUA, you must first run the root configuration script (root.sh) located in the Oracle Database 12c home directory.

11. In the same terminal session, run the root configuration script (root.sh) located in /u01/app/oracle/product/12.1.0/dbhome_1 on c02n01 and c02n02.

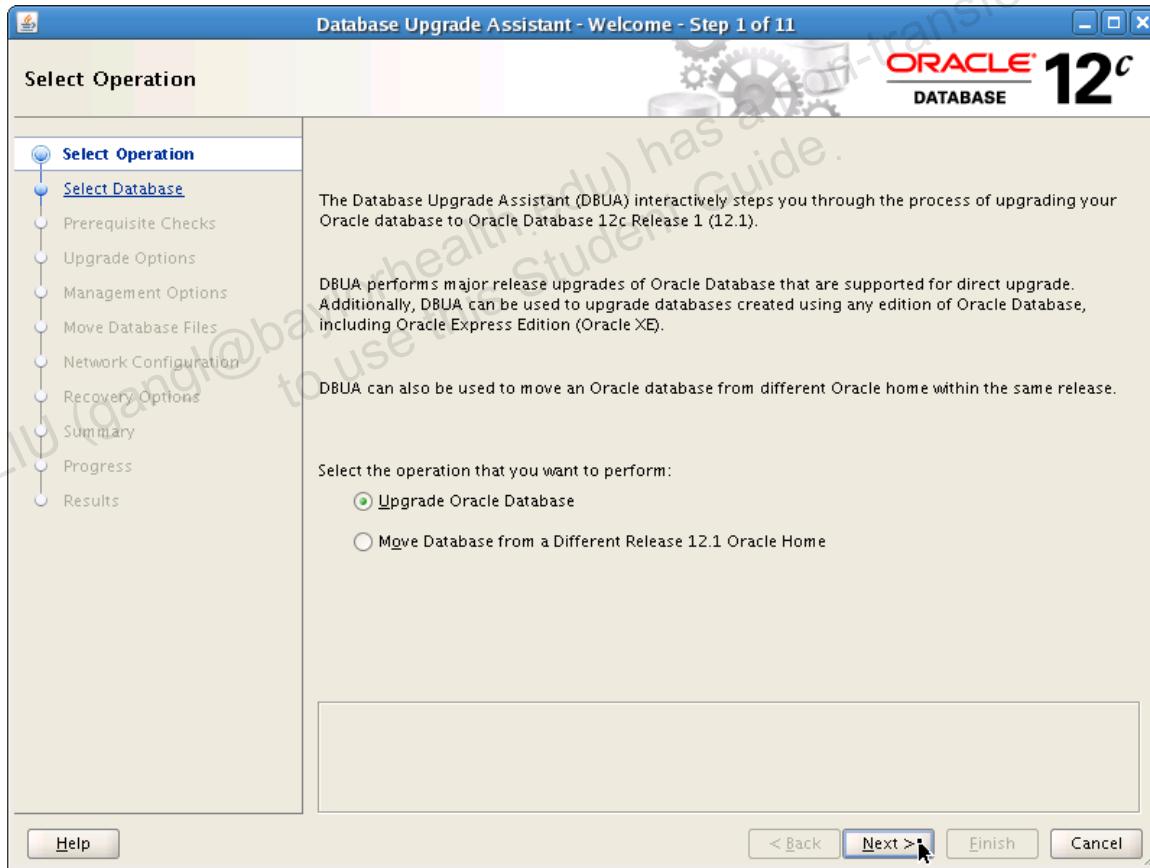
```
[oracle@c02n01 ~]$ su -
Password: <oracle>
[root@c02n01 ~]# /u01/app/oracle/product/12.1.0/dbhome_1/root.sh
Check
/u01/app/oracle/product/12.1.0/dbhome_1/install/root_c02n01.example.com
_2014-08-25_07-06-04.log for the output of root script
[root@c02n01 ~]# ssh c02n02
```

```
[root@c02n02 ~]# /u01/app/oracle/product/12.1.0/dbhome_1/root.sh
Check
/u01/app/oracle/product/12.1.0/dbhome_1/install/root_c02n02.example.com
_2014-08-25_07-06-34.log for the output of root script
[root@c02n02 ~]# exit
logout
Connection to c02n02 closed.
[root@c02n01 ~]# exit
logout
[oracle@c02n01 ~]$
```

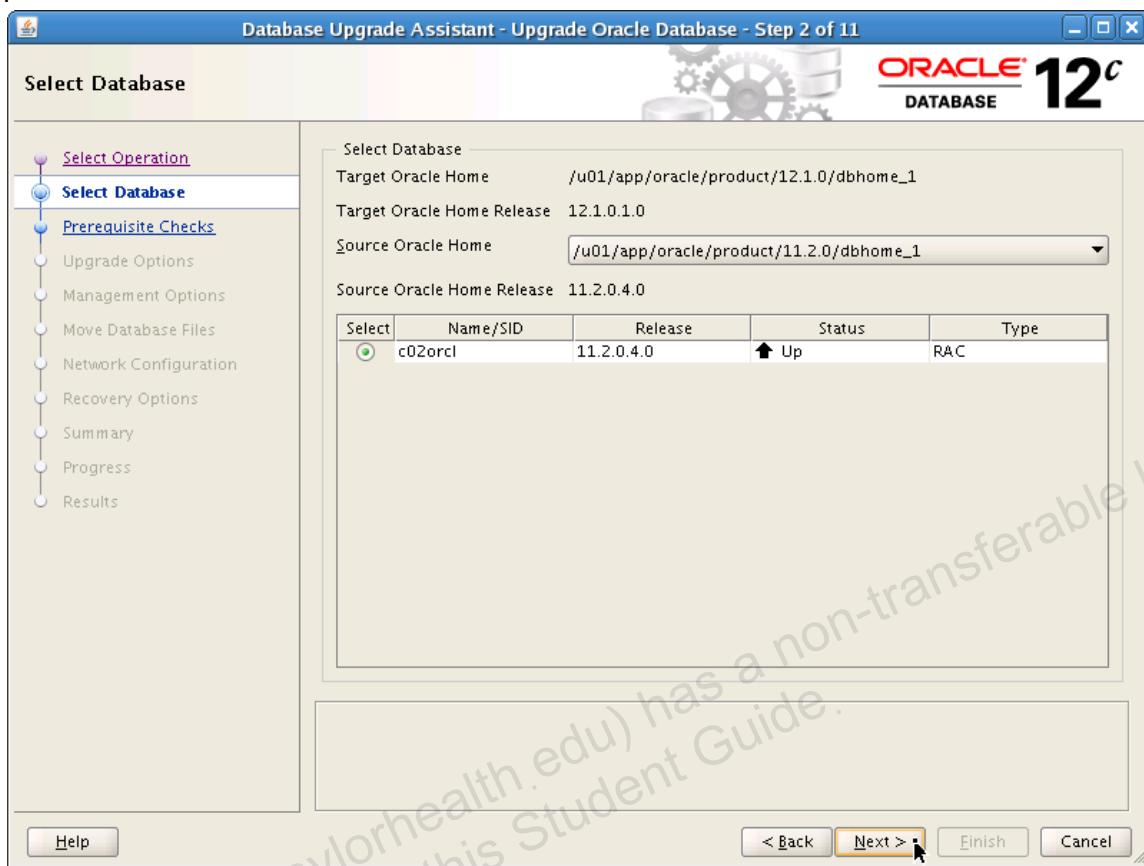
12. In the same terminal session, launch the Oracle Database 12c Database Upgrade Assistant (DBUA).

```
[oracle@c02n01~]$ /u01/app/oracle/product/12.1.0/dbhome_1/bin/dbua &
[1] 19471
[oracle@c02n01~]$
```

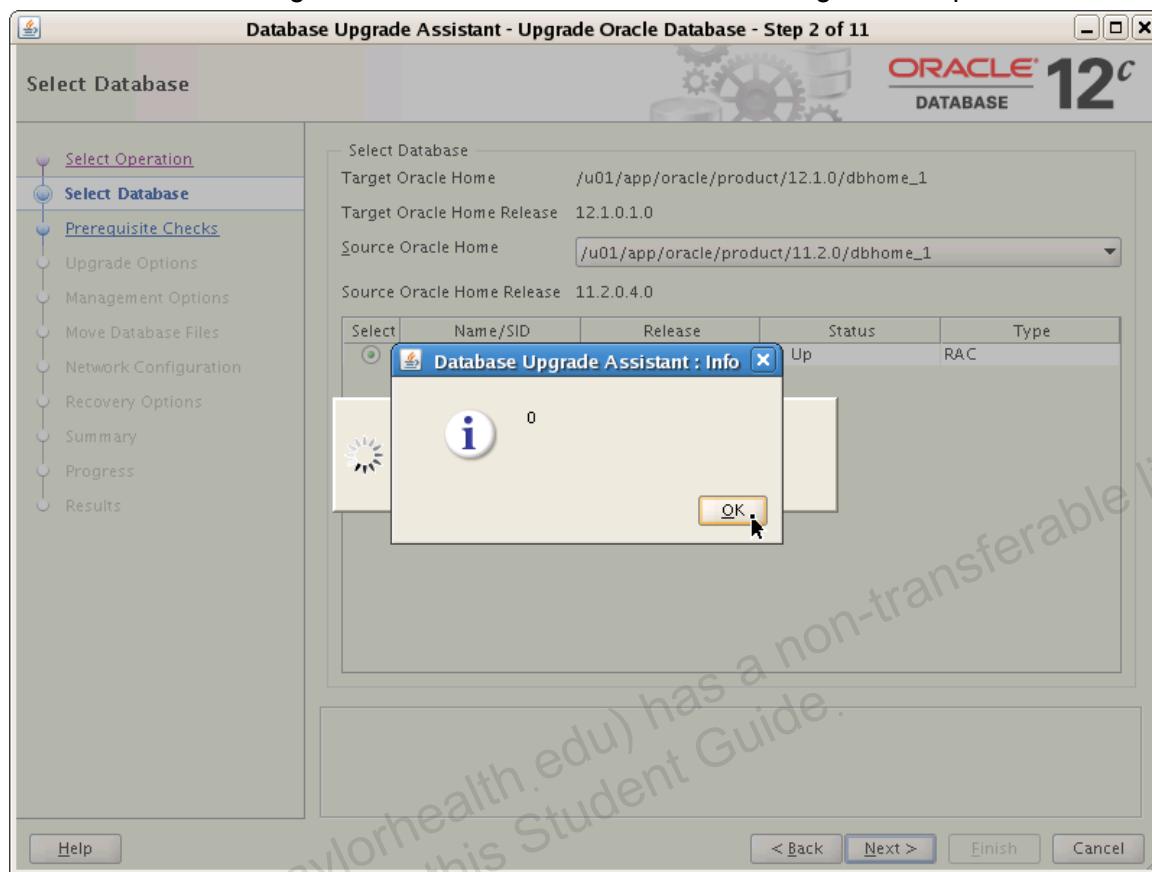
13. On the Select Operation page, ensure that the Upgrade Oracle Database option is selected and click Next to proceed.



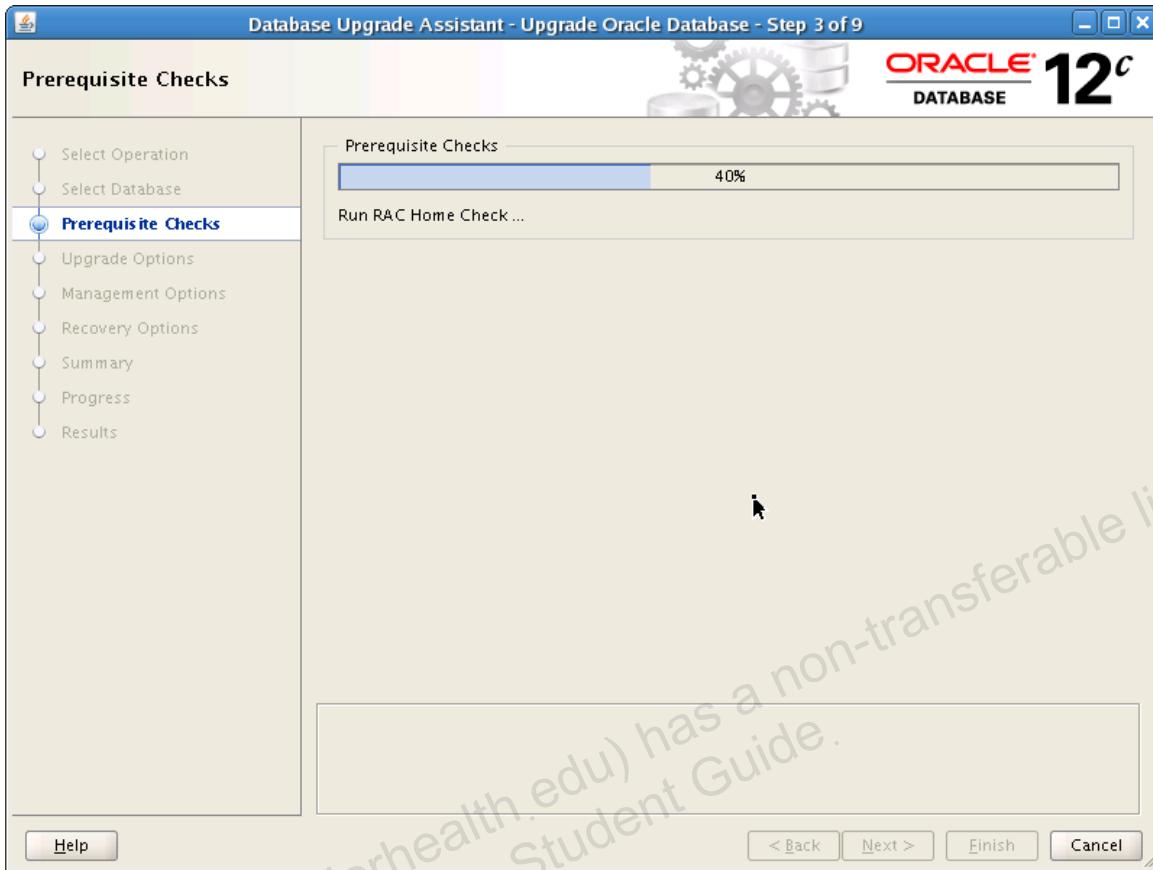
14. On the Select Database page, ensure that the c02orcl database is selected. Click Next to proceed.



15. Because of an undocumented feature of this DBUA version, you may see a dialog box like the one in the following screenshot. Click OK to close the dialog box and proceed.



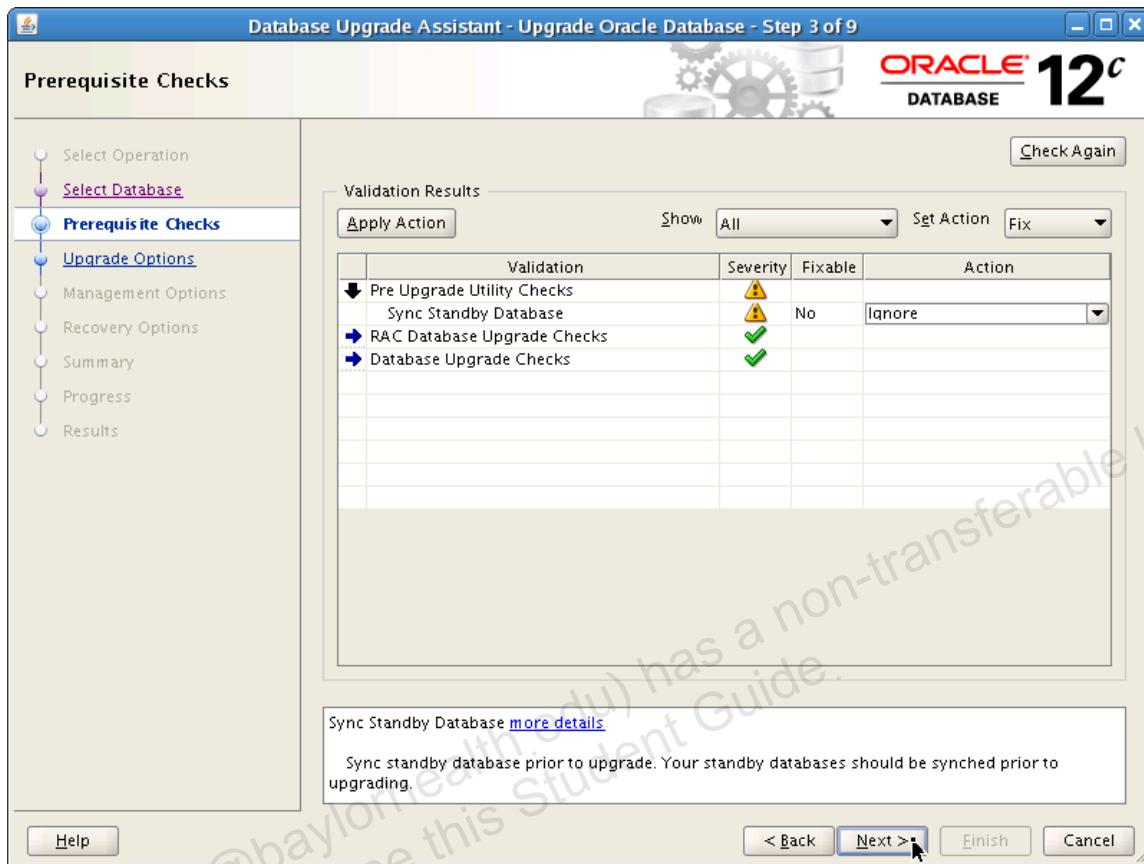
16. Wait while a series of checks are performed.



Notes

- The results of the prerequisite checks will vary depending on the composition of the database being upgraded. It is recommended that you run the DBUA prerequisite checks and understand their output as part of your upgrade planning.
- In your practice environment database, a warning will be generated indicating that your standby database should be synchronized before upgrading. However, you can safely ignore this warning because the `physru.sh` script has already prepared the database to be upgraded.

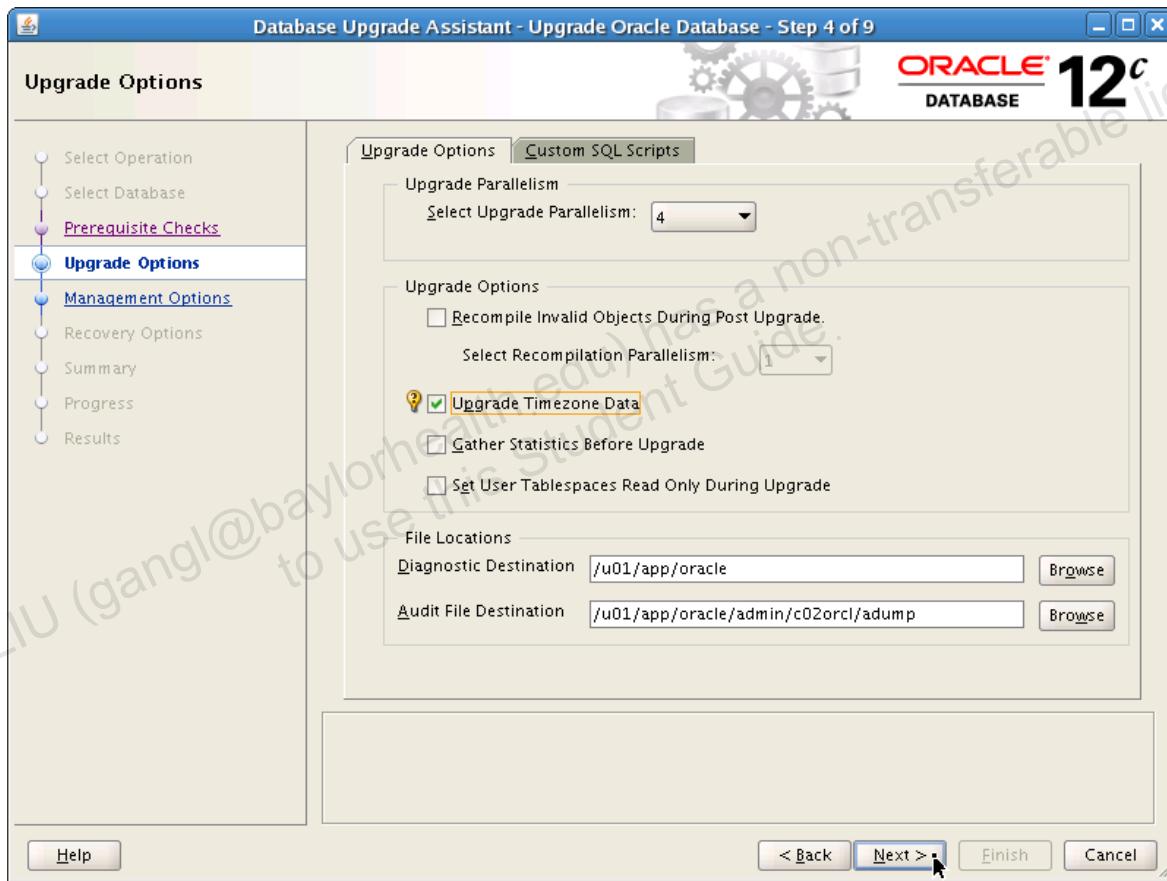
17. Click the arrow beside Pre Upgrade Utility Checks to reveal the Sync Standby Database warning. After you have confirmed that there are no other errors or warnings, click Next to proceed.



18. On the Upgrade Options page, deselect the Recompile Invalid Objects During Post Upgrade check box and select the Upgrade Timezone Data check box. Then click Next to proceed.

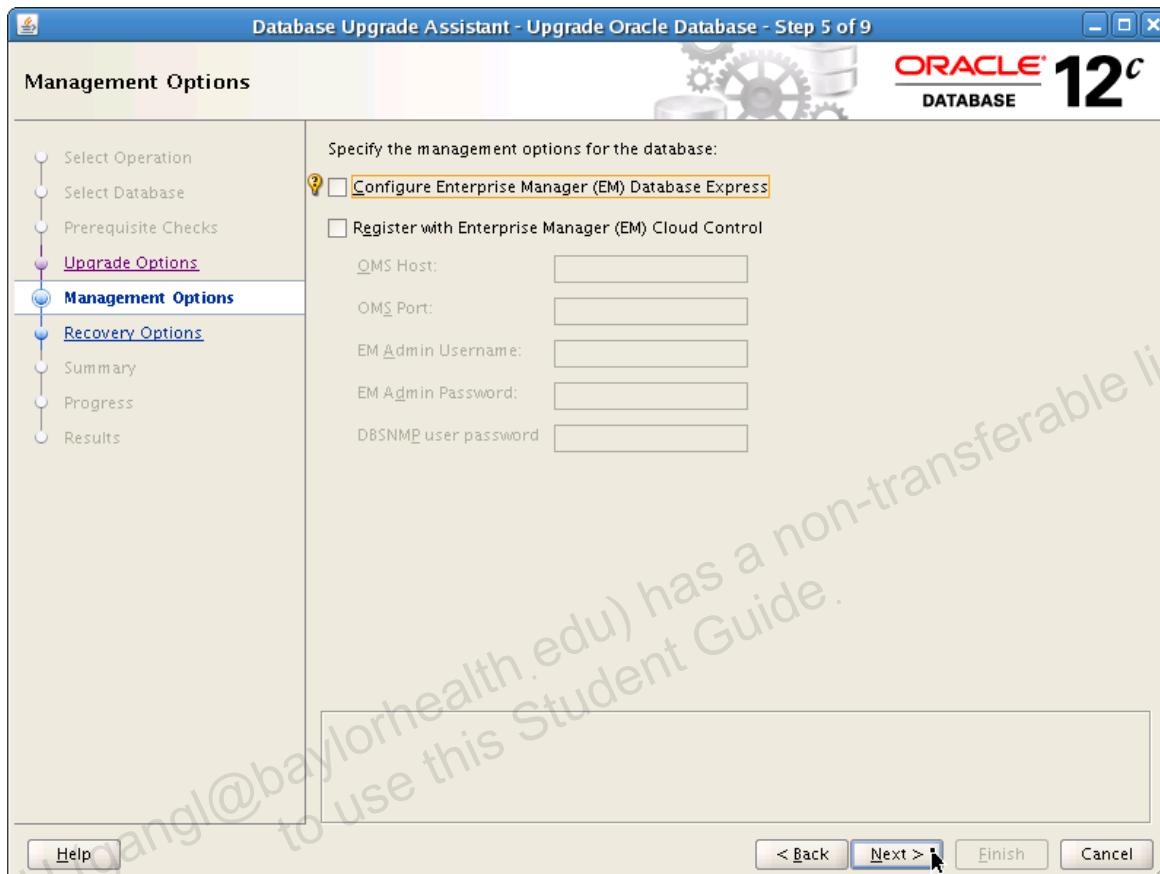
Notes

- It is generally recommended to allow DBUA to recompile invalid database objects during post upgrade processing. However, because this is an optional task, and because it can add approximately 30 minutes to the upgrade process in your practice environment, it is recommended to deselect this option for this practice exercise.
- Regarding the available DBUA options, you should consider the best options to use when performing your own upgrades.



