

Oracle Database 12c: Install and Upgrade Workshop

Activity Guide

D77766GC10
Edition 1.0
July 2013
D82687



Copyright © 2013, 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.

Authors

Donna Keesling, Dominique Jeunot, James Spiller

Technical Contributors and Reviewers

Roy Swonger, Ira Singer, Darryl Balaski, Mark Fuller, Carol Tagliaferri, Harald Van Breederode, Joel Goodman, Prakash Jashnani, Gerlinde Frenzen

This book was published using: *Oracle Tutor*

Table of Contents

Practices for Lesson 1: Oracle Database 12c: Overview.....	1-1
Practices for Lesson 1.....	1-2
Practice 1-1: Listing Existing Instances and Oracle Homes	1-3
Practice 1-2: Determining the Locations of Data Files, Control Files and Redo Log Files	1-6
Practice 1-3: Determining the Character Set of Existing Database(s).....	1-18
Practice 1-4: Determining Existing Listeners.....	1-20
Practices for Lesson 2: Oracle Software Installation Basics	2-1
Practices for Lesson 2.....	2-2
Practices for Lesson 3: Installing Oracle Grid Infrastructure for a Standalone Server and Upgrading ASM.....	3-1
Practices for Lesson 3.....	3-2
Practice 3-1: Installing Oracle Grid Infrastructure and Upgrading ASM	3-4
Practice 3-2: Using the Upgraded ASM Instance	3-15
Practice 3-3: Determining Existing Listeners.....	3-23
Practices for Lesson 4: Installing your Oracle Software	4-1
Practices for Lesson 4.....	4-2
Practice 4-1: Installing the Oracle Database 12c Software	4-3
Practices for Lesson 5: Creating an Oracle Database Using DBCA.....	5-1
Practices for Lesson 5.....	5-2
Practice 5-1: Creating a Non-CDB	5-3
Practice 5-2: Creating a CDB	5-21
Practices for Lesson 6: Using Oracle Restart	6-1
Practices for Lesson 6.....	6-2
Practice 6-1: Starting and Stopping Oracle Restart.....	6-3
Practice 6-2: Starting and Stopping Components	6-6
Practice 6-3: Managing Configuration of Components.....	6-8
Practices for Lesson 7: Introduction to Upgrading to Oracle Database 12c	7-1
Practices for Lesson 7.....	7-2
Practice 7-1: Upgrading an Oracle Database 11g Release 2 Database on File System to Oracle Database 12c	7-4
Solution 7-1: Upgrading an Oracle Database 11g Release 2 Database on File System to Oracle Database 12c	7-5
Practice 7-2: Upgrading an Oracle Database 11g Release 2 Database on ASM to Oracle Database 12c	7-6
Solution 7-2: Upgrading an Oracle Database 11g Release 2 Database on ASM to Oracle Database 12c	7-7
Practice 7-3: Upgrading an Oracle Database 11g Release 2 Database to Oracle Database 12c on a New OS or Platform.....	7-8
Solution 7-3: Upgrading an Oracle Database 11g Release 2 Database to Oracle Database 12c on a New OS or Platform.....	7-9
Practice 7-4: Upgrading an Oracle Database 11g Release 2 Database to Oracle Database 12c with Minimal Downtime	7-10
Solution 7-4: Upgrading an Oracle Database 11g Release 2 Database to Oracle Database 12c with Minimal Downtime	7-11
Practices for Lesson 8: Preparing a Database for Upgrade	8-1
Practices for Lesson 8.....	8-2
Practice 8-1: Executing Preliminary Steps and the Pre-Upgrade Script.....	8-3
Practice 8-2: Implementing the Pre-Upgrade Information Tool Recommendations	8-9
Practice 8-3: Completing Prerequisites Steps Before Upgrade.....	8-19

Practice 8-4: Performing a Full Database Backup.....	8-22
Practice 8-5: Performing a Transportable Tablespace Export.....	8-26
Practices for Lesson 9: Upgrading a Database	9-1
Practices for Lesson 9.....	9-2
Practice 9-1: Upgrading a Database by Using DBUA	9-3
Practice 9-2: Exporting a Non-CDB Application	9-17
Practice 9-3: Plugging a Non-CDB into a CDB	9-19
Practices for Lesson 10: Performing Post-Upgrade Tasks	10-1
Practices for Lesson 10.....	10-2
Practice 10-1: Performing Post-Upgrade Actions.....	10-3
Practices for Lesson 11: Migrating Data by Using Oracle Data Pump	11-1
Practices for Lesson 11.....	11-2
Practice 11-1: Performing a Transportable Tablespace Import.....	11-3
Practice 11-2: Importing a Non-CDB Application Into a CDB	11-8
Appendix: Upgrading a Database With Options.....	12-1
Appendix	12-2
Practice A-1: Executing Preliminary Steps and the Pre-Upgrade Tool.....	12-3
Practice A-2: Implementing the Pre-Upgrade Information Tool Recommendations	12-9
Practice A-3: Completing Prerequisites Steps Before Upgrade	12-23
Practice A-4: Performing a Full Database Backup	12-26
Practice A-5: Upgrading a Database by Using DBUA.....	12-30
Practice A-6: Performing Post-Upgrade Operations.....	12-42

Practices for Lesson 1: Oracle Database 12c: Overview

Chapter 1

Practices for Lesson 1

Practices Overview

Background: In the practices of this course, you assume the role of a database administrator (DBA). The operating system (OS) accounts on your computer are:

- The `oracle` user with a password of `oracle`
- The `grid` user with a password of `oracle`
- The `root` user with a password of `oracle`

In these practices, you explore the existing installation, namely:

- List existing instances
- Determine existing software versions and existing Oracle homes
- Determine locations of data files, control files and redo log files
- Determine the character set of existing database(s)
- Determine existing listeners

Practice 1-1: Listing Existing Instances and Oracle Homes

1. List the existing instances, whether they are running or not. There are two ways to find them all.
 - a. Right-click the desktop and click Open Terminal to open a terminal window. Verify that you are logged in as the `oracle` user.

```
$ id  
uid=54321(oracle) gid=54321(oinstall)  
groups=54321(oinstall),54322(dba),54323(oper),54324(backupdba),54325(dgdba),54326(kmdba),54327(asmdba)  
$
```

- b. To list the running instances, you can search for the SMON background process. Any running instance includes the SMON background process. An instance owns several mandatory background processes including PMON, LGWR, DBW0, CKPT, LREG, MMON, and RECO.

```
$ ps -ef | grep smon  
grid      12992      1  0 13:11 ?          00:00:00  asm_smon_+ASM  
oracle    13115      1  0 13:30 ?          00:00:00  ora_smon_dbupgrd  
oracle    13209 12815  0 13:31 pts/2        00:00:00  grep smon  
$
```

There are two running instances, `dbupgrd` and `+ASM`.

Notice that the user running the `dbupgrd` instance is `oracle` and the user running the `+ASM` instance is `grid`.

- c. There can be instances and related databases that are not currently running. You can view them in the `/etc/oratab` file. This file is used by Oracle utilities. It is created by `root.sh` and updated by the Database Configuration Assistant while creating a database or the ASM Configuration Assistant while creating an ASM instance.

```
$ cat /etc/oratab  
#Backup file is  
/u01/app/grid/product/11.2.0/grid/srvrm/admin/oratab.bak.edrsr32p  
1 line added by Agent  
  
# This file is used by ORACLE utilities. It is created by  
root.sh  
  
# and updated by either Database Configuration Assistant while  
creating  
  
# a database or ASM Configuration Assistant while creating ASM  
instance.  
  
# A colon, ':', is used as the field terminator. A new line  
terminates  
  
# the entry. Lines beginning with a pound sign, '#', are  
comments.
```

```
# Entries are of the form:  
# $ORACLE_SID:$ORACLE_HOME:<N|Y>:  
  
#  
# The first and second fields are the system identifier and home  
# directory of the database respectively. The third field  
# indicates  
# to the dbstart utility that the database should , "Y", or  
# should not,  
# "N", be brought up at system boot time.  
#  
#  
# Multiple entries with the same $ORACLE_SID are not allowed.  
#  
#  
#+ASM:/u01/app/grid/product/11.2.0/grid:N # line added by Agent  
dbupgrd:/u01/app/oracle/product/11.2.0/dbhome_2:N # line added  
by Agent  
$
```

Doubled dbupgrd in oratab

All instances declared in the file are running.

Note: The CREATE DATABASE command does not update the /etc/oratab file

2. In the file, notice that the two instances run in different Oracle homes.
 - The **+ASM** ASM instance is running in **/u01/app/grid/product/11.2.0/grid**, the Oracle home for Oracle Grid Infrastructure.
 - The **dbupgrd** Oracle Database instance is running in **/u01/app/oracle/product/11.2.0/dbhome_2**, the Oracle home for Oracle Database 11g Release 2.
3. The Oracle Grid Infrastructure includes ASM and Oracle Restart components. Check if the Oracle Restart component is started.

```
$ pgrep -lf ohasd  
16152 /bin/sh /etc/init.d/init.ohasd run  
16173 /u01/app/grid/product/11.2.0/grid/bin/ohasd.bin reboot  
$
```

It is currently started.

4. Is it automatically started at reboot?
 - a. To use the CRSCTL utility, you must be connected as the grid user.

```
$ su - grid  
Password:  
$ crsctl config has  
-bash: crsctl: command not found  
$
```

- b. Use the `oraenv` utility to set the `PATH` environment variable to include the Grid home. The `oraenv` script uses `/etc/oratab` to set the `ORACLE_SID`, `ORACLE_BASE`, `ORACLE_HOME` and `PATH` environment variables. You will see that the Oracle base changes from one environment to another.

```
$ . oraenv
ORACLE_SID = [grid] ? +ASM
The Oracle base has been changed from /u01/app/oracle to
/u01/app/grid
$ env | grep ORA
ORACLE_SID=+ASM
ORACLE_BASE=/u01/app/grid
ORACLE_HOME /u01/app/grid/product/11.2.0/grid
$-
$ crsctl config has
CRS-4622: Oracle High Availability Services autostart is
enabled.
$-
```

Practice 1-2: Determining the Locations of Data Files, Control Files and Redo Log Files

Overview

In this practice, you will determine the locations of the data files of the `dbupgrd` Oracle database.

For each of the running instances, connect to the RDBMS instance either as `SYS` or `SYSTEM`, and to the ASM instance as `SYSASM`. Then query the `V$DATAFILE` view.

Tasks

1. Execute the following steps to retrieve information related to the `dbupgrd` Oracle database instance.
 - a. Check that you are connected as the `oracle` UNIX user.

```
$ exit  
logout  
$ id  
uid=54321(oracle) gid=54321(install)  
groups=54321(install), 54322(dba), 54323(oper), 54324(backupdba), 5  
4325(dgdba), 54326(kMDBA), 54327(asmDBA)  
$
```

- b. Use the `oraenv` utility to set the `ORACLE_SID` environment variable to the appropriate value. The utility automatically sets the `ORACLE_HOME` appropriately according to the `ORACLE_SID` entered when prompted. In this case, the `ORACLE_HOME` is set to `/u01/app/oracle/product/11.2.0/dbhome_2`. The `oraenv` utility uses `/etc/oratab` to set the `ORACLE_SID`, `ORACLE_BASE`, `ORACLE_HOME` and `PATH` environment variables.

```
$ . oraenv  
ORACLE_SID = [oracle] ? dbupgrd  
The Oracle base has been set to /u01/app/oracle  
  
$ env | grep ORA  
ORACLE_SID=dbupgrd  
ORACLE_BASE=/u01/app/oracle  
ORACLE_HOME=/u01/app/oracle/product/11.2.0/dbhome_2  
$
```

- c. View the database name.

```
$ sqlplus / as sysdba  
  
Connected to:  
Oracle Database 11g Enterprise Edition Release 11.2.0.3.0 -  
64bit Production
```

```
SQL> show parameter instance_type
```

NAME	TYPE	VALUE
instance_type	string	RDBMS

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

NAME
DBUPGRD

```
SQL>
```

- d. View the data file names. **Note:** the suffix numbers for the datafiles can be different on each database

```
SQL> select name from v$datafile;
```

NAME
+DATA/dbupgrd/datafile/system.260.814012925
+DATA/dbupgrd/datafile/sysaux.261.814012959
+DATA/dbupgrd/datafile/undotbs1.262.814012987
+DATA/dbupgrd/datafile/users.264.814013023
+DATA/dbupgrd/datafile/example.265.814013031

```
SQL> EXIT
$
```

The location for the data files of the **DBUPGRD** database is the **+DATA** disk group.
The disk group is managed by an ASM instance.

2. Execute the following steps to retrieve information related to the **+ASM** ASM instance.
- Check that you are connected as the **grid** UNIX user. Use the **su** (substitute user) UNIX command to start a user session as the **grid** user using the **oracle** password.

```
$ id
uid=54321(oracle) gid=54321(oinstall)
groups=54321(oinstall),54322(dba),54323(oper),54324(backupdba),5
4325(dgdba),54326(kmdba),54327(asmdba)
$
$ su - grid
Password:
$ id
uid=54322(grid) gid=54321(oinstall)
groups=54321(oinstall),54322(dba),54327(asmdba),54329(asmadmin)
$
```

- b. Use the `oraenv` utility to set the `ORACLE_SID` environment variable to the appropriate value. The utility automatically sets `ORACLE_HOME` to `/u01/app/grid/product/11.2.0/grid`.

```
$ . oraenv
ORACLE_SID = [grid] ? +ASM
The Oracle base has been changed from /u01/app/oracle to
/u01/app/grid
$ env | grep ORA
ORACLE_SID=+ASM
ORACLE_BASE=/u01/app/grid
ORACLE_HOME=/u01/app/grid/product/11.2.0/grid
$
```

- c. Connect to the instance with the `SYSASM` privilege.

```
$ sqlplus / as sysasm

Connected to:
Oracle Database 11g Enterprise Edition Release 11.2.0.3.0 -
64bit Production
SQL>
```

- d. View the instance type and then the database name. The unsuccessful query confirms that an ASM instance is only an instance. There is no database attached to it.

```
SQL> show parameter instance_type

NAME                      TYPE        VALUE
-----
instance_type              string      ASM

SQL> select name from v$database;

select name from v$database
*
ERROR at line 1:
ORA-01507: database not mounted
SQL>
```

- e. View the instance name.

```
SQL> select instance_name from v$instance;

INSTANCE_NAME
-----
+ASM

SQL>
```

- f. List the data file names.

```
SQL> select name from v$logfile;
select * from v$logfile
*
ERROR at line 1:
ORA-01507: database not mounted
SQL>
```

An ASM instance manages disk groups and disks for the files of RDBMS databases and has no datafiles of its own.

- g. List the disk groups managed by the ASM instance. Note that the values in TOTAL_MB and FREE_MB will vary depending on the classroom machine configuration.

```
SQL> col name format a4
SQL> select GROUP_NUMBER, NAME, TOTAL_MB, FREE_MB, STATE, TYPE
      from v$asm_diskgroup;
2
GROUP_NUMBER NAME      TOTAL_MB      FREE_MB STATE          TYPE
----- -----
1   DATA        25243       23339 MOUNTED      EXTERN
2   FRA         16829       16595 MOUNTED      EXTERN
SQL>
```

Note: The values of TOTAL_MB and FREE_MB may be different from what is shown in the example

The values of TOTAL_MB and FREE_MB always differ because there is some space taken for the metadata in each disk group. The EXTERNAL type means that there is no copy of allocation units in the disks. If the type was NORMAL or HIGH there would be 2 or 3 copies of the allocation units on different disks. The MOUNTED state means that the disk group is ready to store data for the database files (data files, redo log files, and control files).

- h. List the disks managed by the ASM instance in each disk group.

```
SQL> col path format a27
SQL> col name format a10
SQL> set pagesize 30
SQL> select GROUP_NUMBER, DISK_NUMBER, PATH, NAME
      from v$asm_disk
      order by 1, 2;
2      3
GROUP_NUMBER DISK_NUMBER PATH          NAME
----- -----
0            0   /dev/oracleasm/disks/DISK14
0            1   /dev/oracleasm/disks/DISK13
0            2   /dev/oracleasm/disks/DISK12
```

```

0      3 /dev/oracleasm/disks/DISK11
1      0 /dev/oracleasm/disks/DISK1          DATA_0000
1      1 /dev/oracleasm/disks/DISK2          DATA_0001
1      2 /dev/oracleasm/disks/DISK3          DATA_0002
1      3 /dev/oracleasm/disks/DISK4          DATA_0003
1      4 /dev/oracleasm/disks/DISK5          DATA_0004
1      5 /dev/oracleasm/disks/DISK6          DATA_0005
2      0 /dev/oracleasm/disks/DISK7          FRA_0000
2      1 /dev/oracleasm/disks/DISK8          FRA_0001
2      2 /dev/oracleasm/disks/DISK9          FRA_0002
2      3 /dev/oracleasm/disks/DISK10         FRA_0003

```

14 rows selected.

SQL>

There are six disks in the DATA disk group and four disks in the FRA (Fast Recovery Area reserved for databases backups) disk group. There are also four disks in the 0 disk group. Note that the 0 disk group contains unassigned disks that are candidate disks, available to be assigned to a new disk group.

- i. The disks store files for the RDBMS databases. View the files managed by the ASM instance in each disk group.

```

SQL> col type format a16
SQL> set pages 100
SQL> select GROUP_NUMBER, FILE_NUMBER, TYPE, REDUNDANCY
      from v$asm_file order by type;
2
GROUP_NUMBER FILE_NUMBER TYPE          REDUND
-----
1      253 ASMPARAMETERFILE UNPROT
1      256 CONTROLFILE    UNPROT
2      256 CONTROLFILE    UNPROT
1      262 DATAFILE       UNPROT
1      265 DATAFILE       UNPROT
1      264 DATAFILE       UNPROT
1      261 DATAFILE       UNPROT
1      260 DATAFILE       UNPROT
2      257 ONLINELOG      UNPROT
2      258 ONLINELOG      UNPROT
2      259 ONLINELOG      UNPROT
1      257 ONLINELOG      UNPROT
1      259 ONLINELOG      UNPROT
1      258 ONLINELOG      UNPROT
1      266 PARAMETERFILE   UNPROT

```

```

1          263 TEMPFILE      UNPROT

16 rows selected.

SQL> exit
$
```

You discover that there are three online log files and one control file in the **DATA** disk group and three online log files and one control file in the **FRA** disk group. The disk group number shows which diskgroup by comparing it with the previous query.

- j. Another utility, **ASMCMD**, provides information contained in the metadata of each disk group. The advantage of using **ASMCMD** instead of SQL*Plus is that it can read information in the disk group headers even if the ASM instance is not up. **ASMCMD** uses a Unix-like syntax to view and manage ASM files. Note The '+' is equivalent to the root directory '/' in Unix or Linux.

```

$ asmcmd
ASMCMD> ls
DATA/
FRA/
ASMCMD> ls -l
State    Type     Rebal   Name
MOUNTED  EXTERN   N       DATA/
MOUNTED  EXTERN   N       FRA/
ASMCMD> ls +DATA
ASM/
DBUPGRD/
ASMCMD>
```

The **DATA** disk group stores data for the **DBUPGRD** database and the ASM database instance.

```

ASMCMD> ls +DATA/DBUPGRD
CONTROLFILE/
DATAFILE/
ONLINELOG/
PARAMETERFILE/
TEMPFILE/
control01.ctl
redo01.log
redo02.log
redo03.log
spfiledbupgrd.ora
ASMCMD> ls -l +DATA/DBUPGRD/DATAFILE
Type      Redund  Striped  Time                  Sys  Name
DATAFILE  UNPROT  COARSE   APR 30 05:00:00  Y
EXAMPLE.265.814013031
```

```

DATAFILE UNPROT COARSE APR 30 05:00:00 Y
SYSAUX.261.814012959

DATAFILE UNPROT COARSE APR 30 05:00:00 Y
SYSTEM.260.814012925

DATAFILE UNPROT COARSE APR 30 05:00:00 Y
UNDOTBS1.262.814012987

DATAFILE UNPROT COARSE APR 30 05:00:00 Y
USERS.264.814013023

ASMCMD>

```

What is the relationship between EXAMPLE.265.814013031 and +DATA/dbupgrd/datafile/example.265.814013031 listed in step 1.d?
EXAMPLE.265.814013031 is the unique ASM file managed by the ASM instance and +DATA/dbupgrd/datafile/example.265.814013031 is an alias for the same file.

- k. List the files maintained in the FRA disk group and particularly the control files.

```

ASMCMD> ls +FRA/DBUPGRD/CONTROLFILE/*
Current.256.814012889

ASMCMD>

```

The DBUPGRD database contains two control files which are multiplexed in two distinct disk groups.

```

ASMCMD> ls +*/DBUPGRD/CONTROLFILE/*
Current.256.814012887
Current.256.814012889

ASMCMD>

```

If there were more than two disk groups, and you needed to know in which disk groups the control files are stored, you would enter the following command:

```

ASMCMD> ls -l +*/DBUPGRD/CONTROLFILE/
Type Redund Striped Time Sys Name

+DATA/DBUPGRD/CONTROLFILE/:
CONTROLFILE UNPROT FINE FEB 27 18:00:00 Y
Current.256.814012887

+FRA/DBUPGRD/CONTROLFILE/:
CONTROLFILE UNPROT FINE FEB 27 18:00:00 Y
Current.256.814012889

ASMCMD>

```

- l. List the DBUPGRD database redo log files maintained by the ASM instance.

```

ASMCMD> ls +*/DBUPGRD/ONLINELOG

+DATA/DBUPGRD/ONLINELOG/:
group_1.257.814012897
group_2.258.814012905

```

```
group_3.259.814012913

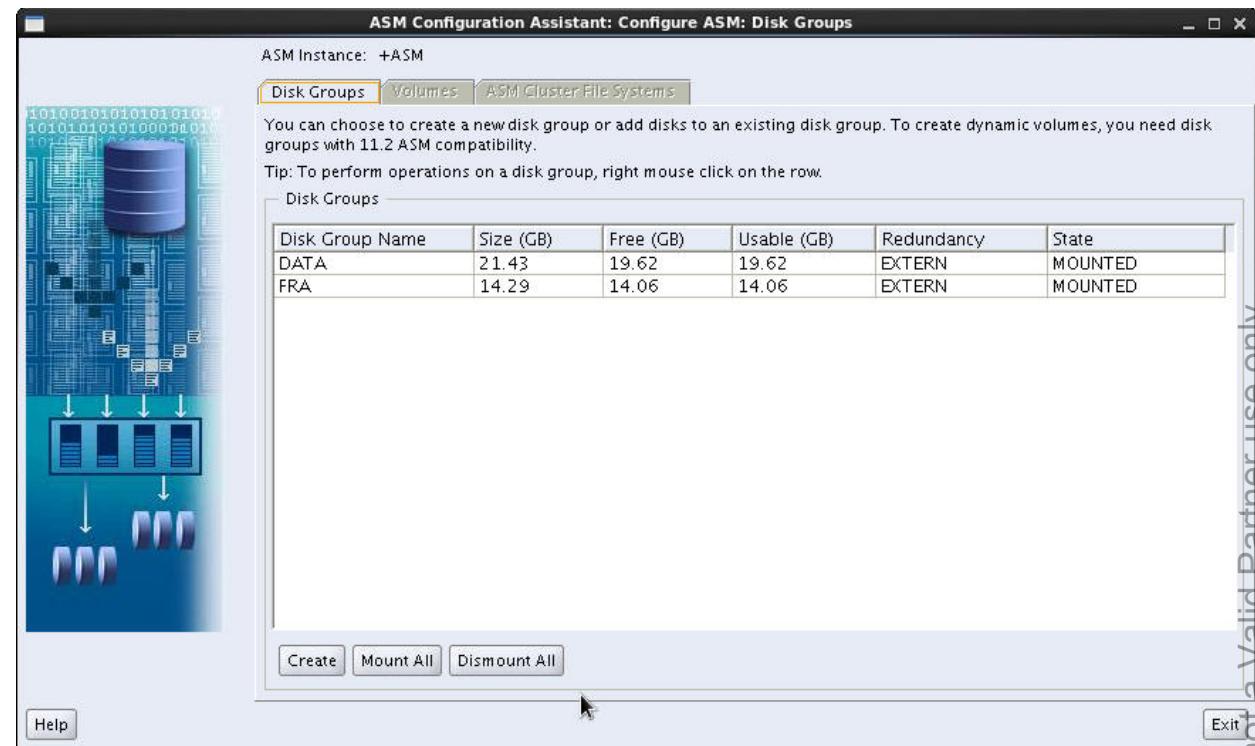
+FRA/DBUPGRD/ONLINELOG/ :
group_1.257.814012901
group_2.258.814012909
group_3.259.814012917
ASMCMD> exit
$
```

The `DBUPGRD` database holds three groups of redo log files multiplexed in two distinct disk groups.

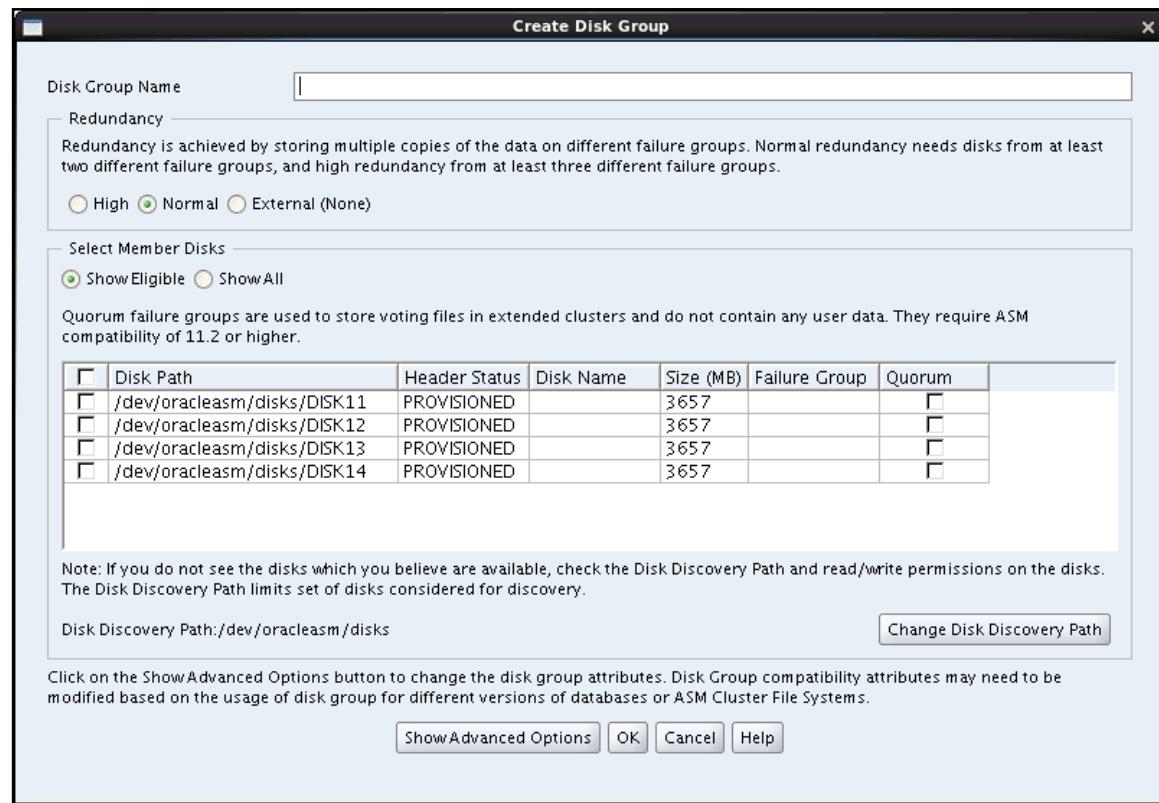
- m. In step 2h, the disk group 0 has four disks assigned to it but are not assigned to any diskgroup that is displayed in the `v$asm_diskgroup` view. These are candidate disks, that are available to be used by ASM. Use another utility, `ASMCA`, to view the candidate disks that do not yet belong to an existing disk group.
- Launch `ASMCA`.

```
$ asmca
```

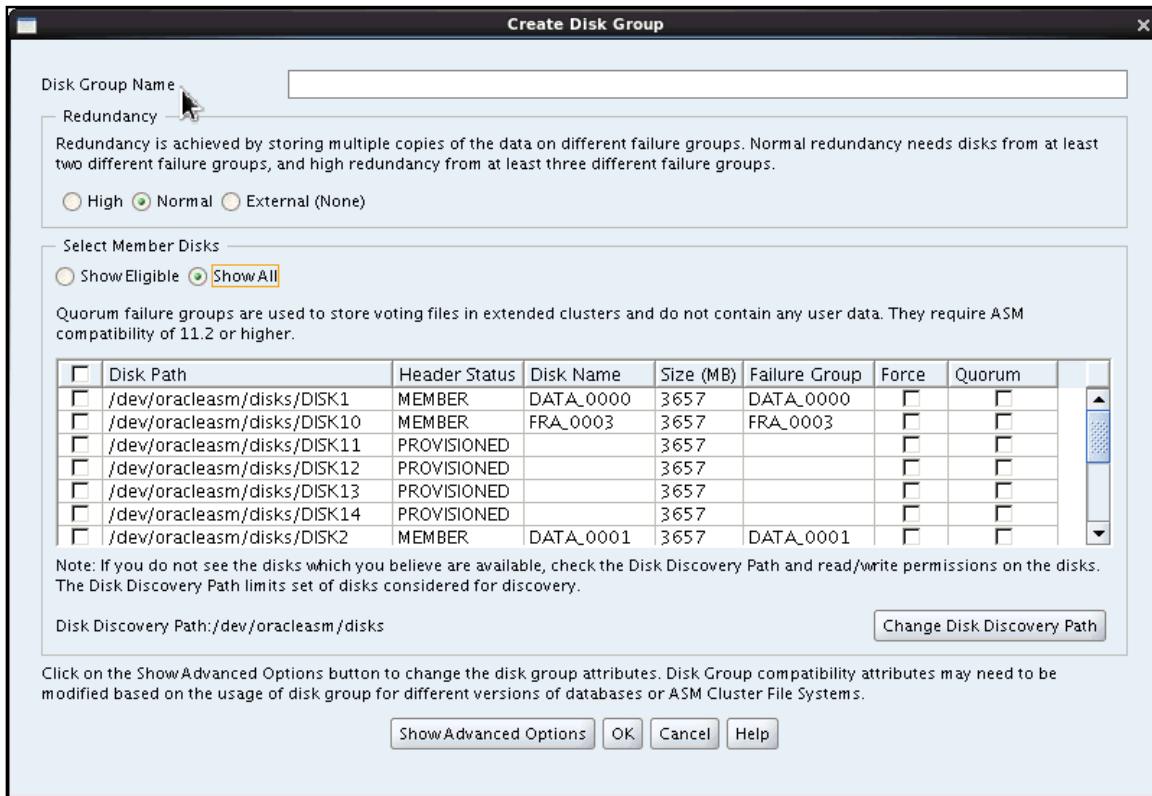
Step	Window/Page Description	Choices or Values
n.	Disk Groups tab page	Verify the two existing disk groups <code>DATA</code> and <code>FRA</code> Click Create .