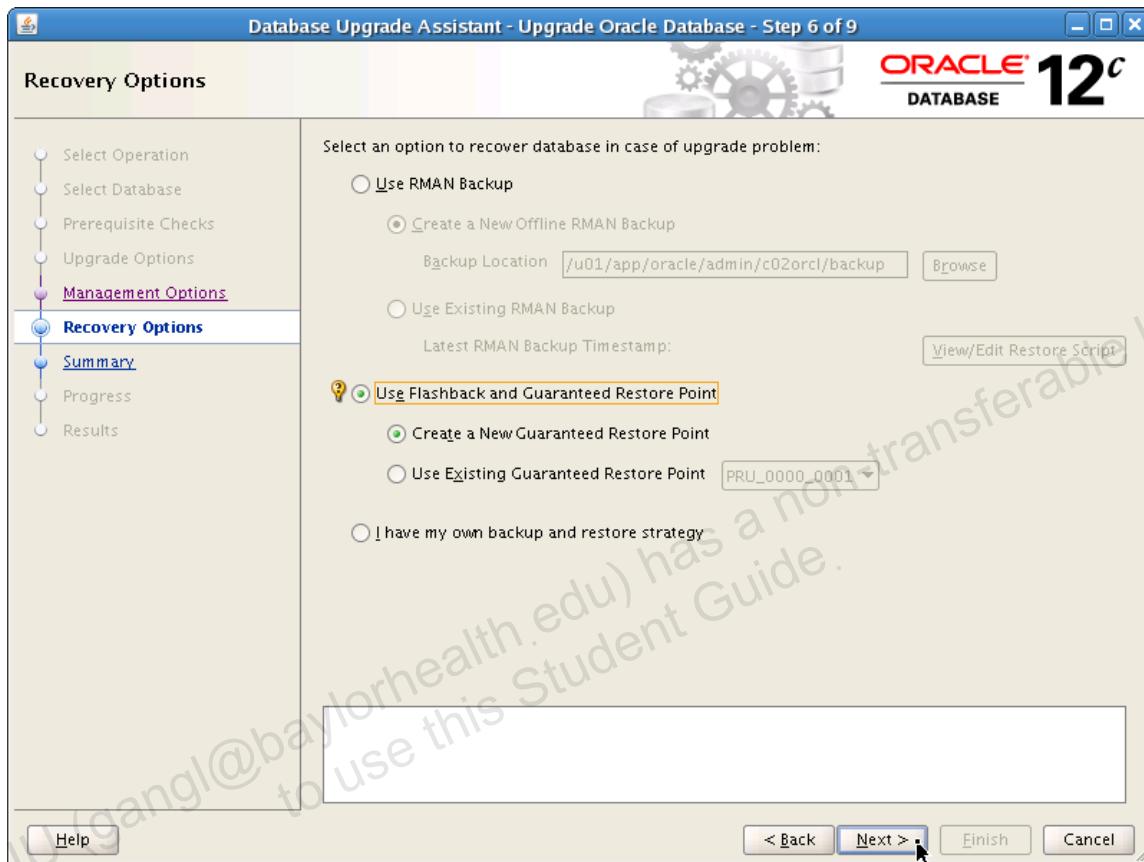
19. On the Management Options page, deselect the Configure Enterprise Manager (EM) Database Express check box. Then click Next to proceed.

Note: If you want to use Enterprise Manager Database Express in your upgraded Oracle Data Guard environment, it should be configured separately on each database after the rolling upgrade process is completed.

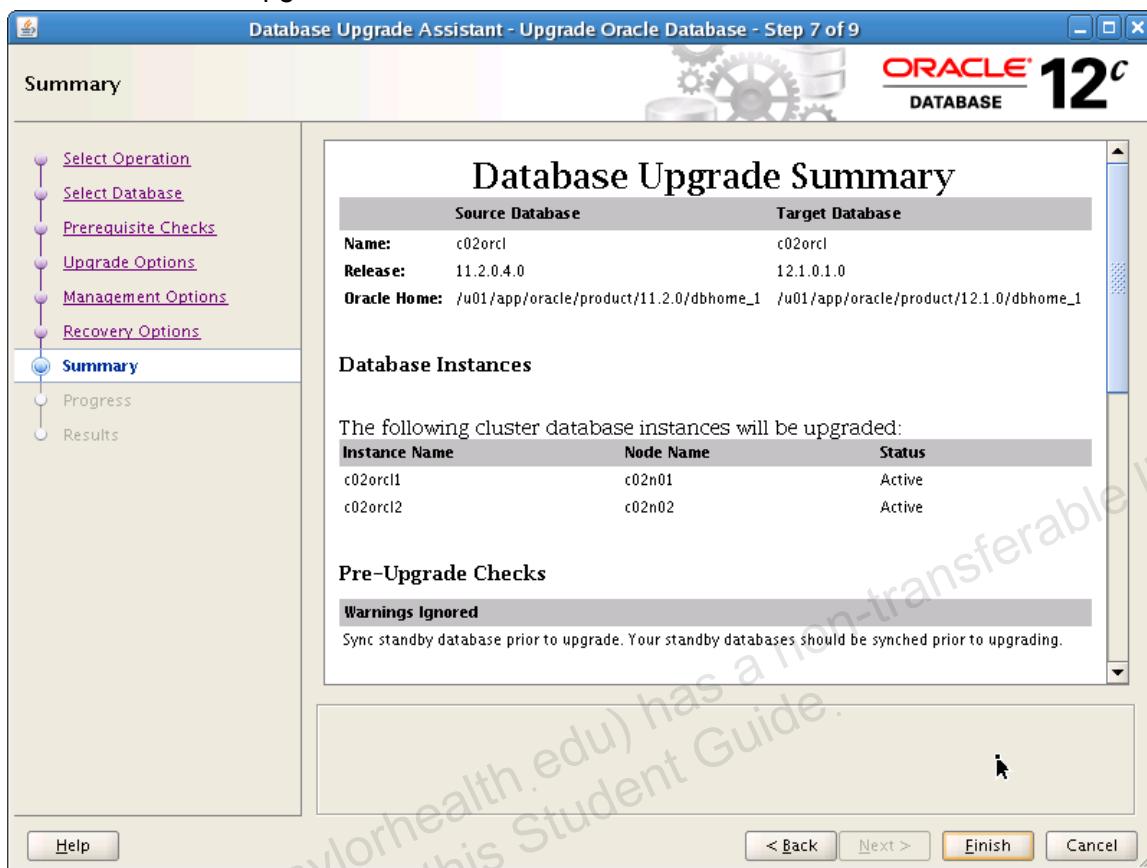


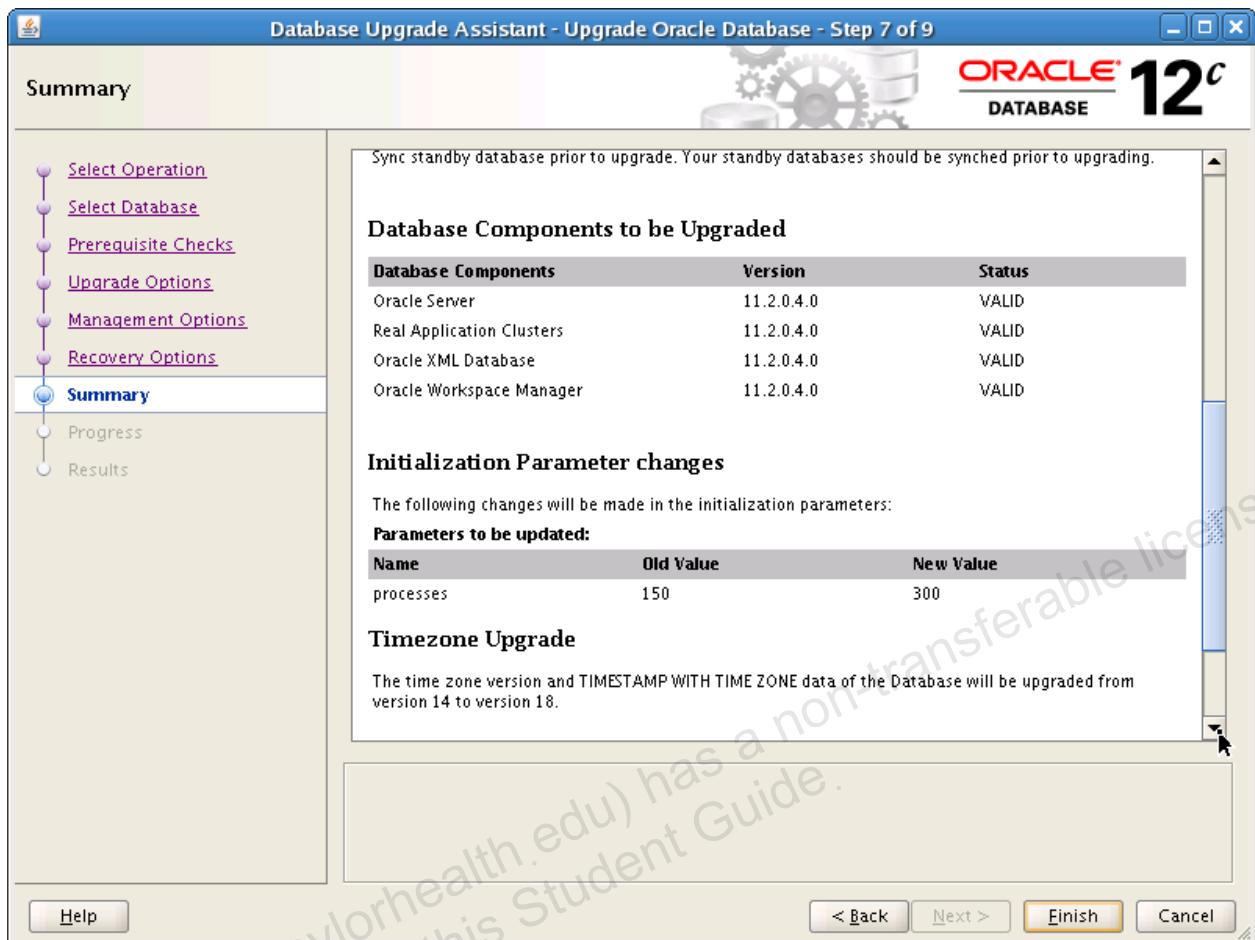
20. On the Recovery Options page, select the “Use Flashback and Guaranteed Restore Point” check box. Then select the “Create a New Guaranteed Restore Point” check box. Finally, click Next to proceed.

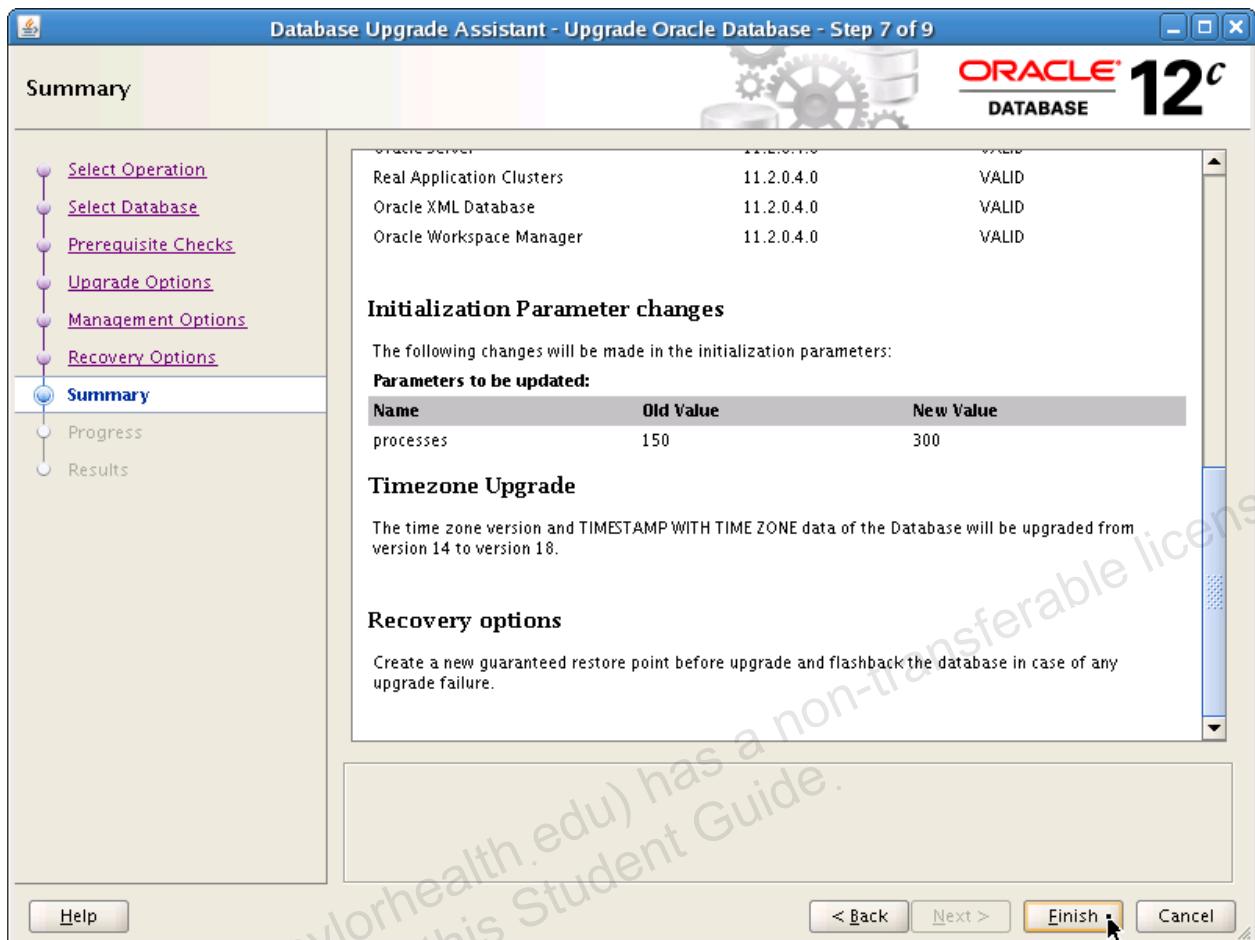
Note: Though not an absolute requirement, a guaranteed restore point provides a quick and easy mechanism to recover the database back to this point in case of a problem during the upgrade.



21. Investigate the information on the Summary page. After you are satisfied, click Finish to start the database upgrade.

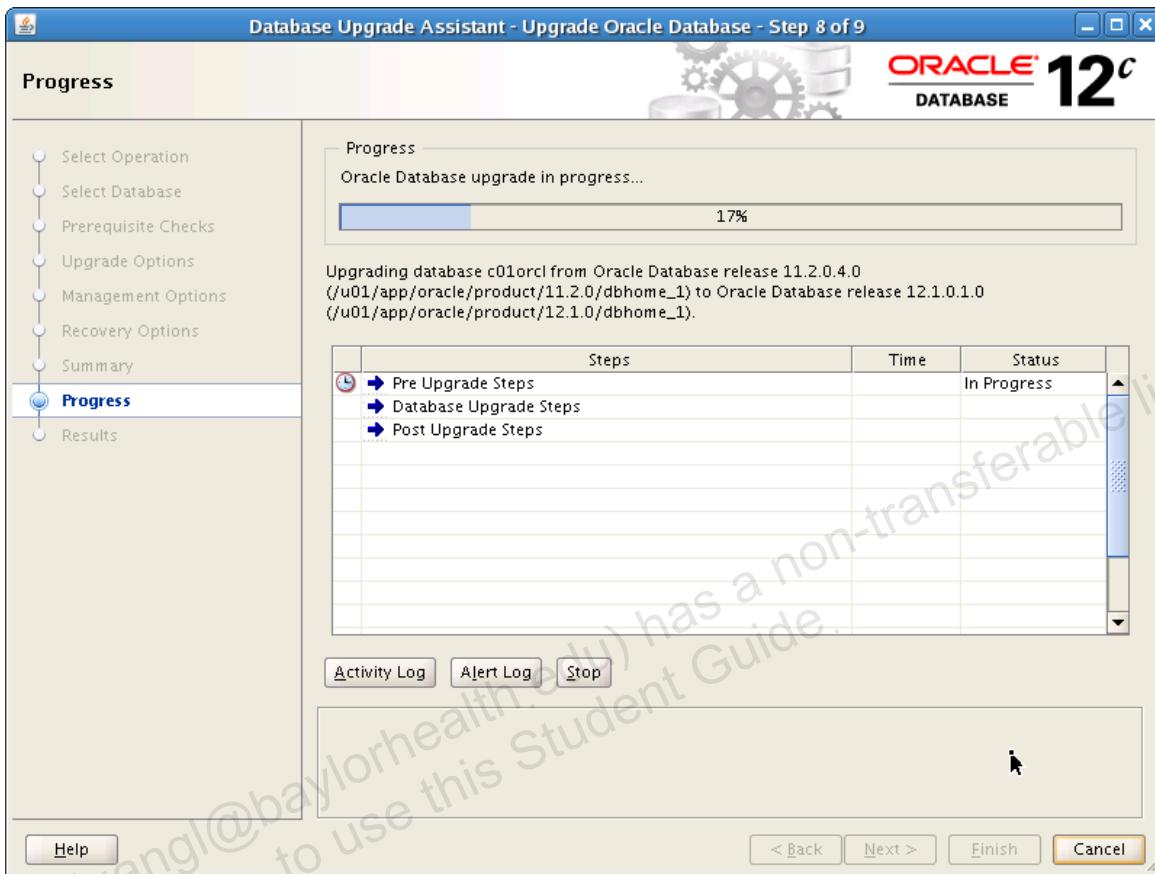




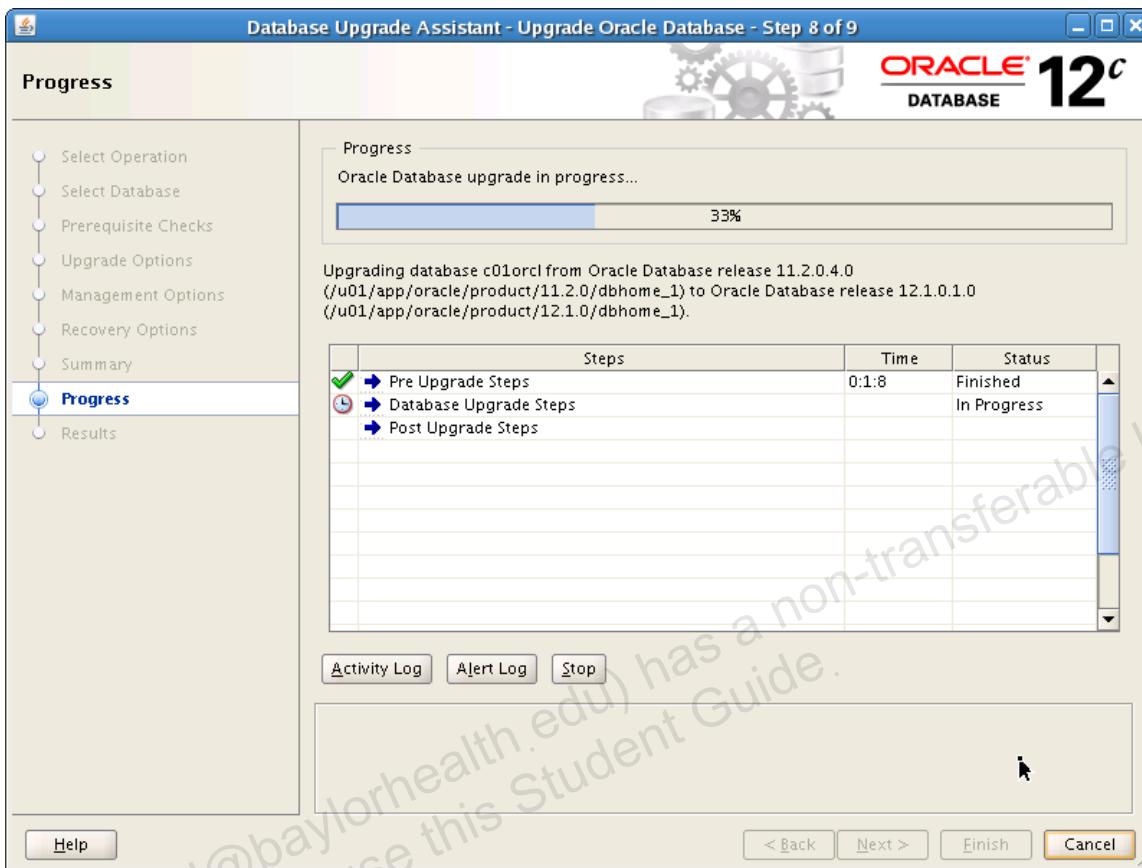


22. Monitor DBUA by viewing the Progress page.

Note: You can also monitor the DBUA session by viewing the log files at /u01/app/oracle/cfgtoollogs/dbua/c02orcl/upgrade1 on c02n01. You can also monitor the database alert log files for c02orcl.



Note: During the Pre Upgrade Steps, the database is shut down and only one instance is restarted to perform the upgrade.



23. **(Session 3)** Return to the terminal session on c01n01. While the standby database (c02orcl) is being upgraded, the primary database (c01orcl) remains open for business, and all the transactions executed on the primary database will be applied to the transient logical standby database after it is upgraded. To illustrate this, use your oracle@c01n01 terminal session to launch SQL*Plus and add some data to the hr.regions table. Later, after the rolling upgrade is finished, you will check to ensure that this database change has been persisted through the entire upgrade process.

```
[oracle@c01n01 ~]$ sqlplus hr/hr
SQL*Plus: Release 11.2.0.4.0 Production ...
SQL> select * from regions;
REGION_ID  REGION_NAME
-----
1 Europe
2 Americas
3 Asia
4 Middle East and Africa
SQL> insert into regions values (5,'Australia');
```

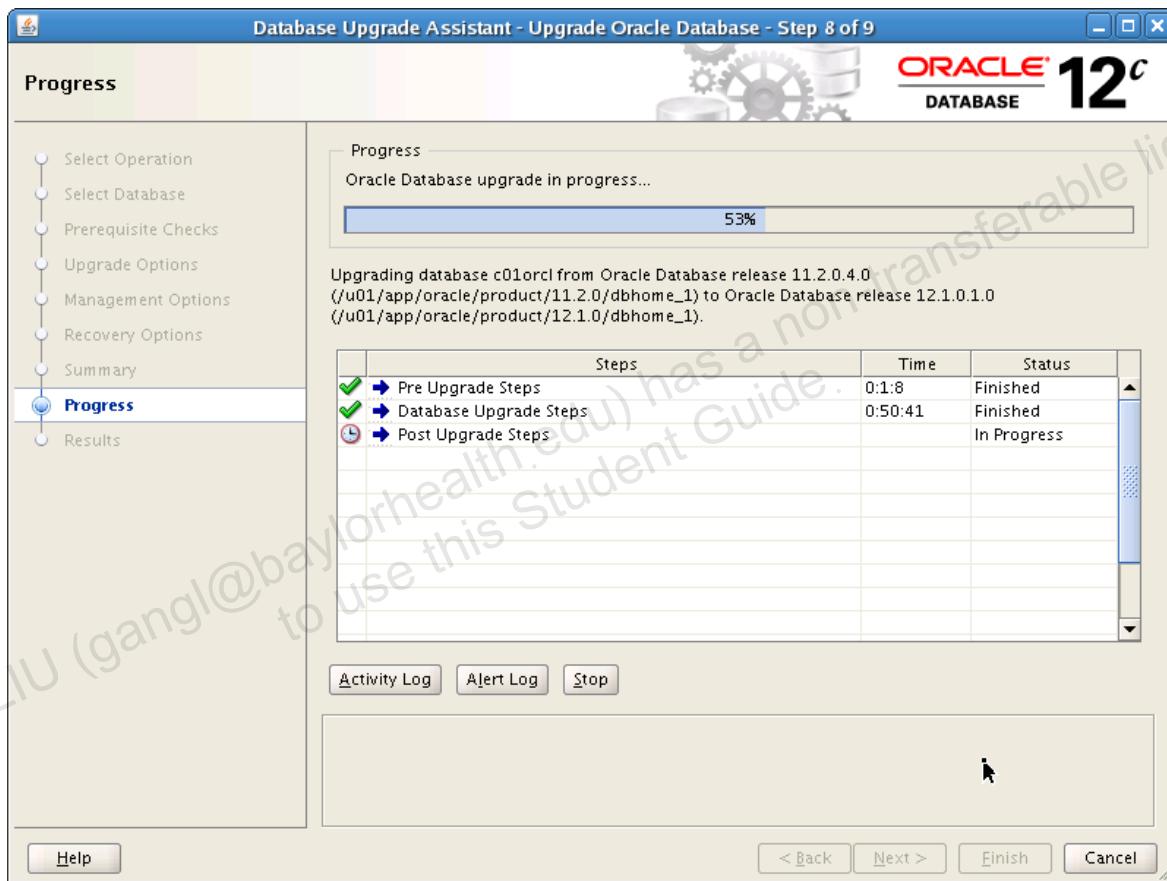
```
1 row created.

SQL> commit;

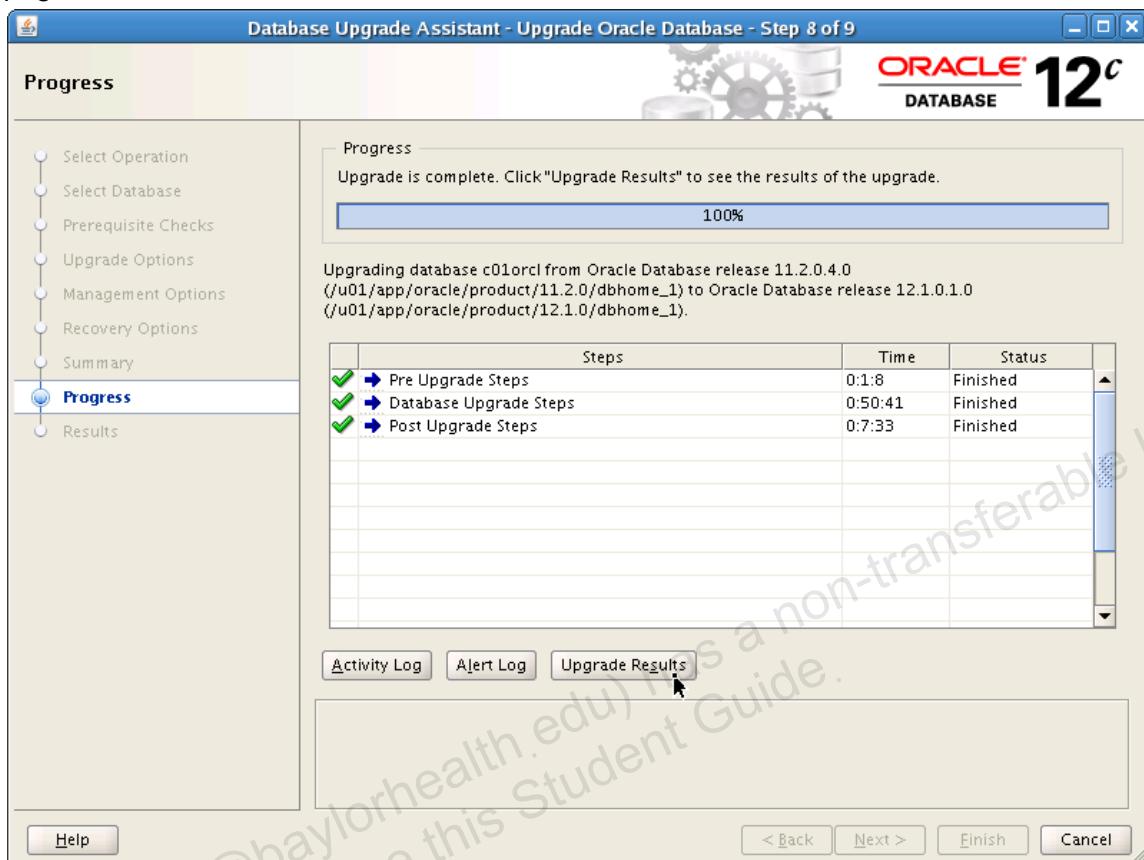
Commit complete.

SQL> exit
Disconnected from Oracle Database 11g Enterprise Edition ...
[oracle@c01n01 ~]$
```

Note: The DBUA Database Upgrade steps can take approximately 50 minutes to complete in your practice environment.

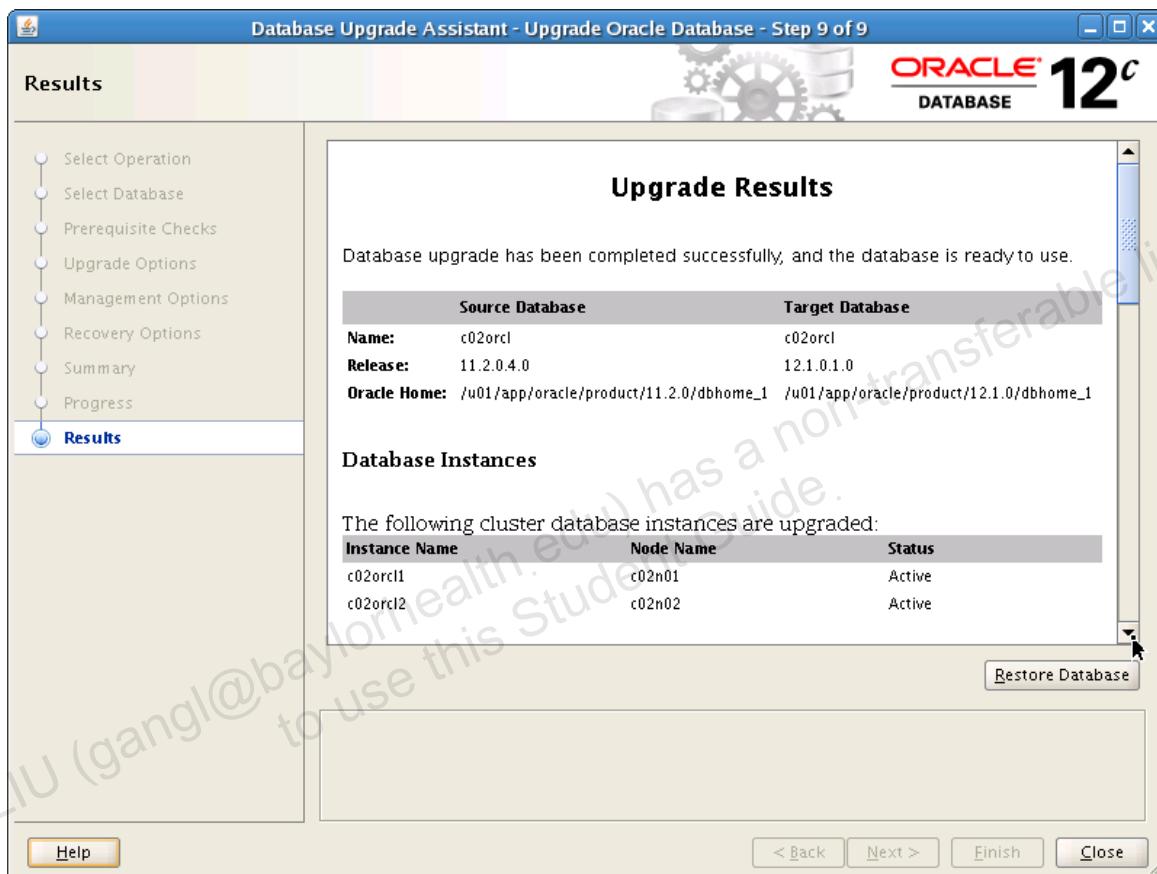


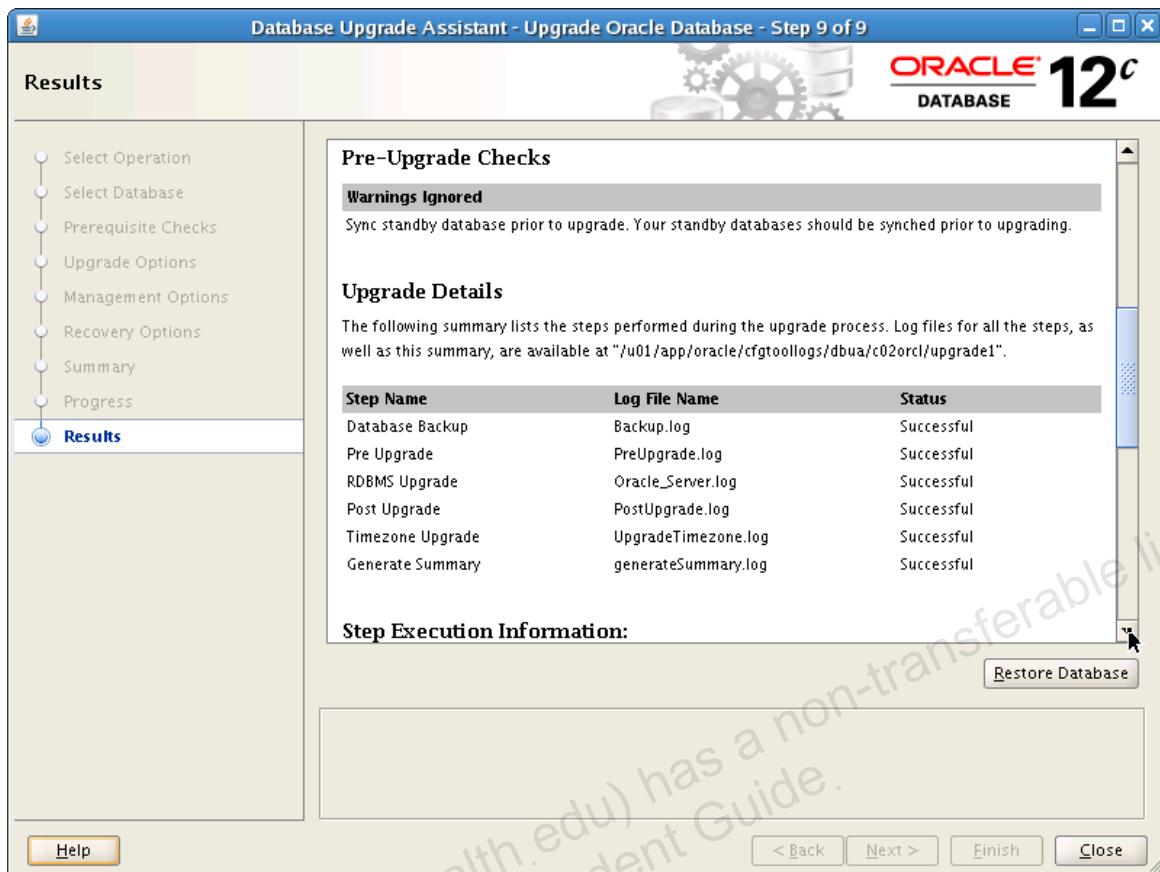
24. When the database upgrade is finished, you should see a page that looks similar to the following. Click Upgrade Results to leave the Progress page and view the detailed results page.

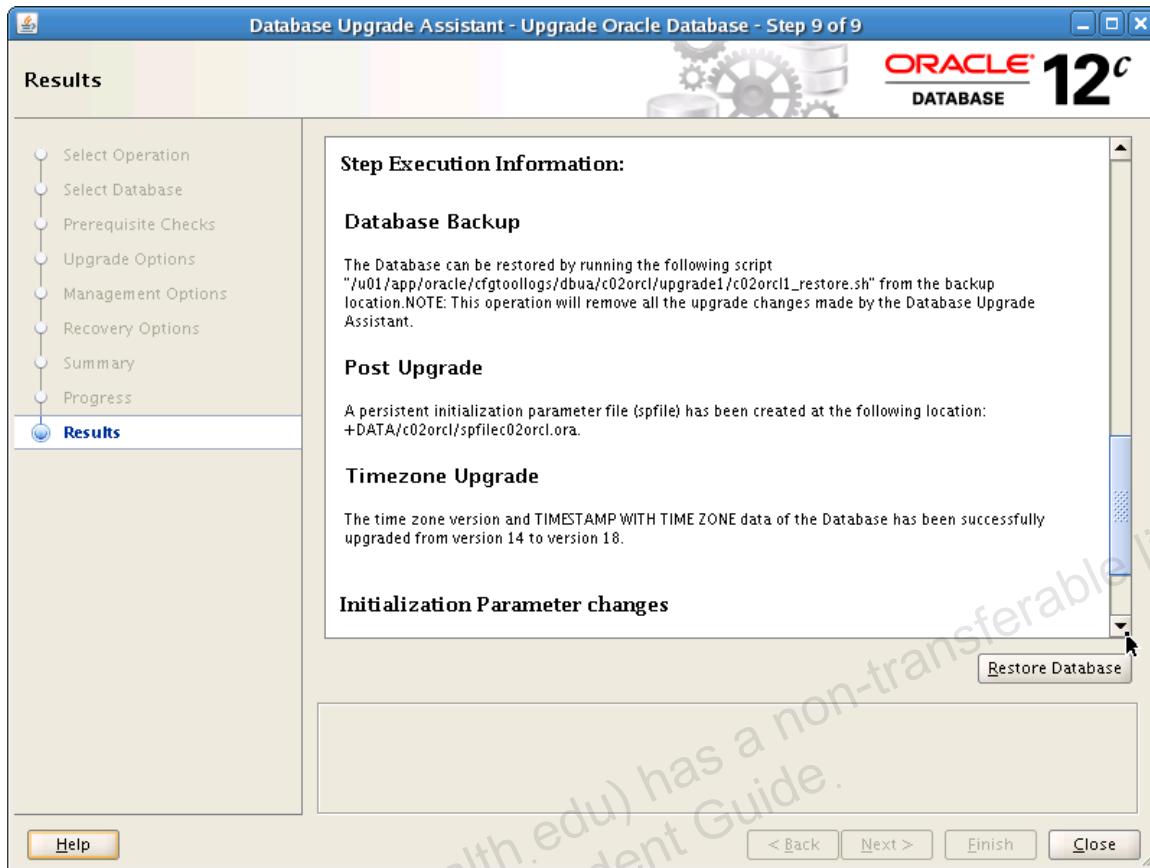


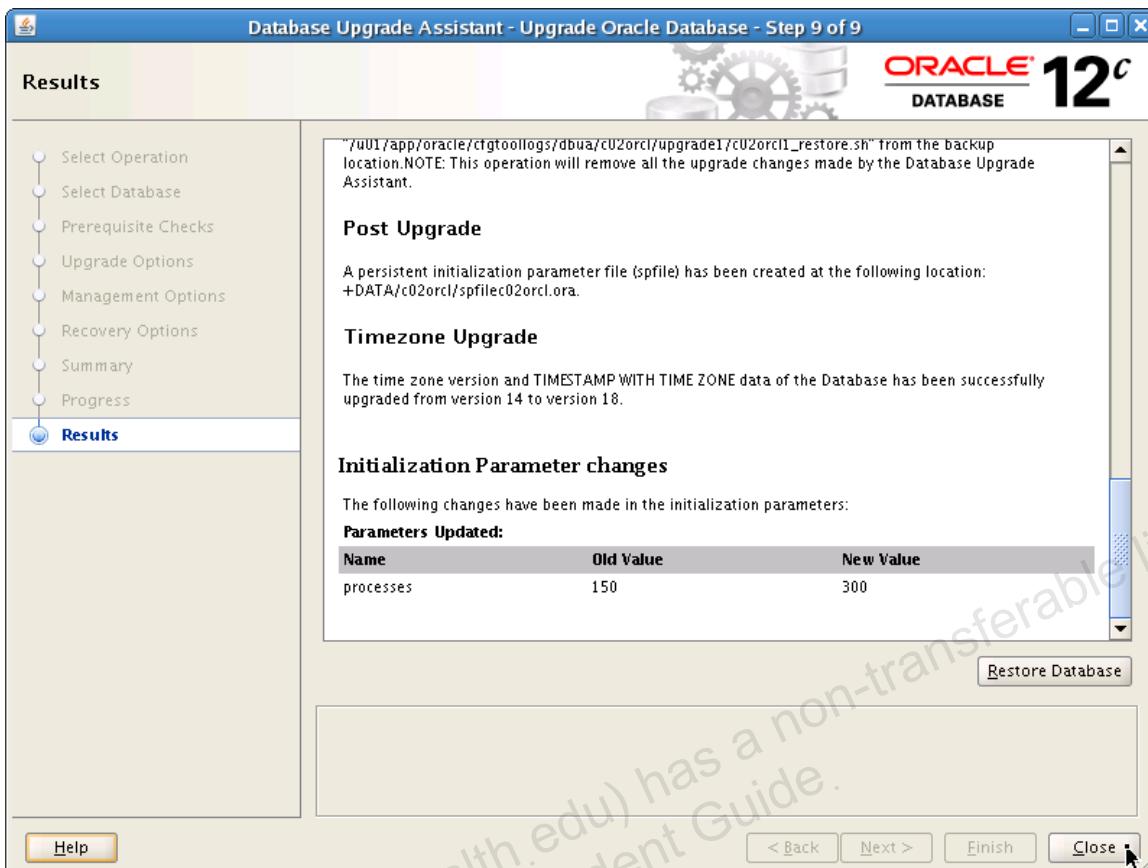
25. Investigate the information on the Results page. After you are satisfied, click Close to exit DBUA.

Note: The Results page may contain information regarding errors or warning generated during the database upgrade. Make sure that you note any information that requires further action. The results are also saved in a file for later examination, if required. The upgrade results for the upgrade you just performed can be found on c02n01 at /u01/app/oracle/cfgtoollogs/dbua/c02orcl/upgrade1/UpgradeResults.html.









26. **(Session 2)** Return to the terminal session on c02n01 to refresh the environment settings in your oracle@c02n01 terminal session to ensure that you use the upgraded Oracle Database 12c software from now on.

```
[oracle@c02n01 ~]$ . oraenv
ORACLE_SID = [c02orcl] ? c02orcl
The Oracle base remains unchanged with value /u01/app/oracle
[oracle@c02n01 ~]$ export ORACLE_SID=c02orcl1
[oracle@c02n01 ~]$
```

27. Check the configuration of the standby database (c02orcl) and confirm that it is now associated with the Oracle Database 12c binaries at /u01/app/oracle/product/12.1.0/dbhome_1.

```
[oracle@c02n01 ~]$ srvctl config database -d c02orcl -v
Database unique name: c02orcl
Database name:
Oracle home: /u01/app/oracle/product/12.1.0/dbhome_1
Oracle user: oracle
Spfile: +DATA/c02orcl/spfilec02orcl.ora
Password file:
Domain: example.com
Start options: open
Stop options: immediate
Database role: PHYSICAL_STANDBY
Management policy: AUTOMATIC
Server pools: c02orcl
Database instances: c02orcl1,c02orcl2
```

```
Disk Groups: DATA
Mount point paths:
Services:
Type: RAC
Start concurrency:
Stop concurrency:
Database is administrator managed
[oracle@c02n01 ~]$
```

28. Confirm that the upgraded standby database is running.

```
[oracle@c02n01 ~]$ srvctl status database -d c02orcl -v
Instance c02orcl1 is running on node c02n01. Instance status:
Open.
Instance c02orcl2 is running on node c02n02. Instance status:
Open.
[oracle@c02n01 ~]$
```

Note: At this point, the standby database (c02orcl) is upgraded; however, the Oracle Net configuration surrounding the database has not been updated. Before physru.sh is restarted, you must perform the following configuration tasks.