Step	Window/Page Description	Choices or Values
0.	Create Disk Group page	<p>Verify the following four ASM disks are listed:</p> <p>/dev/oracleasm/disks/DISK11 /dev/oracleasm/disks/DISK12 /dev/oracleasm/disks/DISK13 /dev/oracleasm/disks/DISK14</p> <p>These four disks have been formatted with oracleasm to create your ASM disks to provision any newly created disk group. Note: any disks that have been made available to ASM will appear here. The oracleasm utility is not required.</p>



Step	Window/Page Description	Choices or Values
p.	Create Disk Group page	View all disks managed by the ASM instance. Click the Show All radio button.



Those with a value of PROVISIONED in “Header Status” are candidates for new disk groups. Those with a value of MEMBER in “Header Status” belong to either the DATA or FRA disk group and are assigned a Disk Name that you displayed in V\$ASM_DISK view.

Step	Window/Page Description	Choices or Values
q.	Create Disk Group page	Click Cancel
r.		Click Exit
s.		Click Yes to leave ASMCA.

- t. There is another way to list all disks available for ASM. Use the ORACLEASM utility and the listdisks command.
- First exit from the grid session. The exit command ends the session belonging to the OS user grid and should bring you back into the session owned by oracle.

```
$ exit
logout
$
```

- Log in as root.

```
$ su -
Password:
# oracleasm listdisks
DISK1
```

```
DISK10  
DISK11  
DISK12  
DISK13  
DISK14  
DISK2  
DISK3  
DISK4  
DISK5  
DISK6  
DISK7  
DISK8  
DISK9  
#
```

- Exit from the `root` session. The `exit` command ends the session belonging to the OS user `root` and brings you back into the session owned by `oracle`.

```
# exit  
logout  
$
```

Practice 1-3: Determining the Character Set of Existing Database(s)

Overview

In this practice, you will determine the character set of the existing Oracle database.

Tasks

1. Verify that your session belongs to the `oracle` user to set up your environment variables to point to the `/u01/app/oracle/product/11.2.0/dbhome_2` Oracle home for your RDBMS database installation.

```
$ whoami  
oracle  
$
```

2. Use the `oraenv` utility to set the `ORACLE_SID` environment variable to the `dbupgrd` instance.

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

3. View the database character set.

```
$ sqlplus / as sysdba  
  
SQL> col VALUE format A16  
SQL> select * from NLS_DATABASE_PARAMETERS  
      where parameter like '%CHARACTERSET%';  
      2  
PARAMETER          VALUE  
-----  
NLS_CHARACTERSET    AL32UTF8  
NLS_NCHAR_CHARACTERSET AL16UTF16  
  
SQL> EXIT  
$
```

Note that there are two character sets defined.

- The `NLS_CHARACTERSET` represents the database character set used to identify and to hold SQL and PL/SQL source code. `AL32UTF8` encoding is the 8-bit encoding of Unicode. It is a variable-width type of encoding and also a strict superset of ASCII. A strict superset means that each and every character in 7-bit ASCII is available in `AL32UTF8` with the same corresponding code point value. One Unicode character can be 1, 2, 3, or 4 bytes in this encoding. Characters from the European scripts are represented in either 1 or 2 bytes; characters from most Asian scripts are represented in 3 bytes, whereas supplementary characters are represented in 4 bytes.

- The **NLS_NCHAR_CHARACTERSET** represents the alternate character set that enables you to store Unicode character data in a database that does not have a Unicode database character set. AL16UTF16 encoding is the 16-bit encoding of Unicode. One Unicode character can be 2 to 4 bytes in this encoding. Characters from both European (including ASCII) and most Asian scripts are represented in 2 bytes. Supplementary characters are represented in 4 bytes.

Practice 1-4: Determining Existing Listeners

Overview

In this practice, you will determine the existing listeners and the environment in which they run.

Tasks

1. Use the listener control utility (LSNRCTL) to determine whether a listener is running.

```
$ lsnrctl status

LSNRCTL for Linux: Version 11.2.0.3.0 - Production on 04-FEB-
2013 17:53:08

Copyright (c) 1991, 2011, Oracle. All rights reserved.

Connecting to (ADDRESS=(PROTOCOL=tcp) (HOST=) (PORT=1521))
STATUS of the LISTENER
-----
Alias                      LISTENER
Version                    TNSLSNR for Linux: Version 11.2.0.3.0
- Production
Start Date                 31-JAN-2013 16:19:48
Uptime                     4 days 17 hr. 33 min. 22 sec
Trace Level                off
Security                   ON: Local OS Authentication
SNMP                       OFF
Listener Parameter File    /u01/app/grid/product/11.2.0/grid/network/admin/listener.ora
Listener Log File          /u01/app/grid/diag/tnslsnr/<yourhostserver>/listener/alert/log.x
ml
Listening Endpoints Summary...
(DESCRIPTION=(ADDRESS=(PROTOCOL=ipc) (KEY=EXTPROC1521)))

(DESCRIPTION=(ADDRESS=(PROTOCOL=tcp) (HOST=<yourhostserver>) (PORT
=1521)))
Services Summary...
Service "+ASM" has 1 instance(s).
  Instance "+ASM", status READY, has 1 handler(s) for this
service...
Service "dbupgrd" has 1 instance(s).
  Instance "dbupgrd", status READY, has 1 handler(s) for this
service...
The command completed successfully
$
```

The listener named LISTENER is running in the /u01/app/grid/product/11.2.0/grid Oracle home with a configuration defined in the /u01/app/grid/product/11.2.0/grid/network/admin/listener.ora configuration file. The LISTENER listener handles requests from both the +ASM and dbupgrd instances.

2. Explore the

/u01/app/grid/product/11.2.0/grid/network/admin/listener.ora configuration file.

```
$ cat  
/u01/app/grid/product/11.2.0/grid/network/admin/listener.ora  
# listener.ora Network Configuration File:  
/u01/app/grid/product/11.2.0/grid/network/admin/listener.ora  
# Generated by Oracle configuration tools.  
  
LISTENER =  
(DESCRIPTION_LIST =  
  (DESCRIPTION =  
    (ADDRESS = (PROTOCOL = IPC) (KEY = EXTPROC1521))  
    (ADDRESS = (PROTOCOL = TCP) (HOST = <yourhostserver>) (PORT  
= 1521))  
  )  
)  
  
ADR_BASE_LISTENER = /u01/app/grid  
  
ENABLE_GLOBAL_DYNAMIC_ENDPOINT_LISTENER=ON      # line added by  
Agent  
$
```

3. Find the user running the listener LISTENER process. grid is the UNIX user running the process in /u01/app/grid/product/11.2.0/grid, the Oracle home of Oracle Grid Infrastructure.

```
$ ps -ef | grep tnslsnr  
grid      16203      1  0 Mar29 ?          00:00:16  
/u01/app/grid/product/11.2.0/grid/bin/tnslsnr LISTENER -inherit  
oracle    27661  31746  0 12:42 pts/1      00:00:00 grep tnslsnr  
$
```


Practices for Lesson 2: Oracle Software Installation Basics

Chapter 2

Practices for Lesson 2

Practices Overview

There is no practice for this lesson.

Practices for Lesson 3: Installing Oracle Grid Infrastructure for a Standalone Server and Upgrading ASM

Chapter 3

Practices for Lesson 3

Practices Overview

In the practices of this course, you assume the role of a database administrator (DBA). The operating system (OS) accounts on your computer are:

- The `oracle` user with a password of `oracle`
- The `grid` user with a password of `oracle`
- The `root` user with a password of `oracle`

The system administrator has set up the OS so that it is ready for your Oracle software installation. You are performing two installations.

1. The first installation is the Oracle Grid Infrastructure for a standalone server.
2. The second installation is the Oracle Database 12c software.

The installation media is staged at:

- `/stage/shiphomes/RDBMS_LINUX.X64_12C/clusterware` for Oracle Grid Infrastructure
- `/stage/shiphomes/RDBMS_LINUX.X64_12C/database` for Oracle Database 12c

Perform most of the following tasks as the `grid` OS user, unless otherwise indicated.

You are going to work as the `oracle` user for some operations and as the `grid` user for other operations. Keep a terminal window opened as the `oracle` user and another terminal window opened as the `grid` user.

- In the `oracle` user terminal window, keep the following environment variables set:

```
$ id  
uid=54321(oracle) gid=54321(oinstall)  
groups=54321(oinstall),54322(dba),54323(oper),54324(backupdba),5  
4325(dgdba),54326(kmdba),54327(asmdba)  
$
```

Set the `ORACLE_SID` environment variable to any instance other than `+ASM`, for example `dbupgrd`.

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

- In the `grid` user terminal window, keep the following environment variables set:

```
$ su - grid  
Password:  
$ id
```

```
uid=54322(grid) gid=54321(oinstall)
groups=54321(oinstall),54322(dba),54327(asmdba),54329(asmadmin)
$
```

Set the ORACLE_SID environment variable to +ASM.

```
$ . oraenv
ORACLE_SID = [grid] ? +ASM
The Oracle base has been changed from /u01/app/oracle to
/u01/app/grid
$
```

Note: Completing this practice is critical for all following practice sessions.

Practice 3-1: Installing Oracle Grid Infrastructure and Upgrading ASM

Overview

In the previous practice, you detected two installed products:

- 11g Oracle Grid Infrastructure with a running ASM instance (+ASM) and Oracle Restart started
- 11g Oracle Database server with a running Oracle RDBMS instance (dbupgrd)

In this practice, you upgrade 11g Oracle Grid Infrastructure to 12c Oracle Grid Infrastructure by using the Oracle Universal Installer.

The Oracle Universal Installer (OUI) will detect the ASM instance during the Oracle Grid Infrastructure upgrade and will upgrade the ASM instance from the 11g Release 2 to the 12c Release 1 version. The ASM instance is managing the storage for the dbupgrd database. The ASM instance will be shut down and therefore the clients need to be shut down as well.

You are going to work as the `oracle` user for some operations and as the `grid` user for other operations. Therefore, keep a terminal window opened as the `oracle` user and another terminal window opened as the `grid` user.

Tasks

1. Before upgrading Oracle Grid Infrastructure and the ASM instance, shut down the dbupgrd instance to close all the database files. The dbupgrd database is a client of the ASM instance.
 - a. From the `oracle` user terminal window, verify that you are logged in as the `oracle` user.

```
$ id  
uid=54321(oracle) gid=54321(oinstall)  
groups=54321(oinstall), 54322(dba), 54323(oper), 54324(backupdba), 5  
4325(dgdba), 54326(kmdba), 54327(asmdba)  
$
```

- b. Set the ORACLE_SID environment variable to dbupgrd.

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

- c. Shut down the dbupgrd instance.

```
$ sqlplus / as sysdba  
  
SQL> shutdown immediate  
Database closed.  
Database dismounted.  
ORACLE instance shut down.  
SQL> exit  
$
```

2. The 11g Oracle Grid Infrastructure includes the Oracle Restart component which is currently running.
 - a. If you did not already open the **grid** user terminal window, right-click the desktop and click Open Terminal to open another terminal window. Check that you are logged as the **grid** UNIX user.

```
$ whoami  
oracle  
$
```

This is not the case. Use the **su** command to switch to the **grid** user.

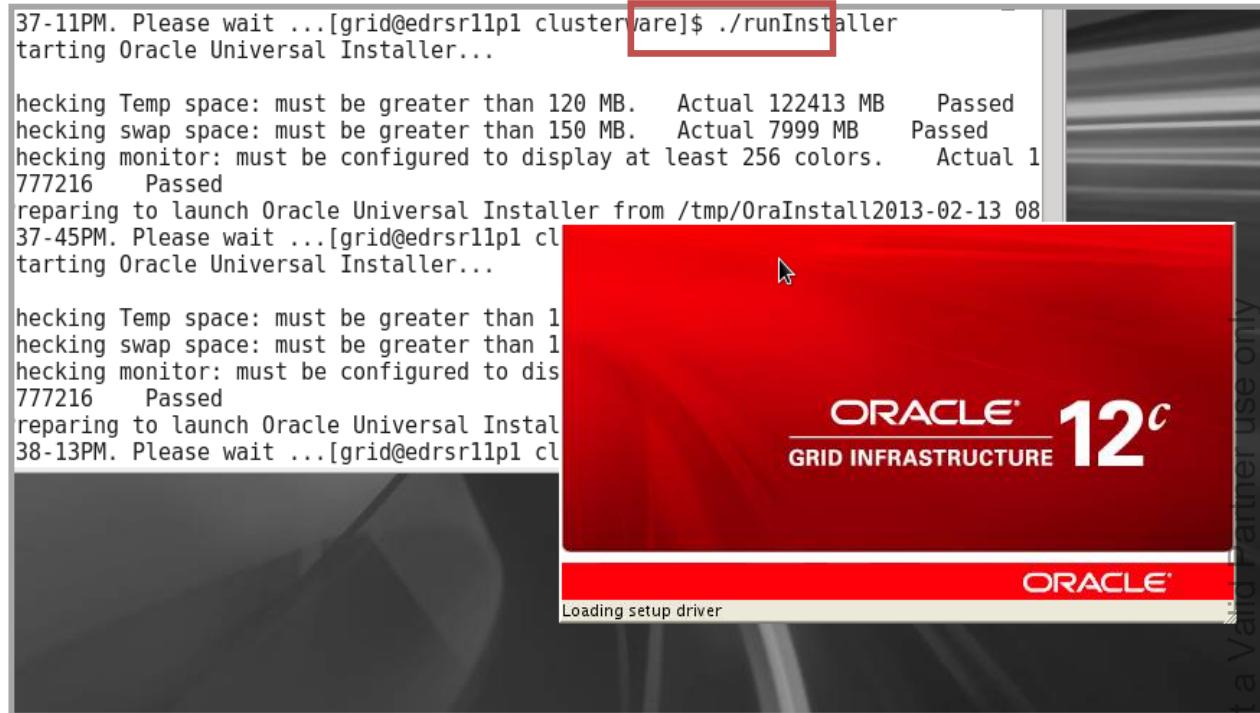
```
$ su - grid  
Password:  
$ id  
uid=54322(grid) gid=54321(oinstall)  
groups=54321(oinstall), 54322(dba), 54327(asmdba), 54329(asmadmin)  
$
```

- b. Set the **ORACLE_BASE** and **ORACLE_HOME** environment variables.

```
$ . oraenv  
ORACLE_SID = [grid] ? +ASM  
The Oracle base has been changed from /u01/app/oracle to  
/u01/app/grid  
$
```

3. Start the Oracle Universal Installer (OUI) for the Oracle Grid Infrastructure. As the **grid** user, navigate to the **/stage/shiphomes/RDBMS_LINUX.X64_12C/clusterware** directory and enter **./runInstaller**.

```
$ cd /stage/shiphomes/RDBMS_LINUX.X64_12C/grid  
$ ./runInstaller
```



Oracle Grid Infrastructure 12c OUI is displayed.

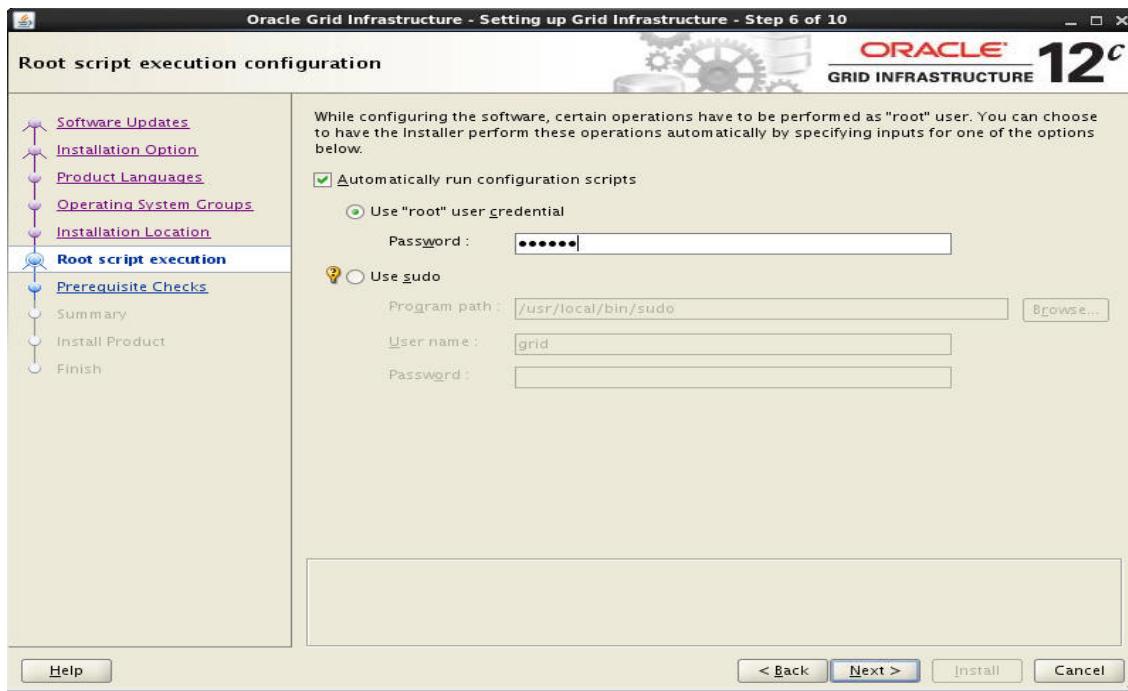
Step	Window/Page Description	Choices or Values
a.	Oracle Grid Infrastructure 12c OUI is displayed.	
b.	Download Software Updates page	Select “ Skip software updates ”. Click Next .
c.	Select Installation Option page	Select “ Upgrade Oracle Grid Infrastructure or Oracle Automatic Storage Management ” Note: This option upgrades the existing 11g Oracle Grid Infrastructure and the ASM instance.
d.	Select Product Languages page	Select all the available languages with the '">>' button. Click Next .
e.	Privileged Operating System Groups page	Verify that role separation is defined for ASM operations. You should see three OS groups, asmadmin, asmdba and asmoper that were created prior to running the installer. These groups may also be viewed in the /etc/group file. Click Next .
f.	Specify Installation Location page	Browse the Oracle base Select /u01/app/grid Set Software Location to /u01/app/grid/product/12.1.0/grid

Step	Window/Page Description	Choices or Values
		Click Next



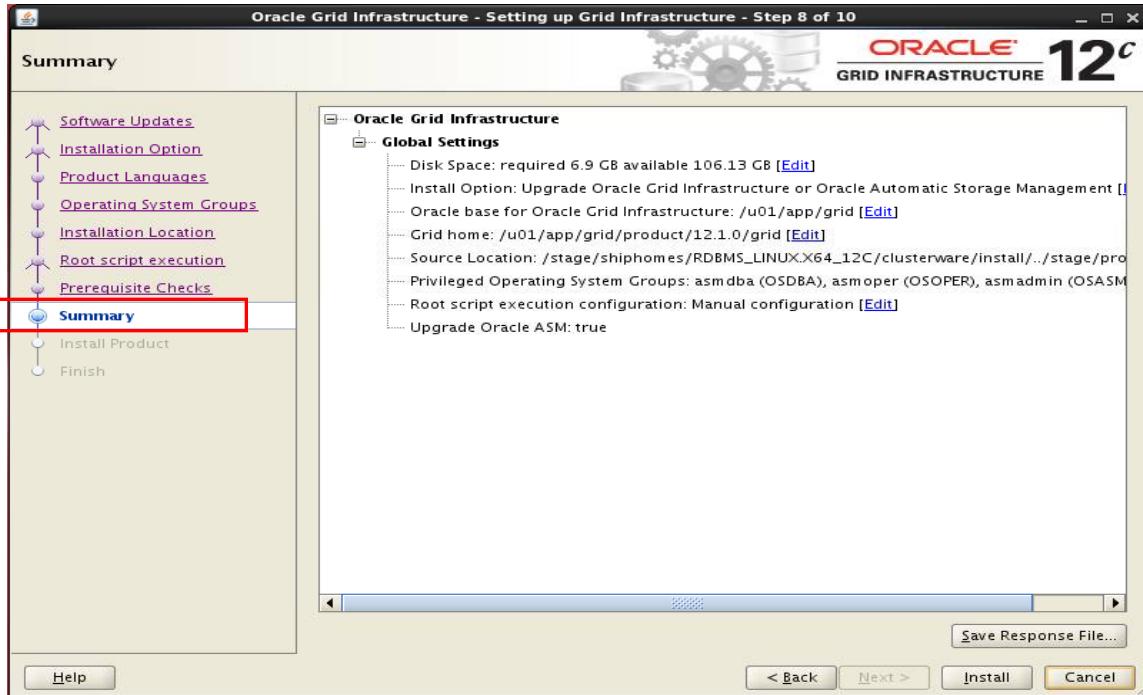
Step	Window/Page Description	Choices or Values
g.	Root script execution configuration page	Configure the root scripts to be executed manually when required. This will allow you to follow the steps more easily. The alternate configuration is described below. Click Next .

You could have chosen the “Automatically run configuration scripts” option as shown in the screenshot below. In this case, the installation would run the required scripts as root when necessary. To run the scripts as root, OUI must know the root password. If a root script failed on the server, then you would have to fix the problem manually and rerun the root script on the server, and continue.

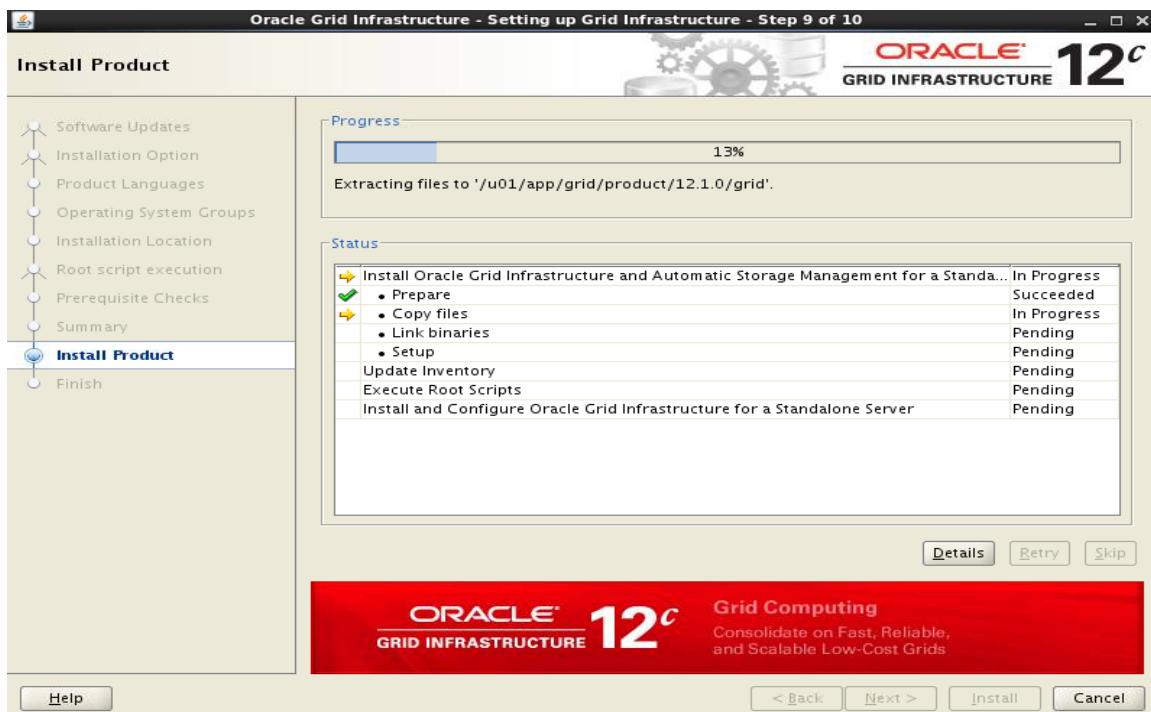


Step	Window/Page Description	Choices or Values
h.	Perform Prerequisites Checks page	Only appears if a check fails. A failure occurs Select the failure message - Check incorrectly sized ASM disks Click more details link in the lower pane
i.	Details	See Message: PRCT-1207 : Failed to set the ORACLE_SID for running asmcmd from CRS home location /u01/app/grid/product/11.2.0/grid This is a known bug: Click Close
j.	Perform Prerequisites Checks page	Click Ignore All Click Next
k.	OracleGrid Infrastructure	Are you sure you want to continue? Click Yes

Step	Window/Page Description	Choices or Values
I.	Summary page	Review the settings and information. Click Install .



Step	Window/Page Description	Choices or Values
m.	Install Product page	The progress of the Oracle Grid Infrastructure installation is displayed and the status of the individual tasks being performed. The installation can take up to 20 minutes to get to Step 3t.



Step	Window/Page Description	Choices or Values
n.	Execute Configuration Scripts page	In a terminal as the root user execute <code>/u01/app/grid/product/12.1.0/grid/rootupgrade.sh</code>



Step	Window/Page Description	Choices or Values
o.	In a new terminal session	<p>Check that you are logged as the root UNIX user.</p> <pre>\$ id uid=54321(oracle) gid=54321(oinstall) groups=54321(oinstall),54322(dba),54323(oper), 54324(backupdba),54325(dgdba),54326(kmdba), 54327(asmdba)</pre>
p.		<p>Use the su command to switch to the root user. Once logged as root, you get the # prompt.</p> <pre>\$ su - Password: #</pre>
q.		<p>Execute the script</p> <pre># /u01/app/grid/product/12.1.0/grid/rootupgrade.sh</pre> <p>Enter RETURN to the four questions</p> <p>See output in code box below</p> <p>Note: This step takes about 12 minutes.</p> <p>Keep the root user terminal session opened.</p>

```
# /u01/app/grid/product/12.1.0/grid/rootupgrade.sh
Performing root user operation for Oracle 12c

The following environment variables are set as:
ORACLE_OWNER= grid
ORACLE_HOME= /u01/app/grid/product/12.1.0/grid

Enter the full pathname of the local bin directory:
[/usr/local/bin]:
The file "dbhome" already exists in /usr/local/bin. Overwrite it? (y/n)
[n]:
The file "oraenv" already exists in /usr/local/bin. Overwrite it? (y/n)
[n]:
The file "coraenv" already exists in /usr/local/bin. Overwrite it? (y/n)
[n]:

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.
```

```
Using configuration parameter file:  
/u01/app/grid/product/12.1.0/grid/crs/install/crsconfig_params  
  
ASM Configuration upgraded successfully.  
  
Creating OCR keys for user 'grid', privgrp 'oinstall'..  
Operation successful.  
LOCAL ONLY MODE  
Successfully accumulated necessary OCR keys.  
Creating OCR keys for user 'root', privgrp 'root'..  
Operation successful.  
CRS-4664: Node <your_hostname> successfully pinned.  
2013/02/25 01:32:09 CLSRSC-329: Replacing Clusterware entries in  
file 'oracle-ohasd.conf'  
  
2013/02/25 01:35:46 CLSRSC-329: Replacing Clusterware entries in  
file 'oracle-ohasd.conf'  
  
  
<your_hostname> 2013/02/25 01:39:55  
/u01/app/grid/product/12.1.0/grid/cdata/<your_hostname>/backup_2  
0130225_013955.olr  
  
<your_hostname> 2013/02/25 04:22:22  
/u01/app/grid/product/11.2.0/grid/cdata/<your_hostname>/backup_2  
0130225_042222.olr  
2013/02/25 01:40:53 CLSRSC-327: Successfully configured Oracle  
Grid Infrastructure for a Standalone Server  
#
```

The ASM configuration is updated. This means that the ASM instance has been upgraded from Oracle Grid Infrastructure 11g to Oracle Grid Infrastructure 12c.