29. On every node in cluster02, copy the existing `tnsnames.ora` files from the current location under the Oracle Database 11g home directory into the corresponding location under the Oracle Database 12c home directory.

```
[oracle@c02n01 ~]$ cp
/u01/app/oracle/product/11.2.0/dbhome_1/network/admin/tnsnames.ora
/u01/app/oracle/product/12.1.0/dbhome_1/network/admin
[oracle@c02n01 ~]$ ssh c02n02 cp
/u01/app/oracle/product/11.2.0/dbhome_1/network/admin/tnsnames.ora
/u01/app/oracle/product/12.1.0/dbhome_1/network/admin
[oracle@c02n01 ~]$
```

30. **(Session 4)** Open a terminal window on c02n01 to connect as the grid user.

```
[Classroom PC]$ ssh grid@c02n01
grid@c02n01's password: <oracle>
[grid@c02n01 ~]$ . oraenv
ORACLE_SID = [grid] ? +ASM1
The Oracle base has been set to /u01/app/grid
[grid@c02n01 ~]$
```

31. View the Oracle Net Listener configuration file (`listener.ora`). Notice that the static registration entry associated with `c02orcl1` still references the Oracle Database 11g home directory.

```
[grid@c02n01 ~]$ cat /u01/app/12.1.0/grid/network/admin/listener.ora
LISTENER= (DESCRIPTION= (ADDRESS_LIST= (ADDRESS= (PROTOCOL=IPC) (KEY=LISTENER)))
# line added by Agent
LISTENER_SCAN3= (DESCRIPTION= (ADDRESS_LIST= (ADDRESS= (PROTOCOL=IPC) (KEY=LISTENER
_SCAN3))) # line added by Agent
LISTENER_SCAN2= (DESCRIPTION= (ADDRESS_LIST= (ADDRESS= (PROTOCOL=IPC) (KEY=LISTENER
_SCAN2))) # line added by Agent
```

```

LISTENER_SCAN1=(DESCRIPTION=(ADDRESS_LIST=(ADDRESS=(PROTOCOL=IPC) (KEY=LISTENER
_SCAN1))) # line added by Agent
ENABLE_GLOBAL_DYNAMIC_ENDPOINT_LISTENER_SCAN1=ON # line added
by Agent
ENABLE_GLOBAL_DYNAMIC_ENDPOINT_LISTENER_SCAN2=ON # line added
by Agent
ENABLE_GLOBAL_DYNAMIC_ENDPOINT_LISTENER_SCAN3=ON # line added
by Agent
SID_LIST_LISTENER=(SID_LIST=(SID_DESC=(GLOBAL_DBNAME=c02orcl_DGMGR
example.co
m) (ORACLE_HOME=/u01/app/oracle/product/11.2.0/dbhome_1) (SID_NAME=c02orc
11)))
VALID_NODE_CHECKING_REGISTRATION_LISTENER_SCAN1=OFF # line added
by Agent
VALID_NODE_CHECKING_REGISTRATION_LISTENER_SCAN2=OFF # line added
by Agent
VALID_NODE_CHECKING_REGISTRATION_LISTENER_SCAN3=OFF # line added
by Agent
ENABLE_GLOBAL_DYNAMIC_ENDPOINT_LISTENER=ON # line added by Agent
VALID_NODE_CHECKING_REGISTRATION_LISTENER=SUBNET # line added
by Agent

[grid@c02n01 ~] $

```

32. Use the following command to update the listener.ora file so that all references to 11.2 are replaced by 12.1.

```

[grid@c02n01 ~]$ sed -i s#11.2#12.1#g
/u01/app/12.1.0/grid/network/admin/listener.ora
[grid@c02n01 ~] $

```

33. View the Oracle Net Listener configuration file again and confirm that the static registration entry associated with c02orcl1 now references the Oracle Database 12c home directory.

```

[grid@c02n01 ~]$ cat /u01/app/12.1.0/grid/network/admin/listener.ora

LISTENER=(DESCRIPTION=(ADDRESS_LIST=(ADDRESS=(PROTOCOL=IPC) (KEY=LISTENER)))
# line added by Agent
LISTENER_SCAN3=(DESCRIPTION=(ADDRESS_LIST=(ADDRESS=(PROTOCOL=IPC) (KEY=LISTENER
_SCAN3))) # line added by Agent
LISTENER_SCAN2=(DESCRIPTION=(ADDRESS_LIST=(ADDRESS=(PROTOCOL=IPC) (KEY=LISTENER
_SCAN2))) # line added by Agent
LISTENER_SCAN1=(DESCRIPTION=(ADDRESS_LIST=(ADDRESS=(PROTOCOL=IPC) (KEY=LISTENER
_SCAN1))) # line added by Agent
ENABLE_GLOBAL_DYNAMIC_ENDPOINT_LISTENER_SCAN1=ON # line added
by Agent
ENABLE_GLOBAL_DYNAMIC_ENDPOINT_LISTENER_SCAN2=ON # line added
by Agent
ENABLE_GLOBAL_DYNAMIC_ENDPOINT_LISTENER_SCAN3=ON # line added
by Agent
SID_LIST_LISTENER=(SID_LIST=(SID_DESC=(GLOBAL_DBNAME=c02orcl_DGMGR
example.co
m) (ORACLE_HOME=/u01/app/oracle/product/12.1.0/dbhome_1) (SID_NAME=c02orc
11))

```

```
11)))
VALID_NODE_CHECKING_REGISTRATION_LISTENER_SCAN1=OFF # line added
by Agent
VALID_NODE_CHECKING_REGISTRATION_LISTENER_SCAN2=OFF # line added
by Agent
VALID_NODE_CHECKING_REGISTRATION_LISTENER_SCAN3=OFF # line added
by Agent
ENABLE_GLOBAL_DYNAMIC_ENDPOINT_LISTENER=ON # line added by Agent
VALID_NODE_CHECKING_REGISTRATION_LISTENER=SUBNET # line added
by Agent

[grid@c02n01 ~]$
```

34. Using the same method as before, modify the `listener.ora` file on c02n02.

```
[grid@c02n01 ~]$ ssh c02n02 sed -i s#11.2#12.1#g
/u01/app/12.1.0/grid/network/admin/listener.ora
[grid@c02n01 ~]$
```

35. Confirm the update that you made in the previous step.

```
[grid@c02n01 ~]$ ssh c02n02 cat
/u01/app/12.1.0/grid/network/admin/listener.ora

LISTENER=(DESCRIPTION=(ADDRESS_LIST=(ADDRESS=(PROTOCOL=IPC)(KEY=LISTENER)))
# line added by Agent
LISTENER_SCAN3=(DESCRIPTION=(ADDRESS_LIST=(ADDRESS=(PROTOCOL=IPC)(KEY=LISTENER
_SCAN3))) # line added by Agent
LISTENER_SCAN2=(DESCRIPTION=(ADDRESS_LIST=(ADDRESS=(PROTOCOL=IPC)(KEY=LISTENER
_SCAN2))) # line added by Agent
LISTENER_SCAN1=(DESCRIPTION=(ADDRESS_LIST=(ADDRESS=(PROTOCOL=IPC)(KEY=LISTENER
_SCAN1))) # line added by Agent
ENABLE_GLOBAL_DYNAMIC_ENDPOINT_LISTENER_SCAN1=ON # line added
by Agent
SID_LIST_LISTENER=(SID_LIST=(SID_DESC=(GLOBAL_DBNAME=c02orcl_DGMGRl.example.co
m)(ORACLE_HOME=/u01/app/oracle/product/12.1.0/dbhome_1)(SID_NAME=c02orc
l2)))
ENABLE_GLOBAL_DYNAMIC_ENDPOINT_LISTENER_SCAN2=ON # line added
by Agent
ENABLE_GLOBAL_DYNAMIC_ENDPOINT_LISTENER_SCAN3=ON # line added
by Agent
VALID_NODE_CHECKING_REGISTRATION_LISTENER_SCAN1=OFF # line added
by Agent
ENABLE_GLOBAL_DYNAMIC_ENDPOINT_LISTENER=ON # line added by Agent
VALID_NODE_CHECKING_REGISTRATION_LISTENER=SUBNET # line added
by Agent

[grid@c02n01 ~]$
```

36. In the same terminal session, stop and restart the Oracle Net listener across cluster02.

```
[grid@c02n01 ~]$ srvctl stop listener
[grid@c02n01 ~]$ srvctl start listener
[grid@c02n01 ~]$
```

37. (**Session 1**) Return to the terminal session on c01n01 to restart physru.sh as shown below to continue the database rolling upgrade process. Enter oracle_4U when you are prompted for the sysdba password.

```
[oracle@c01n01 ~]$ ./physru.sh sys
c01n01:1521/c01orcl_DGMGRL.example.com
c02n01:1521/c02orcl_DGMGRL.example.com c01orcl c02orcl 12.1.0.1.0

Please enter the sysdba password:
<oracle_4U>

### Initialize script to either start over or resume execution

Aug 25 08:35:23 2014 [0-1] Identifying rdbms software version
Aug 25 08:35:23 2014 [0-1] database c01orcl is at version 11.2.0.4.0
Aug 25 08:35:23 2014 [0-1] database c02orcl is at version 12.1.0.1.0
Aug 25 08:35:24 2014 [0-1] verifying flashback database is enabled at
c01orcl
and c02orcl
Aug 25 08:35:25 2014 [0-1] verifying available flashback restore points
Aug 25 08:35:26 2014 [0-1] verifying DG Broker is disabled
Aug 25 08:35:26 2014 [0-1] looking up prior execution history
Aug 25 08:35:27 2014 [0-1] last completed stage [2-4] using script version
0001
Aug 25 08:35:27 2014 [0-1] resuming execution of script
```

Note: This script detects that the last completed stage was Stage 2-4 and resumes at the beginning of Stage 3.

```
### Stage 3: Validate upgraded transient logical standby
```

```
Aug 25 08:35:27 2014 [3-1] database c02orcl is no longer in OPEN MIGRATE
mode
Aug 25 08:35:27 2014 [3-1] database c02orcl is at version 12.1.0.1.0
```

Note: Stage 3 performs validations to ensure that the standby database (c02orcl) is upgraded.

```
### Stage 4: Switch the transient logical standby to be the new primary
```

```
Aug 25 08:35:29 2014 [4-1] waiting for c02orcl to catch up (this could take
a
while)
Aug 25 08:35:29 2014 [4-1] starting logical standby on database c02orcl
Aug 25 08:36:02 2014 [4-1] waiting for apply lag to fall under 30 seconds
Aug 25 08:37:09 2014 [4-1] apply lag measured at 66 seconds
Aug 25 08:37:19 2014 [4-1] apply lag measured at 10 seconds
Aug 25 08:37:20 2014 [4-2] switching c01orcl to become a logical standby
Aug 25 08:38:02 2014 [4-2] c01orcl is now a logical standby
Aug 25 08:38:03 2014 [4-3] waiting for standby c02orcl to process end-of-
redo
from primary
```

```
Aug 25 08:38:05 2014 [4-4] switching c02orcl to become the new primary
Aug 25 08:38:55 2014 [4-4] c02orcl is now the new primary
```

Note: In Stage 4, the transient logical standby database is synchronized with the primary database. During this phase, the standby is updated with the transactions that were captured on the primary database, while the standby database was being upgraded. At the end of Stage 4, the databases switch roles, leaving the updated and upgraded former standby database (c02orcl) as the new primary database. The former primary database (c01orcl) now assumes the standby database role.

```
### Stage 5: Flashback former primary to pre-upgrade restore point and
convert to physical
```

```
Aug 25 08:39:06 2014 [5-1] verifying instance c01orcl1 is the only active
instance
WARN: c01orcl is a RAC database. Before this script can continue, you
must manually reduce the RAC to a single instance. This can be
accomplished with the following step:
1) Shutdown all instances other than instance c01orcl1.
eg: srvctl stop instance -d c01orcl -i c01orcl2 -o abort
Once these steps have been performed, enter 'y' to continue the script.
If desired, you may enter 'n' to exit the script to perform the required
steps, and recall the script to resume from this point.
```

Are you ready to continue? (y/n) :

Note: In Stage 5, the former primary database (c01orcl) flashed back to the pre-upgrade restore point so that it can be synchronized with the updated and upgraded former standby database (c02orcl), which is now a fully functioning primary database. However, before this can occur, c01orcl must be reduced to one instance.

38. (**Session 3**) Return to the terminal session on c01n01 connected as the oracle user. Then shut down all instances of c01orcl except for c01orcl1; that is, shut down c01orcl2.

```
[oracle@c01n01 ~]$ srvctl stop instance -d c01orcl -i c01orcl2 -o abort
[oracle@c01n01 ~]$
```

39. (**Session 1**) Return to your physru.sh session and enter y to continue.

Are you ready to continue? (y/n) : **y**

```
Aug 25 08:41:29 2014 [5-1] continuing
Aug 25 08:41:29 2014 [5-1] verifying instance c01orcl1 is the only active
instance
Aug 25 08:41:29 2014 [5-1] shutting down database c01orcl
Aug 25 08:41:49 2014 [5-1] mounting database c01orcl
Aug 25 08:42:11 2014 [5-2] flashing back database c01orcl to restore point
PRU_0000_0001
Aug 25 08:42:20 2014 [5-3] converting c01orcl into physical standby
Aug 25 08:42:22 2014 [5-4] shutting down database c01orcl
```

NOTE: Database c01orcl has been shutdown, and is now ready to be started
using the newer version Oracle binary. This script requires the
database
to be mounted (on all active instances, if RAC) before calling this

```
script to resume the rolling upgrade.
```

NOTE: Database c01orcl is no longer limited to single instance operation since the database has been successfully converted into a physical standby. For increased availability, Oracle recommends starting all instances in the RAC on the newer binary by performing the following step:

- 1) Startup and mount all instances for database c01orcl
eg: srvctl start database -d c01orcl -o mount

```
[oracle@c01n01 ~]$
```

Note: At this point, the former primary database (c01orcl) has been flashed back to the preupgrade restore point. It has also been converted from a transient logical standby database to a physical standby database. Next, you will manually reconfigure cluster01 so that c01orcl can be opened using Oracle Database 12c binaries, which have been preinstalled on the servers.

40. **(Session 1)** On every node in cluster01, copy the existing Oracle Database password files from the current location under the Oracle Database 11g home directory into the corresponding location under the Oracle Database 12c home directory.

```
[oracle@c01n01 ~]$ cp
/u01/app/oracle/product/11.2.0/dbhome_1/dbs/orapwc01orcl1
/u01/app/oracle/product/12.1.0/dbhome_1/dbs
[oracle@c01n01 ~]$ ssh c01n02 cp
/u01/app/oracle/product/11.2.0/dbhome_1/dbs/orapwc01orcl2
/u01/app/oracle/product/12.1.0/dbhome_1/dbs
```

41. **(Session 1)** On every node in cluster01, copy the existing database initialization files (init.ora files) from the current location under the Oracle Database 11g home directory into the corresponding location under the Oracle Database 12c home directory.

```
[oracle@c01n01 ~]$ cp
/u01/app/oracle/product/11.2.0/dbhome_1/dbs/initc01orcl1.ora
/u01/app/oracle/product/12.1.0/dbhome_1/dbs
[oracle@c01n01 ~]$ ssh c01n02 cp
/u01/app/oracle/product/11.2.0/dbhome_1/dbs/initc01orcl2.ora
/u01/app/oracle/product/12.1.0/dbhome_1/dbs
[oracle@c01n01 ~]$
```

42. **(Session 1)** On every node in cluster01, copy the existing tnsnames.ora files from the current location in the Oracle Database 11g home directory into the corresponding location in the Oracle Database 12c home directory.

```
[oracle@c01n01 ~]$ cp
/u01/app/oracle/product/11.2.0/dbhome_1/network/admin/tnsnames.o
ra /u01/app/oracle/product/12.1.0/dbhome_1/network/admin
[oracle@c01n01 ~]$ ssh c01n02 cp
/u01/app/oracle/product/11.2.0/dbhome_1/network/admin/tnsnames.o
ra /u01/app/oracle/product/12.1.0/dbhome_1/network/admin
[oracle@c01n01 ~]$
```

43. **(Session 1)** On every node in cluster01, execute the `setasmgid` utility so that Oracle Database 12c is able to access ASM.

```
[oracle@c01n01 ~]$ su -
Password: <oracle>
[root@c01n01 ~]# . oraenv
ORACLE_SID = [root] ? +ASM1
The Oracle base has been set to /u01/app/grid
[root@c01n01 ~]# setasmgid
o=/u01/app/oracle/product/12.1.0/dbhome_1/bin/oracle
[root@c01n01 ~]# ssh c01n02
[root@c01n02 ~]# . oraenv
ORACLE_SID = [root] ? +ASM2
The Oracle base has been set to /u01/app/grid
[root@c01n02 ~]# setasmgid
o=/u01/app/oracle/product/12.1.0/dbhome_1/bin/oracle
[root@c01n02 ~]# exit
logout
Connection to c01n02 closed.
[root@c01n01 ~]# exit
logout
[oracle@c01n01 ~]$
```

44. Remove the clusterware service entry for the `c01orcl` database.

```
[oracle@c01n01 ~]$ srvctl remove database -d c01orcl
Remove the database c01orcl? (y/[n]) y
[oracle@c01n01 ~]$
```

45. Refresh the environment settings in your `oracle@c01n01` terminal session to ensure that you use the upgraded Oracle Database 12c software from now on, as shown in the following:

```
[oracle@c01n01 ~]$ . oraenv
ORACLE_SID = [c01orcl1] ? c01orcl1
ORACLE_HOME = [/home/oracle] ?
/u01/app/oracle/product/12.1.0/dbhome_1
The Oracle base remains unchanged with value /u01/app/oracle
[oracle@c01n01 ~]$
```

46. Re-create the clusterware service entry for the `c01orcl` database and its instances.

```
[oracle@c01n01 ~]$ srvctl add database -d c01orcl -o
/u01/app/oracle/product/12.1.0/dbhome_1
[oracle@c01n01 ~]$ srvctl add instance -d c01orcl -i c01orcl1 -n
c01n01
[oracle@c01n01 ~]$ srvctl add instance -d c01orcl -i c01orcl2 -n
c01n02
[oracle@c01n01 ~]$
[oracle@c01n01 ~]$ srvctl start database -d c01orcl -o mount
[oracle@c01n01 ~]$
```

47. **(Session 5)** Open a new terminal window on `c01n01` as `grid` user.

```
[Classroom PC]$ ssh grid@c01n01
grid@c01n01's password: <oracle>
[grid@c01n01 ~]$ . oraenv
ORACLE_SID = [grid] ? +ASM1
The Oracle base has been set to /u01/app/grid
[grid@c01n01 ~]$
```

48. Modify the Oracle Net Listener configuration file (`listener.ora`) on every node in `cluster01` so that the static registration entries associated with `c01orcl` are updated to reference the Oracle Database 12c home directory.

```
[grid@c01n01 ~]$ sed -i s#11.2#12.1#g
/u01/app/12.1.0/grid/network/admin/listener.ora
[grid@c01n01 ~]$ ssh c01n02 sed -i s#11.2#12.1#g
/u01/app/12.1.0/grid/network/admin/listener.ora
[grid@c01n01 ~]$
```

49. Stop and restart the Oracle Net listener across `cluster01`.

```
[grid@c01n01 ~]$ srvctl stop listener
[grid@c01n01 ~]$ srvctl start listener
[grid@c01n01 ~]$
```

50. **(Session 1)** Return to the terminal session on `c01n01` to restart `physru.sh` as shown below to continue the database rolling upgrade process. Enter `oracle_4U` when you are prompted for the `sysdba` password.

```
[oracle@c01n01 ~]$ ./physru.sh sys
c01n01:1521/c01orcl_DGMGRL.example.com
c02n01:1521/c02orcl_DGMGRL.example.com c01orcl c02orcl 12.1.0.1.0

Please enter the sysdba password:
<oracle_4U>

### Initialize script to either start over or resume execution

Aug 25 08:49:05 2014 [0-1] Identifying rdbms software version
Aug 25 08:50:51 2014 [0-1] database c01orcl is at version 12.1.0.1.0
Aug 25 08:50:53 2014 [0-1] database c02orcl is at version 12.1.0.1.0
Aug 25 08:51:02 2014 [0-1] verifying flashback database is enabled at
c01orcl
and c02orcl
Aug 25 08:51:03 2014 [0-1] verifying available flashback restore points
Aug 25 08:51:15 2014 [0-1] verifying DG Broker is disabled
Aug 25 08:51:20 2014 [0-1] looking up prior execution history
Aug 25 08:51:28 2014 [0-1] last completed stage [5-4] using script version
0001
Aug 25 08:51:28 2014 [0-1] resuming execution of script
```

Note: This script detects that the last completed stage was Stage 5-4 and resumes at the beginning of Stage 6.

```
### Stage 6: Run media recovery through upgrade redo

Aug 25 08:51:44 2014 [6-1] upgrade redo region identified as scn range
[412934, 979273]
Aug 25 08:51:54 2014 [6-1] starting media recovery on c01orcl
Aug 25 08:53:36 2014 [6-1] confirming media recovery is running
Aug 25 08:53:45 2014 [6-1] waiting for media recovery to initialize
v$recovery_progress
Aug 25 08:55:39 2014 [6-1] monitoring media recovery's progress
Aug 25 08:55:42 2014 [6-2] last applied scn 400465 is approaching upgrade
redo
start scn 412934
Aug 25 08:55:58 2014 [6-2] last applied scn 402153 is approaching upgrade
```

```
redo
start scn 412934
Aug 25 08:56:34 2014 [6-3] recovery of upgrade redo at 05% - estimated
complete at Aug 25 09:12:39
Aug 25 08:56:50 2014 [6-3] recovery of upgrade redo at 06% - estimated
complete at Aug 25 09:12:35
Aug 25 08:57:26 2014 [6-3] recovery of upgrade redo at 07% - estimated
complete at Aug 25 09:17:30
Aug 25 08:58:01 2014 [6-3] recovery of upgrade redo at 10% - estimated
complete at Aug 25 09:18:00
Aug 25 08:58:18 2014 [6-3] recovery of upgrade redo at 11% - estimated
complete at Aug 25 09:18:33
Aug 25 08:58:52 2014 [6-3] recovery of upgrade redo at 14% - estimated
complete at Aug 25 09:17:04
Aug 25 08:59:08 2014 [6-3] recovery of upgrade redo at 17% - estimated
complete at Aug 25 09:15:50
Aug 25 08:59:26 2014 [6-3] recovery of upgrade redo at 22% - estimated
complete at Aug 25 09:11:49
Aug 25 08:59:44 2014 [6-3] recovery of upgrade redo at 26% - estimated
complete at Aug 25 09:10:51
Aug 25 09:00:01 2014 [6-3] recovery of upgrade redo at 28% - estimated
complete at Aug 25 09:11:02
Aug 25 09:00:18 2014 [6-3] recovery of upgrade redo at 32% - estimated
complete at Aug 25 09:09:57
Aug 25 09:00:35 2014 [6-3] recovery of upgrade redo at 33% - estimated
complete at Aug 25 09:10:09
Aug 25 09:00:52 2014 [6-3] recovery of upgrade redo at 38% - estimated
complete at Aug 25 09:09:12
Aug 25 09:01:13 2014 [6-3] recovery of upgrade redo at 39% - estimated
complete at Aug 25 09:09:33
Aug 25 09:01:57 2014 [6-3] recovery of upgrade redo at 40% - estimated
complete at Aug 25 09:10:25
Aug 25 09:02:16 2014 [6-3] recovery of upgrade redo at 41% - estimated
complete at Aug 25 09:10:53
Aug 25 09:02:33 2014 [6-3] recovery of upgrade redo at 42% - estimated
complete at Aug 25 09:11:19
Aug 25 09:02:50 2014 [6-3] recovery of upgrade redo at 44% - estimated
complete at Aug 25 09:11:26
Aug 25 09:03:06 2014 [6-3] recovery of upgrade redo at 46% - estimated
complete at Aug 25 09:11:13
Aug 25 09:03:23 2014 [6-3] recovery of upgrade redo at 48% - estimated
complete at Aug 25 09:11:07
Aug 25 09:03:40 2014 [6-3] recovery of upgrade redo at 51% - estimated
complete at Aug 25 09:10:56
Aug 25 09:03:58 2014 [6-3] recovery of upgrade redo at 55% - estimated
complete at Aug 25 09:10:23
Aug 25 09:04:14 2014 [6-3] recovery of upgrade redo at 63% - estimated
complete at Aug 25 09:09:03
Aug 25 09:04:31 2014 [6-3] recovery of upgrade redo at 66% - estimated
complete at Aug 25 09:08:49
Aug 25 09:04:48 2014 [6-3] recovery of upgrade redo at 67% - estimated
complete at Aug 25 09:09:04
Aug 25 09:05:06 2014 [6-3] recovery of upgrade redo at 68% - estimated
complete at Aug 25 09:09:13
Aug 25 09:05:23 2014 [6-3] recovery of upgrade redo at 69% - estimated
complete at Aug 25 09:09:24
Aug 25 09:05:40 2014 [6-3] recovery of upgrade redo at 72% - estimated
complete at Aug 25 09:09:22
Aug 25 09:05:57 2014 [6-3] recovery of upgrade redo at 74% - estimated
complete at Aug 25 09:09:20
Aug 25 09:06:13 2014 [6-3] recovery of upgrade redo at 76% - estimated
```

```

complete at Aug 25 09:09:26
Aug 25 09:06:31 2014 [6-3] recovery of upgrade redo at 78% - estimated
complete at Aug 25 09:09:25
Aug 25 09:06:48 2014 [6-3] recovery of upgrade redo at 90% - estimated
complete at Aug 25 09:07:53
Aug 25 09:07:13 2014 [6-3] recovery of upgrade redo at 94% - estimated
complete at Aug 25 09:07:54
Aug 25 09:07:30 2014 [6-3] recovery of upgrade redo at 95% - estimated
complete at Aug 25 09:07:58
Aug 25 09:07:47 2014 [6-3] recovery of upgrade redo at 97% - estimated
complete at Aug 25 09:08:06
Aug 25 09:08:04 2014 [6-3] recovery of upgrade redo at 98% - estimated
complete at Aug 25 09:08:13
Aug 25 09:08:21 2014 [6-4] media recovery has finished recovering through
Upgrade

```

Note: In Stage 6, media recovery is performed on the former primary database (c01orcl). This operation upgrades the database and applies all transactions on user data up to the current time. At the end of Stage 6, both databases are fully upgraded.