- The upgrade script sets the ORACLE_OWNER environment variable to grid and the ORACLE_HOME environment variable to the new /u01/app/grid/product/12.1.0/grid Oracle home.
- The upgrade script upgrades the ASM instance to run in the new /u01/app/grid/product/12.1.0/grid Oracle home.
- The upgrade script replaces clusterware entries in the /u01/app/grid/product/12.1.0/grid/crs/install/oracle-ohasd.conf file to start the OHASD (Oracle High Availability Services Daemon) in the new Oracle home. The Oracle High Availability Services Daemon is covered in another practice.
- The upgrade script creates a new Oracle Local Registry (OLR) in the new Oracle home. The OLR includes the current registered resources that are the listener, the dbupgrd database instance, the ASM instance and the two disk groups. The OLR is finally backed up at the end of the upgrade as

/u01/app/grid/product/11.2.0/grid/cdata/<localhost>/backup_20130208_211335.olr.

- It also backs up the 11g OLR in its original Oracle home,
 /u01/app/grid/product/11.2.0/grid/cdata/<localhost>/backup_20130208_211335.olr. Oracle Restart with the OLR is covered in another practice.

Step	Window/Page Description	Choices or Values
r.	Execute Configuration scripts window	Click OK
s.	Install Product page	Continues
t.	Oracle Grid Infrastructure	Message: [INS-20802] Oracle Cluster Verification Utility failed. Click Details
u.	Oracle Grid Infrastructure	Note: Log File Location In the root user terminal window: <code>less <log file location></code> Scroll up from bottom of file to find error – PRVG-2033 : Permissions of file <code>"'/u01/app/grid/product/12.1.0/grid/cdata/localhost/<your_hostname>.olr"</code> did not match the expected octal value on node <code>"edrsr7p1". [Expected = "0600" ; Found = "0660"]</code> in terminal window as root user change the permissions of this file: <code>chmod 600 /u01/app/grid/product/12.1.0/grid/cdata/localhost/<your_hostname>.olr</code> Click OK
v.	Install Product page	Click Retry
w.	Install Product page	When all items have completed – indicate by a green check Click Next
x.	Finish page	Click Close
y.	Terminal window as grid	press [Enter] The name of the log file of the install is displayed <code>\$ You can find the log of this install session at:</code> <code>u01/app/logs/installActions2013-02-25_01-15-27AM.log</code> <code>\$</code>
z.	Terminal window as root	Quit the root user terminal window. <code># exit</code>

Step	Window/Page Description	Choices or Values
		logout \$ exit

Practice 3-2: Using the Upgraded ASM Instance

Overview

In this practice, you will perform tasks such as resetting the Oracle ASM passwords and configuring disk groups now that the ASM instance has been upgraded.

Tasks

- From the grid user session, verify that the ASM instance entry is updated in /etc/oratab.

```
$ less /etc/oratab
#Backup file is
/u01/app/grid/product/12.1.0/grid/srvvm/admin/oratab.bak.<localho
st> line added by Agent
#
# This file is used by ORACLE utilities. It is created by
root.sh
# and updated by either Database Configuration Assistant while
creating
# a database or ASM Configuration Assistant while creating ASM
instance.
# A colon, ':', is used as the field terminator. A new line
terminates
# the entry. Lines beginning with a pound sign, '#', are
comments.
#
# Entries are of the form:
#   $ORACLE_SID:$ORACLE_HOME:<N|Y>:
#
# The first and second fields are the system identifier and home
# directory of the database respectively. The third filed
indicates
# to the dbstart utility that the database should , "Y", or
should not,
# "N", be brought up at system boot time.
#
# Multiple entries with the same $ORACLE_SID are not allowed.
#
#
dbupgrd:/u01/app/oracle/product/11.2.0/dbhome_2:N          #
line added by Agent
+ASM:/u01/app/grid/product/12.1.0/grid:N:      # line added by
Agent
$
```

The ORACLE_HOME for the ASM instance is updated to
/u01/app/grid/product/12.1.0/grid.

2. Verify that the ASM instance is started.

- a. Reset the environment variables so that `ORACLE_HOME` reflects the new `/u01/app/grid/product/12.1.0/grid` directory.

```
$ . oraenv
ORACLE_SID = [+ASM] ? +ASM
The Oracle base remains unchanged with value /u01/app/grid
$ env | grep ORA
ORACLE_SID=+ASM
ORACLE_BASE=/u01/app/grid
ORACLE_HOME=/u01/app/grid/product/12.1.0/grid
$
```

- b. Verify in the banner that the ASM instance now runs in the 12c environment

```
$ sqlplus / as sysasm

Connected to:
Oracle Database 12c Enterprise Edition Release 12.1.0.1.0 -
64bit Production With the Automatic Storage Management option

SQL> select instance_name, status from v$instance;

INSTANCE_NAME      STATUS
-----
+ASM                STARTED

SQL>
```

- c. Shut down the ASM instance.

```
SQL> shutdown immediate
ASM diskgroups dismounted
ASM instance shutdown
SQL>
```

- d. Start the instance up.

```
SQL> startup
ASM instance started

Total System Global Area  1068937216 bytes
Fixed Size                  2295952 bytes
Variable Size                1041475440 bytes
ASM Cache                     25165824 bytes
ASM diskgroups mounted
SQL>
```

The disk groups are automatically mounted because there is a parameter file including the `ASM_DISKGROUPS` parameter.

3. Determine which initialization parameter file is being used. Note : the suffix of the file name will vary from the example shown below.

```
SQL> show parameter spfile

NAME          TYPE    VALUE
-----
spfile        string  +DATA/asm/asmparamefile/reg
                           istry.253.814078909

SQL> exit
$
```

- a. View the location of the SPFILE in ASMCMD.

```
$ asmcmd ls +DATA/ASM/ASMPARAMEFILE/
REGISTRY.253.814078909
$
```

The SPFILE is stored in the DATA disk group.

- b. Create the ASM text parameter file (PFILE) from the ASM server parameter file (SPFILE).

```
$ sqlplus / as sysasm

SQL> create pfile='?/ dbs/bkup+ASM.ora' from spfile;

File created.

SQL> exit
Disconnected
$
```

- c. View the contents of the ASM text parameter file.

```
$ cd $ORACLE_HOME/dbs
$ ls
ab+_ASM.dat  hc+_ASM.dat  bkup+ASM.ora  init.ora
$ cat bkup+ASM.ora
+ASM.__oracle_base='/u01/app/grid'#ORACLE_BASE set from in
memory value
+ASM.asm_diskgroups='FRA', 'DATA'#Manual Mount
*.asm_diskstring='/dev/oracleasm/disks'
*.core_dump_dest='/u01/app/grid/diag/asm/+asm/+ASM/cdump'
*.diagnostic_dest='/u01/app/grid'
*.memory_max_target=327155712
*.memory_target=327155712
$
```

4. Connect to the ASM instance and verify which disk groups are mounted.

```
$ sqlplus / as sysasm
```

```
SQL> select name, state from v$asm_diskgroup;

NAME                      STATE
-----
FRA                       MOUNTED
DATA                      MOUNTED

SQL> exit
$
```

The DATA disk group must be mounted because the SPFILE is stored there.

- View the alert log file of the ASM instance. You will find all operations performed in the ASM instance. For example, the disks attached to the DATA and FRA disk groups are opened with external redundancy. Both disk groups are mounted, the DATA disk group first because the instance uses “parameter settings in server-side spfile +DATA/asm/asmparamefile/registry.253.814078909”, then the FRA disk group defined in the ASM_DISKGROUPS parameter. The compatible.asm attribute is still set to 11.2.0.0.0 and not yet 12.1.0.0.0.

Use shift-G to move to the bottom of the file then search up until you find the last "Starting up:" string. Note: ?Starting up? will search up in vi.

Use :q to exit vi

```
$ cd /u01/app/grid/diag/asm/+asm/+ASM/trace
$ vi alert_+ASM.log
Starting up:
Oracle Database 12c Enterprise Edition Release 12.1.0.1.0 -
64bit Production
With the Automatic Storage Management option.
ORACLE_HOME = /u01/app/grid/product/12.1.0/grid_1
System name:      Linux
Node name:        <your_hostname>
Release:          2.6.39-200.24.1.el6uek.x86_64
Version:          #1 SMP Sat Jun 23 02:39:07 EDT 2012
Machine:          x86_64
Using parameter settings in server-side spfile
+DATA/asm/asmparamefile/registry.253.814078909
System parameters with non-default values:
memory_target          = 1G
memory_max_target       = 1G
core_dump_dest          =
"/u01/app/grid/diag/asm/+asm/+ASM/cdump"
asm_diskstring          = "/dev/oracleasm/disks"
asm_diskgroups           = "DATA"
asm_diskgroups           = "FRA"
```

```
        diagnostic_dest          = "/u01/app/grid"
NOTE: remote asm mode is local (mode 0x301; from cluster type)
...
SQL> ALTER DISKGROUP ALL MOUNT /* asm agent call crs *//*
{0:0:2} */
Diskgroup with spfile:DATA
NOTE: Diskgroups listed in ASM_DISKGROUP are
      DATA
      (duplicate)
      FRA

NOTE: cache registered group DATA 1/0x07957B80
NOTE: cache began mount (first) of group DATA 1/0x07957B80
NOTE: cache registered group FRA 2/0x07957B81
NOTE: cache began mount (first) of group FRA 2/0x07957B81
NOTE: Loaded library: /opt/oracle/extapi/64/asm/orcl/1/libasm.so
NOTE: Assigning number (1,5) to disk
(/dev/oracleasm/disks/DISK6)
NOTE: Assigning number (1,4) to disk
(/dev/oracleasm/disks/DISK5)
NOTE: Assigning number (1,3) to disk
(/dev/oracleasm/disks/DISK4)
NOTE: Assigning number (1,2) to disk
(/dev/oracleasm/disks/DISK3)
NOTE: Assigning number (1,1) to disk
(/dev/oracleasm/disks/DISK2)
NOTE: Assigning number (1,0) to disk
(/dev/oracleasm/disks/DISK1)
NOTE: Assigning number (2,3) to disk
(/dev/oracleasm/disks/DISK10)
NOTE: Assigning number (2,2) to disk
(/dev/oracleasm/disks/DISK9)
NOTE: Assigning number (2,1) to disk
(/dev/oracleasm/disks/DISK8)
NOTE: Assigning number (2,0) to disk
(/dev/oracleasm/disks/DISK7)
Using default pga_aggregate_limit of 2048 MB
NOTE: GMON heartbeating for grp 1 (DATA)
GMON querying group 1 at 4 for pid 19, osid 24170
NOTE: cache is mounting group DATA created on 2013/04/04
09:00:31
NOTE: cache opening disk 0 of grp 1: DATA_0000
path:/dev/oracleasm/disks/DISK1
NOTE: 04/04/13 11:13:17 DATA.F1X0 found on disk 0 au 2 fcn 0.0
datfmt 1
```

```
NOTE: cache opening disk 1 of grp 1: DATA_0001
path:/dev/oracleasm/disks/DISK2
NOTE: cache opening disk 2 of grp 1: DATA_0002
path:/dev/oracleasm/disks/DISK3
NOTE: cache opening disk 3 of grp 1: DATA_0003
path:/dev/oracleasm/disks/DISK4
NOTE: cache opening disk 4 of grp 1: DATA_0004
path:/dev/oracleasm/disks/DISK5
NOTE: cache opening disk 5 of grp 1: DATA_0005
path:/dev/oracleasm/disks/DISK6
NOTE: cache mounting (first) external redundancy group
1/0x07957B80 (DATA)
NOTE: cache recovered group 1 to fcn 0.2484
NOTE: redo buffer size is 256 blocks (1053184 bytes)
NOTE: LGWR attempting to mount thread 1 for diskgroup 1 (DATA)
NOTE: LGWR found thread 1 closed at ABA 5.214 lock domain=0
inc#=0 instnum=0
NOTE: LGWR mounted thread 1 for diskgroup 1 (DATA)
NOTE: LGWR opened thread 1 (DATA) at fcn 0.2484 ABA 6.215 lock
domain=1 inc#=0 instnum=1
NOTE: cache mounting group 1/0x07957B80 (DATA) succeeded
NOTE: cache mounting group 1/0x07957B80 (DATA) succeeded
NOTE: cache ending mount (success) of group DATA number=1
incarn=0x07957b80
NOTE: GMON heartbeating for grp 2 (FRA)
GMON querying group 2 at 6 for pid 19, osid 24170
NOTE: cache is mounting group FRA created on 2013/04/04 09:00:59
NOTE: cache opening disk 0 of grp 2: FRA_0000
path:/dev/oracleasm/disks/DISK7
NOTE: 04/04/13 11:13:17 FRA.F1X0 found on disk 0 au 2 fcn 0.0
datfmt 1
NOTE: cache opening disk 1 of grp 2: FRA_0001
path:/dev/oracleasm/disks/DISK8
NOTE: cache opening disk 2 of grp 2: FRA_0002
path:/dev/oracleasm/disks/DISK9
NOTE: cache opening disk 3 of grp 2: FRA_0003
path:/dev/oracleasm/disks/DISK10
NOTE: cache mounting (first) external redundancy group
2/0x07957B81 (FRA)
NOTE: cache recovered group 2 to fcn 0.969
NOTE: redo buffer size is 256 blocks (1053184 bytes)
NOTE: LGWR attempting to mount thread 1 for diskgroup 2 (FRA)
NOTE: LGWR found thread 1 closed at ABA 5.158 lock domain=0
inc#=0 instnum=0
NOTE: LGWR mounted thread 1 for diskgroup 2 (FRA)
```

```

NOTE: LGWR opened thread 1 (FRA) at fcn 0.969 ABA 6.159 lock
domain=2 inc#=0 instnum=1
NOTE: cache mounting group 2/0x07957B81 (FRA) succeeded
NOTE: cache ending mount (success) of group FRA number=2
incarn=0x07957b81
NOTE: cache mounting group 2/0x07957B81 (FRA) succeeded
NOTE: cache ending mount (success) of group FRA number=2
incarn=0x07957b81
NOTE: Instance updated compatible.asm to 11.2.0.0.0 for grp 1
SUCCESS: diskgroup DATA was mounted
NOTE: Instance updated compatible.asm to 11.2.0.0.0 for grp 2
SUCCESS: diskgroup FRA was mounted
$
```

6. Start the DBUPGRD Oracle Database instance and verify the locations of all the database files.
 - a. From the oracle user terminal window, set the ORACLE_SID environment variable to dbupgrd.

```

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

- b. Start the dbupgrd instance.

Note: the values of Variable Size, and Database Buffers may vary from the values shown below, but the sum of the two values should be the same. The shared pool will get memory from the buffer pool as needed.

```

$ sqlplus / as sysdba

Connected to an idle instance.

SQL> STARTUP
ORACLE instance started.

Total System Global Area  626327552 bytes
Fixed Size                  2230952 bytes
Variable Size                192939352 bytes
Database Buffers              423624704 bytes
Redo Buffers                  7532544 bytes
Database mounted.
Database opened.
SQL>
```

- c. Verify the location of the data files, the control files and the redo log files.

```
SQL> select name from v$datafile;
```

```
NAME
-----
+DATA/dbupgrd/datafile/system.260.814012925
+DATA/dbupgrd/datafile/sysaux.261.814012959
+DATA/dbupgrd/datafile/undotbs1.262.814012987
+DATA/dbupgrd/datafile/users.264.814013023
+DATA/dbupgrd/datafile/example.265.814013031

SQL> select name from v$controlfile;

NAME
-----
+DATA/dbupgrd/control01.ctl
+FRA/dbupgrd/control02.ctl

SQL> select member from v$logfile;

MEMBER
-----
+DATA/dbupgrd/redo01.log
+FRA/dbupgrd/redo01.log
+DATA/dbupgrd/redo02.log
+FRA/dbupgrd/redo02.log
+DATA/dbupgrd/redo03.log
+FRA/dbupgrd/redo03.log

6 rows selected.

SQL> EXIT
$
```

All files of the dbupgrd database are still stored on ASM disk groups DATA and FRA which are managed by the upgraded ASM instance.

Note: The file numbers and incarnation numbers may be different in individual instances.

Question: Why do datafiles and controlfiles appear with the suffix file and incarnation numbers and the redo log files do not?

Answer: the redo log files were created with an alias specification.

Practice 3-3: Determining Existing Listeners

Overview

In this practice, you will determine the existing listeners and in which environment they run now that the Oracle Grid Infrastructure is installed in a 12c environment. **Note:** setting the environment to either dbupgrd or +ASM will allow you to view the status of the default listener named LISTENER.

Tasks

- From the grid user terminal window, use the listener control utility (LSNRCTL) to determine whether a listener is running.

```
$ lsnrctl status

LSNRCTL for Linux: Version 12.1.0.1.0 - Production on 27-FEB-
2013 17:49:18

Copyright (c) 1991, 2013, Oracle. All rights reserved.

Connecting to
(DESCRIPTION=(ADDRESS=(PROTOCOL=IPC) (KEY=EXTPROC1521)))
STATUS of the LISTENER
-----
Alias                      LISTENER
Version                    TNSLSNR for Linux: Version 12.1.0.1.0
- Production
Start Date                 27-FEB-2013 13:41:17
Uptime                     0 days 4 hr. 8 min. 1 sec
Trace Level                off
Security                   ON: Local OS Authentication
SNMP                       OFF
Listener Parameter File    /u01/app/grid/product/12.1.0/grid/network/admin/listener.ora
Listener Log File          /u01/app/grid/diag/tnslsnr/edrsr32p1/listener/alert/log.xml
Listening Endpoints Summary...
(DESCRIPTION=(ADDRESS=(PROTOCOL=ipc) (KEY=EXTPROC1521)))

(DESCRIPTION=(ADDRESS=(PROTOCOL=tcp) (HOST=edrsr32p1.us.oracle.co
m) (PORT=1521)))
Services Summary...
Service "+ASM" has 1 instance(s).
  Instance "+ASM", status READY, has 1 handler(s) for this
service...
Service "dbupgrd" has 1 instance(s).
```

```
Instance "dbupgrd", status READY, has 1 handler(s) for this
service...
```

```
The command completed successfully
```

```
$
```

The listener, named LISTENER, is running in the /u01/app/grid/product/12.1.0/grid Grid home with a configuration defined in the /u01/app/grid/product/12.1.0/grid/network/admin/listener.ora configuration file. The listener handles requests from both the +ASM and dbupgrd instances.

2. Explore the

/u01/app/grid/product/12.1.0/grid/network/admin/listener.ora configuration file.

```
$ cat $ORACLE_HOME/network/admin/listener.ora
# listener.ora Network Configuration File:
/u01/app/grid/product/11.2.0/grid/network/admin/listener.ora
# Generated by Oracle configuration tools.

LISTENER =
  (DESCRIPTION_LIST =
    (DESCRIPTION =
      (ADDRESS = (PROTOCOL = IPC) (KEY = EXTPROC1521))
      (ADDRESS = (PROTOCOL = TCP) (HOST =
<your_hostname>.us.oracle.com) (PORT = 1521))
    )
  )

ADR_BASE_LISTENER = /u01/app/grid

ENABLE_GLOBAL_DYNAMIC_ENDPOINT_LISTENER=ON          # line added by
Agent
VALID_NODE_CHECKING_REGISTRATION_LISTENER=SUBNET      #
line added by Agent
$
```

3. Find the user running the LISTENER process. grid is the UNIX user running the process in /u01/app/grid/product/12.1.0/grid, the new Grid home of the 12c Oracle Grid Infrastructure.

```
$ ps -ef | grep tnslsnr
grid      4340      1  0 13:12 ?          00:00:00
/u01/app/grid/product/12.1.0/grid_1/bin/tnslsnr LISTENER -
no_crs_notify -inherit
grid      8800  32043  0 13:30 pts/3      00:00:00 grep tnslsnr
$
```

Practices for Lesson 4: Installing your Oracle Software

Chapter 4

Practices for Lesson 4

Practices Overview

The installation of the Oracle Grid Infrastructure software is complete. In this practice you install the Oracle Database 12c software to enable you to create Oracle Database 12c databases.

The installation media is staged at:

- /stage(shiphomes/RDBMS_LINUX.X64_12C/database for Oracle Database 12c

Perform the following tasks as the `oracle` OS user from the `oracle` user terminal window, unless otherwise indicated.

Note: Completing this practice is critical for all following practice sessions.

Practice 4-1: Installing the Oracle Database 12c Software

- From the `oracle` user terminal window, navigate to the `/stage(shiphomes/RDBMS_LINUX.X64_12C/database` directory. Start the Oracle Universal Installer (OUI) by entering `./runInstaller`.
 - Enter the following:

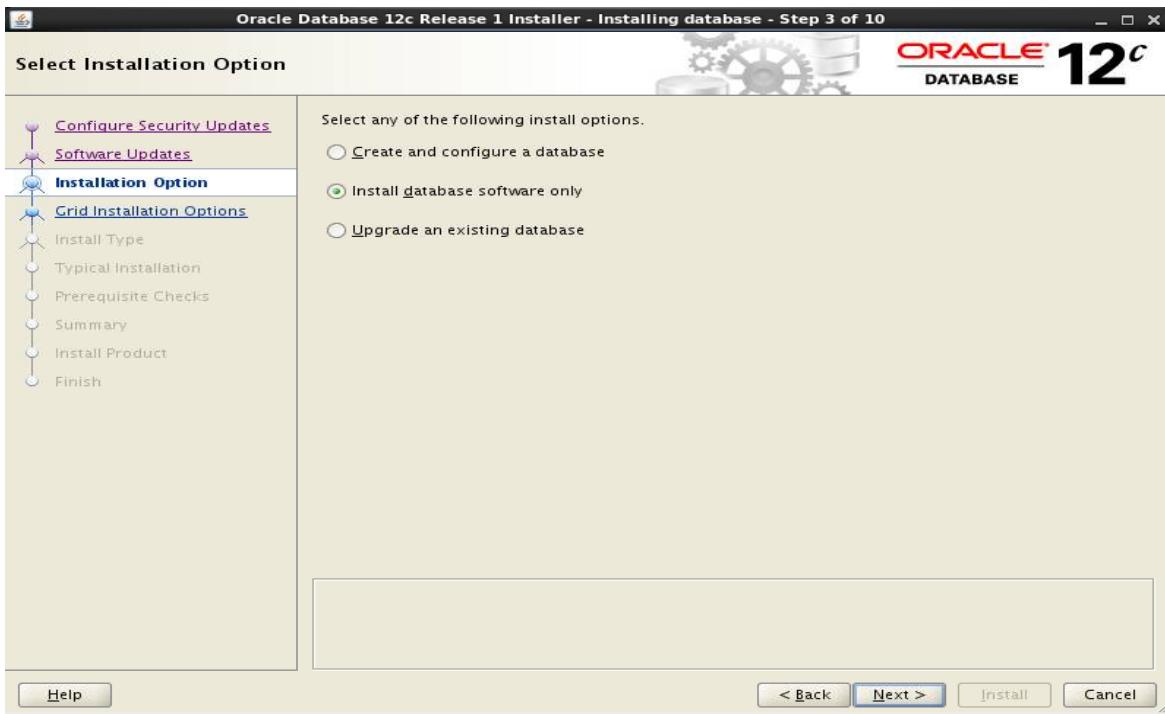
```
$ cd /stage(shiphomes/RDBMS_LINUX.X64_12C/database
$ ./runInstaller
```



Notice that Oracle Database 12c is displayed instead of the Oracle Grid Infrastructure 12c (check in the previous practice). The installation lasts 10 to 15 minutes.

Step	Window/Page Description	Choices or Values
a.	Configure Security Updates	Deselect the option to receive security updates from My Oracle Support. Note: In your real-world environment, you would enter your email address and My Oracle Support password; however, because the classroom is an isolated environment, leave the email and password fields null.
b.	Configure Security Updates	Click Next
c.	Email Address Not Specified warning	Click Yes
d.	Download Software Updates page	Verify Skip Software Updates is selected Click Next .
e.	Select Installation Option page	Select Install database software only Click Next Note: You will create databases by using DBCA in a later practice. The upgrade of the dbupgrd database will be performed in later

Step	Window/Page Description	Choices or Values
		practice.

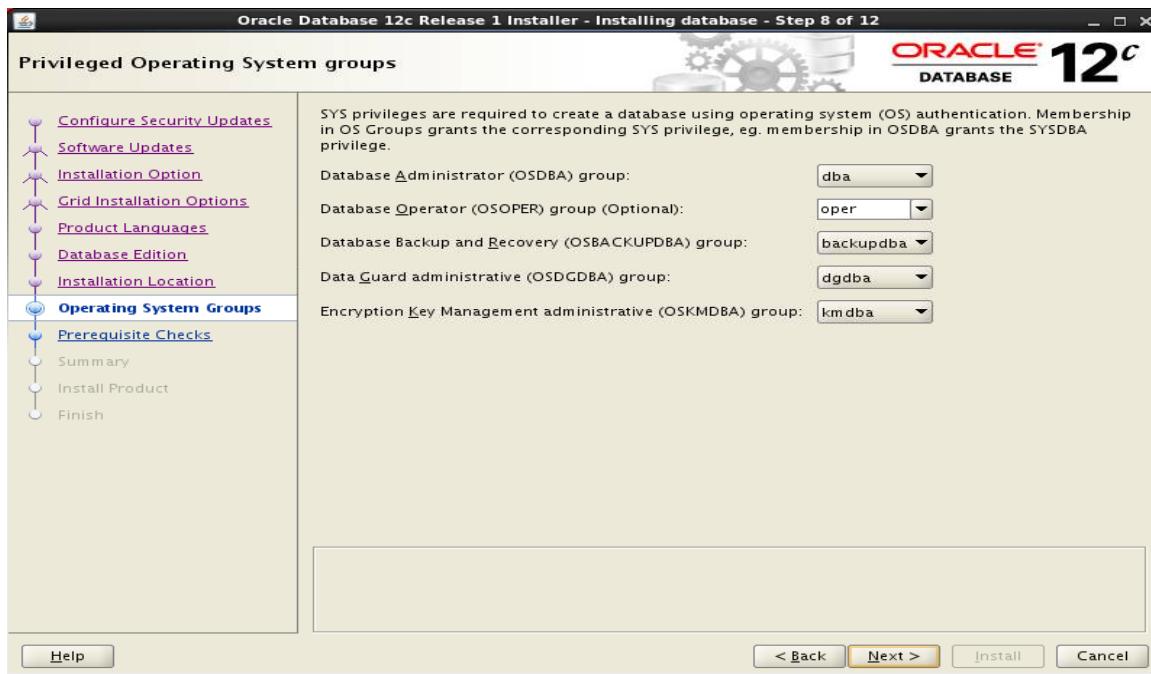


Step	Window/Page Description	Choices or Values
f.	Grid Installation Options page	<p>Verify Single instance database installation is selected</p> <p>Click Next</p> <p>Note: In this course you are not working on a Real Application Cluster system.</p>
g.	Select Product Languages page	Select all the available languages with the '">>>' button Click Next .
h.	Select Database Edition page	<p>Verify Enterprise Edition (6.4GB) is selected.</p> <p>Click Next.</p> <p>Note: With Enterprise Edition, any existing option can be installed which is not the case with Standard Edition. This explains the difference of space used.</p>
i.	Specify Installation Location page	<ol style="list-style-type: none"> Set Oracle Base to <code>/u01/app/oracle</code>. Set Software Location to <code>/u01/app/oracle/product/12.1.0/dbhome_1</code>. Note: You installed the Oracle Grid Infrastructure in <code>/u01/app/grid</code> base directory and you install the Oracle Database in <code>/u01/app/oracle</code> base directory.

Step	Window/Page Description	Choices or Values
		4. Click Next .

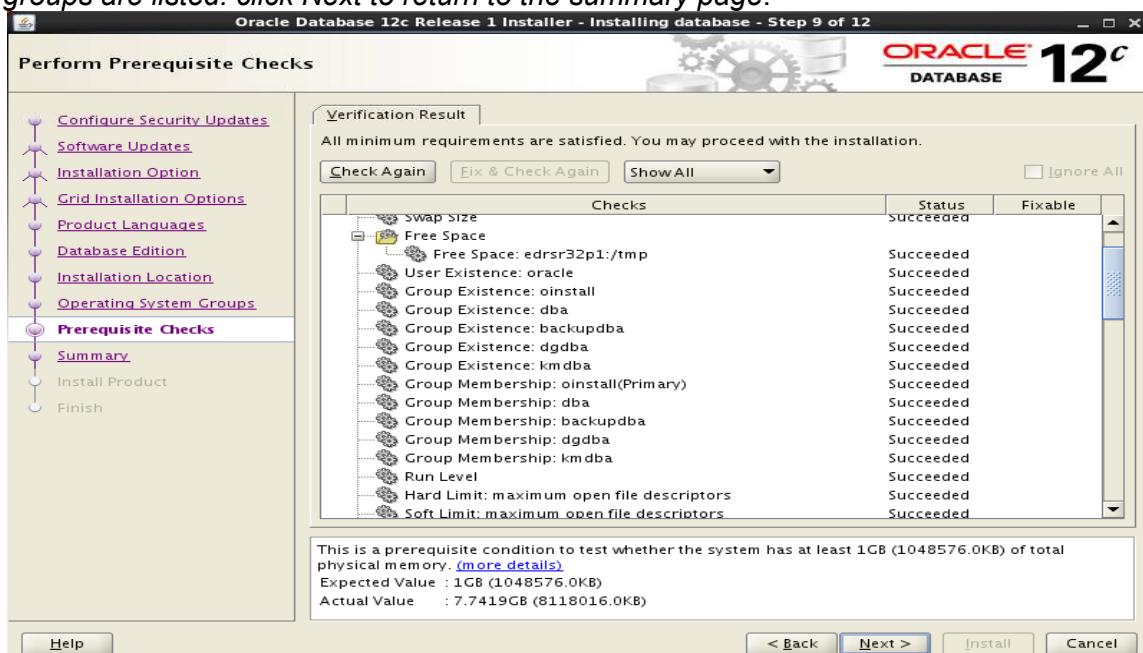


Step	Window/Page Description	Choices or Values
j.	Privileged Operating System groups page	<p>Keep distinct OS groups for separation of duties.</p> <p>Note: The OS groups, dba, oper, backupdba, dgdba and kmdba were created prior to running the installer. You can view them in the /etc/group file.</p> <p>Click Next</p>

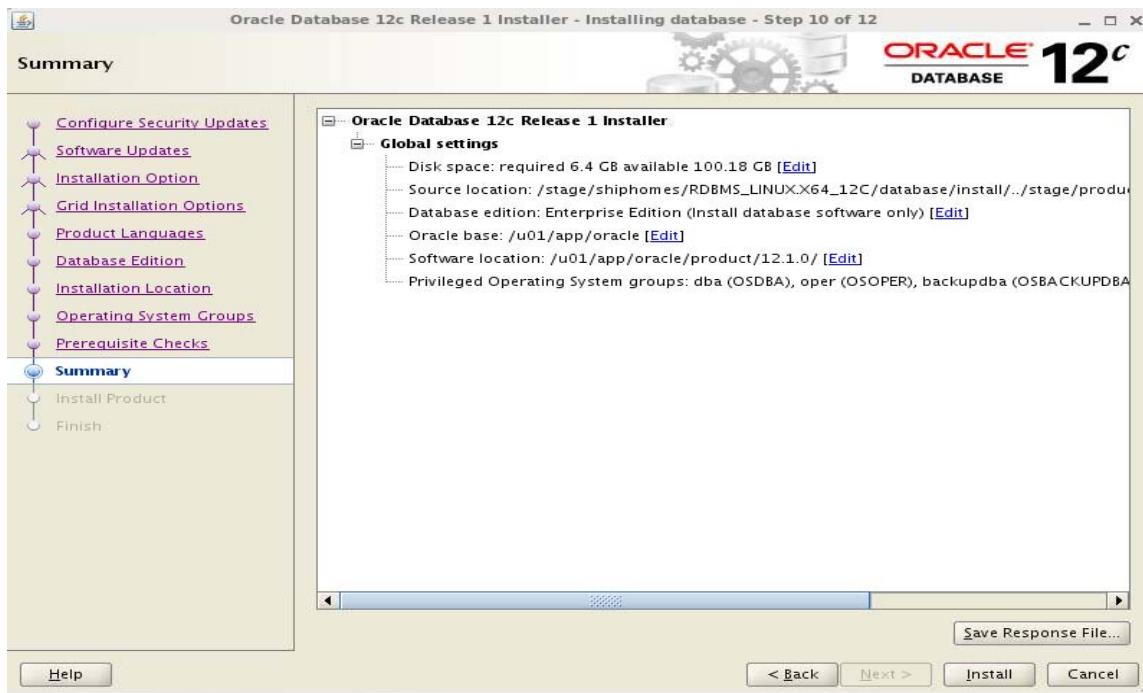


Step	Window/Page Description	Choices or Values
k.	Perform Prerequisites Checks	Appears if any checks failed

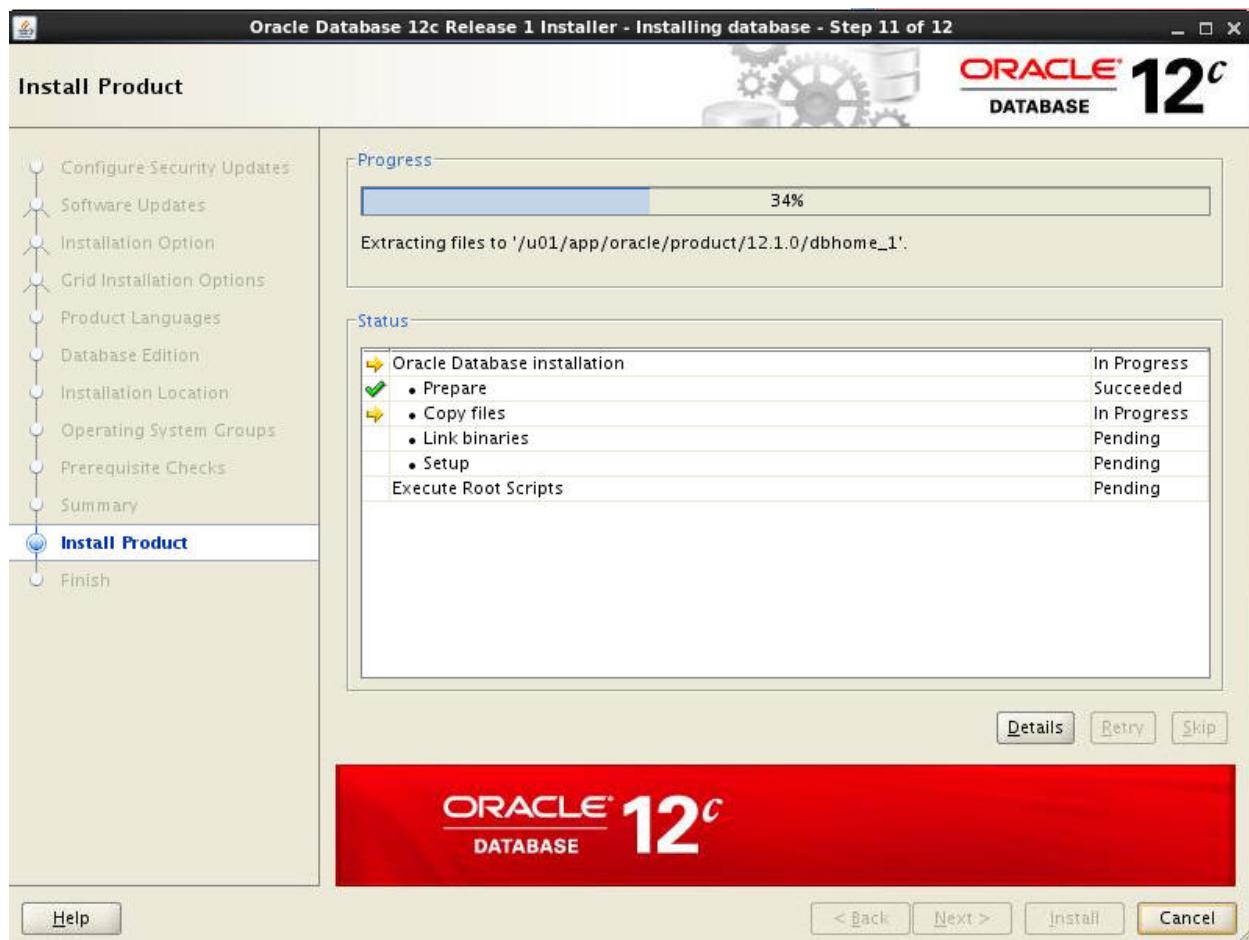
Note: However, if you want to list all the checks, click the Prerequisite Checks in the navigator in the left pane of the OUI. The check for User Existence is oracle. The groups are listed. click Next to return to the summary page.



Step	Window/Page Description	Choices or Values
l.	Summary page	Click Install



Step	Window/Page Description	Choices or Values
m.	Install Product page	Displays the progress of the installation and the status for each individual task being performed



When the Execute Configuration scripts window appears, follow the steps listed in the window.

Step	Window/Page Description	Choices or Values
n.	Execute Configuration scripts	Follow steps in window.

b. Open a terminal window and log in as root .