Stage 7: Switch back to the original roles prior to the rolling upgrade

NOTE: At this point, you have the option to perform a switchover which will restore c01orcl back to a primary database and c02orcl back to a physical standby database. If you answer 'n' to the question below, c01orcl will remain a physical standby database and c02orcl will remain a primary database.

Do you want to perform a switchover? (y/n) :

Note: Stage 7 performs a role switch so that the databases are returned to their original roles. This stage is optional and can be skipped; however, customers often perform this stage.

51. Enter **y** to switch the databases back to their original roles.

Stage 7: Switch back to the original roles prior to the rolling upgrade

NOTE: At this point, you have the option to perform a switchover which will restore c01orcl back to a primary database and c02orcl back to a physical standby database. If you answer 'n' to the question below, c01orcl will remain a physical standby database and c02orcl will remain a primary database.

Do you want to perform a switchover? (y/n) : **y**

```

Aug 25 09:10:30 2014 [7-1] continuing
Aug 25 09:10:30 2014 [7-2] verifying instance c02orcl1 is the only active
instance

```

WARN: c02orcl is a RAC database. Before this script can continue, you must manually reduce the RAC to a single instance. This can be accomplished with the following step:

- 1) Shutdown all instances other than instance c02orcl1.

```
eg: srvctl stop instance -d c02orcl -i c02orcl2
```

Once these steps have been performed, enter 'y' to continue the script. If desired, you may enter 'n' to exit the script to perform the required steps, and recall the script to resume from this point.

Are you ready to continue? (y/n) :

52. **(Session 2)** Return to the terminal session on c02n01 as the oracle user to perform the final switchover. The physru.sh script requires c02orcl to be reduced to one instance. Shut down all instances of c02orcl except for c02orcl1; that is, shut down c02orcl2.

```
[oracle@c02n01 ~]$ srvctl stop instance -d c02orcl -i c02orcl2
[oracle@c02n01 ~]$
```

53. **(Session 1)** Return to your physru.sh session and enter y to continue.

Are you ready to continue? (y/n) : **y**

```
Aug 25 09:12:28 2014 [7-2] continuing
Aug 25 09:12:28 2014 [7-2] verifying instance c02orcl1 is the only active
instance
Aug 25 09:12:30 2014 [7-2] waiting for apply lag to fall under 30 seconds
Aug 25 09:12:37 2014 [7-2] apply lag measured at 7 seconds
Aug 25 09:12:41 2014 [7-3] switching c02orcl to become a physical standby
Aug 25 09:12:56 2014 [7-3] c02orcl is now a physical standby
Aug 25 09:12:56 2014 [7-3] shutting down database c02orcl
Aug 25 09:12:56 2014 [7-3] mounting database c02orcl
Aug 25 09:13:17 2014 [7-4] waiting for standby c01orcl to process end-of-
redo
from primary
Aug 25 09:13:19 2014 [7-5] switching c01orcl to become the new primary
Aug 25 09:13:21 2014 [7-5] c01orcl is now the new primary
Aug 25 09:13:21 2014 [7-5] opening database c01orcl
Aug 25 09:13:56 2014 [7-6] starting media recovery on c02orcl
Aug 25 09:14:05 2014 [7-6] confirming media recovery is running
```

NOTE: Database c01orcl has completed the switchover to the primary role, but

instance c01orcl1 is the only open instance. For increased availability, Oracle recommends opening the remaining active instances which are currently in mounted mode by performing the following steps:

- 1) Shutdown all instances other than instance c01orcl1.
eg: srvctl stop instance -d c01orcl -i c01orcl2
- 2) Startup and open all inactive instances for database c01orcl.
eg: srvctl start database -d c01orcl

NOTE: Database c02orcl is no longer limited to single instance operation since

it has completed the switchover to the physical standby role. For increased availability, Oracle recommends starting the inactive instances in the RAC by performing the following step:

- 1) Startup and mount inactive instances for database c02orcl
eg: srvctl start database -d c02orcl -o mount

Note: At the end of Stage 7, the databases are returned to their original roles. That is, `c01orcl` is the primary database and `c02orcl` is the physical standby database. However, manual intervention is required to restart all the Oracle RAC database instances. The script output contains instructions that include the required commands.

```
### Stage 8: Statistics

script start time: 25-Aug-14 06:16:07
script finish time: 25-Aug-14 09:14:41
total script execution time: +00 02:58:34
wait time for user upgrade: +00 02:03:06
active script execution time: +00 00:55:28
transient logical creation start time: 25-Aug-14 06:22:46
transient logical creation finish time: 25-Aug-14 06:27:43
primary to logical switchover start time: 25-Aug-14 08:37:19
logical to primary switchover finish time: 25-Aug-14 08:39:03
primary services offline for: +00 00:01:44
total time former primary in physical role: +00 00:27:54
time to reach upgrade redo: +00 00:00:34
time to recover upgrade redo: +00 00:12:06
primary to physical switchover start time: 25-Aug-14 09:10:30
physical to primary switchover finish time: 25-Aug-14 09:13:54
primary services offline for: +00 00:03:24

SUCCESS: The physical rolling upgrade is complete

[oracle@c01n01 ~]$
```

Note: Stage 8 outputs statistics for the entire rolling database upgrade process. You can use this information to see how long the entire process took and how long the primary database was unavailable. In the example that forms the basis for this practice, the rolling database upgrade process spanned almost three hours. However, during that time, primary database services were unavailable for a total of 5 minutes and 8 seconds. By using additional custom scripting to automate some of the critical manual tasks, you can expect to reduce primary database down time even further.

54. Close all of your terminal sessions.

Practice 12-5: Post-Upgrade Steps

Overview

At this point, your environment is fully upgraded. However, a few more operations are required to make it fully operational and to leverage the full power of Oracle Database 12c. In the final part of this practice, you will perform the following tasks:

- Start all the Oracle RAC database instances.
- Re-enable Oracle Data Guard Broker.
- Raise the `compatible` database instance parameter to enable all the new features of Oracle Database 12c.
- Confirm that the user data changes made during the upgrade process made it through.

Terminal Sessions

- **Session 1:** `ssh oracle@c01n01`
- **Session 2:** `ssh oracle@c02n01`

Tasks

1. **(Session 1)** Establish a terminal session connected to `c01n01` as the `oracle` user and configure the terminal environment as shown in the following:

```
[Classroom PC]$ ssh oracle@c01n01
oracle@c01n01's password: <oracle>
[oracle@c01n01 ~]$ . oraenv
ORACLE_SID = [oracle] ? c01orcl
The Oracle base has been set to /u01/app/oracle
[oracle@c01n01 ~]$ export ORACLE_SID=c01orcl1
[oracle@c01n01 ~]$
```

2. Confirm the status of `c01orcl` and reconfigure the database so that all the instances are running and open.

```
[oracle@c01n01 ~]$ srvctl status database -d c01orcl -v
Instance c01orcl1 is running on node c01n01. Instance status: Open.
Instance c01orcl2 is running on node c01n02. Instance status: Mounted (Closed).
[oracle@c01n01 ~]$ srvctl stop instance -d c01orcl -i c01orcl2
[oracle@c01n01 ~]$ srvctl start database -d c01orcl
[oracle@c01n01 ~]$ srvctl status database -d c01orcl -v
Instance c01orcl1 is running on node c01n01. Instance status: Open.
Instance c01orcl2 is running on node c01n02. Instance status: Open.
[oracle@c01n01 ~]$
```

3. **(Session 2)** Leave your other terminal session open and establish another terminal session connected to `c02n01` as the `oracle` user. Configure the terminal environment as shown in the following:

```
[Classroom PC]$ ssh oracle@c02n01
oracle@c02n01's password: <oracle>
[oracle@c02n01 ~]$ . oraenv
ORACLE_SID = [oracle] ? c02orcl
The Oracle base has been set to /u01/app/oracle
[oracle@c02n01 ~]$ export ORACLE_SID=c02orcl1
[oracle@c02n01 ~]$
```

4. Confirm the status of c02orcl and reconfigure the database so that all the instances are running and mounted.

```
[oracle@c02n01 ~]$ srvctl status database -d c02orcl -v
Instance c02orcl1 is running on node c02n01. Instance status: Mounted
(Closed).
Instance c02orcl2 is not running on node c02n02
[oracle@c02n01 ~]$ srvctl start database -d c02orcl -o mount
[oracle@c02n01 ~]$ srvctl status database -d c02orcl -v
Instance c02orcl1 is running on node c02n01. Instance status: Mounted
(Closed).
Instance c02orcl2 is running on node c02n02. Instance status: Mounted
(Closed).
[oracle@c02n01 ~]$
```

5. Re-enable Oracle Data Guard Broker on c02orcl.

```
[oracle@c02n01 ~]$ sqlplus / as sysdba
SQL*Plus: Release 12.1.0.1.0 Production ...
SQL> alter system set dg_broker_start=true scope=both;
System altered.
SQL> exit
Disconnected from Oracle Database 12c Enterprise Edition Release
12.1.0.1.0 ...
[oracle@c02n01 ~]$
```

6. **(Session 1)** Return to the terminal session on c01n01 to re-enable Oracle Data Guard Broker on c01orcl.

```
[oracle@c01n01 ~]$ sqlplus / as sysdba
SQL*Plus: Release 12.1.0.1.0 Production ...
SQL> alter system set dg_broker_start=true scope=both;
System altered.
SQL> exit
Disconnected from Oracle Database 12c Enterprise Edition Release
12.1.0.1.0 ...
[oracle@c01n01 ~]$
```

7. Use the Data Guard Broker management utility (dgmgrl) to re-enable the broker configuration. Then, examine the configuration and the databases to confirm that everything is functioning as expected.

```
[oracle@c01n01 ~]$ dgmgrl sys/oracle_4U
DGMGRL for Linux: Version 12.1.0.1.0 - 64bit Production

Copyright (c) 2000, 2012, Oracle. All rights reserved.

Welcome to DGMGRL, type "help" for information.
Connected as SYSDG.
DGMGRL> enable configuration
Enabled.
DGMGRL> show configuration

Configuration - dg_config

Protection Mode: MaxPerformance
Databases:
  c01orcl - Primary database
```

```
c02orcl - Physical standby database

Fast-Start Failover: DISABLED
Configuration Status:

SUCCESS

DGMGRL> show database c01orcl

Database - c01orcl
Role: PRIMARY
Intended State: TRANSPORT-ON
Instance(s):
  c01orcl1
  c01orcl2

Database Status:
SUCCESS

DGMGRL> show database c02orcl

Database - c02orcl

Role: PHYSICAL STANDBY
Intended State: APPLY-ON
Transport Lag: 0 seconds (computed 1 second ago)
Apply Lag: 0 seconds (computed 1 second ago)
Apply Rate: 72.00 KByte/s
Real Time Query: OFF
Instance(s):
  c02orcl1 (apply instance)
  c02orcl2

Database Status:
SUCCESS

DGMGRL> exit
[oracle@c01n01 ~]$
```

Notes

- Next, you will raise the compatible database instance parameter to enable all new features of Oracle Database 12c. Oracle recommends increasing the compatible parameter only after thorough testing of the upgraded database has been performed. This is because the database cannot subsequently be downgraded to releases earlier than what is set for compatibility.
- The Flashback Database feature cannot be used to return a database back to a point prior to raising the compatible parameter. Because of this, you must first remove all guaranteed restore points before raising the value of the compatible parameter.
- In a Data Guard Environment, you must raise the compatible parameter on the standby database before raising it on the primary database.

8. **(Session 2)** Return to the terminal session on c02orcl to perform the following tasks:

- Find the name of any remaining guaranteed restore points. You should expect that the physru.sh script has already removed the guaranteed restore points that it created; however, you should expect to see one remaining guaranteed restore point that was created by the Database Upgrade Assistant (DBUA).
- Drop any remaining guaranteed restore points.
- Confirm the current compatible setting.
- Set compatible='12.1.0.1.0' inside the database server parameter file.

```
[oracle@c02n01 ~]$ sqlplus / as sysdba
SQL*Plus: Release 12.1.0.1.0 Production ...

SQL> select name from v$restore_point;
NAME
-----
GRP_1408952037429

SQL> drop restore point GRP_1408952037429;
Restore point dropped.

SQL> show parameter compatible
NAME          TYPE        VALUE
-----
compatible    string      11.2.0.4.0
noncdb_compatible boolean    FALSE

SQL> alter system set compatible='12.1.0.1.0' scope=spfile;
System altered.

SQL> exit
Disconnected from Oracle Database 12c Enterprise Edition Release
12.1.0.1.0 ...
[oracle@c02n01 ~]$
```

9. Stop and restart the c02orcl database in mount mode. Then confirm that the compatible value is now 12.1.0.1.0.

```
[oracle@c02n01 ~]$ srvctl stop database -d c02orcl
[oracle@c02n01 ~]$ srvctl start database -d c02orcl -o mount
[oracle@c02n01 ~]$ sqlplus / as sysdba
SQL*Plus: Release 12.1.0.1.0 Production ...

SQL> show parameter compatible
NAME          TYPE        VALUE
-----
compatible    string      12.1.0.1.0
noncdb_compatible boolean    FALSE
```

```
SQL> exit
Disconnected from Oracle Database 12c Enterprise Edition Release
12.1.0.1.0 ...
[oracle@c01n01 ~]$
```

10. **(Session 1)** Return to the terminal session on c01n01 to raise the compatible parameter on the c01orcl database. Notice that there should be no guaranteed restore points to drop on this database because DBUA was never run on the database.

```
[oracle@c01n01 ~]$ sqlplus / as sysdba

SQL*Plus: Release 12.1.0.1.0 Production ...

SQL> select name from v$restore_point;
no rows selected

SQL> show parameter compatible

NAME                           TYPE        VALUE
-----
compatible                     string      11.2.0.4.0
noncdb_compatible              boolean    FALSE

SQL> alter system set compatible='12.1.0.1.0' scope=spfile;

System altered.

SQL> exit
Disconnected from Oracle Database 12c Enterprise Edition Release
12.1.0.1.0 ...
[oracle@c01n01 ~]$
```

11. Stop and restart the c01orcl database. Then confirm that the compatible value is now 12.1.0.1.0.

```
[oracle@c01n01 ~]$ srvctl stop database -d c01orcl
[oracle@c01n01 ~]$ srvctl start database -d c01orcl
[oracle@c01n01 ~]$ sqlplus / as sysdba

SQL*Plus: Release 12.1.0.1.0 Production ...

SQL> show parameter compatible

NAME                           TYPE        VALUE
-----
compatible                     string      12.1.0.1.0
noncdb_compatible              boolean    FALSE

SQL> exit
Disconnected from Oracle Database 12c Enterprise Edition Release
12.1.0.1.0
[oracle@c01n01 ~]$
```

12. Finally, connect to your primary database (c01orcl) and query the hr.regions table to confirm that the changes you made during the upgrade process have survived.

```
[oracle@c01n01 ~]$ sqlplus hr/hr

SQL*Plus: Release 12.1.0.1.0 Production ...

SQL> select * from regions;

REGION_ID  REGION_NAME
-----
      5  Australia
      1  Europe
      2  Americas
      3  Asia
      4  Middle East and Africa

SQL> exit
Disconnected from Oracle Database 12c Enterprise Edition Release
12.1.0.1.0 ...
[oracle@c01n01 ~]$
```

13. Close all of your terminal sessions.

Congratulations! You have performed a rolling upgrade from Oracle Database release 11.2.0.4 to Oracle Database 12.1.0.1 by using Oracle RAC and the Oracle Data Guard transient logical standby feature to maximize the availability of the primary database during the upgrade process.

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

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

Appendix

Chapter 13

Appendix 13: Overview

Overview

This appendix provides the troubleshooting steps to resolve the commonly observed issues during the practices caused by user errors or unknown reasons.

- Practice 13-1: Troubleshooting for Case 1
- Practice 13-2: Troubleshooting for Case 2
- Practice 13-3: Troubleshooting for Case 3
- Practice 13-4: Troubleshooting for Case 4
- Practice 13-5: Troubleshooting for Case 5
- Practice 13-6: Troubleshooting for Case 6
- Practice 13-7: Troubleshooting for Case 7
- Practice 13-8: Troubleshooting for Case 8

Practice 13-1: Troubleshooting (Case 1)

Overview

Troubleshooting Case 1 shows how to fix the following unknown issue occurred in *the case where a sequence number is associated with a missing archived log file.*

- **Observation 1 (on the primary database)**

```
ORA-12543: TNS:destination host unreachable
FAL[server, ARC3]: Error 12543 creating remote archivelog file
'westdb'
ARC3: FAL archive failed with error 12543. See trace for details
Fri Sep 25 08:09:15 2015
Errors in file
/u01/app/oracle/diag/rdbms/eastdb/eastdb1/trace/eastdb1_arc3_2302
5.trc:
ORA-16055: FAL request rejected
ARCH: FAL archive failed. Archiver continuing
Fri Sep 25 08:09:15 2015
ORACLE Instance eastdb1 - Archival Error. Archiver continuing.
```

- **Observation 2 (on the standby database)**

```
RFS[5]: Possible network disconnect with primary database
[oracle@wnode03 trace]$ tail -f alert_westdb1.log
RFS[5]: Assigned to RFS process (PID:1611)
RFS[5]: Opened log for thread 1 sequence 8 dbid 812862977 branch
890730627
CORRUPTION DETECTED: In redo blocks starting at block 8193 count
2048 for thread 1 sequence xx
Deleted Oracle managed file
+FRA/WESTDB/ARCHIVELOG/2015_09_18/thread_1_seq_8.312.890743407
RFS[5]: Possible network disconnect with primary database
```

Terminal Sessions

- **Session 1:** ssh oracle@enode01
- **Session 2:** ssh oracle@wnode03



Tasks

1. **(Session 1)** Open a terminal window on enode01 to connect as the oracle user.

```
[Classroom PC]$ ssh oracle@enode01
oracle@enode01's password: <oracle>
[oracle@enode01 ~]$ . oraenv
ORACLE_SID = [oracle] ? eastdb
The Oracle base has been set to /u01/app/oracle
[oracle@enode01 ~]$ export ORACLE_SID=eastdb1
[oracle@enode01 ~]$
[oracle@enode01 ~]$
```

2. Review the alert log file in the primary database.

```
[oracle@enode01 ~]$ tail -f
/u01/app/oracle/diag/rdbms/eastdb/eastdb1/trace/alert_eastdb1.log

ORACLE Instance eastdb1 - Archival Error. Archiver continuing.
Wed Oct 28 08:03:03 2015
ARC0: Error 3135 archiving log 0 to 'westdb'
ARC0: FAL archive failed with error 3135. See trace for details
Wed Oct 28 08:03:03 2015
Errors in file
/u01/app/oracle/diag/rdbms/eastdb/eastdb1/trace/eastdb1_arc0_24766.trc:
ORA-16055: FAL request rejected
ARCH: FAL archive failed. Archiver continuing
Wed Oct 28 08:03:03 2015
ORACLE Instance eastdb1 - Archival Error. Archiver continuing.
```

3. **(Session 2)** Open another terminal on wnode03 to connect as the oracle user.

```
[oracle@enode01 ~]$ ssh wnode03
[oracle@wnode03 ~]$ . oraenv
ORACLE_SID = [oracle] ? westdb
The Oracle base has been set to /u01/app/oracle
[oracle@wnode03 ~]$ export ORACLE_SID=westdb1
[oracle@wnode03 ~]$
```

4. Review the missing archived log file caused by the unknown issue in the physical standby database.

Note: You may have multiple missing archived log files.

```
[oracle@wnode03 ~]$ sqlplus / as sysdba

SQL> select * from v$archive_gap;

THREAD#  LOW_SEQUENCE#  HIGH_SEQUENCE#      CON_ID
-----  -----
          1            22            22          1

SQL> exit
[oracle@wnode03 ~]$
```

5. If there is no output in step 4, review the alert log file in the physical standby database.

```
[oracle@wnode03 ~]$ tail -f
/u01/app/oracle/diag/rdbms/westdb/westdb1/trace/alert_westdb1.log

RFS [5] : Possible network disconnect with primary database
[oracle@wnode03 trace]$ tail -f alert_westdb1.log
RFS [5] : Assigned to RFS process (PID:1611)
RFS [5] : Opened log for thread 1 sequence 8 dbid 812862977 branch
890730627
CORRUPTION DETECTED: In redo blocks starting at block 8193 count
2048 for thread 1 sequence 22
Deleted Oracle managed file
+FRA/WESTDB/ARCHIVELOG/2015_09_18/thread_1_seq_22.312.890743407
RFS [5] : Possible network disconnect with primary database
```

6. (**Session 1**) Return to the terminal session on the primary database to identify the missing archive log files.

```
[oracle@enode01 ~]$ sqlplus / as sysdba

SQL> select name from v$archived_log where thread#=1 and sequence#= 22
and name like '+FRA/EASTDB%';

NAME
-----
+FRA/EASTDB/ARCHIVELOG/2015_10_27/thread_1_seq_22.276.894233603

SQL> exit
```

7. In the same terminal session, switch to grid user to copy the identified archived log files to the /tmp directory.

Note: The file name in the /tmp directory can be any meaningful name.

```
[oracle@enode01 ~]$ su - grid
Password: <oracle>
[grid@enode01 ~]$ . oraenv
ORACLE_SID = [grid] ? +ASM1
The Oracle base has been set to /u01/app/grid
[grid@enode01 ~]$ asmcmd cp
+FRA/EASTDB/ARCHIVELOG/2015_10_27/thread_1_seq_22.276.894233603
/tmp/seq122
copying +FRA/EASTDB/ARCHIVELOG/2015_10_27/thread_1_seq_22.276.894233603
-> /tmp/seq122
[grid@enode01 ~]$
```

8. Transfer the archive log files in the /tmp directory to the standby host.

```
[grid@enode01 ~]$ scp /tmp/seq122* wnode03:/tmp
grid@wnode03's password: <oracle>
seq122                                         100%    41MB   20.3MB/s   00:02
[grid@enode01 ~]$ exit
[oracle@enode01 ~]$
```

9. **(Session 2)** Return to the terminal session on wnode03 to register the missing archived log file in the physical standby database.

```
[oracle@wnode03 ~]$ sqlplus / as sysdba  
  
SQL> alter database register logfile '/tmp/seq122';  
  
Database altered.  
  
SQL>
```

10. Verify that the redo gap issue has been resolved.

```
SQL> select * from v$archive_gap;  
  
no rows selected  
  
SQL> exit  
[oracle@wnode03 ~]$
```

11. If any additional missing log files are detected, go back to step 3 and repeat the steps again.

```
SQL> select * from v$archive_gap;  
  
THREADD# LOW_SEQUENCE# HIGH_SEQUENCE# CON_ID  
-----  
1 26 26 1  
  
SQL> exit
```

12. If you have completed practice 5, you can verify the Data Guard configuration.

Note: The Transport Lag and Apply Lag must be 0 seconds.

```
[oracle@enode01 ~]$ dgmgrl sysdg/oracle_4U@eastdb  
  
DGMGRL> show database westdb  
  
Database - westdb  
  
Role: PHYSICAL STANDBY  
Intended State: APPLY-ON  
Transport Lag: 0 seconds (computed 0 seconds ago)  
Apply Lag: 0 seconds (computed 0 seconds ago)  
Average Apply Rate: 14.00 KByte/s  
Real Time Query: ON  
Instance(s):  
westdb1 (apply instance)  
westdb2  
  
Database Status:  
SUCCESS  
  
DGMGRL>
```

13. Close all of your terminal sessions.

Practice 13-2: Troubleshooting (Case 2)

Overview

Troubleshooting Case 2 shows how to fix the following unknown issue occurred in *the case where a sequence number is associated with multiple archived log files*.

- **Observation 1 (on the primary database)**

```
ORA-12543: TNS:destination host unreachable
FAL[server, ARC3]: Error 12543 creating remote archivelog file
'westdb'
ARC3: FAL archive failed with error 12543. See trace for details
Fri Sep 25 08:09:15 2015
Errors in file
/u01/app/oracle/diag/rdbms/eastdb/eastdb1/trace/eastdb1_arc3_2302
5.trc:
ORA-16055: FAL request rejected
ARCH: FAL archive failed. Archiver continuing
Fri Sep 25 08:09:15 2015
ORACLE Instance eastdb1 - Archival Error. Archiver continuing.
```

- **Observation 2 (on the standby database)**

```
RFS[5]: Possible network disconnect with primary database
[oracle@wnode03 trace]$ tail -f alert_westdb1.log
RFS[5]: Assigned to RFS process (PID:1611)
RFS[5]: Opened log for thread 1 sequence 8 dbid 812862977 branch
890730627
CORRUPTION DETECTED: In redo blocks starting at block 8193count
2048 for thread 1 sequence xx
Deleted Oracle managed file
+FRA/WESTDB/ARCHIVELOG/2015_09_18/thread_1_seq_8.312.890743407
RFS[5]: Possible network disconnect with primary database
```

Terminal Sessions

- **Session 1:** ssh oracle@enode01
- **Session 2:** ssh oracle@wnode03



Tasks

1. **(Session 1)** Open a terminal window on enode01 to connect as the oracle user.

```
[Classroom PC]$ ssh oracle@enode01
oracle@enode01's password: <oracle>
[oracle@enode01 ~]$ . oraenv
ORACLE_SID = [oracle] ? eastdb
The Oracle base has been set to /u01/app/oracle
[oracle@enode01 ~]$ export ORACLE_SID=eastdb1
[oracle@enode01 ~]$
```

2. Review the alert log file in the primary database.

```
[oracle@enode01 ~]$ tail -f
/u01/app/oracle/diag/rdbms/eastdb/eastdb1/trace/alert_eastdb1.log
ORACLE Instance eastdb1 - Archival Error. Archiver continuing.
Wed Oct 28 08:03:03 2015
ARC0: Error 3135 archiving log 0 to 'westdb'
ARC0: FAL archive failed with error 3135. See trace for details
Wed Oct 28 08:03:03 2015
Errors in file
/u01/app/oracle/diag/rdbms/eastdb/eastdb1/trace/eastdb1_arc0_24766.trc:
ORA-16055: FAL request rejected
ARCH: FAL archive failed. Archiver continuing
Wed Oct 28 08:03:03 2015
ORACLE Instance eastdb1 - Archival Error. Archiver continuing.
```

3. **(Session 2)** Open another terminal on wnode03 to connect as the oracle user.

```
[oracle@enode01 ~]$ ssh wnode03
[oracle@wnode03 ~]$ . oraenv
ORACLE_SID = [oracle] ? westdb
The Oracle base has been set to /u01/app/oracle
[oracle@wnode03 ~]$ export ORACLE_SID=westdb1
[oracle@wnode03 ~]$
```

4. Review the missing archived log files caused by the unknown issue in the physical standby database.

```
[oracle@wnode03 ~]$ sqlplus / as sysdba

SQL> select * from v$archive_gap;

  THREAD#  LOW_SEQUENCE# HIGH_SEQUENCE#      CON_ID
-----  -----  -----  -----
          1          16          16          1
          2          15          15          1

SQL>
```

5. **(Session 1)** Return to the terminal session on the primary database to identify the missing archived log files.

Note: This is the case where a sequence number is assigned to multiple archived log files for unknown reasons. Ignore the westdb entries.

```
[oracle@enode01 ~]$ sqlplus / as sysdba

SQL> select name from v$archived_log where thread#=1 and sequence#=16
```

```

and name like '+FRA/EASTDB%';

NAME
-----
+FRA/EASTDB/ARCHIVELOG/2015_10_27/thread_1_seq_16.295.894203115
+FRA/EASTDB/ARCHIVELOG/2015_10_28/thread_1_seq_16.476.894281025
+FRA/EASTDB/ARCHIVELOG/2015_10_28/thread_1_seq_16.539.894312459

SQL> select name from v$archived_log where thread#=2 and sequence#=15;

NAME
-----
+FRA/EASTDB/ARCHIVELOG/2015_10_27/thread_2_seq_15.283.894237001
+FRA/EASTDB/ARCHIVELOG/2015_10_28/thread_2_seq_15.470.894279337
+FRA/EASTDB/ARCHIVELOG/2015_10_28/thread_2_seq_15.538.894312451

SQL> exit

```

- In the same terminal session, switch to the grid user to copy the identified archived log files to the /tmp directory.

Note: The file name in the /tmp directory can be any meaningful name.

```

[oracle@enode01 ~]$ su - grid
Password: <oracle>
[grid@enode01 ~]$ . oraenv
ORACLE_SID = [grid] ? +ASM1
The Oracle base has been set to /u01/app/grid
[grid@enode01 ~]$ asmcmd cp
+FRA/EASTDB/ARCHIVELOG/2015_10_27/thread_1_seq_16.29
5.894203115 /tmp/seq1161
copying +FRA/EASTDB/ARCHIVELOG/2015_10_27/thread_1_seq_16.295.894203115
-> /tmp/
      seq1161
[grid@enode01 ~]$ asmcmd cp
+FRA/EASTDB/ARCHIVELOG/2015_10_28/thread_1_seq_16.47
6.894281025 /tmp/seq1162
copying +FRA/EASTDB/ARCHIVELOG/2015_10_28/thread_1_seq_16.476.894281025
-> /tmp/
      seq1162
[grid@enode01 ~]$ asmcmd cp
+FRA/EASTDB/ARCHIVELOG/2015_10_28/thread_1_seq_16.53
9.894312459 /tmp/seq1163
copying +FRA/EASTDB/ARCHIVELOG/2015_10_28/thread_1_seq_16.539.894312459
-> /tmp/
      seq1163

[grid@enode01 ~]$ asmcmd cp
+FRA/EASTDB/ARCHIVELOG/2015_10_27/thread_2_seq_15.28
3.894237001 /tmp/seq2151
copying +FRA/EASTDB/ARCHIVELOG/2015_10_27/thread_2_seq_15.283.894237001
-> /tmp/
      seq2151
[grid@enode01 ~]$ asmcmd cp
+FRA/EASTDB/ARCHIVELOG/2015_10_28/thread_2_seq_15.47
0.894279337 /tmp/seq2152
copying +FRA/EASTDB/ARCHIVELOG/2015_10_28/thread_2_seq_15.470.894279337
-> /tmp/
      seq2152
[grid@enode01 ~]$ asmcmd cp
+FRA/EASTDB/ARCHIVELOG/2015_10_28/thread_2_seq_15.53
8.894312451 /tmp/seq2153

```

```
copying +FRA/EASTDB/ARCHIVELOG/2015_10_28/thread_2_seq_15.538.894312451
-> /tmp/
      seq2153
[grid@enode01 ~]$
```

7. Transfer the archived log files in the /tmp directory to the standby host.

```
[grid@enode01 ~]$ scp /tmp/seq116* wnode03:/tmp
grid@wnode03's password: <oracle>
seq1161                      100%   122KB 122.0KB/s  00:00
seq1162                      100%    11KB 11.0KB/s  00:00
seq1163
      100%   22MB 11.1MB/s  00:02
[grid@enode01 ~]$ scp /tmp/seq215* wnode03:/tmp
grid@wnode03's password: <oracle>
seq2151                      100%   267KB 266.5KB/s  00:00
seq2152                      100%    15KB 15.0KB/s  00:00
seq2153
      100%   41MB 20.3MB/s  00:02
[grid@enode01 ~]$ exit
[oracle@enode01 ~]$
```

8. (Session 2) Return to the terminal session on wnode03 to register the missing archived log file in the physical standby database.

```
SQL> alter database register logfile '/tmp/seq1161';
alter database register logfile '/tmp/seq1161'
*
ERROR at line 1:
ORA-16089: archive log has already been registered

SQL> alter database register logfile '/tmp/seq1162';
alter database register logfile '/tmp/seq1162'
*
ERROR at line 1:
ORA-16089: archive log has already been registered

SQL> alter database register logfile '/tmp/seq1163';
Database altered.

SQL> alter database register logfile '/tmp/seq2151';
alter database register logfile '/tmp/seq2151'
*
ERROR at line 1:
ORA-16089: archive log has already been registered

SQL> alter database register logfile '/tmp/seq2152';
alter database register logfile '/tmp/seq2152'
*
ERROR at line 1:
ORA-16089: archive log has already been registered

SQL> alter database register logfile '/tmp/seq2153';
Database altered.

SQL>
```

9. Verify that the redo gap issue has been resolved.

```
SQL> select * from v$archive_gap;
no rows selected

SQL> exit
[oracle@wnode03 ~] $
```

10. Restart the physical standby database.

```
[oracle@wnode03 ~] $ srvctl stop database -db westdb
[oracle@wnode03 ~] $ srvctl start database -db westdb
[oracle@wnode03 ~] $
```

11. In the same terminal session, repeat step 8.

Note: The previous step indicated there is no a missing archived log file. However, the alert log file may report BLOCK CORRUPTION issue, which indicates the redo gap issue for unknown reasons. This is the case where a sequence number is assigned to multiple log files. You will notice that the physical standby database accepts the missing archived log files that have been registered in step 8.

```
SQL> alter database register logfile '/tmp/seq1161';
alter database register logfile '/tmp/seq1161'
*
ERROR at line 1:
ORA-16089: archive log has already been registered

SQL> alter database register logfile '/tmp/seq1162';
alter database register logfile '/tmp/seq1162'
*
ERROR at line 1:
ORA-16089: archive log has already been registered

SQL> alter database register logfile '/tmp/seq1163';
Database altered.

SQL> alter database register logfile '/tmp/seq2151';
alter database register logfile '/tmp/seq2151'
*
ERROR at line 1:
ORA-16089: archive log has already been registered

SQL> alter database register logfile '/tmp/seq2152';
alter database register logfile '/tmp/seq2152'
*
ERROR at line 1:
ORA-16089: archive log has already been registered

SQL> alter database register logfile '/tmp/seq2153';
Database altered.

SQL> alter database recover managed standby database disconnect;
Database altered.

SQL> exit
[oracle@wnode03 ~] $
```

12. **(Session 1)** Return to the terminal session on enode01 to enable the redo transport service.

```
[oracle@enode01 ~]$ sqlplus / as sysdba
SQL> alter system set log_archive_dest_state_2=enable;
SQL>
```

13. If you have completed practice 5, you can verify the Data Guard configuration.

Note: The Transport Lag and Apply Lag must be 0 seconds.

```
[oracle@enode01 ~]$ dgmgrl sysdg/oracle_4U@eastdb

DGMGRL> show database westdb

Database - westdb

Role: PHYSICAL STANDBY
Intended State: APPLY-ON
Transport Lag: 0 seconds (computed 0 seconds ago)
Apply Lag: 0 seconds (computed 0 seconds ago)
Average Apply Rate: 14.00 KByte/s
Real Time Query: ON
Instance(s):
    westdb1 (apply instance)
    westdb2

Database Status:
SUCCESS

DGMGRL>
```

14. Close all of your terminal sessions.

Practice 13-3: Troubleshooting (Case 3)

Overview

Troubleshooting Case 3 shows how to fix primary database performance issues. At times, your primary database can be extremely slow. The alert log file reports ORA-600 error like observation 3 in the following. Apply the following workaround in your primary database.

- Observation 3 (on the primary database)**

```
Errors in file
/u01/app/oracle/diag/rdbms/eastdb/eastdb1/trace/eastdb1_q004_25404.trc
(incident=94543) :

ORA-00600: internal error code, arguments: [kwqitnmphe:ltbagi], [1],
[0], [], [], [], [], [], [], [], []

Incident details in:
/u01/app/oracle/diag/rdbms/eastdb/eastdb1/incident/incdir_94543/eastdb1
_q004_25404_i94543.trc

Mon Oct 26 16:34:29 2015

Dumping diagnostic data in directory=[cdmp_20151026163429], requested
by (instance=1, osid=25404 (Q004)), summary=[incident=94543].
```

Tasks

1. **(Session 1)** Open a terminal window on enode01 to connect as the oracle user.

```
[Classroom PC]$ ssh oracle@enode01
oracle@enode01's password: <oracle>
[oracle@enode01 ~]$ . oraenv
ORACLE_SID = [oracle] ? eastdb
The Oracle base has been set to /u01/app/oracle
[oracle@enode01 ~]$ export ORACLE_SID=eastdb1
[oracle@enode01 ~]$
```

2. On the primary database, perform the following workaround.

```
[oracle@enode01 ~]$ sqlplus / as sysdba
SQL> DECLARE
  po dbms_aqadm.aq$purge_options_t;
BEGIN
  po.block := FALSE;
  DBMS_AQADM.PURGE_QUEUE_TABLE(
    queue_table => 'SYS.SYS$SERVICE_METRICS_TAB',
    purge_condition => NULL,
    purge_options => po);
END;
/
      2      3      4      5      6      7      8      9     10

PL/SQL procedure successfully completed.

SQL>
```

3. Close all of your terminal sessions.

Practice 13-4: Re-creating Data Guard Broker Configuration (Case 4)

Overview:

The following steps show how to re-create the Data Guard configuration.

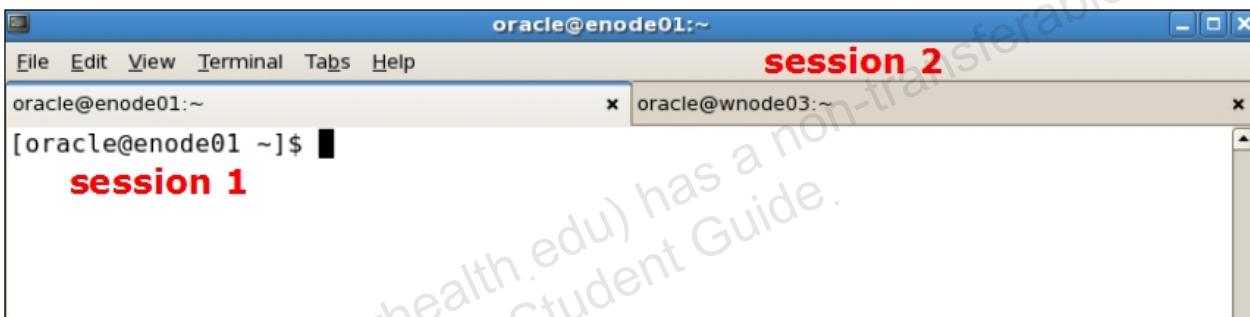
Estimated Time: 10 mins

Assumption

- The primary and standby databases are in the normal state, but the Data Guard broker does not show the correct status.

Terminal Sessions

- Session 1:** ssh oracle@enode01
- Session 2:** ssh oracle@wnode03



Tasks

- (**Session 1**) From your classroom PC desktop, execute `ssh oracle@enode01` to open a terminal session on enode01 as the oracle user.

```
[Classroom PC] $ ssh oracle@enode01
oracle@enode01's password: <oracle>
[oracle@enode01 ~] $
```

- Make sure that you set up your environment variables correctly for the `oracle` user.

```
[oracle@enode01 ~] $ . oraenv
ORACLE_SID = [oracle]? eastdb
The Oracle base has been set to /u01/app/oracle
[oracle@enode01 ~] $ export ORACLE_SID=eastdb1
[oracle@enode01 ~] $
```

- Connect to the primary database by using SQL*Plus to perform the following tasks.

```
[oracle@enode01 ~] $ sqlplus / as sysdba
System altered
SQL> alter system set log_archive_dest_2='' scope=both sid='*';
System altered
SQL> alter system set dg_broker_start=false scope=both sid='*';
System altered
SQL> exit
[oracle@enode01 ~] $
```

4. **(Session 2)** Open a terminal window on wnode03 connected as the oracle with the environment variables set to westdb1.

```
[Classroom PC]$ ssh oracle@wnode03
oracle@wnode03's password: <oracle>
[oracle@wnode03 ~]$ . oraenv
ORACLE_SID = [oracle]? westdb
The Oracle base has been set to /u01/app/oracle
[oracle@wnode03 ~]$ export ORACLE_SID=westdb1
[oracle@wnode03 ~]$
```

5. Connect to the physical standby by using SQL*Plus to perform the following tasks.

```
[oracle@wnode03 ~]$ sqlplus / as sysdba
SQL> alter database recover managed standby database cancel;
Database altered.
SQL> alter system set log_archive_dest_2='' scope=both sid='*';
System altered.
SQL> alter system set dg_broker_start=false scope=both sid='*';
stem altered
SQL> exit
[oracle@wnode03 ~]$
```

6. **(Session 1)** Return to the terminal window on enode01 to remove the existing broker configuration files in the primary database.

```
[oracle@enode01 ~]$ su - grid
Password: <oracle>
[grid@enode01 ~]$ . oraenv
ORACLE_SID = [grid] ? +ASM1
The Oracle base has been set to /u01/app/grid
[grid@enode01 ~]$ asmcmd rm +DATA/eastdb/dr1config.dat
[grid@enode01 ~]$ asmcmd rm +FRA/eastdb/dr2config.dat
[grid@enode01 ~]$ exit
logout
[oracle@enode01 ~]$
```

7. **(Session 2)** Return to the terminal window on wnode03 to remove the existing broker configuration files in the standby database.

```
[oracle@wnode03 ~]$ su - grid
Password: <oracle>
[grid@enode01 ~]$ . oraenv
ORACLE_SID = [grid] ? +ASM1
The Oracle base has been set to /u01/app/grid
[grid@wnode03 ~]$ asmcmd rm +DATA/westdb/dr1config.dat
[grid@wnode03 ~]$ asmcmd rm +FRA/westdb/dr2config.dat
[grid@wnode03 ~]$ exit
logout
[oracle@wnode03 ~]$
```

8. **(Session 1)** Return to the terminal window on enode01 to start the Data Guard broker.

```
[oracle@enode01 ~]$ sqlplus / as sysdba
SQL> alter system set
dg_broker_config_file1='+data/eastdb/dr1config.dat' scope=both
sid='*';
System altered.
```

```

SQL> alter system set
dg_broker_config_file2='+fra/eastdb/dr2config.dat' scope=both
sid='*';
System altered.
SQL> alter system set dg_broker_start=true scope=both sid='*';
System altered.
SQL> exit
[oracle@enode01 ~] $

```

9. (Session 2) Return to the terminal window on wnode03 to start the Data Guard broker.

```

[oracle@wnode03 ~]$ sqlplus / as sysdba
SQL> alter system set
dg_broker_config_file1='+data/westdb/dr1config.dat' scope=both
sid='*';
System altered.
SQL> alter system set
dg_broker_config_file2='+fra/westdb/dr2config.dat' scope=both
sid='*';
System altered.
SQL> alter system set dg_broker_start=true scope=both sid='*';
System altered.
SQL> exit
[oracle@wnode03 ~] $

```

10. Launch DGMGRL and attempt to show the configuration.

```

[oracle@wnode03 ~]$ dgmgrl
DGMGRL for Linux: Version 12.1.0.2.0 - 64bit Production
Copyright (c) 2000, 2012, Oracle. All rights reserved.
Welcome to DGMGRL, type "help" for information.
DGMGRL> connect sys/oracle_4U@eastdb
Connected as SYSDG.
DGMGRL> show configuration
ORA-16532: Data Guard broker configuration does not exist

Configuration details cannot be determined by DGMGRL
DGMGRL>

```

11. Create the Data Guard broker configuration and then show the configuration.

```

DGMGRL> create configuration 'dg_config' as primary database is
'eastdb' connect identifier is eastdb;
Configuration "dg_config" created with primary database "eastdb"

DGMGRL> show configuration

Configuration - dg_config

Protection Mode: MaxPerformance
Members:
eastdb - Primary database

Fast-Start Failover: DISABLED

Configuration Status:
DISABLED

DGMGRL>

```

12. Add the physical standby database westdb to the configuration and show the results.

```
DGMGRL> add database 'westdb' as connect identifier is westdb;
Database "westdb" added
DGMGRL> show configuration

Configuration - dg_config

Protection Mode: MaxPerformance
Members:
eastdb - Primary database
westdb - Physical standby database

Fast-Start Failover: DISABLED

Configuration Status:
DISABLED

DGMGRL>
```

13. Set the CommunicationTimeout property and set it to a value of 300.

```
DGMGRL> edit configuration set property CommunicationTimeout = 300;
Property "communicationtimeout" updated
DGMGRL>
```

14. Set the OperationTimeout property and set it to the maximum value of 600.

```
DGMGRL> edit configuration set property OperationTimeout = 600;
Property "operationtimeout" updated
DGMGRL>
```

15. Enable the Data Guard broker configuration and show the results.

Note: To enable the configuration, you might need to wait for a minute.

```
DGMGRL> enable configuration
Enabled.
DGMGRL> show configuration

Configuration - dg_config

Protection Mode: MaxPerformance
Members:
eastdb - Primary database
westdb - Physical standby database

Fast-Start Failover: DISABLED

Configuration Status:
SUCCESS      (status updated 75 seconds ago)

DGMGRL>
```

16. Wait until the physical standby database is synchronized. It may take a few moments for the Lag for the standby database to clear.

Note: Do not continue with labs until the Apply Lag is cleared.

```
DGMGRL> show database westdb

Database - westdb

Role: PHYSICAL STANDBY
Intended State: APPLY-ON
Transport Lag: 0 seconds (computed 1 second ago)
Apply Lag: 0 seconds (computed 1 second ago)
Average Apply Rate: 40.00 KByte/s
Real Time Query: OFF
Instance(s):
    westdb1 (apply instance)
    westdb2

Database Status:
SUCCESS

DGMGRL>
```

17. Close all of your terminal sessions.

Practice 13-5: Resynchronizing a Physical Standby Database by Using RMAN Incremental Backup (Case 5)

Overview

The following steps show how to resolve problems of missing or corrupted archive log file, an unrecoverable archive gap, or need to roll standby forward in time without applying a large number of archivelog files.

Estimated Time: 35 mins

Assumption

- The primary database is in the normal state, but the physical standby database is behind due to the unrecoverable archive gap.

Tasks

1. **(Session 1)** Return to the terminal window on enode01 to stop the Data Guard broker in the primary database. Then exit SQL*Plus.

```
[oracle@enode01 ~]$ sqlplus / as sysdba
SQL> alter system set dg_broker_start=false;
System altered.
SQL> exit
[oracle@enode01 ~]$
```

2. **(Session 2)** Return to the terminal session on wnode03 to stop the Data Guard broker in the standby database.

```
SQL> alter system set dg_broker_start=false;
System altered.
SQL>
```

3. Stop the managed recovery process (MRP) on the STANDBY database.

```
SQL> ALTER DATABASE RECOVER MANAGED STANDBY DATABASE CANCEL;
Database altered.
SQL> exit
[oracle@wnode03 ~]$
```

4. Stop and mount the standby database in a MOUNT state in the remaining steps.

```
[oracle@wnode03 ~]$ srvctl stop database -db westdb
[oracle@wnode03 ~]$ srvctl start database -db westdb -startoption mount
[oracle@wnode03 ~]$
```

5. Determine the `checkpoint_change#` of the STANDBY (`westdb`) database. On the standby database, find the SCN that will be used for the incremental backup at the primary database: You need to use the lowest SCN from the queries below. Then exit SQL*Plus.

```
[oracle@wnode03 ~]$ sqlplus / as sysdba

SQL> select min(checkpoint_change#) from v$logfile_header;

MIN(CHECKPOINT_CHANGE#)
-----
3802036

SQL> select min(f.fhscn) from x$kcvfh f, v$datafile d
where f.hxfil =d.file# and d.enabled != 'READ ONLY';

MIN(CHECKPOINT_CHANGE#)
-----
3802036

SQL> exit
[oracle@wnode03 ~]$
```

You need to use the “lowest SCN” from the queries; in this example SCN is 3802036. Therefore, from the above, you need to back up from SCN 3802036.