```
i. $ su -
ii. Password:
iii. #
```

c. Run the script shown in the Execute Configuration scripts window. Accept the default for the local bin directory and do not overwrite any files (you can just press [Enter] because the default option is to not overwrite).

```
# /u01/app/oracle/product/12.1.0/dbhome_1/root.sh
```

Performing root user operation for Oracle 12c

The following environment variables are set as:

ORACLE_OWNER= oracle

ORACLE_HOME= /u01/app/oracle/product/12.1.0/dbhome_1

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

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

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

- d. Exit from the root session.

```
# exit  
logout  
$ exit
```

Step	Window/Page Description	Choices or Values
o.	Execute Configuration	Click OK
p.	Finish page	Click Close

2. Return to the oracle user Unix session where runInstaller was launched and press [Enter].

```
$ You can find the log of this install session at:  
/u01/app/logs/installActions2013-02-25_02-27-55AM.log  
$
```

The name of the log file of the install is displayed.

Practices for Lesson 5: Creating an Oracle Database Using DBCA

Chapter 5

Practices for Lesson 5

Practices Overview

You will create two types Oracle databases in this practice:

- A non container database (non-CDB): `orcl`
- A multitenant container database (CDB): `cdb1`

You anticipate that several similar non-CDB databases will be needed in the near future. Therefore, you decide to create your `orcl` database, as well as a database template and the database creation scripts. Locate the scripts in the `/home/oracle/labs` directory (which is the directory that you use most often throughout this course).

The CDB named `cdb1` will contain sample schemas and therefore one pluggable database named `pdb1`.

Practice 5-1: Creating a Non-CDB

In this practice, you create the non-container database named `orcl`. You use the Database Configuration Assistant (DBCA) to create the database.

- From the `oracle` user terminal window, set your `ORACLE_HOME` environment variable by using `oraenv`. Enter `orcl` for the `SID` and then enter `/u01/app/oracle/product/12.1.0/dbhome_1` for the new `ORACLE_HOME`.

Note: You enter the full `ORACLE_HOME` path at this time because the `orcl` database does not exist yet. The entry is not yet registered in the `/etc/oratab` file and therefore the `ORACLE_HOME` is not known. After the database is created, you will only have to enter `orcl` as the `SID` and it will determine the appropriate `ORACLE_HOME` from the `/etc/oratab` file.

```
$ . oraenv
ORACLE_SID = [dbupgrd] ? orcl
ORACLE_HOME = [/home/oracle] ?
/u01/app/oracle/product/12.1.0/dbhome_1
The Oracle base remains unchanged with value /u01/app/oracle
$
```

The Oracle base remains `/u01/app/oracle` because you are creating a database and therefore still working in the Oracle Database environment.

- Start the Database Configuration Assistant (DBCA).

```
$ dbca
```

Step	Window/Page Description	Choices or Values
a.	Database Operation page	Select Create Database Click Next
b.	Creation Mode page	Select Advanced Mode . <i>Note: This option allows more customization.</i> Click Next .
c.	Database Template page	Select the General Purpose or Transaction Processing template. Click Show Details
d.	Template Details	Answer the following questions:

- Question 1:** How many control files are created?

Answer: Two

Note: The location is by default `{ORACLE_BASE}/oradata/{DB_UNIQUE_NAME}` for the first control file and `{ORACLE_BASE}/fast_recovery_area/{DB_UNIQUE_NAME}` for the second control file.

- ii) **Question 2:** How many redo log groups are created?

Answer: Three

Note: The location will change later in this practice when we choose to use filesystem as our storage technique.

- iii) **Question 3:** What is the database block size (`db_block_size`)?

Answer: 8 KB

- iv) **Question 4:** What is the value of Sample Schemas?

Answer: Sample Schemas is set to False.

Note: You will change this setting later in this practice so that the `HR` sample schema is included in your database.

- v) **Question 5:** What is the template default for the Database Character Set?

Answer: WE8MSWIN1252

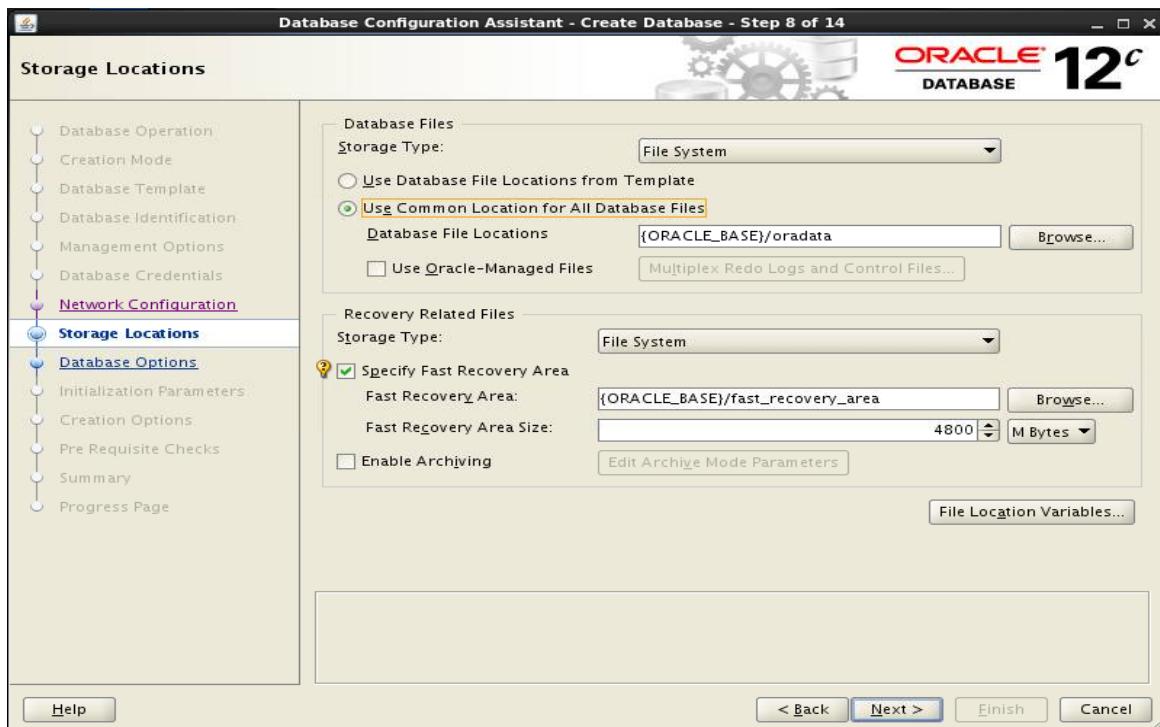
Note: You will change this setting later in this practice to use a Unicode database character set.

Step	Window/Page Description	Choices or Values
e.	Template Details window.	Click OK
f.	Database Template page	Click Next
g.	Database Identification page	<p>Set Global Database Name = <code>orcl</code>.</p> <p>Note: The SID defaults to the database name <code>orcl</code>.</p> <p>Verify that the Create As Container Database check box is NOT checked so that you will create a non-CDB.</p> <p>Click Next.</p>

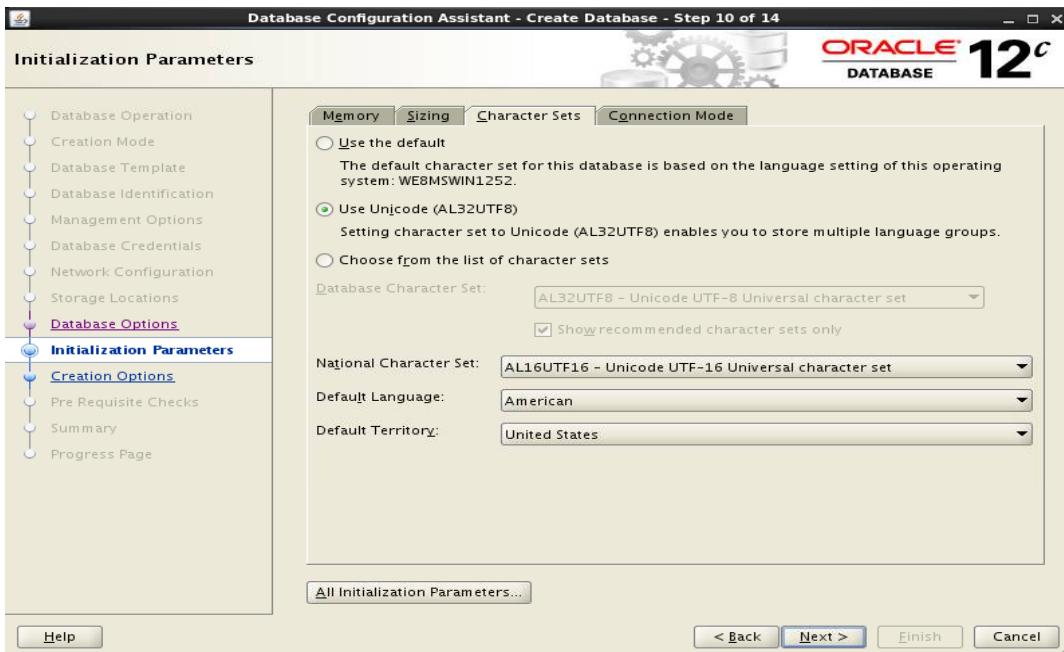
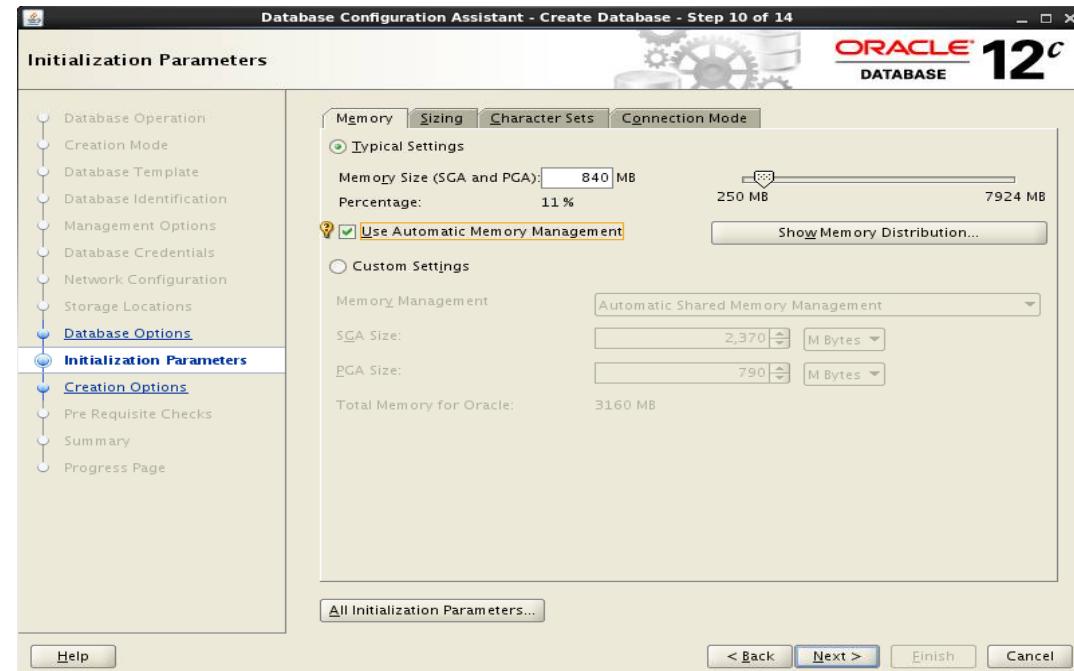
Note: it is common to input a fully qualified name for the global database name, for example: `orcl.example.com`. The global name often needs to be unique across the domain which in this case is `example.com`. Notice the SID excludes the domain name automatically.

Step	Window/Page Description	Choices or Values
h.	Management Options page	<p>Select Configure Enterprise Manager (EM) Database Express</p> <p>Note: Enterprise Manager Database Express allows you to perform DBA tasks through a graphical user interface</p> <p>Click Next</p>
i.	Database Credentials page	<i>Best practice tip: use separate passwords used for each account to help maintain separation of duties.</i>

Step	Window/Page Description	Choices or Values
		<p>Note: In this class, you are using the same password to minimize disruption to the practices due to a forgotten password.</p> <p>Select Use the Same Administrative Password for All Accounts</p> <p>Set Password = <code>oracle_4U</code> Set Confirm Password = <code>oracle_4U</code></p> <p>Click Next.</p>
j.	Network Configuration page Listener Selection tab	<p>Verify Listener named LISTENER with Port 1521 is selected and is Up, and running from the /u01/app/grid/product/12.1.0/grid directory</p> <p>Click Next</p>
k.	Storage Locations page	<p>Set Storage Type = File System</p> <p>Select Use Common Location for All Database Files</p> <p>Note: All the database files of the orcl non-CDB will be created in /u01/app/oracle/oradata/orcl directory. The RMAN backup files will be stored in /u01/app/oracle/fast_recovery_area/orcl. The orcl directories are created automatically - the {ORACLE_BASE}/oradata listed in the Database File Locations field is sufficient. There is no need to add the orcl directory. The same applies to the Fast Recovery Area field.</p> <p>Click Next.</p>



Step	Window/Page Description	Choices or Values
l.	Database Options page	Select Sample Schemas Note: This will create several sample schemas such as HR, OE, SH and others. Best Practice Tip: Do NOT create Sample Schemas in production databases. Sample Schemas are for training and testing. Click Next
m.	Initialization Parameters page Memory tab	Set Memory Size = 840 MB Select Use Automatic Memory Management
n.	Initialization Parameters page Character Sets tab	Select Use Unicode (AL32UTF8) .
o.	Initialization Parameters page Sizing tab	Review Settings, Do not change anything
p.	Initialization Parameters page Connection Mode tab	Review Settings, Do not change anything Click Next



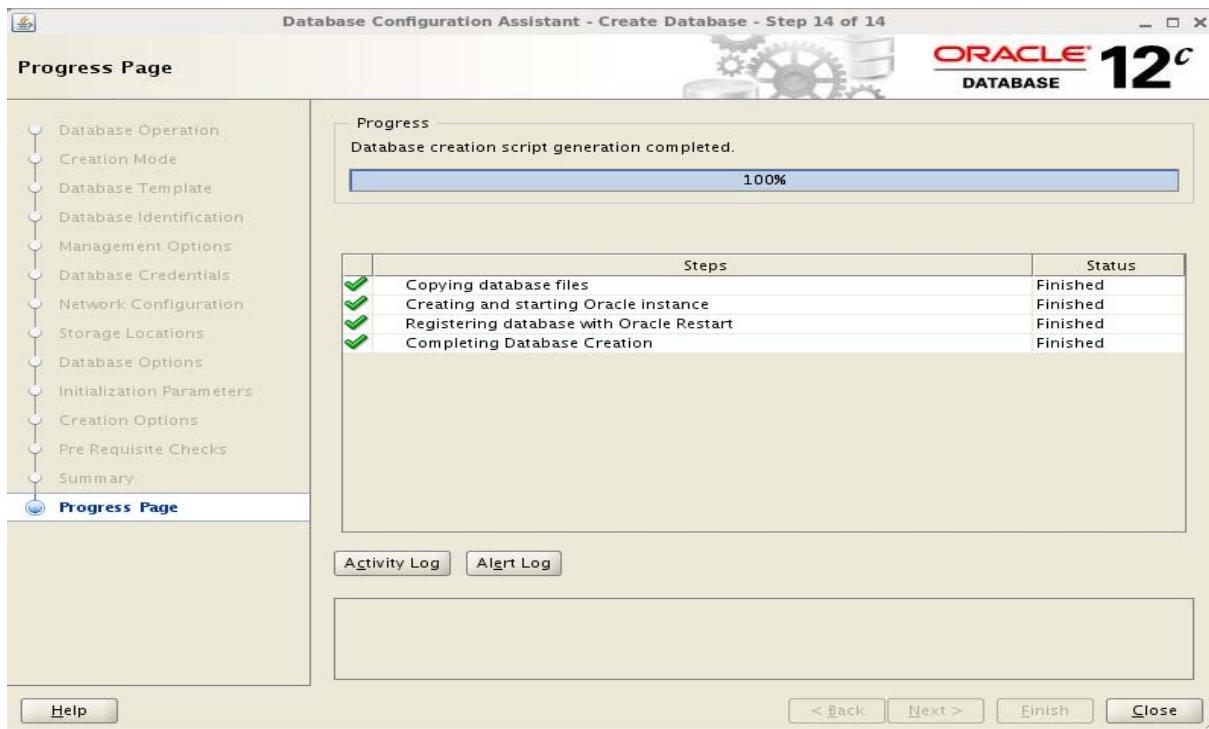
Step	Window/Page Description	Choices or Values
q.	Creation Options page	<p>Select Create Database</p> <p>Select Save as a Database Template option.</p> <ul style="list-style-type: none"> • Set Name = orcl as the Name for the database template • Set Description to orcl Database Template <p>Select Generate Database Creation Scripts</p>

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

Step	Window/Page Description	Choices or Values
		<p>Set Destination Directory = /home/oracle/labs Note: this directory must exist. Click Next.</p>
r.	Pre Requisite Checks	If there are no failed prerequisites, the DBCA moves immediately to the next page
s.	Summary page	Review options, parameters, locations and settings, specifically the ones in the table below.

Name	Value or Location Value
Sample Schemas	true
db_name	orcl
db_block_size	8KB
Data Files	{ORACLE_BASE}/oradata/{DB_UNIQUE_NAME}/*
Control Files	{ORACLE_BASE}/oradata/{DB_UNIQUE_NAME}/control01.ctl {ORACLE_BASE}/fast_recovery_area/{DB_UNIQUE_NAME}/control02.ctl
memory_target	840MB
Database Character Set	AL32UTF8

Step	Window/Page Description	Choices or Values
t.	Summary page	Click Finish
u.	Template has been created	Click OK
v.	Generation of the database scripts	Click OK
w.	Progress page	Displays the progress of the various installation steps. You can see that the database is automatically registered with Oracle Restart. You will learn about Oracle Restart in a later practice. This step takes about 15 minutes.

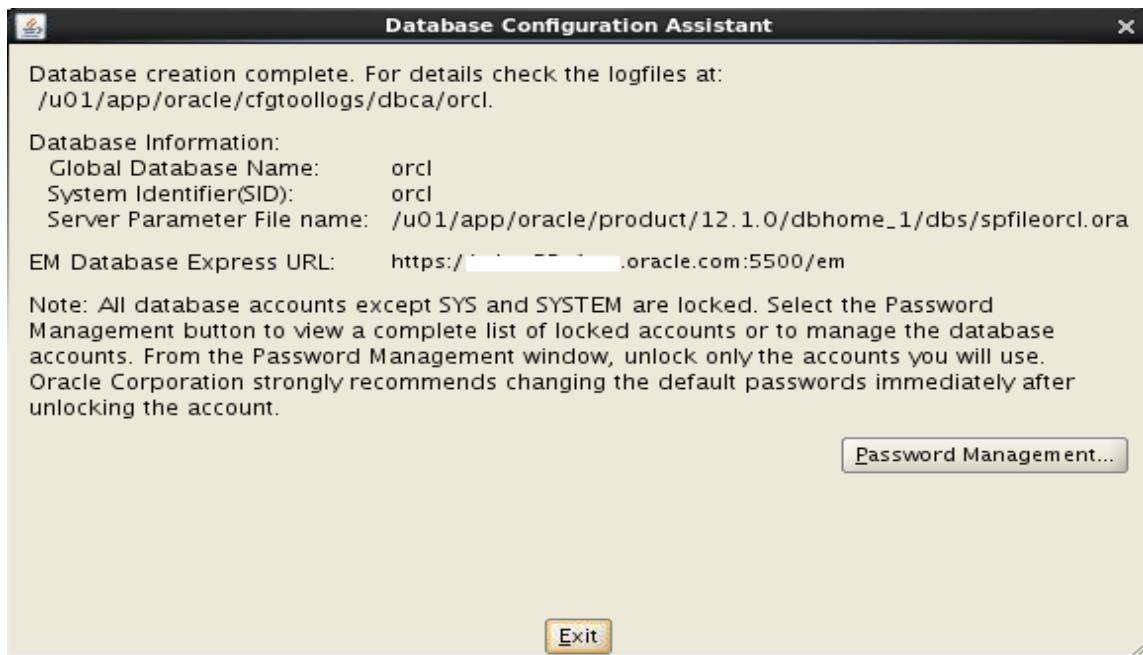


Step	Window/Page Description	Choices or Values
X.	Database creation complete	Displays essential information about the database. Make note Enterprise Manager Express URL, because will be used in several of the following practice sessions

Important: Make note of your Enterprise Manager Express URL here:

https://_____ : _____ /em

You will be using this URL many times throughout the remainder of the course.



Step	Window/Page Description	Choices or Values
y.	Database creation complete	Click Password Management
z.	Password Management page	Scroll down to find the HR user Deselect Lock Account? Set New Password: oracle_4U Confirm Password: oracle_4U Click OK
aa.	Database creation complete	Click Exit
bb.	Database Creation Assistant	Click Close

3. You have completed the creation of the non-CDB, of the `orcl` database template, and the database generation scripts. When you create another database, the `orcl` template will appear on the templates page. Check that the `orcl.sh` shell script exists.

Note: the `ls` command below has a “one” not “el” as the last character.

```
$ cd /home/oracle/labs
$ ls -tr -1
init.ora
orcl.sh
tempControl.ctl
initorclTempOMF.ora
rmanRestoreDatafiles.sql
CloneRmanRestore.sql
initorclTemp.ora
cloneDBCreation.sql
```

```
postScripts.sql  
lockAccount.sql  
orcl.sql  
postDBCreation.sql  
$
```

4. Verify that the `orcl` database instance entry is added in `/etc/oratab`.

```
$ cat /etc/oratab  
...  
+ASM:/u01/app/grid/product/12.1.0/grid:N: # line added by  
Agent  
dbupgrd:/u01/app/oracle/product/11.2.0/dbhome_2:N: #  
line added by  
Agent  
orcl:/u01/app/oracle/product/12.1.0/dbhome_1:N: # line  
added by Agent  
$
```

5. Verify that you can connect as `SYSTEM` and that the database name is `ORCL`. Do not mix up instance and database names.

- a. Use views to verify instance and database names. The banner displays “Oracle Database 12c Enterprise Edition Release **12.1.0.1.0**”.

```
$ sqlplus system/oracle_4U  
  
Connected to:  
Oracle Database 12c Enterprise Edition Release 12.1.0.1.0 -  
64bit Production  
  
SQL> select instance_name from v$instance;  
  
INSTANCE_NAME  
-----  
orcl  
  
SQL> select name from v$database;  
  
NAME  
-----  
ORCL  
  
SQL> EXIT  
$
```

- b. View the `ORACLE_SID` environment variable.

```
$ echo $ORACLE_SID
```

```
orcl
$
```

6. Verify the existence of initialization parameter files.

```
$ ls $ORACLE_HOME/dbs/*orcl*.ora
/u01/app/oracle/product/12.1.0/dbhome_1/dbs/spfileorcl.ora
$
```

Only the server parameter file exists. DBCA removes the parameter file after the creation of the database is complete. It is a good practice to have a parameter file to edit it when necessary to modify initialization parameters. But do not create the parameter file with the default name. A non-default name will prevent accidentally starting the database instance with a possibly out-of-date parameter file.

- a. Re-create the parameter file.

```
$ sqlplus system/oracle_4U

SQL> create pfile='bkuporcl.ora' from spfile;
create pfile from spfile
*
ERROR at line 1:
ORA-01031: insufficient privileges

SQL>
```

The SYSTEM user is not privileged enough to create a parameter file. Connect as SYSDBA.

```
SQL> connect / as sysdba
Connected.
SQL> create pfile='bkuporcl.ora' from spfile;

File created.

SQL> exit
$
```

- b. Edit the parameter file using gedit or your preferred editor to set the db_recovery_file_dest_size to **6G** instead of **4800m**. Save the changes and quit gedit.

```
$ gedit $ORACLE_HOME/dbs/bkuporcl.ora
```

```

orcl._data_transfer_cache_size=0
orcl._db_cache_size=306184192
orcl._java_pool_size=4194304
orcl._large_pool_size=8388608
orcl._oracle_base='/u01/app/oracle'#ORACLE_BASE set from environment
orcl._pga_aggregate_target=352321536
orcl._sga_target=528482304
orcl._shared_io_pool_size=20971520
orcl._shared_pool_size=176160768
orcl._streams_pool_size=0
*.audit_file_dest='/u01/app/oracle/admin/orcl/adump'
*.audit_trail='db'
*.compatible='12.0.0.0'
*.control_files='/u01/app/oracle/oradata/orcl/control01.ctl', '/u01/app/oracle/fast_recovery_area/orcl/control02.ctl'
*.db_block_size=8192
*.db_domain=''
*.db_name='orcl'
*.db_recovery_file_dest='/u01/app/oracle/fast_recovery_area'
*.db_recovery_file_dest_size=6G
*.diagnostic_dest='/u01/app/oracle'
*.dispatchers='(PROTOCOL=TCP) (SERVICE=orclXDB)'
*.local_listener='LISTENER_ORCL'
*.memory_target=840M
*.open_cursors=300
*.processes=300
*.remote_login_passwordfile='EXCLUSIVE'
*.undo_tablespace='UNDOTBS1'

```

- c. Be aware that if you shut the instance down and restart it without naming the initialization parameter file, the SPFILE is used by default. If you want the PFILE to be used, use the following steps:

- Shut the instance down.

```

$ sqlplus / as sysdba

SQL> shutdown immediate
Database closed.
Database dismounted.
ORACLE instance shut down.
SQL>

```

- Start the instance up with the PFILE clause.

```

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

Total System Global Area  876859392 bytes
Fixed Size                  2294360 bytes
Variable Size                62070488 bytes
Database Buffers              247463936 bytes
Redo Buffers                   6340608 bytes
Database mounted.
Database opened.

SQL> show parameter spfile
NAME                      TYPE          VALUE
-----
spfile                    string

SQL> show parameter db_recovery_file_dest_size

```

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

NAME	TYPE	VALUE
db_recovery_file_dest_size	big integer	6G

- iii) Restart the instance without specifying the initialization file and you find out that the server parameter file is used.

```
SQL> shutdown immediate
Database closed.
Database dismounted.
ORACLE instance shut down.

SQL> startup
ORACLE instance started.

Total System Global Area  881037312 bytes
Fixed Size                  2294408 bytes
```

```
Variable Size            620760440 bytes
Database Buffers        251658240 bytes
Redo Buffers             6324224 bytes
Database mounted.
Database opened.

SQL> show parameter spfile
NAME          TYPE        VALUE
-----
spfile        string     /u01/app/oracle/product/12.1.0/dbhome_1/dbs/spfileorcl.ora

SQL>
SQL> show parameter db_recovery_file_dest_size

NAME          TYPE        VALUE
-----
db_recovery_file_dest_size  big integer  4800M
SQL> EXIT
$
```

7. View the alert.log file. You can use the Automatic Diagnostic Repository Command Interface (ADRCI), any text editor, or UNIX command to read the alert.log.
 a. Use ADRCI.

```
$ adrci
ADRCI: Release 12.1.0.1.0 - Production on Wed Feb 27 20:10:48
2013
```

```
Copyright (c) 1982, 2013, Oracle and/or its affiliates. All rights reserved.
```

```
ADR base = "/u01/app/oracle"
adrci> show alert -tail
DIA-48449: Tail alert can only apply to single ADR home

adrci> EXIT
$
```

- b. Set the ADR home to the ADR of the `orcl` database instance. If you do not know it, you can find it using the `V$DIAG_INFO` view.

```
$ sqlplus / as sysdba

SQL> col value format A54
SQL> col name format A22
SQL> select name, value from v$diag_info;

NAME          VALUE
-----
Diag Enabled    TRUE
ADR Base        /u01/app/oracle
ADR Home        /u01/app/oracle/diag/rdbms/orcl/orcl
Diag Trace      /u01/app/oracle/diag/rdbms/orcl/orcl/trace
Diag Alert       /u01/app/oracle/diag/rdbms/orcl/orcl/alert
Diag Incident   /u01/app/oracle/diag/rdbms/orcl/orcl/incident
Diag Cdmp       /u01/app/oracle/diag/rdbms/orcl/orcl/cdmp
Health Monitor  /u01/app/oracle/diag/rdbms/orcl/orcl/hm
Default Trace File
/u01/app/oracle/diag/rdbms/orcl/orcl/trace/orcl_ora_28
                           725.trc

Active Problem Count  0
Active Incident Count 0

11 rows selected.

SQL> EXIT
$
```

- c. Set the ADR home to the ADR of the `orcl` database instance. The text of the alert file will vary from the example shown below.

```
$ adrci
adrci> set homepath diag/rdbms/orcl/orcl
```

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

```
adrci> show alert -tail
2013-02-27 20:09:20.484000 -08:00
No Resource Manager plan active
Starting background process FBDA
FBDA started with pid=7, OS id=28197
replication_dependency_tracking turned off (no async multimaster
replication found)
2013-02-27 20:09:23.566000 -08:00
Starting background process AQPC
AQPC started with pid=28, OS id=28204
Starting background process CJQ0
Completed: ALTER DATABASE OPEN
CJQ0 started with pid=36, OS id=28221
2013-02-27 20:09:24.812000 -08:00
db_recovery_file_dest_size of 4800 MB is 0.00% used. This is a
user-specified limit on the amount of space that will be used by
this
database for recovery-related files, and does not reflect the
amount of
space available in the underlying filesystem or ASM diskgroup.
adrci>
```

- d. Find any existing error.

```
adrci> show alert -p "MESSAGE_TEXT LIKE '%ORA-%'"
```

```
2013-02-27 19:26:00.850000 -08:00
ORA-1109 signalled during: ALTER DATABASE CLOSE NORMAL...
2013-02-27 19:26:20.309000 -08:00
Errors in file /u01/app/oracle/diag/rdbms/orcl/orcl/trace/orcl_ora_24939.trc:
ORA-00313: open failed for members of log group 1 of thread 1
ORA-00312: online log 1 thread 1: '/u01/app/oracle/oradata/orcl redo01.log'
ORA-27037: unable to obtain file status
Linux-x86_64 Error: 2: No such file or directory
Additional information: 3
Errors in file /u01/app/oracle/diag/rdbms/orcl/orcl/trace/orcl_ora_24939.trc:
ORA-00313: open failed for members of log group 1 of thread 1
ORA-00312: online log 1 thread 1: '/u01/app/oracle/oradata/orcl redo01.log'
ORA-27037: unable to obtain file status
Linux-x86_64 Error: 2: No such file or directory
Additional information: 3
Errors in file /u01/app/oracle/diag/rdbms/orcl/orcl/trace/orcl_ora_24939.trc:
ORA-00313: open failed for members of log group 1 of thread 1
ORA-00312: online log 1 thread 1: '/u01/app/oracle/oradata/orcl redo01.log'
ORA-27037: unable to obtain file status
Linux-x86_64 Error: 2: No such file or directory
Additional information: 3
2013-02-27 19:26:21.816000 -08:00
Errors in file /u01/app/oracle/diag/rdbms/orcl/orcl/trace/orcl_ora_24939.trc:
ORA-00313: open failed for members of log group 2 of thread 1
"/tmp/alert_29068_1397_orcl_2.ado" 59L, 3173C
```

If you need to read all errors, scroll down the temporary file. If you need to keep the information in a file, enter :w /tmp/mylogfile. This creates a permanent file.

Enter :q to quit the temporary file.

```
ADR Home = /u01/app/oracle/diag/rdbms/orcl/orcl:  
*****  
Output the results to file: /tmp/alert_29068_1397_orcl_1.ado  
adrci> EXIT  
$
```

- e. Use any text editor or UNIX command to read the alert.log. The text of the alert file will vary from the example shown below.

Note: the tail command in the example shows the last 30 lines in the file.

```
$ cd $ORACLE_BASE/diag/rdbms/orcl/orcl/trace  
$ tail -30 alert_orcl.log  
Current log# 1 seq# 10 mem# 0:  
/u01/app/oracle/oradata/orcl/redo01.log  
Successful open of redo thread 1  
MTTR advisory is disabled because FAST_START_MTTR_TARGET is not  
set  
SMON: enabling cache recovery  
[16330] Successfully onlined Undo Tablespace 2.  
Undo initialization finished serial:0 start:55183694  
end:55183774 diff:80 ms (0.1 seconds)  
Verifying file header compatibility for 11g tablespace  
encryption..  
Verifying 11g file header compatibility for tablespace  
encryption completed  
SMON: enabling tx recovery  
Starting background process SMCO  
Mon Feb 25 03:35:25 2013  
SMCO started with pid=25, OS id=16336  
Database Characterset is AL32UTF8  
No Resource Manager plan active  
Starting background process FBDA  
Mon Feb 25 03:35:25 2013  
FBDA started with pid=26, OS id=16338  
replication_dependency_tracking turned off (no async multimaster  
replication found)  
Starting background process AQPC  
Mon Feb 25 03:35:26 2013  
AQPC started with pid=27, OS id=16340  
Starting background process CJQ0  
Completed: ALTER DATABASE OPEN  
Mon Feb 25 03:35:29 2013  
CJQ0 started with pid=37, OS id=16363
```

```
Mon Feb 25 03:35:33 2013
db_recovery_file_dest_size of 4800 MB is 0.00% used. This is a
user-specified limit on the amount of space that will be used by
this
database for recovery-related files, and does not reflect the
amount of
space available in the underlying filesystem or ASM diskgroup.
$
```

8. Verify that you can connect as the HR user, one of the sample schemas.

```
$ sqlplus hr/oracle_4U

SQL> show user
USER is "HR"
SQL> select table_name from user_tables;

TABLE_NAME
-----
COUNTRIES
JOB_HISTORY
EMPLOYEES
JOBS
REGIONS
DEPARTMENTS
LOCATIONS

7 rows selected.

SQL>
```

9. List all data files, control files and redo log files of the orcl database.

- a. List all data files.

```
SQL> connect system/oracle_4U
Connected.
SQL> select name from v$datafile;

NAME
-----
/u01/app/oracle/oradata/orcl/system01.dbf
/u01/app/oracle/oradata/orcl/example01.dbf
/u01/app/oracle/oradata/orcl/sysaux01.dbf
/u01/app/oracle/oradata/orcl/undotbs01.dbf
/u01/app/oracle/oradata/orcl/users01.dbf
```

```
SQL>
```

The data files are all stored in the \$ORACLE_BASE/oradata/orcl directory.

- b. List all control files.

```
SQL> select name from v$controlfile;
```

NAME

```
-----  
/u01/app/oracle/oradata/orcl/control01.ctl  
/u01/app/oracle/fast_recovery_area/orcl/control02.ctl
```

```
SQL>
```

One control file resides in the \$ORACLE_BASE/oradata/orcl directory and another control file resides in the \$ORACLE_BASE/fast_recovery_area/orcl directory.

- c. List all redo log files.

```
SQL> select member from v$logfile;
```

MEMBER

```
-----  
/u01/app/oracle/oradata/orcl/redo03.log  
/u01/app/oracle/oradata/orcl/redo02.log  
/u01/app/oracle/oradata/orcl/redo01.log
```

```
SQL>
```

There are three redo log groups. Each redo log group owns a single member residing in the \$ORACLE_BASE/oradata/orcl directory. The default configuration does not create multiplexed redo log groups. Best practice is for each group to have two members, each on a different disk drive.

- d. If you had another disk on /u02, you could multiplex the redo log groups to this disk. On your server you only have /u01. You use it to multiplex.

```
SQL> alter database add logfile member  
'/u01/app/oracle/oradata/orcl/redo01b.log' to group 1;
```

Database altered.

```
SQL> alter database add logfile member  
'/u01/app/oracle/oradata/orcl/redo02b.log' to group 2;
```

Database altered.

```
SQL> alter database add logfile member  
'/u01/app/oracle/oradata/orcl/redo03b.log' to group 3;
```

Database altered.

```
SQL> col member format A50
SQL> select group#, member from v$logfile order by 1;

GROUP# MEMBER
-----
1 /u01/app/oracle/oradata/orcl/redo01.log
1 /u01/app/oracle/oradata/orcl/redo01b.log
2 /u01/app/oracle/oradata/orcl/redo02b.log
2 /u01/app/oracle/oradata/orcl/redo02.log
3 /u01/app/oracle/oradata/orcl/redo03b.log
3 /u01/app/oracle/oradata/orcl/redo03.log

6 rows selected.

SQL> exit
$
```

Practice 5-2: Creating a CDB

Overview

In this practice, you create the `cdb1` multitenant container database. You use the Database Configuration Assistant (DBCA) to create the CDB.

- Still from the `oracle` user terminal window, set your `ORACLE_HOME` environment variable by using `oraenv`. Enter `cdb1` for the `SID` and then enter `/u01/app/oracle/product/12.1.0/dbhome_1` for `ORACLE_HOME`. The entry is not yet registered in the `/etc/oratab` file and therefore the `ORACLE_HOME` is not known.

```
$ . oraenv
ORACLE_SID = [orcl] ? cdb1
ORACLE_HOME = [/home/oracle] ?
/u01/app/oracle/product/12.1.0/dbhome_1
The Oracle base remains unchanged with value /u01/app/oracle
$
```

- Start the Database Configuration Assistant (DBCA) and perform the following steps:

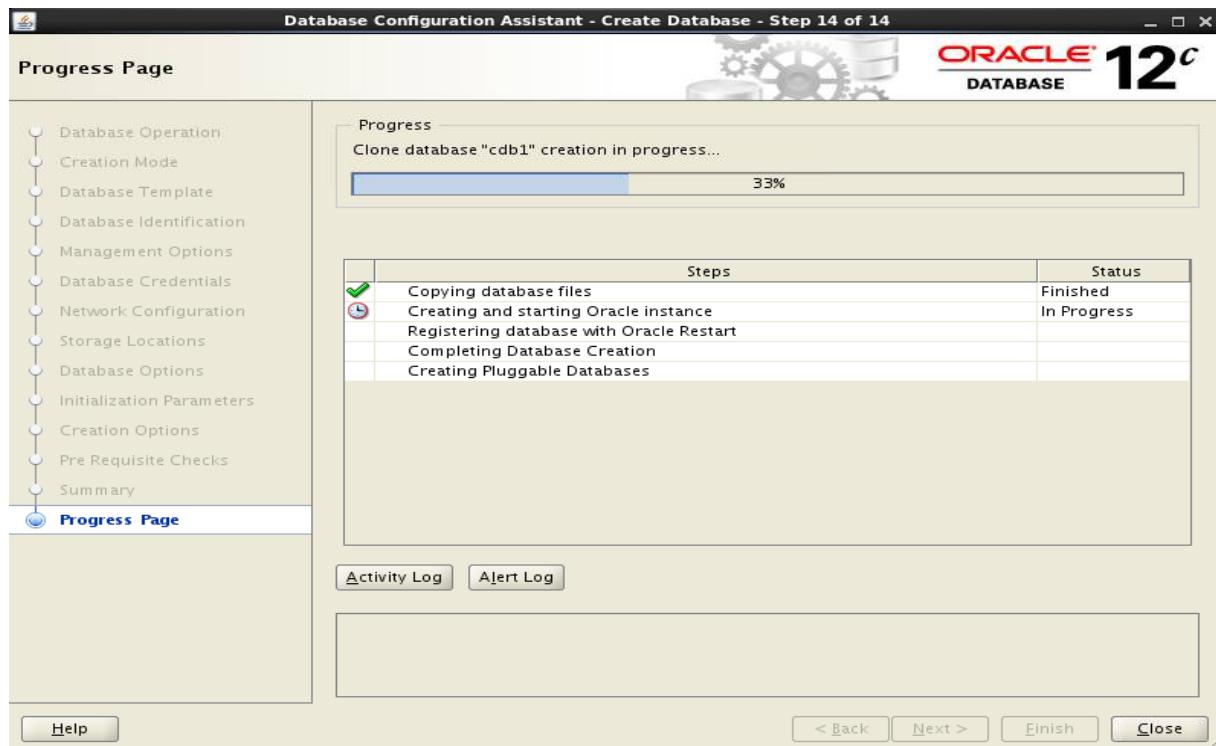
```
$ dbca
```

Step	Window/Page Description	Choices or Values
a.	Step 1: Database Operation	Select “ Create Database ”. Click Next .
b.	Step 2: Creation Mode	Select “ Advanced Mode ”. Click Next .
c.	Step 3: Database Template	Notice that the template <code>orcl</code> created in practice 5-1 is available. Select “ General Purpose or Transaction Processing ”. Click Next .
d.	Step 4: Database Identification	Enter Global Database Name: <code>cdb1</code> SID: <code>cdb1</code> Select “ Create As Container Database ” Select “ Create a Container Database with one or more PDBs ” Enter PDB Name: <code>pdb1</code> Click Next .
e.	Step 5: Management Options	Deselect “Configure Enterprise Manager (EM) Database Express”. Note: each database using EM Database Express requires a separate port. The default port of 5500 has already been assigned so it can be used again for this database.

Step	Window/Page Description	Choices or Values
		Click Next .
f.	Step 6: Database Credentials	Notice that a new User Name appears: PDBADMIN, this user could be the DBA of the pdb1 PDB. Select “ Use same Administrative password... ” Enter: Password: oracle_4U Confirm password: oracle_4U Click Next .
g.	Step 7: Network Configuration	Listener Selection: Click Next
h.	Step 8: Storage Locations	Confirm Storage type is “ File System ”. Select “ Use Common Location for All Database Files ”. Click Next .
i.	Step 9: Database Options	Select Sample Schemas . Click Next .
j.	Step 10: Initialization Parameters Memory Tab	Enter Memory Size (SGA and PGA): 844 MB Check the Use Automatic Memory Management Select “Character Sets” tab Select “ Use Unicode (AL32UTF8) ”. Click Next .
k.	Step 11: Creation Options	Select “ Create Database ”. Click Next .
l.	Step 12: Pre Requisite Checks	Note if there are no warnings or errors, this page automatically advances to the next.
m.	Step 13: Summary	Review options, parameters, locations and settings Verify Number of Pluggable Databases is one Sample Schema PDB is named pdb1 Parameter enable_pluggable_database is true. Click Finish .
n.	Step 14: Progress Page	The DBCA displays the progress of the installation steps. This operation takes about 25 minutes.

Step 14 is detailed below.

- You can see that the database is automatically registered with Oracle Restart. You will learn about Oracle Restart in another practice. After the CDB is created, not only is the CDB is created but also a PDB inside the CDB. There are two PDBs created: the PDB\$SEED PDB, automatically created in any CDB as a PDB template to create other PDBs, and pdb1.



- When the database is created, the DBCA displays essential information about the database. Click the **Password Management** button. Scroll down the Password Management page to find the `HR` username. You will not find the `HR` username because it is created only in the PDB, not in the root container of the CDB. Click **Cancel**.



- Click **Exit** to close the Database Configuration Assistant window. Then click **Close** to quit DBCA.
3. Verify that the `cdb1` database instance entry is added in `/etc/oratab`.

```
$ cat /etc/oratab
...
+ASM:/u01/app/grid/product/12.1.0/grid:N:          # line added by
Agent
dbupgrd:/u01/app/oracle/product/11.2.0/dbhome_2:N:      #
line added by
Agent
orcl:/u01/app/oracle/product/12.1.0/dbhome_1:N:          # line
added by Agent
cdb1:/u01/app/oracle/product/12.1.0/dbhome_1:N:          # line
added by Agent
$
```

4. You have completed the creation of the CDB, including a PDB. Verify that you can connect as `SYSTEM` and that the database name is `CDB1`. Use views to verify the instance and database names. The `CON_NAME` and `CON_ID` values allow you to verify in which container in the CDB you are connected to.

```
$ sqlplus system/oracle_4U

Connected to:
Oracle Database 12c Enterprise Edition Release 12.1.0.1.0 -
64bit Production

SQL> show con_name
```

```
CON_NAME
-----
CDB$ROOT
SQL> show con_id

CON_ID
-----
1
SQL> select instance_name from v$instance;

INSTANCE_NAME
-----
cdb1

SQL> select name from v$database;

NAME
-----
CDB1

SQL>
```

The container ID for the root is 1.

5. Display the status of the pdb1 PDB.

```
SQL> select name, open_mode from v$pdb$;

NAME          OPEN_MODE
-----
PDB$SEED      READ ONLY
PDB1          READ WRITE

SQL>
```

The PDB\$SEED PDB is automatically created in any CDB as a PDB template to create other PDBs. The seed PBD is always READ ONLY.

The PDB1 PDB has been created and DBCA automatically put the PDB in READ WRITE mode.

6. Connect to pdb1 as SYSDBA.

```
SQL> connect sys/oracle_4U@localhost:1521/pdb1 as sysdba
Connected.
SQL> show con_name

CON_NAME
-----
```

```
PDB1
SQL> show con_id

CON_ID
-----
3
SQL>
```

The container ID for pdb1 is 3. The container ID 2 is PDB\$SEED PDB.

7. Connect to pdb1 as HR.

- a. Use the Easy Connect syntax to connect to pdb1 as HR.

```
SQL> connect hr/oracle_4U@localhost:1521/pdb1
ERROR:
ORA-28000: the account is locked

Warning: You are no longer connected to ORACLE.

SQL>
```

- b. Connect as the administrator of the PDB to unlock the HR account. The HR user is a local user only known in its PDB. Therefore, only the administrator of the PDB can unlock the user.

```
SQL> connect sys/oracle_4U@localhost:1521/pdb1 as sysdba
Connected.
SQL> alter user hr identified by oracle_4U account unlock;

User altered.

SQL>
```

- c. Reattempt the connection to pdb1 as HR.

```
SQL> connect hr/oracle_4U@localhost:1521/pdb1
Connected.

SQL>
```

- d. Verify that HR is connected in pdb1.

```
SQL> show con_name

CON_NAME
-----
PDB1
SQL>
```

- e. List the tables owned by the HR sample schema.

```
SQL> select table_name from user_tables;
```

```
TABLE_NAME
-----
COUNTRIES
JOB_HISTORY
EMPLOYEES
JOBS
REGIONS
DEPARTMENTS
LOCATIONS

7 rows selected.
```

```
SQL> exit
$
```

- Verify the existence of initialization parameter files.

```
$ ls $ORACLE_HOME/dbs/*cdb1*.ora
/u01/app/oracle/product/12.1.0/dbhome_1/dbs/spfilecdb1.ora
$
```

Only the SPFILE server parameter file exists. DBCA removes the PFILE parameter file after the creation of the database is complete. It is always a good habit to have a parameter file to edit when necessary to modify initialization parameters.

Also Notice pdb1 does not have an SPFILE.

- Re-create the parameter file with a non-default name. The SYSTEM user is not privileged enough to create a parameter file. Connect as SYSDBA.

```
$ sqlplus / as sysdba

SQL> create pfile='bkupcdb1.ora' from spfile;

File created.

SQL> exit
$ 
$ ls $ORACLE_HOME/dbs/*cdb1*.ora
/u01/app/oracle/product/12.1.0/dbhome_1/dbs/bkupcdb1.ora
/u01/app/oracle/product/12.1.0/dbhome_1/dbs/spfilecdb1.ora
$
```

- Be aware that if you shut down the instance and restart it, the PDBs (only one in our case excluding PDB\$SEED which remains in READ ONLY mode) are all closed or MOUNTED by default.

- Shut the instance down.

```
$ sqlplus / as sysdba
```

```
SQL> shutdown immediate
Database closed.
Database dismounted.
ORACLE instance shut down.
SQL>
```

- b. Start the instance up.

```
SQL> startup
ORACLE instance started.

Total System Global Area  881037312 bytes
Fixed Size                  2294408 bytes
Variable Size                620760440 bytes
Database Buffers            251658240 bytes
Redo Buffers                 6324224 bytes
Database mounted.
Database opened.
SQL> select name, open_mode from v$pdbs;

NAME          OPEN_MODE
-----
PDB$SEED      READ ONLY
PDB1          MOUNTED

SQL>
```

- c. If you want the PDBs to be opened when the CDB instance is started and the root container opened, create a trigger as follows:

```
CREATE TRIGGER open_all_PDBs
  AFTER STARTUP ON DATABASE
begin
    execute immediate 'alter pluggable database all open';
end open_all_PDBs;
/
SQL> CREATE TRIGGER open_all_PDBs
  2  after startup on database
  3  begin
  4    execute immediate 'alter pluggable database all open';
  5  end open_all_PDBs;
  6  /
Trigger created.

SQL>
```

Use the trigger to open all PDBs or only some PDBs.

- d. Shut down cdb1.

```
SQL> shutdown immediate
Database closed.
Database dismounted.
ORACLE instance shut down.
SQL>
```

- e. Start up cdb1.

```
SQL> startup
ORACLE instance started.

Total System Global Area  881037312 bytes
Fixed Size                  2294408 bytes
Variable Size                620760440 bytes
Database Buffers            251658240 bytes
Redo Buffers                 6324224 bytes
Database mounted.
Database opened.
SQL>
```

- f. Notice that the pdb1 is now in READ WRITE open mode.

```
SQL> select CON_ID, NAME, OPEN_MODE from v$pdbs;

  CON_ID NAME          OPEN_MODE
----- -----
    2     PDB$SEED      READ ONLY
    3     PDB1          READ WRITE

SQL> exit
$
```

11. View the alert log file to list the operations performed on the cdb1, pdb\$seed and pdb1 databases by using ADRCI.