6. **(Session 1)** Return to the terminal session on `enode01` to take an incremental backup of the PRIMARY database. In RMAN, connect to the primary database and create an incremental backup from the SCN derived in the previous step.

```
[oracle@enode01 ~]$ rman
RMAN> connect target /

connected to target database: EASTDB (DBID=814727097)

RMAN> BACKUP INCREMENTAL FROM SCN 3802036 DATABASE FORMAT
' /tmp/ForStandby_%U' tag 'FORSTANDBY';

Starting backup at 25-OCT-15
using target database control file instead of recovery catalog
allocated channel: ORA_DISK_1
channel ORA_DISK_1: SID=1 instance=eastdb1 device type=DISK
channel ORA_DISK_1: starting full datafile backup set
channel ORA_DISK_1: specifying datafile(s) in backup set
input datafile file number=00003
name=+DATA/EASTDB/DATAFILE/sysaux.291.892718045
input datafile file number=00005
name=+DATA/EASTDB/DATAFILE/example.279.89271829
9
input datafile file number=00001
name=+DATA/EASTDB/DATAFILE/system.290.892718111
input datafile file number=00002
name=+DATA/EASTDB/DATAFILE/undotbs2.285.8927212
43
input datafile file number=00004
name=+DATA/EASTDB/DATAFILE/undotbs1.276.8927181
79
input datafile file number=00007 name=+DATA/eastdb/bartbs.dbf
input datafile file number=00008 name=+FRA/eastdb/online_tbs01.dbf
```

```

input datafile file number=00006
name=+DATA/EASTDB/DATAFILE/users.289.892929237
channel ORA_DISK_1: starting piece 1 at 25-OCT-15
channel ORA_DISK_1: finished piece 1 at 25-OCT-15
piece handle=/tmp/ForStandby_0vqkjvgj_1_1 tag=FORSTANDBY comment=NONE
channel ORA_DISK_1: backup set complete, elapsed time: 00:04:17
channel ORA_DISK_1: starting full datafile backup set
channel ORA_DISK_1: specifying datafile(s) in backup set
including current control file in backup set
channel ORA_DISK_1: starting piece 1 at 25-OCT-15
channel ORA_DISK_1: finished piece 1 at 25-OCT-15
piece handle=/tmp/ForStandby_10qkjvol_1_1 tag=FORSTANDBY comment=NONE
channel ORA_DISK_1: backup set complete, elapsed time: 00:00:01
Finished backup at 25-OCT-15

RMAN> exit
[oracle@enode01 ~] $

```

7. Transfer all backup sets to wnode03. All backups created in the previous step must be transferred from the primary to the standby system.

Note: If the backup is written to the NFS device in your production environment, this step can be skipped.

```

[oracle@enode01 ~]$ scp /tmp/ForStandby_* wnode03:/tmp
oracle@wnode03's password: <oracle>
ForStandby_0vqkjvgj_1_1          100%   151MB   30.2MB/s   00:05
ForStandby_10qkjvol_1_1          100%    13MB   13.0MB/s   00:01
[oracle@enode01 ~] $

```

8. **(Session 2)** Return to the terminal session on wnode03 to catalog the backups in STANDBY control file. For the standby database to know about the backups, catalog the backup pieces on the STANDBY database. Then exit RMAN.

```

[oracle@wnode03 ~]$ rman
RMAN> connect target /

connected to target database: EASTDB (DBID=814727097)

RMAN> CATALOG START WITH '/tmp/ForStandby';

using target database control file instead of recovery catalog
searching for all files that match the pattern /tmp/ForStandby

List of Files Unknown to the Database
=====
File Name: /tmp/ForStandby_0vqkjvgj_1_1
File Name: /tmp/ForStandby_10qkjvol_1_1

Do you really want to catalog the above files (enter YES or NO)? yes
cataloging files...
cataloging done

List of Cataloged Files
=====
File Name: /tmp/ForStandby_0vqkjvgj_1_1
File Name: /tmp/ForStandby_10qkjvol_1_1

RMAN>

```

9. Use the RMAN RECOVER command with the NOREDO option to apply the incremental backup to the standby database. All changed blocks captured in the incremental backup are updated at the standby database, bringing it up-to-date with the primary database.

```
RMAN> RECOVER DATABASE NOREDO;

Starting recover at 25-OCT-15
allocated channel: ORA_DISK_1
channel ORA_DISK_1: SID=53 instance=westdb1 device type=DISK
channel ORA_DISK_1: starting incremental datafile backup set restore
channel ORA_DISK_1: specifying datafile(s) to restore from backup set
destination for restore of datafile 00001:
+DATA/westdb/DATAFILE/system.284.892725097
destination for restore of datafile 00002:
+DATA/westdb/DATAFILE/undotbs2.286.892725109
destination for restore of datafile 00003:
+DATA/westdb/DATAFILE/sysaux.287.892725103
destination for restore of datafile 00004:
+DATA/westdb/DATAFILE/undotbs1.276.892725221
destination for restore of datafile 00005:
+DATA/westdb/DATAFILE/example.281.892725093
destination for restore of datafile 00006:
+DATA/westdb/DATAFILE/users.294.892725321
destination for restore of datafile 00007:
+DATA/westdb/DATAFILE/bartbs.303.892930675
destination for restore of datafile 00008: +FRA/WESTDB/online_tbs.dbf
channel ORA_DISK_1: reading from backup piece
/tmp/ForStandby_1cqkk6oi_1_1
channel ORA_DISK_1: piece handle=/tmp/ForStandby_1cqkk6oi_1_1
tag=FORSTANDBY
channel ORA_DISK_1: restored backup piece 1
channel ORA_DISK_1: restore complete, elapsed time: 00:00:45

Finished recover at 25-OCT-15

RMAN> exit
[oracle@wnode03 ~] $
```

10. **(Session 1)** Return to the terminal session on enode01. In RMAN, connect to the PRIMARY database and create a standby control file backup. Then exit RMAN.

```
[oracle@enode01 ~]$ rman
RMAN> connect target /

connected to target database: EASTDB (DBID=814727097)

RMAN> BACKUP CURRENT CONTROLFILE FOR STANDBY FORMAT
'./tmp/ForStandbyCTRL.bck';

Starting backup at 25-OCT-15
using target database control file instead of recovery catalog
allocated channel: ORA_DISK_1
channel ORA_DISK_1: SID=104 instance=eastdb1 device type=DISK
channel ORA_DISK_1: starting full datafile backup set
channel ORA_DISK_1: specifying datafile(s) in backup set
including standby control file in backup set
channel ORA_DISK_1: starting piece 1 at 25-OCT-15
channel ORA_DISK_1: finished piece 1 at 25-OCT-15
```

```

piece handle=/tmp/ForStandbyCTRL.bck tag=TAG20151025T170317
comment=NONE
channel ORA_DISK_1: backup set complete, elapsed time: 00:00:03
Finished backup at 25-OCT-15

RMAN> exit
[oracle@enode01 ~] $

```

11. Copy the standby control file backup to wnode03.

```

[oracle@enode01 ~]$ scp /tmp/ForStandbyCTRL.bck wnode03:/tmp
oracle@wnode03's password: <oracle>
ForStandbyCTRL.bck                                100%    10MB   9.9MB/s   00:00
[oracle@enode01 ~] $

```

12. (**Session 2**) Return to the terminal session on wnode03. Capture datafile information in the STANDBY database (westdb). You now need to refresh the standby control file from the primary control file (for standby) backup. However, because the datafile names are likely to be different from the primary, save the name of the datafiles on standby first, which you can refer to after restoring the control file from the primary backup to verify if there is any discrepancy. Run the following query from standby and save the results for further use. Then exit SQL*Plus.

Note: The output may be different.

```

[oracle@wnode03 ~]$ sqlplus / as sysdba
SQL> spool datafile_names_step29.txt
SQL> set lines 200
SQL> col name format a60
SQL> select file#, name from v$datafile order by file#;

FILE# NAME
-----
1 +DATA/WESTDB/DATAFILE/system.284.892725097
2 +DATA/WESTDB/DATAFILE/undotbs2.286.892725109
3 +DATA/WESTDB/DATAFILE/sysaux.287.892725103
4 +DATA/WESTDB/DATAFILE/undotbs1.276.892725221
5 +DATA/WESTDB/DATAFILE/example.281.892725093
6 +DATA/WESTDB/DATAFILE/users.294.892725321
7 +DATA/WESTDB/DATAFILE/bartbs.303.892930675
8 +FRA/WESTDB/online_tbs.dbf

8 rows selected.

SQL> spool off
SQL> exit
[oracle@wnode03 ~] $

```

13. Start the STANDBY database (westdb) in NOMOUNT state to restore the standby control file using RMAN.

```

[oracle@wnode03 ~]$ srvctl stop database -db westdb
[oracle@wnode03 ~]$ srvctl start database -db westdb -startoption
nomount
[oracle@wnode03 ~]$ rman
RMAN> connect target /
connected to target database: EASTDB (not mounted)

```

```
RMAN> RESTORE STANDBY CONTROLFILE FROM '/tmp/ForStandbyCTRL.bck';

Starting restore at 25-OCT-15
using target database control file instead of recovery catalog
allocated channel: ORA_DISK_1
channel ORA_DISK_1: SID=41 instance=westdb1 device type=DISK

channel ORA_DISK_1: restoring control file
channel ORA_DISK_1: restore complete, elapsed time: 00:00:03
output file name=+DATA/WESTDB/CONTROLFILE/current.293.893009367
output file name=+FRA/WESTDB/CONTROLFILE/current.327.893009369
Finished restore at 25-OCT-15

RMAN> exit
[oracle@wnode03 ~] $
```

14. Shut down the STANDBY database (westdb) and start it up in MOUNT state:

```
[oracle@wnode03 ~] $ srvctl stop database -db westdb
[oracle@wnode03 ~] $ srvctl start database -db westdb -startoption mount
[oracle@wnode03 ~] $
```

15. Catalog datafiles in STANDBY because the primary and standby databases have different directory structure and datafile names. If the primary and standby databases have identical directory structure and datafile names, this step can be skipped. Perform the following step in STANDBY for each disk group where the datafile directory structure between primary and standby is different.

```
[oracle@wnode03 ~] $ rman

RMAN> connect target /
connected to target database: EASTDB (DBID=814727097, not open)

RMAN> CATALOG START WITH '+DATA/westdb/datafile/';

Starting implicit crosscheck backup at 25-OCT-15
using target database control file instead of recovery catalog
allocated channel: ORA_DISK_1
Crosschecked 7 objects
Finished implicit crosscheck backup at 25-OCT-15

Starting implicit crosscheck copy at 25-OCT-15
using channel ORA_DISK_1
Crosschecked 2 objects
Finished implicit crosscheck copy at 25-OCT-15

searching for all files in the recovery area
cataloging files...
cataloging done

List of Cataloged Files
=====
File Name: +FRA/WESTDB/online_tbs.dbf
File Name:
+FRA/WESTDB/ARCHIVELOG/2015_10_25/thread_1_seq_74.596.893985301
File Name:
```

```
+FRA/WESTDB/ARCHIVELOG/2015_10_25/thread_1_seq_75.473.894040281
File Name:
+FRA/WESTDB/ARCHIVELOG/2015_10_25/thread_2_seq_75.550.893997699
File Name:
+FRA/WESTDB/ARCHIVELOG/2015_10_25/thread_2_seq_79.600.894042015
File Name:
+FRA/WESTDB/ARCHIVELOG/2015_10_25/thread_2_seq_80.599.894040899
File Name:
+FRA/WESTDB/ARCHIVELOG/2015_10_25/thread_1_seq_77.555.894042355
File Name:
+FRA/WESTDB/ARCHIVELOG/2015_10_24/thread_2_seq_72.601.893939325
File Name:
+FRA/WESTDB/ARCHIVELOG/2015_10_24/thread_1_seq_71.584.893938645
File Name:
+FRA/WESTDB/ARCHIVELOG/2015_10_24/thread_1_seq_72.586.893940017
File Name:
+FRA/WESTDB/ARCHIVELOG/2015_10_24/thread_2_seq_73.543.893940547
File Name:
+FRA/WESTDB/ARCHIVELOG/2015_10_24/thread_1_seq_73.579.893966087

searching for all files that match the pattern +DATA/westdb/datafile/

List of Files Unknown to the Database
=====
File Name: +DATA/westdb/DATAFILE/example.281.892725093
File Name: +DATA/westdb/DATAFILE/system.284.892725097
File Name: +DATA/westdb/DATAFILE/sysaux.287.892725103
File Name: +DATA/westdb/DATAFILE/undotbs2.286.892725109
File Name: +DATA/westdb/DATAFILE/undotbs1.276.892725221
File Name: +DATA/westdb/DATAFILE/users.294.892725321
File Name: +DATA/westdb/DATAFILE/bartbs.303.892930675

Do you really want to catalog the above files (enter YES or NO)? yes
cataloging files...
cataloging done

List of Cataloged Files
=====
File Name: +DATA/westdb/DATAFILE/example.281.892725093
File Name: +DATA/westdb/DATAFILE/system.284.892725097
File Name: +DATA/westdb/DATAFILE/sysaux.287.892725103
File Name: +DATA/westdb/DATAFILE/undotbs2.286.892725109
File Name: +DATA/westdb/DATAFILE/undotbs1.276.892725221
File Name: +DATA/westdb/DATAFILE/users.294.892725321
File Name: +DATA/westdb/DATAFILE/bartbs.303.892930675

RMAN>
```

16. (**Session 1**) Return to the terminal session on enode01. If any datafiles have been added to Primary AFTER SCN 3802036, they will also have to be restored to the standby host (see Note 1531031.1 Steps to perform for Rolling forward a standby database using RMAN incremental backup when datafile is added to primary) and catalogued as shown above before doing the switch. To determine if any files have been added to Primary since the standby current SCN:

```
[oracle@enode01 ~]$ sqlplus / as sysdba
SQL> SELECT FILE#, NAME FROM V$DATAFILE WHERE CREATION_CHANGE# >
```

```
3802036;

no rows selected

SQL>
```

17. **(Session 2)** Return to the terminal session on wnode03. If the above query returns with zero rows, you can switch the datafiles. This will rename the datafiles to its correct name at the standby site:

```
RMAN> SWITCH DATABASE TO COPY;

datafile 1 switched to datafile copy
"+DATA/westdb/DATAFILE/system.284.892725097"
datafile 2 switched to datafile copy
"+DATA/westdb/DATAFILE/undotbs2.286.892725109"
datafile 3 switched to datafile copy
"+DATA/westdb/DATAFILE/sysaux.287.892725103"
datafile 4 switched to datafile copy
"+DATA/westdb/DATAFILE/undotbs1.276.892725221"
datafile 5 switched to datafile copy
"+DATA/westdb/DATAFILE/example.281.892725093"
datafile 6 switched to datafile copy
"+DATA/westdb/DATAFILE/users.294.892725321"
datafile 7 switched to datafile copy
"+DATA/westdb/DATAFILE/bartbs.303.892930675"
datafile 8 switched to datafile copy "+FRA/WESTDB/online_tbs.dbf"

RMAN>
```

At this point, you can compare the query output from step 29 for any discrepancy (other than newly added datafiles) to ensure you have all the datafiles added in standby.

18. On the STANDBY database (westdb), clear all standby redo log groups:

```
RMAN> select group# from v$standby_log order by 1;

GROUP#
-----
      5
      6
      7
      8
      9
     10
     13
     14

8 rows selected

RMAN> ALTER DATABASE CLEAR LOGFILE GROUP 5;
Statement processed

RMAN> ALTER DATABASE CLEAR LOGFILE GROUP 6;
Statement processed

RMAN> ALTER DATABASE CLEAR LOGFILE GROUP 7;
```

```
Statement processed

RMAN> ALTER DATABASE CLEAR LOGFILE GROUP 8;
Statement processed

RMAN> ALTER DATABASE CLEAR LOGFILE GROUP 9;
Statement processed

RMAN> ALTER DATABASE CLEAR LOGFILE GROUP 10;
Statement processed

RMAN> ALTER DATABASE CLEAR LOGFILE GROUP 13;
Statement processed

RMAN> ALTER DATABASE CLEAR LOGFILE GROUP 14;
Statement processed

RMAN>
```

19. Open the STANDBY database in READ ONLY mode.

```
RMAN> alter database open read only;
Statement processed

RMAN>
```

20. On the STANDBY database, start the MRP.

```
RMAN> ALTER DATABASE RECOVER MANAGED STANDBY DATABASE DISCONNECT;
Statement processed

RMAN>
```

21. Restart the Data Guard broker.

```
RMAN> alter system set dg_broker_start=true;
Statement processed

RMAN> exit
[oracle@wnode03 ~] $
```

22. Restart the physical standby database.

```
[oracle@wnode03 ~] $ srvctl stop database -db westdb
[oracle@wnode03 ~] $ srvctl start database -db westdb
[oracle@wnode03 ~] $
```

23. (**Session 1**) Return to the terminal session on enode01 to restart the Data Guard broker.

```
SQL> alter system set dg_broker_start=true;

System altered.

SQL> exit
[oracle@enode01 ~] $
```

24. Using DGMGRL, display the status of the data guard configuration.

Note: You may or may not see warning messages such as ORA-16525, ORA-16665, ORA-16810, or ORA-16817 due to the lag and overall performance of the virtual machine environment. The warning messages will be cleared in a few minutes.

```
[oracle@enode01 ~]$ dgmgrl
DGMGRL> connect sysdg/oracle_4U@eastdb
Connected as SYSDG.
DGMGRL> show configuration

Configuration - dg_config

Protection Mode: MaxPerformance
Members:
  eastdb - Primary database
  westdb - Physical standby database

Fast-Start Failover: DISABLED

Configuration Status:
  SUCCESS      (status updated 20 seconds ago)

DGMGRL>
```

25. Wait until the physical standby database is synchronized. It may take a few moments for the Lag for the standby database to clear.

Note: Do not continue with the practices until the Apply Lag is cleared.

```
DGMGRL> show database westdb

Database - westdb

  Role:          PHYSICAL STANDBY
  Intended State: APPLY-ON
  Transport Lag:   0 seconds (computed 1 second ago)
  Apply Lag:    0 seconds (computed 1 second ago)
  Average Apply Rate: 40.00 KByte/s
  Real Time Query: OFF
  Instance(s):
    westdb1 (apply instance)
    westdb2

  Database Status:
  SUCCESS

DGMGRL>
```

26. Exit all of your terminal sessions.

Practice 13-6: Recovering the Primary Database by Using Archival Backup (Case 6)

Overview

The following steps show how to restore your primary database back to the time when the archival backup was created (practice 6-1). After recovering the primary database, you will have to resynchronize the physical standby database.

Estimated Time: 25 mins

Assumption

- Completed practice 5
- Flashback Database is enabled in the physical standby database

Task 1: Recovering the Primary Database

1. From your classroom PC desktop, execute `ssh oracle@enode01` to open a terminal session on enode01 as the `oracle` user.

```
[Classroom PC] $ ssh oracle@enode01
oracle@enode01's password: <oracle>
[oracle@enode01 ~] $
```

2. Make sure that you set up your environment variables correctly for the `oracle` user.

```
[oracle@enode01 ~] $ . oraenv
ORACLE_SID = [oracle]? eastdb
The Oracle base has been set to /u01/app/oracle
[oracle@enode01 ~] $ export ORACLE_SID=eastdb1
[oracle@enode01 ~] $
```

3. Mount one of the primary database instances.

```
[oracle@enode01 ~] $ srvctl stop database -db eastdb
[oracle@enode01 ~] $ srvctl start instance -db eastdb -instance eastdb1
-startoption mount
[oracle@enode01 ~] $
```

4. Connect to the target database using RMAN.

```
[oracle@enode01 ~] $ rman target /
RMAN>
```

5. Confirm the name of the restore point to be used, `KEEPDB` in this example:

```
RMAN> LIST RESTORE POINT ALL;

SCN          RSP Time   Type        Time      Name
-----  -----
1914331                                26-MAR-16  KEEPDB6_1

RMAN>
```

6. At the **MOUNT** state, restore and recover your database by using your restore point:

```
RMAN> RESTORE DATABASE UNTIL RESTORE POINT 'KEEPDB6_1';

Starting restore at 26-MAR-16
using target database control file instead of recovery catalog
allocated channel: ORA_DISK_1
channel ORA_DISK_1: SID=43 instance=eastdb1 device type=DISK

channel ORA_DISK_1: starting datafile backup set restore
channel ORA_DISK_1: specifying datafile(s) to restore from backup set
channel ORA_DISK_1: restoring datafile 00001 to
+DATA/EASTDB/DATAFILE/system.290.905548821
channel ORA_DISK_1: restoring datafile 00002 to
+DATA/EASTDB/DATAFILE/undotbs2.285.907524789
channel ORA_DISK_1: restoring datafile 00003 to
+DATA/EASTDB/DATAFILE/sysaux.291.905548767
channel ORA_DISK_1: restoring datafile 00004 to
+DATA/EASTDB/DATAFILE/undotbs1.276.905548889
channel ORA_DISK_1: restoring datafile 00005 to
+DATA/EASTDB/DATAFILE/example.279.905549011
channel ORA_DISK_1: restoring datafile 00006 to
+DATA/EASTDB/DATAFILE/users.289.905548887
channel ORA_DISK_1: reading from backup piece
/home/oracle/backup/0gr1flck_1_1
channel ORA_DISK_1: piece handle=/home/oracle/backup/0gr1flck_1_1
tag=KEEPDB
channel ORA_DISK_1: restored backup piece 1
channel ORA_DISK_1: restore complete, elapsed time: 00:03:45
Finished restore at 26-MAR-16

RMAN> RECOVER DATABASE UNTIL RESTORE POINT 'KEEPDB6_1';

Starting recover at 26-MAR-16
using channel ORA_DISK_1

starting media recovery

archived log for thread 1 with sequence 25 is already on disk as file
+FRA/EASTDB/ARCHIVELOG/2016_03_26/thread_1_seq_25.256.907532001
archived log for thread 1 with sequence 26 is already on disk as file
+FRA/EASTDB/ARCHIVELOG/2016_03_26/thread_1_seq_26.276.907532469
archived log for thread 2 with sequence 9 is already on disk as file
+FRA/EASTDB/ARCHIVELOG/2016_03_26/thread_2_seq_9.289.907532001
archived log file
name=+FRA/EASTDB/ARCHIVELOG/2016_03_26/thread_1_seq_25.256.907532001
thread=1 sequence=25
archived log file
name=+FRA/EASTDB/ARCHIVELOG/2016_03_26/thread_2_seq_9.289.907532001
thread=2 sequence=9
media recovery complete, elapsed time: 00:00:04
Finished recover at 26-MAR-16

RMAN>
```

7. Because your database is now at an earlier point in time, open it with the RESETLOGS option, and display the newly created DBID:

```
RMAN> ALTER DATABASE OPEN RESETLOGS;

Statement processed

RMAN> SELECT DBID FROM V$DATABASE;

DBID
-----
827681294

RMAN> exit
[oracle@enode01 ~] $
```

8. Restart the recovered database.

```
[oracle@enode01 ~] $ srvctl stop database -db eastdb
[oracle@enode01 ~] $ srvctl start database -db eastdb
[oracle@enode01 ~] $
```

Note: In the Data Guard environments, the physical standby database must be synchronized. In most of the environments, Oracle also recommends to perform a new backup after a recovery.