```
$ adrci
adrci> set homepath diag/rdbms/cdb1/cdb1
adrci> show alert -tail 30
2013-02-27 21:31:42.637000 -08:00
MMNL started with pid=20, OS id=3066
starting up 1 shared server(s) ...
ORACLE_BASE from environment = /u01/app/oracle
ALTER DATABASE MOUNT
2013-02-27 21:31:45.472000 -08:00
Using default pga_aggregate_limit of 2048 MB
```

```
2013-02-27 21:31:47.377000 -08:00
Successful mount of redo thread 1, with mount id 782251263
Database mounted in Exclusive Mode
Lost write protection disabled
Ping without log force is disabled.
Completed: ALTER DATABASE MOUNT
ALTER DATABASE OPEN
Starting background process TMON
TMON started with pid=23, OS id=3092
Thread 1 opened at log sequence 21
  Current log# 3 seq# 21 mem# 0:
/u01/app/oracle/oradata/cdb1/redo03.log
Successful open of redo thread 1
MTTR advisory is disabled because FAST_START_MTTR_TARGET is not
set
SMON: enabling cache recovery
[3090] Successfully onlined Undo Tablespace 2.
Undo initialization finished serial:0 start:292530234
end:292530314 diff:80 ms (0.1 seconds)
Verifying file header compatibility for 11g tablespace
encryption..
Verifying 11g file header compatibility for tablespace
encryption completed
SMON: enabling tx recovery
Starting background process SMCO
SMCO started with pid=24, OS id=3096
Database Characterset is AL32UTF8
2013-02-27 21:31:48.478000 -08:00
No Resource Manager plan active
replication_dependency_tracking turned off (no async multimaster
replication found)
Starting background process AQPC
AQPC started with pid=26, OS id=3105
2013-02-27 21:31:49.861000 -08:00
Due to limited space in shared pool (need 6094848 bytes, have
3981120 bytes), limiting Resource Manager entities from 2048 to
32
Opening pdb PDB$SEED (2) with no Resource Manager plan active
db_recovery_file_dest_size of 4800 MB is 0.00% used. This is a
user-specified limit on the amount of space that will be used by
this
database for recovery-related files, and does not reflect the
amount of
space available in the underlying filesystem or ASM diskgroup.
```

```
2013-02-27 21:31:52.744000 -08:00
alter pluggable database all open
2013-02-27 21:31:53.724000 -08:00
Due to limited space in shared pool (need 6094848 bytes, have
3981120 bytes), limiting Resource Manager entities from 2048 to
32
Opening pdb PDB1 (3) with no Resource Manager plan active
Pluggable database PDB1 opened read write
Completed: alter pluggable database all open
2013-02-27 21:31:54.782000 -08:00
Starting background process CJQ0
CJQ0 started with pid=27, OS id=3161
Completed: ALTER DATABASE OPEN
adrci> exit
$
```

12. List all data files, control files and redo log files of the cdb1 database.

- a. List all the data files.

```
$ sqlplus system/oracle_4U

SQL> select name from v$datafile;

NAME
-----
/u01/app/oracle/oradata/cdb1/system01.dbf
/u01/app/oracle/oradata/cdb1/sysaux01.dbf
/u01/app/oracle/oradata/cdb1/undotbs01.dbf
/u01/app/oracle/oradata/cdb1/pdbseed/system01.dbf
/u01/app/oracle/oradata/cdb1/users01.dbf
/u01/app/oracle/oradata/cdb1/pdbseed/sysaux01.dbf
/u01/app/oracle/oradata/cdb1/pdb1/system01.dbf
/u01/app/oracle/oradata/cdb1/pdb1/sysaux01.dbf
/u01/app/oracle/oradata/cdb1/pdb1/SAMPLE_SCHEMA_users01.dbf
/u01/app/oracle/oradata/cdb1/pdb1/example01.dbf

10 rows selected.

SQL>
```

The data files are all stored in the \$ORACLE_BASE/oradata/cdb1 directory. There are:

- Four data files for the root container:
 - One for the SYSTEM tablespace
 - One for the SYSAUX tablespace

- One for the UNDO tablespace
 - One for the USERS tablespace
 - Two data files for the seed container under a subdirectory pdbseed:
 - One for the SYSTEM tablespace
 - One for the SYSAUX tablespace
 - Four data files for the pdb1 container under a subdirectory pdb1:
 - One for the SYSTEM tablespace
 - One for the SYSAUX tablespace
 - One for the USERS tablespace
 - One for the EXAMPLE tablespace
- b. List all the control files.

```
SQL> select name from v$controlfile;

NAME
-----
/u01/app/oracle/oradata/cdb1/control01.ctl
/u01/app/oracle/fast_recovery_area/cdb1/control02.ctl

SQL>
```

One control file resides in the \$ORACLE_BASE/oradata/cdb1 directory and another control file resides in the \$ORACLE_BASE/fast_recovery_area/cdb1 directory. The control files are mounted at the instance start up. There is only one instance for the whole CDB including the PDBs, and therefore the control files are mounted for the single instance.

- c. List all the redo log files.

```
SQL> select member from v$logfile;

MEMBER
-----
/u01/app/oracle/oradata/cdb1/redo03.log
/u01/app/oracle/oradata/cdb1/redo02.log
/u01/app/oracle/oradata/cdb1/redo01.log

SQL>
```

There are three redo log groups whose members reside in the \$ORACLE_BASE/oradata/cdb1 directory. The LGWR process works for the redo stream of the whole CDB including the PDBs. The default configuration does not create multiplexed redo log groups.

Best Practice Tip: Create two members for each group each on a different disk drive. If you had another disk on /u02, you could multiplex the redo log groups to this disk.

- d. On your server you only have /u01. You use it to multiplex.

```
SQL> alter database add logfile member  
'/u01/app/oracle/oradata/cdb1/redo01b.log' to group 1;  
  
Database altered.  
  
SQL> alter database add logfile member  
'/u01/app/oracle/oradata/cdb1/redo02b.log' to group 2;  
  
Database altered.  
  
SQL> alter database add logfile member  
'/u01/app/oracle/oradata/cdb1/redo03b.log' to group 3;  
  
Database altered.  
  
SQL> col member format A50  
SQL> select group#, member from v$logfile order by 1;  
  
GROUP# MEMBER  
-----  
1 /u01/app/oracle/oradata/cdb1/redo01.log  
1 /u01/app/oracle/oradata/cdb1/redo01b.log  
2 /u01/app/oracle/oradata/cdb1/redo02b.log  
2 /u01/app/oracle/oradata/cdb1/redo02.log  
3 /u01/app/oracle/oradata/cdb1/redo03b.log  
3 /u01/app/oracle/oradata/cdb1/redo03.log  
  
6 rows selected.  
  
SQL>
```

13. You did not configure Enterprise Manager (EM) Database Express while the `cdb1` database was created with DBCA. Each database instance requires a separate port for EM Database Express. You are now going to configure it with the port 5501. 5500 is being used by the `orcl` instance.

- a. Verify that the `DISPATCHERS` instance parameter has at least one dispatcher configured for the XMLDB service with the TCP protocol.

```
SQL> show parameter DISPATCHERS

NAME          TYPE        VALUE
-----
dispatchers    string      (PROTOCOL=TCP) (SERVICE=cdb1XDB)
max_dispatchers integer
```

- b. Choose between a secured and an unsecured connection. If you chose, a secured connection, use the `setHTTPsPort` procedure in the `DBMS_XDB_CONFIG` package to configure the port 5501 (5500 is used for the `orcl` EM Database Express), else use the `setHTTPPort` procedure.

Note: The https protocol uses Secure Socket Layer (SSL)

```
SQL> exec DBMS_XDB_CONFIG.setHTTPsPort(5501)

PL/SQL procedure successfully completed.

SQL>
```

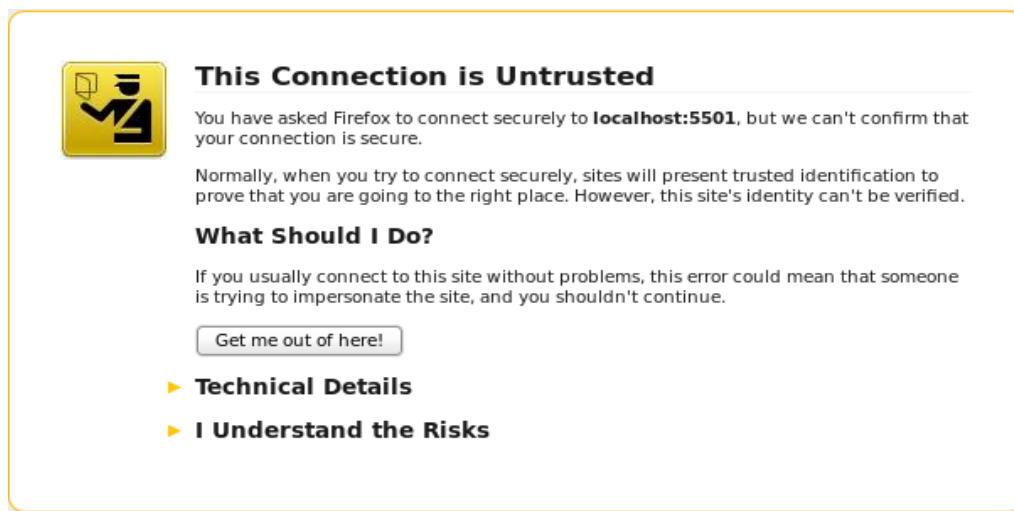
- c. Verify the port allocation.

```
SQL> SELECT dbms_xdb_config.gethttpsport FROM DUAL;

GETHTTPSPORT
-----
5501

SQL> exit
$
```

- d. In a browser, use the URL: <https://localhost:5501/em>. This Connection is Untrusted page appears. You need to add a security exception.

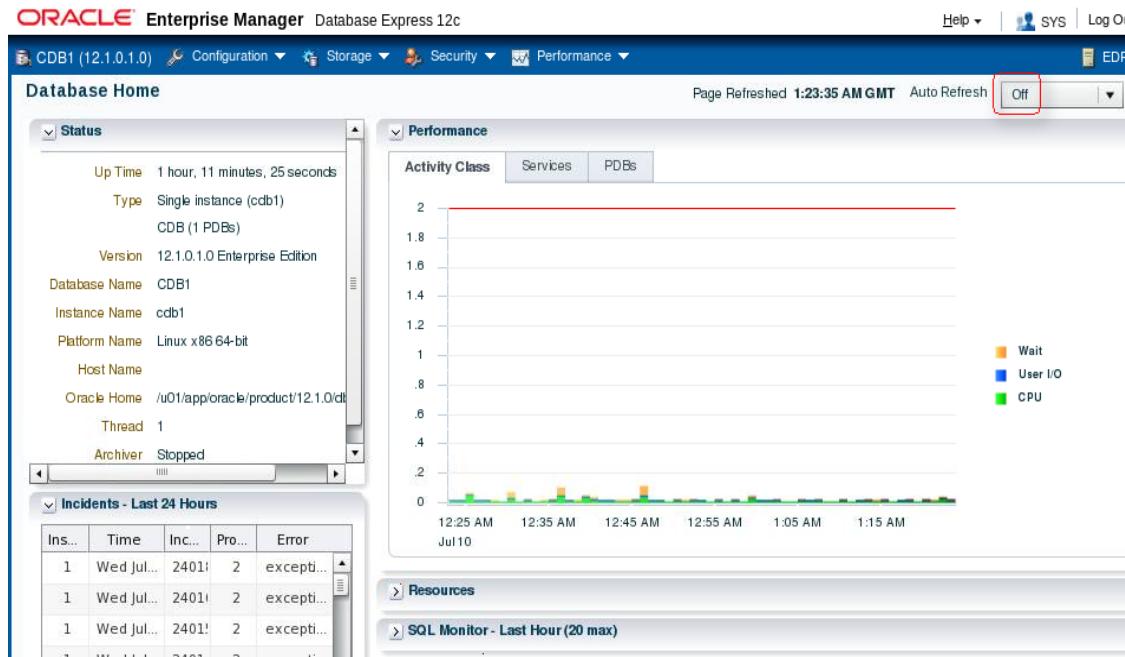


- At the end of the alert box, click **I Understand the Risks**.
 - At the bottom of the page, click **Add Exception**.
 - In the Add Security Exception pop-up window, click **Get Certificate**.
 - Confirm that “Permanently store this exception” is selected in your training environment and click **Confirm Security Exception**.
- e. The Enterprise Manager Database Express console appears. In the Login page, enter sys in the User Name field, oracle_4U in the Password field, select “as sysdba” and click **Login**.

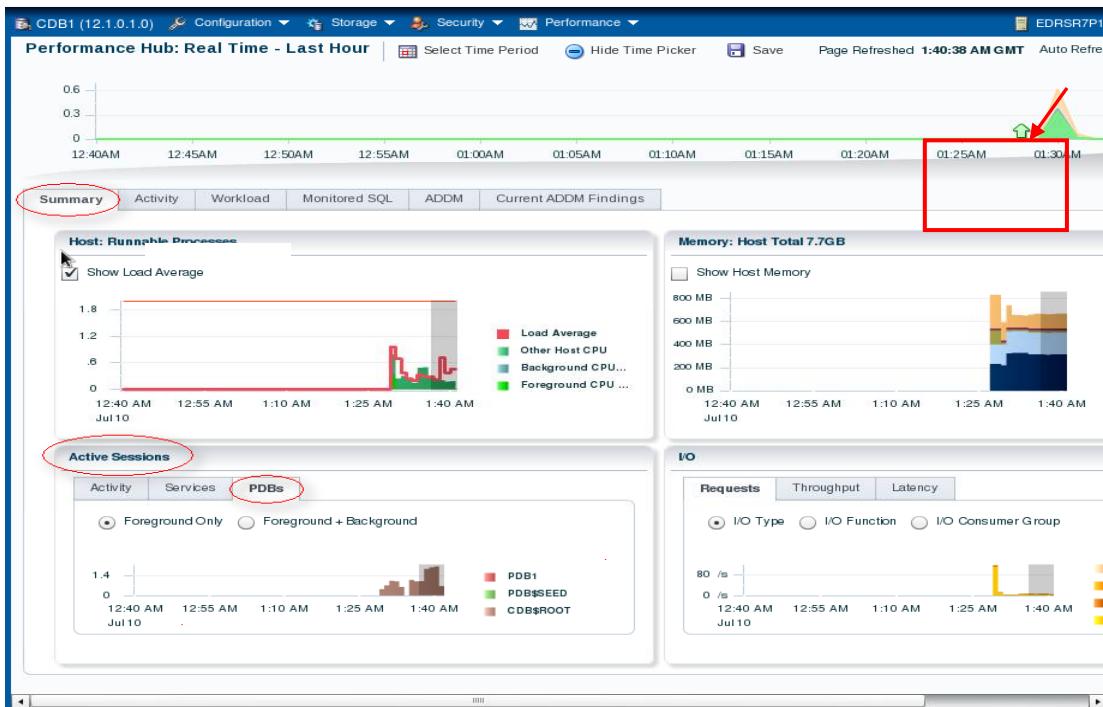
A screenshot of the Oracle Database Express (DBA) Login page. The title bar says "Login". There are two input fields: "User Name" containing "sys" and "Password" containing "*****". Below the password field is a checked checkbox labeled "as sysdba". At the bottom is a "Login" button.

- f. Now the Database Home page of the cdb1 instance appears. To avoid the current issues displaying the **“Resources”** and **“SQL Monitor- Last Hour (20 max)”**

information, collapse the two tabs and switch the Auto Refresh to Off.



- You can view the list of PDBs in the CDB1. Click the Performance tab from the top menu, then Performance Hub from the options. In the Summary tab, click the PDBs tab from the Active Sessions section.



- Exit EM Database Express. Click Log Out.
- Close the browser window.

Practices for Lesson 6: Using Oracle Restart

Chapter 6

Practices for Lesson 6

Practices Overview

In the previous practices, you upgraded the ASM instance by using OUI and you created two databases by using DBCA, a non-CDB and a CDB including one PDB.

You can manage these instances and databases by using:

- SQL*Plus to start up or shut down the ASM and database instances, create and drop non-CDBs and CDBs, create and drop PDBs, and open and close PDBs
- DBCA to create and delete non-CDBs and CDBs, and add new PDBs in CDBs
- SQL Developer to start up or shut down the instances, and create new PDBs in CDBs
- ASMCA to create and drop ASM disk groups, configure ASM disk group attributes, manage templates, upgrade the ASM instance
- ASMCMD to manage ASM disk groups, ASM files, and ASM aliases and templates.

In this practice, you will manage Oracle Restart by using:

- CRSCTL to monitor the Oracle Restart daemon
- SRVCTL to manage components of the database instances, ASM instance, and listeners

You are going to work as the `oracle` user for some operations and as the `grid` user for other operations. Keep a terminal window opened as the `oracle` user and another terminal window opened as the `grid` user.

Assumptions

The `orcl` non-CDB was successfully created in the previous practice.

The `cdb1` CDB was successfully created in the previous practice.

Practice 6-1: Starting and Stopping Oracle Restart

Overview

Oracle Grid Infrastructure includes the Oracle Restart component. In this practice, you will check if Oracle Restart is started. You will also stop and restart Oracle Restart.

Tasks

1. Check if Oracle Restart is started.
 - a. From the `grid` user terminal window, check the status of the HAS process (Oracle High Availability Services daemon), `ohasd`.

```
$ pgrep -lf ohasd
3296 /bin/sh /etc/init.d/init.ohasd run
3326 /u01/app/grid/product/12.1.0/grid/bin/ohasd.bin reboot
$
```

It is currently started.

- b. Another way to determine if the HAS process is started is to use a CRSCTL command.

```
$ crsctl check has
CRS-4638: Oracle High Availability Services is online
$
```

2. Determine the HAS version.

```
$ crsctl query has softwareversion
Oracle High Availability Services version on the local node is
[12.1.0.1.0]
$
```

3. Is Oracle Restart automatically started at reboot?

```
$ crsctl config has
CRS-4622: Oracle High Availability Services autostart is
enabled.
$
```

4. Stop Oracle Restart.

```
$ crsctl stop has
CRS-2791: Starting shutdown of Oracle High Availability
Services-managed resources on '<localhost>'

CRS-2673: Attempting to stop 'ora.dbupgrd.db' on
'<your_hostname>'

CRS-2673: Attempting to stop 'ora.LISTENER.lsnr' on
'<your_hostname>'

CRS-2673: Attempting to stop 'ora.orcl.db' on '<your_hostname>'

CRS-2673: Attempting to stop 'ora.cdb1.db' on '<your_hostname>'

CRS-2677: Stop of 'ora.LISTENER.lsnr' on '<your_hostname>' succeeded

CRS-2677: Stop of 'ora.orcl.db' on '<your_hostname>' succeeded

CRS-2677: Stop of 'ora.cdb1.db' on '<your_hostname>' succeeded
```

```

CRS-2677: Stop of 'ora.dbupgrd.db' on '<your_hostname>' succeeded
CRS-2673: Attempting to stop 'ora.DATA.dg' on '<your_hostname>'
CRS-2673: Attempting to stop 'ora.FRA.dg' on '<your_hostname>'
CRS-2677: Stop of 'ora.evmd' on '<your_hostname>' succeeded
CRS-2677: Stop of 'ora.FRA.dg' on '<your_hostname>' succeeded
CRS-2677: Stop of 'ora.DATA.dg' on '<your_hostname>' succeeded
CRS-2673: Attempting to stop 'ora.asm' on '<your_hostname>'
CRS-2677: Stop of 'ora.asm' on '<your_hostname>' succeeded
CRS-2673: Attempting to stop 'ora.cssd' on '<your_hostname>'
CRS-2677: Stop of 'ora.cssd' on '<your_hostname>' succeeded
CRS-2793: Shutdown of Oracle High Availability Services-managed resources on '<your_hostname>' has completed
CRS-4133: Oracle High Availability Services has been stopped.
$
```

You can view all resources stopped one by one:

- The dbupgrd, orcl and cdb1 databases are registered as local resources in the OLR.
- The listener is also registered in the OLR.
- The DATA and FRA disk groups and the ASM instance are also registered in the OLR.

They are stopped automatically by Oracle Restart in the right sequence, the client database instances first then the ASM instance. The disk groups are dismounted before the ASM instance is shut down.

These resources have been registered automatically by DBCA for the first three database instances and by ASMCA for the disk groups and the ASM instance.

5. Check that all database instances and the ASM instance are shut down.

```

$ ps -ef | grep smon
grid      17257  8535  0 18:09 pts/3    00:00:00 grep smon
$
```

6. Restart Oracle Restart. It will take some time to start all resources. However if you want to see the resources started one after the other, the commands in step 7 should be run as soon as the current crsctl command has completed.

```

$ crsctl start has

CRS-4123: Oracle High Availability Services has been started.
$
```

7. Check that all the database and ASM instances are automatically started up. Oracle Restart knows that there are some dependencies between the resources. For example, the ASM instance must be started before the disk groups can be mounted and the client database instances such as dbupgrd can start up to connect to the ASM instance. If you check with the following Unix command, you will have to repeat the command to see the resources started up one after the other.

```
$ ps -ef | grep smon
```

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

```
oracle 17802 1 0 18:14 ? 00:00:00 ora_smon_cdb1
oracle 17812 1 0 18:14 ? 00:00:00 ora_smon_orcl
grid 18021 8535 0 18:15 pts/3 00:00:00 grep smon
$ ps -ef | grep smon
oracle 17802 1 0 18:14 ? 00:00:00 ora_smon_cdb1
oracle 17812 1 0 18:14 ? 00:00:00 ora_smon_orcl
grid 18072 1 0 18:15 ? 00:00:00 asm_smon_+ASM
grid 18150 8535 0 18:15 pts/3 00:00:00 grep smon
$ ps -ef | grep smon
oracle 17802 1 0 18:14 ? 00:00:00 ora_smon_cdb1
oracle 17812 1 0 18:14 ? 00:00:00 ora_smon_orcl
grid 18072 1 0 18:15 ? 00:00:00 asm_smon_+ASM
oracle 18199 1 0 18:15 ? 00:00:00 ora_smon_dbupgrd
grid 18450 8535 0 18:17 pts/3 00:00:00 grep smon
$
```

Practice 6-2: Starting and Stopping Components

Overview

In this practice, you use Oracle Restart to view the status, start and stop the `orcl` database, the ASM instance, the disk groups, and the listener.

Tasks

1. Invoke the SRVCTL utility from the Oracle Database home when working with the database instance. You will check the status of the `orcl` database instance.
 - a. From the `oracle` user terminal window, set the `ORACLE_BASE` and `ORACLE_HOME` environment variables for your `orcl` database instance.

```
$ . oraenv
ORACLE_SID = [cdb1] ? orcl
The Oracle base has been set to /u01/app/oracle
$
```

- b. Check the status of the `orcl` instance by using the SRVCTL utility.

```
$ srvctl status database -d orcl
Database is running.
$
```

- c. Kill the LGWR process. After you kill the LGWR process, the instance is shut down. After a little while, the instance is automatically restarted by Oracle Restart because the `orcl` resource is registered in the configuration file of Oracle Restart. The lgwr process may take a few seconds to restart. Repeat the `pgrep` command until it has started.

```
$ pgrep -fl ora_lgwr_orcl
7955 ora_lgwr_orcl
$ pkill -SIGKILL -f ora_lgwr_orcl
$ pgrep -fl ora_lgwr_orcl
...
$ pgrep -fl ora_lgwr_orcl
8792 ora_lgwr_orcl
$
```

2. From the `grid` user terminal window, invoke the SRVCTL utility from the Oracle Grid Infrastructure home when working with the ASM instance.

- a. Check the status of the ASM instance by using the SRVCTL utility.

```
$ srvctl status asm
ASM is running on <your_hostname>
$
```

- b. Kill the LGWR process. After you kill the LGWR process, the instance is shut down. After a little while, the instance is automatically restarted by Oracle Restart. notice that the processes have a new process id when they are restarted.

```
$ pgrep -lf asm_lgwr
2267 asm_lgwr_+ASM
```

```
$ pkill -SIGKILL -f asm_lgwr
$ pgrep -lf asm_lgwr
$ pgrep -lf asm_lgwr
4961 asm_lgwr_+ASM
$
```

3. Invoke the SRVCTL utility from the Oracle Grid Infrastructure home when working with the disk groups. You will check the status of the DATA disk group.

```
$ srvctl status diskgroup -g DATA
Disk Group DATA is running on <your_hostname>
$
```

4. Invoke the SRVCTL utility from the Oracle Grid Infrastructure home when working with the listener.
 - a. You will check the status of the listener.

```
$ srvctl status listener
Listener LISTENER is enabled
Listener LISTENER is running on node(s) : <your_hostname>
$
```

- b. Kill the listener process. When you kill the listener process, the listener is shut down. Very quickly the listener is automatically restarted by Oracle Restart. Notice the process id will be different after the listener is restarted.

```
$ pgrep -lf tnslsnr
1921 /u01/app/grid/product/12.1.0/grid/bin/tnslsnr LISTENER -
no_crs_notify -inherit
$ pkill -SIGKILL -f tnslsnr
$ pgrep -lf tnslsnr
...
$ pgrep -lf tnslsnr
8014 /u01/app/grid/product/12.1.0/grid/bin/tnslsnr LISTENER -
no_crs_notify -inherit
$
```

Practice 6-3: Managing Configuration of Components

In this practice, you use Oracle Restart to view and modify the configuration of the `orcl` database and the listener.

1. The component you will manage in this step is the `orcl` database instance. You have to use `SRVCTL` in the `ORACLE_HOME` of the database instance.

- a. From the `oracle` user terminal window, set the ORACLE environment variables.

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

- b. List the database instances configured with Oracle Restart.

```
$ srvctl config  
cdb1  
dbupgrd  
orcl  
$
```

- c. Display the configuration of the `orcl` database instance.

```
$ srvctl config database -d orcl  
Database unique name: orcl  
Database name: orcl  
Oracle home: /u01/app/oracle/product/12.1.0/dbhome_1  
Oracle user: oracle  
Spfile:  
Password file:  
Domain:  
Start options: open  
Stop options: immediate  
Database role: PRIMARY  
Management policy: AUTOMATIC  
Database instance: orcl  
Disk Groups:  
Services:  
$
```

- d. Remove your `orcl` database instance from the Oracle Restart configuration.

```
$ srvctl remove database -d orcl  
PRKO-3141 : Database orcl could not be removed because it was  
running  
$
```

- e. Stop the `orcl` database instance.

```
$ srvctl stop database -d orcl  
$
```

- f. Check that the database instance is shut down.

```
$ pgrep -lf orcl  
$
```

- g. Remove your `orcl` database instance from the Oracle Restart configuration.

```
$ srvctl remove database -d orcl  
Remove the database orcl? (y/[n]) y  
$
```

- h. Check that the `orcl` database is no longer a resource known to Oracle Restart.

```
$ srvctl config database -d orcl  
PRCD-1120 : The resource for database orcl could not be found.  
PRCR-1001 : Resource ora.orcl.db does not exist  
$
```

- i. Check that the database instance is no longer automatically started.

```
$ pgrep -lf orcl  
$ pgrep -lf orcl  
...  
$
```

- j. You can perform manual startup and shutdown operations by using SQL*Plus.

```
$ sqlplus / as sysdba  
  
SQL*Plus: Release 12.1.0.1.0 Production on Thu Feb 14 02:01:27  
2013  
  
Copyright (c) 1982, 2013, Oracle. All rights reserved.  
  
Connected to an idle instance.  
  
SQL> startup  
ORACLE instance started.  
  
Total System Global Area 876859392 bytes  
Fixed Size 2294360 bytes  
Variable Size 620760488 bytes  
Database Buffers 247463936 bytes  
Redo Buffers 6340608 bytes  
Database mounted.  
Database opened.  
SQL> exit  
$
```

- k. Once again, kill the LGWR process of your `orcl` database instance. What do you observe?

The database instance is shut down, but is no longer automatically restarted.

```
$ pgrep -fl ora_lgwr_orcl  
7955 ora_lgwr_orcl  
$ pkill -SIGKILL -f ora_lgwr_orcl  
$ pgrep -fl ora_lgwr_orcl  
$ pgrep -fl ora_lgwr_orcl  
$
```

- l. Add the `orcl` database instance and its dependencies back to the Oracle Restart configuration.

```
$ srvctl add database -d orcl -o  
/u01/app/oracle/product/12.1.0/dbhome_1  
$
```

- m. The database resource is added back to the Oracle Restart configuration. You can verify by using SRVCTL.

```
$ srvctl config database -d orcl  
Database unique name: orcl  
Database name:  
Oracle home: /u01/app/oracle/product/12.1.0/dbhome_1  
Oracle user: oracle  
Spfile:  
Password file:  
Domain:  
Start options: open  
Stop options: immediate  
Database role: PRIMARY  
Management policy: AUTOMATIC  
Database instance: orcl  
Disk Groups:  
Services:  
$
```

- n. Restart the `orcl` database instance.

```
$ srvctl status database -d orcl  
Database is not running.  
$ srvctl start database -d orcl  
$ srvctl status database -d orcl  
Database is running.  
$
```

- o. Modify the configuration of the `orcl` database instance to add the following information: the SPFILE is

```
/u01/app/oracle/product/12.1.0/dbhome_1/dbs/spfileorcl.ora.
```

```
$ srvctl modify database -d orcl -p
/u01/app/oracle/product/12.1.0/dbhome_1/dbs/spfileorcl.ora
$ srvctl config database -d orcl
Database unique name: orcl
Database name:
Oracle home: /u01/app/oracle/product/12.1.0/dbhome_1
Oracle user: oracle
Spfile:
/u01/app/oracle/product/12.1.0/dbhome_1/dbs/spfileorcl.ora
Password file:
Domain:
Start options: open
Stop options: immediate
Database role: PRIMARY
Management policy: AUTOMATIC
Database instance: orcl
Disk Groups:
Services:
$
```

- p. Make sure Oracle Restart is functioning properly with your `orcl` database instance. Once again kill the LGWR process and verify that it is automatically restarted.

```
$ pgrep -f1 ora_lgwr_orcl
10310 ora_lgwr_orcl
$ pkill -SIGKILL -f ora_lgwr_orcl
$ pgrep -f1 ora_lgwr_orcl
$ pgrep -f1 ora_lgwr_orcl
10787 ora_lgwr_orcl
$
```

- q. Disable Oracle Restart on the `orcl` database instance. This does not remove the resource configuration from the registry. It disables it until you decide to reenable it.

```
$ srvctl disable database -d orcl
$ srvctl status database -d orcl -f
Database is disabled
Database is running.
$
```

- r. Verify that Oracle Restart does not restart the `orcl` database instance. Again kill the LGWR process and then verify that the instance is not restarted automatically.

```
$ pgrep -f1 ora_lgwr_orcl
```

```
10787 ora_lgwr_orcl
```

```
$ pkill -SIGKILL -f ora_lgwr_orcl
$ pgrep -fl ora_lgwr_orcl
$ pgrep -fl ora_lgwr_orcl
$
```

- s. Attempt to start the orcl resource.

```
$ srvctl start database -d orcl
PRCR-1079 : Failed to start resource ora.orcl.db
CRS-2501: Resource 'ora.orcl.db' is disabled
$
```

- t. Enable Oracle Restart again on your orcl database instance.

Note: the first status command may show the database is running depending on the time between the enable command and the status command.

```
$ srvctl enable database -d orcl
$ srvctl status database -d orcl -f
Database is not running.
$ pgrep -fl ora_lgwr_orcl
11793 ora_lgwr_orcl
$ srvctl status database -d orcl -f
Database is running.
$
```

2. The component you will manage in this step is the listener. You have to use SRVCTL in the ORACLE_HOME of the listener.

- a. From the grid user terminal window, display the configuration of the listener.

```
$ srvctl config listener
Name: LISTENER
Home: /u01/app/grid/product/12.1.0/grid
End points: TCP:1521
$
```

- b. You can specify a non-default location for the listener's networking files. In our case, the location is the default, but the procedure is the same. This command sets an environment variable to used whenever the listener is managed.

```
$ srvctl setenv listener -l LISTENER -t
TNS_ADMIN=/u01/app/grid/product/12.1.0/grid
$
```

3. The component you will manage in this step is the cdb1 database. You have to use SRVCTL in the ORACLE_HOME of the database instance.

- a. *If you attempt to execute the SRVCTL command in the ORACLE_HOME of the Grid home, you get the following message.*

```
$ whoami
grid
$ srvctl add service -db cdb1 -service pdb1bis -pdb pdb1
PRCD-1026 : Failed to create service pdb1bis for database cdb1
```

```
PRKH-1014 : Current user "grid" is not the oracle owner user  
"oracle" of oracle home  
"/u01/app/oracle/product/12.1.0/dbhome_1"  
$
```

- b. From the oracle user terminal window, set the ORACLE_SID to cdb1.

```
$ . oraenv  
ORACLE_SID = [orcl] ? cdb1  
The Oracle base remains unchanged with value /u01/app/oracle  
$
```

- c. Display the configuration of the cdb1 database.

```
$ srvctl config database -d cdb1  
Database unique name: cdb1  
Database name: cdb1  
Oracle home: /u01/app/oracle/product/12.1.0/dbhome_1  
Oracle user: oracle  
Spfile:  
Password file:  
Domain:  
Start options: open  
Stop options: immediate  
Database role: PRIMARY  
Management policy: AUTOMATIC  
Database instance: cdb1  
Disk Groups:  
Services:  
$
```

- d. Add another pdb1bis service for the pdb1 PDB in the cdb1 CDB.

```
$ srvctl add service -db cdb1 -service pdb1bis -pdb pdb1  
$
```

- e. Display the configuration of the cdb1 database.

```
$ srvctl config database -d cdb1  
Database unique name: cdb1  
Database name: cdb1  
Oracle home: /u01/app/oracle/product/12.1.0/dbhome_1  
Oracle user: oracle  
Spfile:  
Password file:  
Domain:  
Start options: open  
Stop options: immediate  
Database role: PRIMARY
```

```
Management policy: AUTOMATIC
Database instance: cdb1
Disk Groups:
Services: pdb1bis
$
```

- f. Start the service.

```
$ srvctl start service -db cdb1 -service pdb1bis
$
```

- g. Check the status of the new service.

```
$ srvctl status service -db cdb1
Service pdb1bis is running
$
```

- h. Verify that you can connect to the PDB using the new service.

```
$ sqlplus sys/oracle_4U@localhost:1521/pdb1bis as sysdba

SQL> select name, con_id from v$services;

NAME                                     CON_ID
-----
pdb1bis                                  3
pdb1                                     3

SQL> exit
$
```

- i. Use LSNRCTL to show the new service.

```
$ lsnrctl services

LSNRCTL for Linux: Version 12.1.0.1.0 - Production on 27-FEB-
2013 23:31:51

Copyright (c) 1991, 2013, Oracle. All rights reserved.

Connecting to (ADDRESS=(PROTOCOL=tcp) (HOST=) (PORT=1521))
Services Summary...
Service "+ASM" has 1 instance(s).
  Instance "+ASM", status READY, has 1 handler(s) for this
  service...
    Handler(s):
      "DEDICATED" established:60 refused:0 state:ready
        LOCAL SERVER
Service "cdb1" has 1 instance(s).
```

```
Instance "cdb1", status READY, has 1 handler(s) for this
service...
Handler(s):
  "DEDICATED" established:1 refused:0 state:ready
    LOCAL SERVER
...
Service "pdb1" has 1 instance(s).
Instance "cdb1", status READY, has 1 handler(s) for this
service...
Handler(s):
  "DEDICATED" established:1 refused:0 state:ready
    LOCAL SERVER
Service "pdb1bis" has 1 instance(s).
Instance "cdb1", status READY, has 1 handler(s) for this
service...
Handler(s):
  "DEDICATED" established:1 refused:0 state:ready
    LOCAL SERVER
The command completed successfully
$
```


Practices for Lesson 7: Introduction to Upgrading to Oracle Database 12c

Chapter 7

Practices for Lesson 7

Practices Overview

In these practices, you will consider several upgrade scenarios.

A general checklist for any upgrade scenario is:

1. For all of these cases the first step is to become familiar with the Oracle Database Upgrade Guide 12c Release 1 (12.1).
2. Survey your database to determine if the current configuration is compatible with Oracle Database 12c. Document any issues that you find.
3. What-if: You have discovered that your database is not a supported release.
 - a. For any patch release before 10.2.0.5, 11.1.0.7, 11.2.0.2, you cannot upgrade directly to Oracle Database 12c. You must upgrade to a supported patch level (10.2.0.5, 11.1.0.7, 11.2.0.2 or later) and then upgrade to 12c.
 - b. For database versions older than 10.2.x, you must upgrade to a supported version. For example, 9.2.0.8 could be upgraded to 11.2.0.2 or 11.2.0.3, and then to 12.1.
 - c. For other than supported patch releases, and older versions of the Oracle Database software from 10.1.0.1 and up, the database can be migrated using Oracle Data Pump export/import. See chapter 2 of the Oracle Database Upgrade Guide 12c Release 1 (12.1) for more information about Data Pump use.
 - d. For versions of the database earlier than 10.1.0.1, you can use the original Export and Import utilities.
4. Determine the upgrade/migration method.
 - a. Direct upgrade using DBUA
 - Requires supported version level
 - Requires that source and target ORACLE_HOME directories have compatible permission. Both must be owned by the same OS user, or by the same OS group. Log files must be directed to a directory accessible by both source and target OS users.
 - Requires that the database be in restricted mode during the upgrade.
 - Time required for the upgrade is dependent on installed components, and number of objects, not the volume of data.
 - Database files remain in place by default.
 - Downgrade possible after an upgrade, if COMPATIBLE parameter has NOT been set to a higher level.
 - **Note:** If the upgrade fails, you must recover the database to the pre-upgrade state. A tested backup and recovery plan to restore database files is essential.
 - b. Direct Upgrade using manual method. (same requirements as using DBUA)
 - more flexibility
 - More opportunities for error
 - Requires additional post-upgrade steps
 - c. Data Pump export/import
 - Allows you to move database to different OS, hardware, and version
 - Allows network transfer of data (no dump files created or moved)
 - Allows data transfer from any version from 10.1.0.1 and higher to 12.1.0.1

- Allows transfer of selected schemas, and objects
 - Time to transfer is dependent on amount of data
 - Source database must be in restricted mode while exporting the data (no data updates allowed)
- d. Original export/import (same restrictions as Data Pump export with exceptions)
- Allows export from older versions of Oracle Database software older than 10.1.0.1 to Oracle Database 12c
 - Does not allow network transfer of data, except through custom scripting (e.g. pipes)

For all upgrade methods to 12c:

- Install Oracle Database 12c in a new ORACLE_HOME directory.
- The current ORACLE_HOME directory is not removed. DO NOT attempt to open the upgraded database with the older software.

Best Practice: Practice whatever upgrade method you decide to use. Create a test database instance as similar as possible to the source instance. Use this instance to test applications and performance after the upgrade, before upgrading the production instance.

5. Perform pre-upgrade steps, and correct issues.
6. Perform upgrade using method of choice
 - a. Using DBUA (recommended)
 - b. Using manual upgrade
 - c. Using Data Pump export/import
 - d. Using original export/import
7. Perform post-upgrade steps
 - a. Consider resetting password to be case sensitive. Note the –ignorecase parameter for orapwd to create the oracle password file is deprecated.
 - b. Add new features. Several new features require setting the COMPATIBLE parameter to 12.0.0.0.0. **Note:** If this parameter is changed, a downgrade operation is not possible. The source database must be restored from backup to return to the prior version.
 - c. Migrate to use Unified Auditing.
8. Test the upgraded database
 - a. Test for correct application behavior.
 - b. Test performance.

Practice 7-1: Upgrading an Oracle Database 11g Release 2 Database on File System to Oracle Database 12c

Overview

In this practice, you will outline the steps to upgrade an existing Oracle Database 11g Release 2 database on file system to Oracle Database 12c.

Tasks

Write an outline of the major steps in upgrading an Oracle Database 11g Release 2 database to Oracle Database 12c.

1. _____
2. _____
3. _____
4. _____
5. _____
6. _____
7. _____
8. _____
9. _____
10. _____

Solution 7-1: Upgrading an Oracle Database 11g Release 2 Database on File System to Oracle Database 12c

Suggested Answers

Your answers may vary

- a. Determine upgrade method
- b. Use Database Upgrade Assistant (DBUA)
- c. Install Oracle Database 12c
- d. Perform preupgrade tasks
- e. Upgrade using chosen method
- f. Perform post-upgrade tasks
- g. Test upgraded instance

Practice 7-2: Upgrading an Oracle Database 11g Release 2 Database on ASM to Oracle Database 12c

Overview

In this practice you develop an outline for upgrading an Oracle Database 11g Release 2 database on standalone ASM to Oracle Database 12c.

Tasks

1. _____
2. _____
3. _____
4. _____
5. _____
6. _____
7. _____
8. _____
9. _____
10. _____
11. _____
12. _____
13. _____
14. _____
15. _____

Solution 7-2: Upgrading and Oracle Database 11g Release 2 Database on ASM to Oracle Database 12c

Suggested Answers

- a. Stop the database instances using the 11g ASM instance.
- b. Stop any listener running out of the current ORACLE_HOME for grid (if it exists).
- c. Install Oracle Grid Infrastructure 12c for standalone server in a new ORACLE_HOME.
 - Upgrade ASM instance
 - Create ASM SPFILE
 - Adjust ASM parameters
- d. Restart listener running out of Grid Infrastructure 12c new ORACLE_HOME (if it exists).
- e. Restart Database instances.
- f. Install Oracle Database software in new ORACLE_HOME for Oracle 12c Database.
- g. Perform preupgrade tasks.
- h. Upgrade using DBUA.
- i. Perform post-upgrade tasks.
- j. Test upgraded instance.
- k. Adjust ASM diskgroup parameters, set diskgroup compatible level if necessary.

Practice 7-3: Upgrading an Oracle Database 11g Release 2 Database to Oracle Database 12c on a New OS or Platform

Overview

In this practice you consider the differences in this scenario from the previous scenarios.

Tasks

1. _____
2. _____
3. _____
4. _____
5. _____
6. _____
7. _____
8. _____
9. _____
10. _____
11. _____
12. _____
13. _____
14. _____
15. _____

Solution 7-3: Upgrading an Oracle Database 11g Release 2 Database to Oracle Database 12c on a New OS or Platform

Suggested Answers

Determine compatibility of datafiles and OS versions.

- Is endianness the same (check)
- Yes, datafiles can be copied with OS utilities
- No datafiles must be converted to target OS endianness. **Note:** If the source platform and the target platform are of different endianness, then you cannot use the RMAN CONVERT DATABASE command. This process requires both the source and target platform to be the same endian value. Your available options are Data Pump replication, Data Pump export/import, or Transportable Tablespace, with an RMAN CONVERT TABLESPACE. If the platforms are of the same endianness, then no conversion is necessary and data can be transported as if on the same platform.
- Does either the source or target OS support both Oracle database versions
 - Yes, consider migrating to new platform and upgrading to new version as separate operations. Upgrade as in practice 7-1 or 7-3 then migrate to new platform or migrate then upgrade.
 - No, Use Data Pump export/import **Note:** this situation may also have a Golden Gate solution. See Practice 7-7 for more information.
- Is there a difference in bit-width of the OS, that is one 32 bit and one 64 bit?
 - Follow the notes given in MOS document "Can you / How to Upgrade RDBMS and Convert From 32-bit to 64-bit Binaries Directly on Linux or Windows based Intel Platforms Using the Database Upgrade Assistant (DBUA) [ID 757245.1]"
 - Perform the migration to 64 bit separate from the upgrade, **Note:** This may require the 32-bit binaries to be installed for a time on the 64-bit OS to perform the migration in place.

Practice 7-4: Upgrading an Oracle Database 11g Release 2 Database to Oracle Database 12c with Minimal Downtime

Overview

In this practice you consider options to reduce downtime while upgrading from Oracle Database 11g Release 2 to Oracle Database 12c.

Tasks

1. _____
2. _____
3. _____
4. _____
5. _____
6. _____
7. _____
8. _____
9. _____
10. _____
11. _____
12. _____
13. _____
14. _____
15. _____

Solution 7-4: Upgrading an Oracle Database 11g Release 2 Database to Oracle Database 12c with Minimal Downtime

Suggested Answers

Practice upgrade as many times as needed to automate pre-upgrade and post-upgrade tasks.

Reduce the time for DBUA to run. Remove any unused components from the database before upgrade. For example Oracle Text, Oracle Spatial.

Script all pre-upgrade tasks.

Perform upgrade with DBUA.

Script all post-upgrade tasks.

Alternates to using DBUA

- Create a target database and use Oracle GoldenGate to synchronize the source with the target. When the synchronization is complete and applications can be run on the target, stop the applications on the source, restart the applications on the target. Shutdown the source database. The detailed procedures for this operation are beyond the scope of this course. See Oracle Database Global Data Services Concepts and Administration Guide 12c Release 1 (12.1) for details.

Practices for Lesson 8: Preparing a Database for Upgrade

Chapter 8

Practices for Lesson 8

Practices Overview

In the previous practices, you upgraded the 11g ASM instance to 12c using OUI and you created two databases in 12c.

In these practices, you will prepare the dbupgrd database instance currently running in 11g to upgrade to 12c. The upgrade process will use Database Upgrade Assistant (DBUA). The upgrade process will be covered in the next practice. In these practices, you will:

- Execute the preupgrd.sql script
- Review the information displayed by the pre-upgrade information tool
- Resolve issues if any
- Perform a full database backup before the upgrade process starts

In your upgrade plan, you do not plan to relocate the data files, control files, redo log files and archive log files in another location.

You are going to work as the oracle user to perform operations in the 11g environment and as the oracle user for other operations in the 12c environment. Therefore, keep a terminal window opened as the oracle user in the 11g environment and another terminal window opened as the oracle user in the 12c environment.

Practice 8-1: Executing Preliminary Steps and the Pre-Upgrade Script

Overview

In this practice, you will now prepare your `dbupgrd` database instance for an upgrade to 12c. You have already installed the Oracle 12c Database software. You need to check that the `dbupgrd` database instance is compatible with an upgrade from 11.2.0.3.0 to 12.1.0.1.0 before upgrading.

Tasks

1. List the options that require a specific treatment before upgrade.
 - a. From the `oracle` user terminal window (we will name this window the `oracle` user 11g terminal window), check if Oracle Label Security or/and Oracle Database Vault are installed in the 11g `dbupgrd` database instance.

```
$ . oraenv
ORACLE_SID = [cdb1] ? dbupgrd
The Oracle base remains unchanged with value /u01/app/oracle
$ env | grep ORA
ORACLE_SID=dbupgrd
ORACLE_BASE=/u01/app/oracle
ORACLE_HOME=/u01/app/oracle/product/11.2.0/dbhome_2
$ sqlplus / as sysdba

SQL> col parameter format A30
SQL> col value format A10
SQL> select * from v$option
      where parameter in
          ('Oracle Label Security', 'Oracle Database Vault');
2      3
PARAMETER                               VALUE
-----
Oracle Label Security                   FALSE
Oracle Database Vault                  FALSE

SQL>
```

If one of these two options were enabled, you would have to execute the `$ORACLE_HOME/rdbms/admin/olspreupgrade.sql` script copied from the 12c database `ORACLE_HOME` environment to the 11g database `ORACLE_HOME` environment. The script moves the `AUD$` table from the `SYSTEM` schema to the `SYS` schema. You would have to disable Oracle Database Vault if the option is enabled.

- b. View the `olspreupgrade.sql` script. This script is available in the 12c `ORACLE_HOME`. Open another `oracle` user terminal window in which you set the environment variables to the 12c Oracle database `ORACLE_HOME`. We will name this window the `oracle` user 12c terminal window. Set the `ORACLE_SID` environment variable to `orcl`. (The `orcl` database instance runs in the 12c database environment)

```
$ . oraenv
ORACLE_SID = [oracle] ? orcl
The Oracle base remains unchanged with value /u01/app/oracle
$ env | grep ORA
ORACLE_SID=orcl
ORACLE_BASE=/u01/app/oracle
ORACLE_HOME=/u01/app/oracle/product/12.1.0/dbhome_1
$
```

- c. Display the content of the \$ORACLE_HOME/rdbms/admin/olspreupgrade.sql script.

```
$ less $ORACLE_HOME/rdbms/admin/olspreupgrade.sql
Rem
Rem $Header: rdbms/admin/olspreupgrade.sql /st_rdbms_12.1.0.1/1
Rem
Rem olspreupgrade.sql
Rem
Rem Copyright (c) 2012, 2013 Oracle and/or its affiliates. All
rights reserved
Rem
Rem      NAME olspreupgrade.sql - OLS Pre processing script prior
to upgrade for 12.1
Rem
Rem      DESCRIPTION
Rem
This is a mandatory OLS preprocess upgrade script that
needs to be run by Label Security and Database Vault customers
when upgrading to 12.1.
Rem
Rem      This script is NOT NEEDED if you DO NOT have Database
Vault or Label Security.
...
```

It is clearly stated that this is a mandatory OLS preprocess upgrade script that needs to be run by Label Security and Database Vault customers when upgrading to 12.1.0.0, but this script is NOT NEEDED if you DO NOT have Database Vault or Label Security enabled.

- d. It is clearly described in the script to know how to proceed:

```
Rem      STEPS TO RUN THIS SCRIPT
Rem -----
Rem      To run the Oracle Label Security preprocess upgrade
script, copy the $ORACLE_HOME/rdbms/admin/olspreupgrade.sql
script to the old ORACLE_HOME. Run the script as SYSDBA in the
old ORACLE_HOME.

Rem      However, if you have Database Vault, the following
steps need to be done after copying the olspreupgrade.sql to the
old ORACLE_HOME:
```

```

Rem      To run the OLS preprocess script on a release 11.1.0.7
database before upgrading:
Rem
Rem      1. Start SQL*Plus and connect to the database to be
upgraded as DVOWNER.
Rem      2. Execute the following statement:
Rem          SQL>EXEC dbms_macadm.add_auth_to_realm('Database
Vault','SYS',NULL,0);
Rem      3. Run the OLS preprocess script, at
Rem          ORACLE_HOME/rdbms/admin/olspreupgrade.sql
Rem      4. After the olspreupgrade.sql has been successfully
run, start
Rem          SQL*Plus and connect to the database as DVOWNER.
Rem      5. Execute the following statement:
Rem          SQL> EXEC
dbms_macadm.delete_auth_from_realm('Database Vault','SYS');
Rem
...

```

- Quit the less page browser.

Question: How would you disable Oracle Database Vault?

Answer:

- Log in as the Oracle Database Owner (`DV_OWNER`) account.
`sqlplus dvo/password`
- Execute the following procedure:
`exec DVSYS.DBMS_MACADM.DISABLE_DV`
- Restart the instance
`connect / as sysdba`
`shutdown immediate`
`startup`

- The upgrade will fail if the tablespace that are not schema-based such as SYSAUX, SYSTEM, XDB, HTMLDB, CTXSYS are set to READ ONLY or OFFLINE. Check that the tablespaces set to READ ONLY or OFFLINE mode are not any of these tablespaces. In the rare case where queue tables reside in a tablespace that has been set to READ ONLY for the upgrade, then that tablespace should be set back to READ WRITE.

In the terminal session that is set to the 11g `ORACLE_HOME`, run the following statement:

```
SQL> select tablespace_name, status from dba_tablespaces;
```

TABLESPACE_NAME	STATUS
SYSTEM	ONLINE
SYSAUX	ONLINE
UNDOTBS1	ONLINE
TEMP	ONLINE

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

```

USERS          ONLINE
EXAMPLE        ONLINE

6 rows selected.

SQL>

```

3. Remove the AUDSYS schema and the AUDIT_ADMIN and AUDIT_VIEWER roles. At this stage, there should be no AUDSYS schema. These user and roles will be created as predefined user and roles when the database will have been upgraded to 12c and migrated to Unified Auditing. Unified auditing is the new auditing capability in Oracle database 12c. It is covered in another practice after the database upgrade completion.

```

SQL> DROP USER AUDSYS CASCADE;
DROP USER AUDSYS CASCADE
*
ERROR at line 1:
ORA-01918: user 'AUDSYS' does not exist

SQL> DROP ROLE AUDIT_ADMIN;
DROP ROLE AUDIT_ADMIN
*
ERROR at line 1:
ORA-01919: role 'AUDIT_ADMIN' does not exist

SQL> DROP ROLE AUDIT_VIEWER;
DROP ROLE AUDIT_VIEWER
*
ERROR at line 1:
ORA-01919: role 'AUDIT_VIEWER' does not exist

SQL> exit
$ 

```

4. Run the Pre-Upgrade Information Tool using the \$ORACLE_HOME/rdbms/admin/preupgrd.sql script.
 - a. Copy the \$ORACLE_HOME/rdbms/admin/preupgrd.sql and \$ORACLE_HOME/rdbms/admin/utlupppkg.sql scripts from the 12c database ORACLE_HOME environment to the 11g database ORACLE_HOME environment. Run the following statement:

```

$ cp
/u01/app/oracle/product/12.1.0/dbhome_1/rdbms/admin/preupgrd.sql
/u01/app/oracle/product/11.2.0/dbhome_2/rdbms/admin/preupgrd.sql

```

```
$ cp  
/u01/app/oracle/product/12.1.0/dbhome_1/rdbms/admin/utlupppkg.sql  
/u01/app/oracle/product/11.2.0/dbhome_2/rdbms/admin/utlupppkg.sql  
$
```

- b. The execution of the preupgrd.sql script will generate a log file and two SQL scripts in the /u01/app/oracle/cfgtoollogs/dbupgrd/preupgrade directory. Create the directory.

```
$ mkdir -p /u01/app/oracle/cfgtoollogs/dbupgrd/preupgrade  
$
```

- c. From the oracle user 11g terminal window, run the preupgrd.sql script. This

```
$ sqlplus / as sysdba  
...  
SQL>  
@/u01/app/oracle/product/11.2.0/dbhome_2/rdbms/admin/preupgrd.sql  
1  
Loading Pre-Upgrade Package...  
  
Executing Pre-Upgrade Checks...  
Pre-Upgrade Checks Complete.  
  
*****  
  
Results of the checks are located at:  
/u01/app/oracle/cfgtoollogs/dbupgrd/preupgrade/preupgrade.log  
  
Pre-Upgrade Fixup Script (run in source database environment):  
/u01/app/oracle/cfgtoollogs/dbupgrd/preupgrade/preupgrade_fixups.sql  
  
Post-Upgrade Fixup Script (run shortly after upgrade):  
/u01/app/oracle/cfgtoollogs/dbupgrd/preupgrade/postupgrade_fixups.sql  
  
*****  
Fixup scripts must be reviewed prior to being executed.  
*****  
  
*****  
===== >> USER ACTION REQUIRED <<=====  
*****  
  
The following are *** ERROR LEVEL CONDITIONS *** that must be  
addressed prior to attempting your upgrade.  
Failure to do so will result in a failed upgrade.
```

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

1) Check Tag: PURGE_RECYCLEBIN

Check Summary: Check that recycle bin is empty prior to upgrade

Fixup Summary:

"The recycle bin will be purged."

You MUST resolve the above errors prior to upgrade

SQL> exit

\$

Practice 8-2: Implementing the Pre-Upgrade Information Tool Recommendations

Overview

In this practice, you will examine the Pre-Upgrade Information Tool output log file and implement the recommended actions. The actions and recommendations resulting from your environment and the one displayed below may differ.

1. Examine the

/u01/app/oracle/cfgtoollogs/dbupgrd/preupgrade/preupgrade.log, output created from the Pre-Upgrade Information Tool. Generally you will access this from the environment where you ran the script. From the oracle user 11g terminal window, execute the following:

```
$ cat
/u01/app/oracle/cfgtoollogs/dbupgrd/preupgrade/preupgrade.log
Oracle Database Pre-Upgrade Information Tool 02-14-2013 19:13:10
Script Version: 12.1.0.1.0 Build: 006
*****
Database Name: DBUPGRD
Version: 11.2.0.3.0
Compatible: 11.2.0.0.0
Blocksize: 8192
Platform: Linux x86 64-bit
Timezone file: V14
*****
[Renamed Parameters]
[No Renamed Parameters in use]
*****
[Obsolete/Deprecated Parameters]
[No Obsolete or Desupported Parameters in use]
*****
[Component List]
*****
--> Oracle Catalog Views [upgrade] VALID
--> Oracle Packages and Types [upgrade] VALID
--> Oracle Workspace Manager [upgrade] VALID
--> Oracle Enterprise Manager Repository [upgrade] VALID
*****
[Tablespaces]
*****
--> SYSTEM tablespace is adequate for the upgrade.
    minimum required size: 748 MB
--> SYSAUX tablespace is adequate for the upgrade.
    minimum required size: 500 MB
```

```
--> UNDOTBS1 tablespace is adequate for the upgrade.  
    minimum required size: 400 MB  
--> TEMP tablespace is adequate for the upgrade.  
    minimum required size: 60 MB
```

[No adjustments recommended]

```
*****  
[Pre-Upgrade Checks]  
*****
```

WARNING: --> Process Count may be too low

Database has a maximum process count of 150 which is lower than the default value of 300 for this release.

You should update your processes value prior to the upgrade to a value of at least 300.

For example:

```
ALTER SYSTEM SET PROCESSES=300 SCOPE=SPFILE
```

or update your init.ora file.

WARNING: --> Enterprise Manager Database Control repository found in the database

In Oracle Database 12c, Database Control is removed during the upgrade. To save time during the Upgrade, this action can be done prior to upgrading using the following steps after copying **rdbms/admin/emremove.sql** from the new Oracle home

- Stop EM Database Control:

```
$> emctl stop dbconsole
```

- Connect to the Database using the SYS account AS SYSDBA:

```
SET ECHO ON;  
SET SERVEROUTPUT ON;  
@emremove.sql
```

Without the set echo and serveroutput commands you will not be able to follow the progress of the script.

INFORMATION: --> Older Timezone in use

Database is using a time zone file older than version 18. After the upgrade, it is recommended that DBMS_DST package

be used to upgrade the 11.2.0.3.0 database time zone version
to the latest version which comes with the new release.
Please refer to My Oracle Support note number 977512.1 for details.

ERROR: --> RECYCLE_BIN not empty.

Your recycle bin contains 2 object(s).

It is REQUIRED that the recycle bin is empty prior to upgrading.

Immediately before performing the upgrade, execute the following command:

```
EXECUTE dbms_prep.purge_recyclebin_fixup;
```

[Pre-Upgrade Recommendations]

Dictionary Statistics *****

Please gather dictionary statistics 24 hours prior to upgrading the database.

To gather dictionary statistics execute the following command while connected as SYSDBA:

```
EXECUTE dbms_stats.gather_dictionary_stats;
```

^^^ MANUAL ACTION SUGGESTED ^^^

[Post-Upgrade Recommendations]

Fixed Object Statistics *****

Please create stats on fixed objects two weeks after the upgrade using the command:

```
EXECUTE DBMS_STATS.GATHER_FIXED_OBJECTS_STATS;
```

```
^^^ MANUAL ACTION SUGGESTED ^^^  
*****  
***** Summary *****  
  
1 ERROR exist that must be addressed prior to performing your  
upgrade.  
2 WARNINGS that Oracle suggests are addressed to improve  
database performance.  
1 INFORMATIONAL message that should be reviewed prior to your  
upgrade.  
  