Task 2: Resynchronizing the Physical Standby Database

9. On the primary database, determine an SCN that is at least two SCNs prior to the SCN where the OPEN RESETLOGS command was issued. This is necessary to enable the standby to recover properly through OPEN RESETLOGS. Use the following query to find the “BEFORE RESETLOGS” SCN.

```
[oracle@enode01 ~] $ sqlplus / as sysdba
SQL> SELECT TO_CHAR(resetlogs_change# - 2) FROM v$database;
TO_CHAR (RESETLOGS_CHANGE#-2)
-----
00000000
SQL>
```

10. From your classroom PC desktop, execute `ssh oracle@wnode03` to open a terminal session on enode01 as the oracle user.

```
[Classroom PC] $ ssh oracle@wnode03
oracle@wnode03's password: <oracle>
[oracle@wnode03 ~] $
```

11. Make sure that you set up your environment variables correctly for the `oracle` user.

```
[oracle@wnode03 ~] $ . oraenv
ORACLE_SID = [oracle]? eastdb
The Oracle base has been set to /u01/app/oracle
[oracle@wnode03 ~] $ export ORACLE_SID=westdb1
[oracle@wnode03 ~] $
```

12. In the same terminal, obtain the current SCN by using the following query.

```
SQL> SELECT to_char(current_scn) FROM v$database;  
TO_CHAR(CURRENT_SCN)  
-----  
00000000  
SQL>
```

13. Flash back the standby database to the “before RESETLOGS” SCN that you queried in step 10.

Note: Use your SCN number.

```
SQL> FLASHBACK STANDBY DATABASE TO SCN 00000000;  
Flashback complete.  
SQL>
```

14. Restart managed recovery on the standby database. The standby database will be ready to receive and apply logs from the primary database.

```
SQL> ALTER DATABASE RECOVER MANAGED STANDBY DATABASE DISCONNECT;  
Database altered.  
SQL>
```

15. **(Session 1)** Return to the terminal session on enode01 to start the redo transport service.

Note: This step is to speed up the synchronization and is optional.

```
SQL> alter system set log_archive_dest_state_2=enable;  
System altered.  
SQL> exit  
[oracle@enode01 ~]$
```

16. Determine the status of the Data Guard configuration.

Note: Make sure that your Data Guard configuration status is still fine. If ORA-16525, ORA-16810, and ORA-16857 appear, wait a few minutes or restart the standby database to clear the error message.

```
[oracle@enode01 ~]$ dgmgrl  
DGMGRL> connect sysdg/oracle_4U@eastdb  
Connected as SYSDG.  
DGMGRL> show configuration  
Configuration - dg_config  
  
Protection Mode: MaxPerformance  
Members:  
eastdb - Primary database  
westdb - Physical standby database  
  
Fast-Start Failover: DISABLED  
  
Configuration Status:  
SUCCESS (status updated 33 seconds ago)  
  
DGMGRL>
```

17. Wait until the physical standby database is synchronized. It may take a few moments for the Lag for the standby database to clear.

Note: Do not continue with the practices until the Apply Lag is cleared.

```
DGMGRL> show database westdb

Database - westdb

Role: PHYSICAL STANDBY
Intended State: APPLY-ON
Transport Lag: 0 seconds (computed 1 second ago)
Apply Lag: 0 seconds (computed 1 second ago)
Average Apply Rate: 40.00 KByte/s
Real Time Query: OFF
Instance(s):
    westdb1 (apply instance)
    westdb2

Database Status:
SUCCESS

DGMGRL>
```

18. Exit your terminal sessions.

Practice 13-7: Re-creating the Physical Standby Database (Case 7)

Overview

The following steps show how to re-create your physical standby database in case of unrecoverable issues.

Estimated Time: 60 mins

Assumption

- Flashback Database is NOT enabled in the physical standby database.
- The primary database is in the normal state. However, the physical standby database is in an unrecoverable state.

Task 1: Clean Up the Physical Standby Environment

1. **(ednode01)** Disable the redo transport service to the physical standby database.

```
[Classroom PC ~]$ ssh oracle@ednode01
oracle@ednode01's password: <oracle>
[oracle@ednode01 ~]$ . oraenv
ORACLE_SID = [oracle] ? eastdb
The Oracle base has been set to /u01/app/oracle
[oracle@ednode01 ~]$ 
[oracle@ednode01 ~]$ export ORACLE_SID=eastdb1
[oracle@ednode01 ~]$ sqlplus / as sysdba
SQL> alter system set log_archive_dest_state_2='DEFER' scope=both
sid='*';
System altered.
SQL>
```

2. **(wnode03)** Drop westdb using DBCA.

```
[Classroom PC ~]$ ssh -X oracle@wnode03
oracle@wnode03's password: <oracle>
[oracle@wnode03 ~]$ . oraenv
ORACLE_SID = [oracle] ? westdb
The Oracle base has been set to /u01/app/oracle
[oracle@wnode03 ~]$ dbca
[oracle@wnode03 ~]$
```

3. **(wnode03)** Clean up the following directories.

```
[oracle@wnode03 ~]$ ssh wnode03 rm -rf
/u01/app/oracle/product/12.1.0/dbhome_1/dbs/*stdb*
[oracle@wnode03 ~]$ ssh wnode04 rm -rf
/u01/app/oracle/product/12.1.0/dbhome_1/dbs/*stdb*
[oracle@wnode03 ~]$
```

4. **(wnode03)** Clean up the ASM Disk Group.

```
[oracle@wnode03 ~]$ su - grid
Password: <oracle>
[grid@wnode03 ~]$ . oraenv
ORACLE_SID = [grid] ? +ASM1
The Oracle base has been set to /u01/app/grid
```

```
[grid@wnode03 ~]$ asmcmd rm -rf data/*STDB*
[grid@wnode03 ~]$ asmcmd rm -rf fra/*STDB*
[grid@wnode03 ~]$ exit
[oracle@wnode03 ~]$
```

Task 2: Catchup Practice 3

5. (enode01) Catchup: Practice 3-1 (tnsnames.ora)

```
[oracle@enode01 ~]$ scp /home/oracle/labs/tnsnames_lab31.ora
enode01:/u01/app/oracle/product/12.1.0/dbhome_1/network/admin/tns
names.ora
tnsnames_lab31.ora                                100% 2363      2.3KB/s  00:00

[oracle@enode01 ~]$ scp /home/oracle/labs/tnsnames_lab31.ora
enode02:/u01/app/oracle/product/12.1.0/dbhome_1/network/admin/tns
names.ora
tnsnames_lab31.ora                                100% 2363      2.3KB/s  00:00

[oracle@enode01 ~]$ scp /home/oracle/labs/tnsnames_lab31.ora
wnode03:/u01/app/oracle/product/12.1.0/dbhome_1/network/admin/tns
names.ora
oracle@wnode03's password: <oracle>
tnsnames_lab31.ora                                100% 2363      2.3KB/s  00:00

[oracle@enode01 ~]$ scp /home/oracle/labs/tnsnames_lab31.ora
wnode04:/u01/app/oracle/product/12.1.0/dbhome_1/network/admin/tns
names.ora
oracle@wnode04's password: <oracle>
tnsnames_lab31.ora
[oracle@enode01 ~]$
```

6. (enode01) Review the following steps in Practice 3-3 (listener.ora)

- Step 8 (enode01)
- Step 18 (enode02)
- Step 28 (wnode03)
- Step 37 (wnode04)

Task 3: Catchup Practice 4

7. (enode01) Review: Practice 4-1

Note: The output of the standby log file configuration may be different depending on how far you have completed.

```
[oracle@enode01 ~]$ sqlplus / as sysdba
SQL> @/home/oracle/labs/review41.sql
SQL> exit
[oracle@enode01 ~]$
[oracle@enode01 ~]$ echo
eastdb1:/u01/app/oracle/product/12.1.0/dbhome_1:N >> /etc/oratab
[oracle@enode01 ~]$ ssh enode02
```

```
[oracle@enode02 ~]$ echo
eastdb2:/u01/app/oracle/product/12.1.0/dbhome_1:N >> /etc/oratab
[oracle@enode02 ~]$ exit
[oracle@enode01 ~]$
```

8. (enode01) Catchup: Practice 4-2

- Review step 12 on wnode03
- Complete the following tasks.

```
[oracle@enode01 ~]$ ssh wnode03
Password: <oracle>
[oracle@wnode03 ~]$ echo
westdb1:/u01/app/oracle/product/12.1.0/dbhome_1:N >> /etc/oratab
[oracle@wnode03 ~]$ mkdir -p /u01/app/oracle/admin/westdb/adump
[oracle@enode01 ~]$ ssh wnode04
[oracle@wnode04 ~]$ echo
westdb2:/u01/app/oracle/product/12.1.0/dbhome_1:N >> /etc/oratab
[oracle@wnode04 ~]$ mkdir -p /u01/app/oracle/admin/westdb/adump
[oracle@wnode04 ~]$ exit
logout
Connection to wnode04 closed.
[oracle@wnode03 ~]$ exit
logout
Connection to wnode03 closed.
[oracle@enode01 ~]$ su - grid
Password: <oracle>
[grid@enode01 ~]$ . oraenv
ORACLE_SID = [grid] ? +ASM1
The Oracle base has been set to /u01/app/grid
[grid@enode01 ~]$ asmcmd pwcOPY +DATA/eastdb/orapweastdb
/tmp/orapweastdb
copying +DATA/eastdb/orapweastdb -> /tmp/orapweastdb
[grid@enode01 ~]$ exit
logout
[oracle@enode01 ~]$ scp enode01:/tmp/orapweastdb
wnode03:/u01/app/oracle/product/12.1.0/dbhome_1/dbs/orapwwestdb1
oracle@wnode03's password: <oracle>
orapweastdb                                100%    10KB   10.0KB/s   00:00
Connection to enode01 closed.
[oracle@enode01 ~]$ scp enode01:/tmp/orapweastdb
wnode04:/u01/app/oracle/product/12.1.0/dbhome_1/dbs/orapwwestdb2
oracle@wnode04's password: <oracle>
orapweastdb                                100%    10KB   10.0KB/s   00:00
Connection to enode01 closed.
[oracle@enode01 ~]$
```

9. (wnode03) Perform the following tasks:

- Create a starter initialization file for the physical standby instance
- Start up the temporary standby instance on wnode03 in nomount state.
- Create a physical standby using the RMAN utility.

```
[oracle@wnode03 ~]$ cd $ORACLE_HOME/dbs
[oracle@wnode03 dbs]$ vi initwestdb.ora
```

```

db_name=eastdb
db_unique_name=westdb
db_domain=example.com

:wq!
[oracle@wnode03 dbs]$ export ORACLE_SID=westdb1
[oracle@wnode03 dbs]$ sqlplus / as sysdba
SQL> startup nomount pfile=$ORACLE_HOME/dbs/initwestdb.ora
ORACLE instance started.

Total System Global Area 243269632 bytes
Fixed Size                  2923000 bytes
Variable Size              184550920 bytes
Database Buffers           50331648 bytes
Redo Buffers                5464064 bytes
SQL> exit
[oracle@wnode03 dbs]$ rman target sys/oracle_4U@eastdb auxiliary
sys/oracle_4U@clone
connected to target database: EASTDB (DBID=812282532)
connected to auxiliary database: EASTDB (not mounted)

RMAN> @/home/oracle/labs/cr_phys_standby
                                -- Output Truncated --
RMAN> exit
[oracle@wnode03 ~] $

```

10. (enode01) Enable LOG_ARCHIVE_DEST_2.

Note: If an error appears in the ERROR column, resolve the issue first working with your instructor.

```

[oracle@enode01 ~]$ sqlplus / as sysdba
SQL> alter system set log_archive_dest_state_2=ENABLE;
System altered.
SQL> select dest_id, status, destination, archiver, error, valid_now
from v$archive_dest where dest_id = 2;

  DEST_ID STATUS      DESTINATIO ARCHIVER      E VALID_NOW
-----  -----      -----
          2 VALID       westdb        LGWR          YES

SQL> exit
[oracle@enode01 ~] $

```

11. (wnode03) Create a temporary pfile from spfile on wnode03. Exit SQL*Plus when done.

```

SQL> create pfile='/tmp/init.ora' from spfile;
File created.
SQL> exit
[oracle@wnode03 ~] $

```

12. (wnode03) Modify the temporary pfile to match the standby instance names.

```
[oracle@wnode03 ~]$ vi /tmp/init.ora
```

- Remove the westdb1._* entries.

```
:g/westdb1._/d
```

- b. Replace the eastdb._ entries with the westdb._ entries.

```
:%s/eastdb._/westdb._/g
10 substitutions on 10 lines
```

- c. Replace the eastdb1 entries with the westdb1 entries.

```
:%s/eastdb1/westdb1/g
13 substitutions on 13 lines
```

- d. Replace the eastdb2 entries with the westdb2 entries.

```
:%s/eastdb2/westdb2/g
13 substitutions on 13 lines
```

- e. Update the following entries and save the changes.

```
*.cluster_database=TRUE
*.dispatchers='(PROTOCOL=TCP) (SERVICE=westdbXDB)'
*.log_archive_dest_1='location=USE_DB_RECOVERY_FILE_DEST
valid_for=(ALL_LOGFILES,ALL_ROLES) db_unique_name=westdb'
*.log_archive_dest_2='SERVICE=eastdb SYNC
valid_for=(ONLINE_LOGFILES,PRIMARY_ROLE) db_unique_name=eastdb'
*.log_archive_dest_state_2='enable'

:wq!
[oracle@wnode03 ~]$
```

13. (wnode03) Perform the following tasks.

- Create an spfile in a shared storage for the standby database in RAC.
- Modify the \$ORACLE_HOME/dbs/initwestdb1.ora file on wnode03.
- Create the \$ORACLE_HOME/dbs/initwestdb2.ora file on wnode04.
- Shut down the first standby instance named westdb1.

```
[oracle@wnode03 ~]$ sqlplus / as sysdba
SQL> create spfile='+DATA/westdb/spfilewestdb.ora' from
pfile='/tmp/init.ora';
File created.
SQL> exit
oracle@wnode03 ~]$ vi $ORACLE_HOME/dbs/initwestdb1.ora

spfile='+DATA/westdb/spfilewestdb.ora'

:wq!
[oracle@wnode03 ~]$ scp $ORACLE_HOME/dbs/initwestdb1.ora
wnode04:/u01/app/oracle/product/12.1.0/dbhome_1/dbs/initwestdb2.ora
initwestdb1.ora          100%   39      0.0KB/s  00:00
[oracle@wnode03 ~]$ sqlplus / as sysdba
SQL> shutdown immediate
SQL> exit
[oracle@wnode03 ~]$
```

14. (wnode03) Perform the following tasks.

- Create the clusterware entries for the newly created standby database.
- Start the physical standby database.

- Start the Managed Recovery Process (MRP).

```
[oracle@wnode03 ~]$ sh /home/oracle/labs/register_rac.sh
[oracle@wnode03 ~]$ srvctl start database -db westdb -startoption mount
[oracle@wnode03 ~]$ srvctl status database -db westdb
Instance westdb1 is running on node wnode03
Instance westdb2 is running on node wnode04
[oracle@wnode03 ~]$ sqlplus / as sysdba
SQL> alter database recover managed standby database disconnect;
Database altered.
SQL> exit
[oracle@wnode03 ~]$
```

Note: If you haven't started practice 5, then stop at this point and do practice 5, but if you encountered the issue after practice 5, then move on to the following task to catch up.

Task 4: Catchup Practice 5

15. (enode01) Catchup 5-1

```
[oracle@enode01 ~]$ sqlplus / as sysdba
SQL> alter user sysdg identified by oracle_4U;
User altered.
SQL> alter user sysdg account unlock;
User altered.
SQL> exit;
[oracle@enode01 ~]$ su - grid
Password: <oracle>
[grid@enode01 ~]$ . oraenv
ORACLE_SID = [grid] ? +ASM1
The Oracle base has been set to /u01/app/grid
[grid@enode01 ~]$ asmcmd pwcOPY +DATA/eastdb/orapweastdb
/tmp/orapweastdb
copying +DATA/eastdb/orapweastdb -> /tmp/orapwestdb
[grid@enode01 ~]$ exit
[oracle@enode01 ~]$ scp /tmp/orapweastdb wnode03:/tmp/orapwestdb
oracle@wnode03's password: <oracle>
orapweastdb                                         100%  7680      7.5KB/s   00:00
[oracle@enode01 ~]$
```

16. (wnode03) Catchup 5-1

```
[oracle@wnode03 ~]$ su - grid
Password: <oracle>
[grid@wnode03 ~]$ . oraenv
ORACLE_SID = [grid] ? +ASM1
The Oracle base has been set to /u01/app/grid
[grid@wnode03 ~]$ asmcmd pwcOPY /tmp/orapwestdb
+data/westdb/orapwestdb
copying /tmp/orapwestdb -> +data/westdb/orapwestdb [grid@wnode03 ~]$ exit
[oracle@wnode03 ~]$ srvctl modify database -db westdb -pwfile
+data/westdb/orapwestdb
[oracle@wnode03 ~]$
```

17. Catchup Practice 5-2

- Perform tasks in Appendix (Practice 13-4: Case 4).

18. **Catchup Practice 5-3:** Switch the apply instance to `westdb1` and make it as a preferred instance. Exit DGMGRL.

```
DGMGRL> edit database westdb set state='APPLY-ON' with apply instance =
  westdb1;
Succeeded.
DGMGRL> edit database westdb set property 'ApplyInstanceTimeout' =
  '10';
Property "ApplyInstanceTimeout" updated
DGMGRL> exit
[oracle@wnode03 ~]$
```

19. **Catchup Practice 5-4:** Perform the following tasks properly:

- Step 5
- Step 12 ~ Step 16

20. **Catchup Practice 5-5:** Perform the following tasks properly:

- Step 5 ~ Step 8

21. **Catchup Practice 5-7:** Make sure that fast-start failover is disabled and reset your environment.

Note: You may receive a warning message like "Error: ORA-16525: The Oracle Data Guard broker is not yet available." This warning message will be cleared in a minute.

```
DGMGRL> show fast_start failover;

Fast-Start Failover: DISABLED

Threshold:      50 seconds
Target:        (none)
Observer:       (none)
Lag Limit:     30 seconds
Shutdown Primary: TRUE
Auto-reinstate: TRUE
Observer Reconnect: (none)
Observer Override: FALSE

Configurable Failover Conditions
  Health Conditions:
    Corrupted Controlfile      YES
    Corrupted Dictionary       YES
    Inaccessible Logfile       NO
    Stuck Archiver             NO
    Datafile Offline           YES

  Oracle Error Conditions:
    (none)

DGMGRL> EDIT CONFIGURATION SET PROTECTION MODE AS MaxPerformance;
Succeeded.
DGMGRL> EDIT DATABASE 'eastdb' SET PROPERTY 'RedoRoutes' =
  '(eastdb:westdb ASYNC)';
Property "RedoRoutes" updated
DGMGRL> EDIT DATABASE 'westdb' SET PROPERTY 'RedoRoutes' =
```

```
' (westdb:eastdb ASYNC) ';
Property "RedoRoutes" updated
DGMGRL> EDIT DATABASE eastdb SET PROPERTY MaxConnections=1;
Property "maxconnections" updated
DGMGRL> EDIT DATABASE westdb SET PROPERTY MaxConnections=1;
Property "maxconnections" updated
DGMGRL>
DGMGRL> show configuration
Configuration - dg_config

Protection Mode: MaxPerformance
Members:
  eastdb - Primary database
  westdb - Physical standby database

Fast-Start Failover: DISABLED

Configuration Status:
SUCCESS
DGMGRL>
```

22. Wait until the physical standby database is synchronized. It may take a few moments for the Lag for the standby database to clear.

Note: Do not continue with labs until the Apply Lag is cleared.

```
DGMGRL> show database westdb

Database - westdb

Role: PHYSICAL STANDBY
Intended State: APPLY-ON
Transport Lag: 0 seconds (computed 1 second ago)
Apply Lag: 0 seconds (computed 1 second ago)
Average Apply Rate: 40.00 KByte/s
Real Time Query: OFF
Instance(s):
  westdb1 (apply instance)
  westdb2

Database Status:
SUCCESS

DGMGRL>
```

23. Exit all of your terminal sessions.

Practice 13-8: Reconfiguring Your Practice Environment (Case 8)

Overview

This practice is to show how to reconfigure your practice environment catching up many of the tasks up through practice 5-7. Because it reconfigures the entire virtual machines, it would take up to 130 minutes. Whenever possible, consider the other troubleshooting cases first working with instructors.

Estimated Time: 130 mins (**Step 1:** 100 mins, **Step 2-4:** 30 mins)

Tasks

1. Open a terminal window and execute the catchup script to reconfigure your environment up through practice 5-7.

```
[Classroom PC ~]$ su -
Password: <oracle>
[root@Classrom PC]# /ovs/seed_pool/catchup_RACnDG.sh
Removing current VMs...
Extracting GDS VMs...
Starting VMs...
VM status...
Name ID Mem VCPUs State Time(s)
Domain-0 0 1024 2 r----- 54609.2
enode01 1 3800 1 -b---- 98.0
enode02 2 3400 1 -b---- 97.2
wnode03 3 3400 1 -b---- 96.9
wnode04 4 3400 1 -b---- 96.1
[root@Classroom PC ~]#
```

Note: Even though VMs are up and running, you may have to wait for 15-20 minutes until the clusterware, database, Data Guard broker, and other components are ready.

2. Determine the status of the Data Guard configuration.

Note: You may or may not see warning messages such as ORA-00604, ORA-01219, ORA-12514, ORA-16525, ORA-16665, ORA-16810, ORA-16782, or ORA-16817 due to the lag and overall performance of the virtual machine environment. The warning messages will be cleared in a few minutes. Make sure that your Data Guard configuration status is fine.

```
[root@Classroom PC ~]# ssh oracle@enode01
oracle@enode01's password: <oracle>
[oracle@enode01 ~]$ . oraenv
ORACLE_SID = [oracle]? eastdb
The Oracle base has been set to /u01/app/oracle
[oracle@enode01 ~]$ dgmgrl
DGMGRL> connect sysdg/oracle_4U@eastdb
Connected as SYSDG.
DGMGRL> show configuration

Configuration - dg_config

Protection Mode: MaxPerformance
```

```

Members:
eastdb - Primary database
westdb - Physical standby database

Fast-Start Failover: DISABLED

Configuration Status:
SUCCESS      (status updated 17 seconds ago)
DGMGRL>

```

- Wait until the physical standby database is synchronized. It may take a few moments for the Lag for the standby database to clear.

Note: Do not continue with the practices until the Apply Lag is cleared.

```

DGMGRL> show database westdb
Database - westdb

Role:          PHYSICAL STANDBY
Intended State:   APPLY-ON
Transport Lag:    0 seconds (computed 1 second ago)
Apply Lag:    0 seconds (computed 1 second ago)
Average Apply Rate: 40.00 KByte/s
Real Time Query: OFF
Instance(s):
  westdb1 (apply instance)
  westdb2

Database Status:
SUCCESS

DGMGRL>

```

- Close all of your terminal sessions.
- Now, your environment is ready for the following scenarios.

Catchup	First, Complete Pre-Tasks	To Start from
Scenario 1	• Practice 6-1	• Practice 6-2
Scenario 2	• Practice 6-1	• Practice 7-1
Scenario 3	• Practice 6-1	• Practice 8-1
Scenario 4	• Practice 6-1 • Skip Practice 9-1	• Practice 9-2
Scenario 5	• Practice 6-1	• Practice 10-1
Scenario 6	• Practice 6-1	• Practice 11-1

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

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