After your database is upgraded and open in normal mode you  
must run rdbms/admin/catuppst.sql which executes several  
required tasks and completes the upgrade process.  
  
You should follow that with the execution of  
rdbms/admin/utlrp.sql, and a comparison of invalid objects  
before and after the upgrade using rdbms/admin/utluobj.sql  
  
If needed you may want to upgrade your timezone data using the  
process described in My Oracle Support note 977512.1  
*****  
$
```

2. You have two ways to fix the warnings and recommendations. You run the `$ORACLE_BASE/cfgtoollogs/dbupgrd/preupgrade/preupgrade_fixups.sql` script which will attempt to resolve issues reported, or you can perform the recommend actions manually. Issues that cannot be resolved automatically by the fixup script are flagged with ** USER ACTION REQUIRED **. You will have to fix them manually one by one.
3. From the oracle user 11g terminal window, execute the `/u01/app/oracle/cfgtoollogs/dbupgrd/preupgrade/preupgrade_fixups.sql` script and check if all required actions were fixed..

```
$ sqlplus / as sysdba  
  
SQL>  
@/u01/app/oracle/cfgtoollogs/dbupgrd/preupgrade/preupgrade_fixups.sql  
  
Pre-Upgrade Fixup Script Generated on 2013-02-14 19:13:04  
Version: 12.1.0.1 Build: 006  
Beginning Pre-Upgrade Fixups...  
  
PL/SQL procedure successfully completed.
```

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

```
*****
```

Check Tag: DEFAULT_PROCESS_COUNT

Check Summary: Verify min process count is not too low

Fix Summary: **Review and increase if needed, your PROCESSES value.**

```
*****
```

Fixup Returned Information:

WARNING: --> Process Count may be too low

Database has a maximum process count of 150 which is lower than the default value of 300 for this release.

You should update your processes value prior to the upgrade to a value of at least 300.

For example:

```
ALTER SYSTEM SET PROCESSES=300 SCOPE=SPFILE  
or update your init.ora file.
```

```
*****
```

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

```
*****
```

Check Tag: EM_PRESENT

Check Summary: Check if Enterprise Manager is present

Fix Summary: **Execute emremove.sql prior to upgrade.**

```
*****
```

Fixup Returned Information:

WARNING: --> Enterprise Manager Database Control repository found in the database

In Oracle Database 12c, Database Control is removed during the upgrade. To save time during the Upgrade, this action can be done prior to upgrading using the following steps after copying rdbms/admin/emremove.sql from the new Oracle home

- Stop EM Database Control:

```
$> emctl stop dbconsole
```

- Connect to the Database using the SYS account AS SYSDBA:

```
SET ECHO ON;
```

```
SET SERVEROUTPUT ON;
```

```
@emremove.sql
Without the set echo and serveroutput commands you will not
be able to follow the progress of the script.
*****
PL/SQL procedure successfully completed.

*****
Check Tag: PURGE_RECYLEBIN
Check Summary: Check that recycle bin is empty prior to upgrade
Fix Summary: The recycle bin will be purged.
*****
Fixup Succeeded
*****
PL/SQL procedure successfully completed.

*****
[Pre-Upgrade Recommendations]
*****

PL/SQL procedure successfully completed.

*****
Dictionary Statistics *****
*****
Please gather dictionary statistics 24 hours prior to
upgrading the database.
To gather dictionary statistics execute the following command
while connected as SYSDBA:
  EXECUTE dbms_stats.gather_dictionary_stats;

^^^ MANUAL ACTION SUGGESTED ^^^

PL/SQL procedure successfully completed.

*****
***** Fixup Summary *****
```

```
1 fixup routine was successful.  
2 fixup routines returned INFORMATIONAL text that should be  
reviewed.  
PL/SQL procedure successfully completed.  
  
***** Pre-Upgrade Fixup Script Complete *****  
  
PL/SQL procedure successfully completed.  
  
SQL>
```

The fixup routine that was succeeded is the recycle bin dump.

Verify that the recycle bin is now empty.

```
SQL> select * from dba_recyclebin;  
  
no rows selected  
  
SQL>
```

4. Perform the non-fixed issues.

- a. First update the PROCESSES value prior to the upgrade to a value of at least 300.

```
SQL> show parameter processes  
  
NAME                                     TYPE        VALUE  
-----  
...  
processes                                integer     150  
  
SQL> ALTER SYSTEM SET PROCESSES=300 SCOPE=SPFILE;  
  
System altered.  
  
SQL> EXIT  
$
```

- b. Perform the second recommended action. In Oracle Database 12c, Enterprise Manager Database Control is removed during the upgrade. Enterprise Manager Database Control does not exist anymore with Oracle Database 12c. It is replaced by Enterprise Manager Database Express. To save time during the Upgrade, this action can be done prior to upgrading using the following steps:

- i) Copy the \$ORACLE_HOME/rdbms/admin/emremove.sql from the new 12c Oracle home.

```
$ cp  
/u01/app/oracle/product/12.1.0/dbhome_1/rdbms/admin/emremove.sql  
/u01/app/oracle/product/11.2.0/dbhome_2/rdbms/admin/emremove.sql  
$
```

- ii) Stop EM Database Control in case it is running.

```
$ emctl stop dbconsole  
Oracle Enterprise Manager 11g Database Control Release  
11.2.0.3.0  
Copyright (c) 1996, 2011 Oracle Corporation. All rights  
reserved.  
https://<your\_hostname>:1158/em/console/aboutApplication  
Stopping Oracle Enterprise Manager 11g Database Control ...  
... Stopped.  
$
```

- iii) Execute the removal script. This takes about 6 minutes

```
$ sqlplus / as sysdba  
  
SQL> SET ECHO ON  
SQL> SET SERVEROUTPUT ON  
SQL> @$ORACLE_HOME/rdbms/admin/emremove.sql  
...  
Dropping synonym : MGMT$APPLIED_PATCHSETS ...  
Dropping synonym : MGMT$APPLIED_PATCHES ...  
Dropping synonym : MGMT$ALERT_NOTIF_LOG ...  
Dropping synonym : MGMT$ALERT_HISTORY ...  
Dropping synonym : MGMT$ALERT_CURRENT ...  
Dropping synonym : MGMT$ALERT_ANNOTATIONS ...  
Dropping synonym : EMD_MNTR ...  
Dropping synonym : ECM_UTIL ...  
Finished phase 5  
Starting phase 6 : Dropping Oracle Enterprise Manager related  
other roles ...  
Finished phase 6  
The Oracle Enterprise Manager related schemas and objects are  
dropped.  
Do the manual steps to shutdown the DB Control if not done  
before running this  
script and then delete the DB Control configuration files  
  
PL/SQL procedure successfully completed.
```

SQL>

- c. The last recommendation is to gather dictionary statistics 24 hours prior to upgrading the database.

SQL> exec dbms_stats.gather_dictionary_stats

PL/SQL procedure successfully completed.

SQL> exit

\$

5. Examine the

\$ORACLE_BASE/cfgtoollogs/dbupgrd/preupgrade/postupgrade_fixups.sql
script created to address issues that can be fixed after the database will have been upgraded.

```
$ cat
$ORACLE_BASE/cfgtoollogs/dbupgrd/preupgrade/postupgrade_fixups.sql
REM Post Upgrade Script Generated on: 2013-02-14 19:13:04
REM Generated by Version: 12.1.0.1 Build: 006
SET ECHO OFF SERVEROUTPUT ON FORMAT WRAPPED TAB OFF LINESIZE
750;
BEGIN
    dbms_output.put_line ('Post Upgrade Fixup Script Generated on
2013-02-14 19:13:04 Version: 12.1.0.1 Build: 006');
    dbms_output.put_line ('Beginning Post-Upgrade Fixups...');

END;
/
BEGIN
dbms_preup.clear_run_flag(FALSE);
END;
/
BEGIN
-- ***** Fixup Details *****
-- Name:          OLD_TIME_ZONES_EXIST
-- Description:  Check for use of older timezone data file
-- Severity:     Informational
-- Action:        ^^^ MANUAL ACTION REQUIRED ^^^
-- Fix Summary:
--               Update the timezone using the DBMS_DST package after
upgrade is complete.

dbms_preup.run_fixup_and_report('OLD_TIME_ZONES_EXIST');
END;
/
```

```
BEGIN
  dbms_output.put_line
  ('*****');
  dbms_output.put_line ('[Post-Upgrade Recommendations]');
  dbms_output.put_line
  ('*****');
  dbms_output.put_line ('');
END;
/
BEGIN
  dbms_output.put_line (
  '*****');
  dbms_output.put_line (' ***** Fixed Object Statistics
*****');
  dbms_output.put_line (
  '*****');
  dbms_output.put_line ('');
  dbms_output.put_line ('Please create stats on fixed objects two
weeks');
  dbms_output.put_line ('after the upgrade using the command:');
dbms_output.put_line ('EXECUTE
DBMS_STATS.GATHER_FIXED_OBJECTS_STATS');
  dbms_output.put_line ('');
  dbms_output.put_line ('^^^ MANUAL ACTION SUGGESTED ^^^');
  dbms_output.put_line ('');
END;
/
BEGIN dbms_prepup.fixup_summary(FALSE); END;
/
BEGIN
  dbms_output.put_line ('* Post Upgrade Fixup Script Complete
**');
END;
/
REM Post Upgrade Script Closed At: 2013-02-14 19:13:16
$
```

Recommendations are suggested for after upgrade such as updating the time zone using the DBMS_DST package.

Practice 8-3: Completing Prerequisites Steps Before Upgrade

Overview

In this practice, you perform the last actions and verifications before the upgrade.

Tasks

1. The Pre-Upgrade Information Tool creates and populates the `registry$sys_inv_objs` and `registry$nonsys_inv_objs` tables. If you did not execute the Pre-Upgrade Information Tool script, you can execute the `utluiobj.sql` script before and after upgrading. The script execution outputs the difference between the invalid objects that existed prior the upgrade and invalid objects that exist after to upgrade. Consider that you did not execute the Pre-Upgrade Information Tool because you know that DBUA will automatically execute it. You are still logged in the session with 11g `ORACLE_HOME` set.

```
$ sqlplus / as sysdba

SQL> @$ORACLE_HOME/rdbms/admin/utluiobj.sql
.
.
.
Oracle Database 11.1 Post-Upgrade Invalid Objects Tool 02-14-
2013 21:42:13
.

This tool lists post-upgrade invalid objects that were not
invalid prior to upgrade (it ignores pre-existing pre-upgrade
invalid objects).

.

Owner          Object Name          Object Type
.

PL/SQL procedure successfully completed.

SQL> select * from REGISTRY$sys_inv_objs;
no rows selected

SQL> select * from REGISTRY$nonsys_inv_objs;
no rows selected

SQL>
```

2. Verify that materialized view refreshes have completed. If materialized view refreshes have not completed, perform the refresh manually using the `DBMS_MVIEW` package.

```
SQL> select o.name from sys.obj$ o, sys.user$ u, sys.sum$ s
      where o.type# = 42
        and bitand(s.mflags, 8) =8;
2      3
```

```
no rows selected  
SQL>
```

3. Ensure that no files need media recovery.

```
SQL> select * from v$recover_file;  
  
no rows selected  
  
SQL>
```

If there are files that need recovery, perform the required recovery using RMAN.

4. Resolve outstanding distributed transactions.

```
SQL> select * from dba_2pc_pending;  
  
no rows selected  
  
SQL>
```

If the query in the previous step returns any rows, then run the following statements:

```
SELECT local_tran_id FROM dba_2pc_pending;  
EXECUTE dbms_transaction.purge_lost_db_entry('');  
COMMIT;
```

5. If a standby database exists, then you must synchronize it with the primary database.

- a. Check if a standby database exists.

```
SQL> select  
SUBSTR(value,INSTR(value,'=',INSTR(UPPER(value), 'SERVICE'))+1)  
from v$parameter  
where name like 'log_archive_dest%'  
and UPPER(value) like 'SERVICE%';  
2      3      4  
no rows selected  
  
SQL>
```

- b. *If the query in the previous step returns a row, then synchronize the standby database with the primary database.*
 - *Make sure all the logs are transported to the standby server after a final log switch in the primary.*
 - *Start the recovery of the standby database with the NODELAY option.*

6. The upgrade process uses a large amount of archive log files. Increase the DB_RECOVERY_FILE_DEST_SIZE to 30GB.

```
SQL> alter system set DB_RECOVERY_FILE_DEST_SIZE=30G scope=BOTH;  
System altered.  
  
SQL> exit  
$
```

Practice 8-4: Performing a Full Database Backup

Overview

In this practice, you will perform a full database backup of the dbupgrd database before you start the upgrade.

Tasks

1. Verify that the database is in ARCHIVELOG mode.
 - a. Stay in the oracle user 11g terminal window with the environment set to the dbupgrd instance. Connect to the database instance as SYSDBA.

```
$ sqlplus / as sysdba

SQL> select log_mode from v$database;

LOG_MODE
-----
NOARCHIVELOG

SQL>
```

- b. Set the database instance in ARCHIVELOG mode.
 - i) Shut the database instance down.

```
SQL> shutdown immediate
Database closed.
Database dismounted.
ORACLE instance shut down.

SQL>
```

- ii) Start the database instance in MOUNT mode.

```
SQL> startup mount
ORACLE instance started.

Total System Global Area  663908352 bytes
Fixed Size                  2231312 bytes
Variable Size                239076336 bytes
Database Buffers              415236096 bytes
Redo Buffers                  7364608 bytes
Database mounted.

SQL>
```

- iii) Alter the database instance in ARCHIVELOG mode and open.

```
SQL> alter database archivelog;
Database altered.
```

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

```
SQL> alter database open;
```

```
Database altered.
```

```
SQL> exit
```

```
$
```

2. Use RMAN to perform the database full backup.

- a. Verify that the configuration includes the automatic control file and SPFILE backup. If this is not the case, enable the automatic control file and SPFILE backup.

```
$ rman target /
```

```
RMAN> show all;
```

```
using target database control file instead of recovery catalog
RMAN configuration parameters for database with db_unique_name
DBUPGRD are:
```

```
CONFIGURE RETENTION POLICY TO REDUNDANCY 1; # default
```

```
CONFIGURE BACKUP OPTIMIZATION OFF; # default
```

```
CONFIGURE DEFAULT DEVICE TYPE TO DISK; # default
```

```
CONFIGURE CONTROLFILE AUTOBACKUP OFF; # default
```

```
...
```

```
CONFIGURE SNAPSHOT CONTROLFILE NAME TO
```

```
'/u01/app/oracle/product/11.2.0/dbhome_2/dbs/snapcf_dbupgrd.f';
# default
```

```
RMAN> CONFIGURE CONTROLFILE AUTOBACKUP ON;
```

```
new RMAN configuration parameters:
```

```
CONFIGURE CONTROLFILE AUTOBACKUP ON;
```

```
new RMAN configuration parameters are successfully stored
```

```
RMAN>
```

- b. Perform the full database backup including all data files, control files, SPFILE and archive log files.

```
RMAN> backup database plus archivelog;
```

```
Starting backup at 14-FEB-13
```

```
current log archived
```

```
allocated channel: ORA_DISK_1
```

```
channel ORA_DISK_1: SID=147 device type=DISK
```

```
channel ORA_DISK_1: starting archived log backup set
```

```
channel ORA_DISK_1: specifying archived log(s) in backup set
input archived log thread=1 sequence=12 RECID=1 STAMP=807374917
channel ORA_DISK_1: starting piece 1 at 14-FEB-13
channel ORA_DISK_1: finished piece 1 at 14-FEB-13
piece
handle=+FRA/dbupgrd/backupset/2013_02_14/annnf0_tag20130214t1448
38_0.261.807374919 tag=TAG20130214T144838 comment=NONE
channel ORA_DISK_1: backup set complete, elapsed time: 00:00:03
Finished backup at 14-FEB-13

Starting backup at 14-FEB-13
using channel ORA_DISK_1
channel ORA_DISK_1: starting full datafile backup set
channel ORA_DISK_1: specifying datafile(s) in backup set
input datafile file number=00001
name=+DATA/dbupgrd/datafile/system.256.807214393
input datafile file number=00002
name=+DATA/dbupgrd/datafile/sysaux.257.807214393
input datafile file number=00005
name=+DATA/dbupgrd/datafile/example.261.808328869
input datafile file number=00003
name=+DATA/dbupgrd/datafile/undotbs1.258.807214395
input datafile file number=00004
name=+DATA/dbupgrd/datafile/users.258.808326885
channel ORA_DISK_1: starting piece 1 at 14-FEB-13
channel ORA_DISK_1: finished piece 1 at 14-FEB-13
piece
handle=+FRA/dbupgrd/backupset/2013_02_14/nnndf0_tag20130214t1448
41_0.262.807374923 tag=TAG20130214T144841 comment=NONE
channel ORA_DISK_1: backup set complete, elapsed time: 00:01:45
Finished backup at 14-FEB-13

Starting backup at 14-FEB-13
current log archived
using channel ORA_DISK_1
channel ORA_DISK_1: starting archived log backup set
channel ORA_DISK_1: specifying archived log(s) in backup set
input archived log thread=1 sequence=13 RECID=2 STAMP=807375027
channel ORA_DISK_1: starting piece 1 at 14-FEB-13
channel ORA_DISK_1: finished piece 1 at 14-FEB-13
piece
handle=+FRA/dbupgrd/backupset/2013_02_14/annnf0_tag20130214t1450
27_0.264.807375029 tag=TAG20130214T145027 comment=NONE
channel ORA_DISK_1: backup set complete, elapsed time: 00:00:01
```

```
Finished backup at 14-FEB-13

Starting Control File and SPFILE Autobackup at 14-FEB-13
piece
handle=+FRA/dbupgrd/autobackup/2013_02_14/s_807375029.265.807375
031 comment=NONE
Finished Control File and SPFILE Autobackup at 14-FEB-13

RMAN> exit
$
```

At the end of the full database backup, you have:

- Two backup sets of archived logs: one completed BEFORE the data files backup and one completed AFTER the data files backup because modifications in the data files may have been applied by applications. The backup sets are stored in +FRA/dbupgrd/backupset/20yy_mm_dd
- One backup set of the five data files stored in +FRA/dbupgrd/backupset/20yy_mm_dd
- One backup set of the control file and SPFILE stored in +FRA/dbupgrd/autobackup/20yy_mm_dd

Practice 8-5: Performing a Transportable Tablespace Export

Overview

In anticipation of a later practice, you will perform a transportable tablespace export before the database is upgraded. This export will be used in a later practice to experiment another method of upgrade using Data Pump export and import.

Tasks

1. Perform a transportable tablespace export of the dbupgrd database. The database instance still runs in the Oracle Database 11g version. Connect to the dbupgrd database instance.

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

$ sqlplus / as sysdba

SQL>
```

2. Put the user-defined tablespaces in the source database dbupgrd in read-only mode.
 - a. Find the list of user-defined tablespaces to be put in read-only mode.

```
SQL> SELECT tablespace_name FROM dba_tablespaces ORDER BY 1;

TABLESPACE_NAME
-----
EXAMPLE
SYSAUX
SYSTEM
TEMP
UNDOTBS1
USERS

6 rows selected.

SQL>
```

- b. The list may be different from yours according to the tablespaces created during the training session. Make all tablespaces except SYSTEM, SYSAUX, TEMP, and UNDOTBS1 read-only.

```
SQL> ALTER TABLESPACE example READ ONLY;

Tablespace altered.

SQL> ALTER TABLESPACE users READ ONLY;
```

```
Tablespace altered.
```

```
SQL>
```

- c. Find the list of data files associated to the read-only tablespaces that need to be transported.

```
SQL> SELECT file_name FROM dba_data_files
      WHERE tablespace_name IN ('EXAMPLE','USERS');
      2
FILE_NAME
-----
+DATA/dbupgrd/datafile/users.258.808326885
+DATA/dbupgrd/datafile/example.261.808328869

SQL> exit
$
```

3. Export the dbupgrd database in transportable tablespace mode.

```
$ rm /u01/app/oracle/admin/dbupgrd/dpdump/expTTS.dmp
rm: cannot remove
`/u01/app/oracle/admin/dbupgrd/dpdump/expTTS.dmp': No such file
or directory
$ expdp system/oracle_4U DUMPFILE=expTTS.dmp
TRANSPORT_TABLESPACES=example, users TRANSPORT_FULL_CHECK=YES
LOGFILE=expTTS.log

Export: Release 11.2.0.3.0 - Production on Wed Feb 20 22:02:47
2013

Copyright (c) 1982, 2011, Oracle and/or its affiliates. All
rights reserved.

Connected to: Oracle Database 11g Enterprise Edition Release
11.2.0.3.0 - 64bit Production
With the Partitioning, Automatic Storage Management, OLAP, Data
Mining
and Real Application Testing options
Starting "SYSTEM"."SYS_EXPORT_TRANSPORTABLE_01":
system/******** DUMPFILE=expTTS.dmp
TRANSPORT_TABLESPACES=example, users TRANSPORT_FULL_CHECK=YES
LOGFILE=expTTS.log
Processing object type TRANSPORTABLE_EXPORT/PLUGTS_BLK
Processing object type TRANSPORTABLE_EXPORT/TABLE
Processing object type TRANSPORTABLE_EXPORT/TABLE_STATISTICS
Processing object type
TRANSPORTABLE_EXPORT/POST_INSTANCE/PLUGTS_BLK
Master table "SYSTEM"."SYS_EXPORT_TRANSPORTABLE_01" successfully
loaded/unloaded
```

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

```
*****
Dump file set for SYSTEM.SYS_EXPORT_TRANSPORTABLE_01 is:
/u01/app/oracle/admin/dbupgrd/dpdump/expTTS.dmp
*****
Datafiles required for transportable tablespace EXAMPLE:
+DATA/dbupgrd/datafile/example.261.808328869
Datafiles required for transportable tablespace USERS:
+DATA/dbupgrd/datafile/users.258.808326885
Job "SYSTEM"."SYS_EXPORT_TRANSPORTABLE_01" successfully
completed at 22:06:27
$
```

4. Create a /home/oracle/labs/dbupgrd12 directory to store the dump file.

```
$ mkdir -p /home/oracle/labs/dbupgrd12
$
```

5. Move the dump file to the /home/oracle/labs/dbupgrd12 directory.

```
$ mv /u01/app/oracle/admin/dbupgrd/dpdump/expTTS.dmp
/home/oracle/labs/dbupgrd12
$ ls /home/oracle/labs/dbupgrd12
expTTS.dmp
$
```

6. From the grid user terminal window, copy the data files required for transportable tablespace import of the 11g dbupgrd database to the FRA disk group.

```
$ asmcmd
ASMCMD> ls +DATA/DBUPGRD/DATAFILE
EXAMPLE.261.808328869
SYSAUX.264.808326845
SYSTEM.263.808326819
UNDOTBS1.269.808326865
USERS.258.808326885
ASMCMD> cd +DATA/DBUPGRD/DATAFILE
ASMCMD> cp USERS.* +FRA/USERS
copying +DATA/DBUPGRD/DATAFILE/USERS.258.808326885 -> +FRA/USERS
ASMCMD>
ASMCMD> cp EXAMPLE.* +FRA/EXAMPLE
copying +DATA/DBUPGRD/DATAFILE/EXAMPLE.261.808328869 ->
+FRA/EXAMPLE
ASMCMD> exit
$
```

7. From the oracle user 11g terminal window, put the user-defined tablespaces in the database dbupgrd back in read-write mode.

```
$ sqlplus / as sysdba

SQL> ALTER TABLESPACE example READ WRITE;

Tablespace altered.

SQL> ALTER TABLESPACE users  READ WRITE;

Tablespace altered.

SQL> exit
$
```


Practices for Lesson 9: Upgrading a Database

Chapter 9

Practices for Lesson 9

Practices Overview

In the previous practices, you prepared the `dbupgrd` database instance currently running in `11g` to upgrade to `12c`. The upgrade process will use Database Upgrade Assistant (DBUA) which is recommended as the best method for upgrading your database.

In these practices, you will:

- Upgrade the Oracle Database `11g` Release 2 `dbupgrd` database to Oracle Database `12c` by using DBUA
- Plug the `orcl` non-CDB into the `cdb1` CDB as a new PDB

In your upgrade plan, you do not plan to relocate the data files, control files, redo log files and archive log files to another location.

You are going to work as the `oracle` user to perform operations in the `12c` environment only. Therefore, keep a terminal window opened as the `oracle` user in the `12c` environment.

Practice 9-1: Upgrading a Database by Using DBUA

Overview

In this practice, you will upgrade your 11g dbupgrd database instance to 12c.

Assumptions

- You successfully installed the Oracle 12c Database software in Practice 4-1.
- In Practice 8-1, you checked that the dbupgrd database instance is compatible with an upgrade from 11.2.0.3.0 to 12.1.0.1.0 before upgrading.
- In Practices 8-2 and 8-3, you performed all necessary actions to get the database ready for upgrade.
- The full database backup was successfully completed in Practice 8-4.

Tasks

1. From the oracle user 12c terminal window, release resources by shutting down instances other than the ASM instance and the dbupgrd database instance. Why is the ASM instance still required? It is required because the dbupgrd database instance is a client storing its database files in ASM disks.
 - a. Shut down orcl.

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

$ sqlplus / as sysdba

SQL> SHUTDOWN IMMEDIATE
Database closed.
Database dismounted.
ORACLE instance shut down.

SQL> exit
$
```

- b. Shut down cdb1.

```
$ . oraenv
ORACLE_SID = [orcl] ? cdb1
The Oracle base remains unchanged with value /u01/app/oracle

$ sqlplus / as sysdba

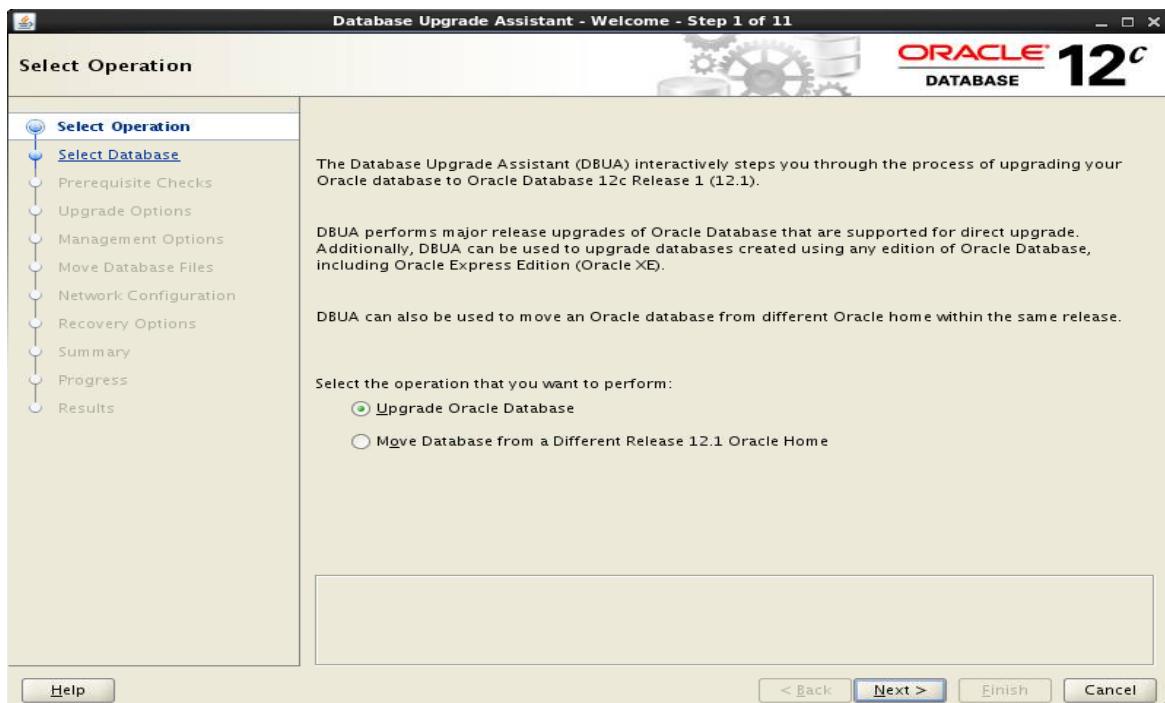
SQL> SHUTDOWN IMMEDIATE
Database closed.
Database dismounted.
ORACLE instance shut down.
```

```
SQL> exit
$
```

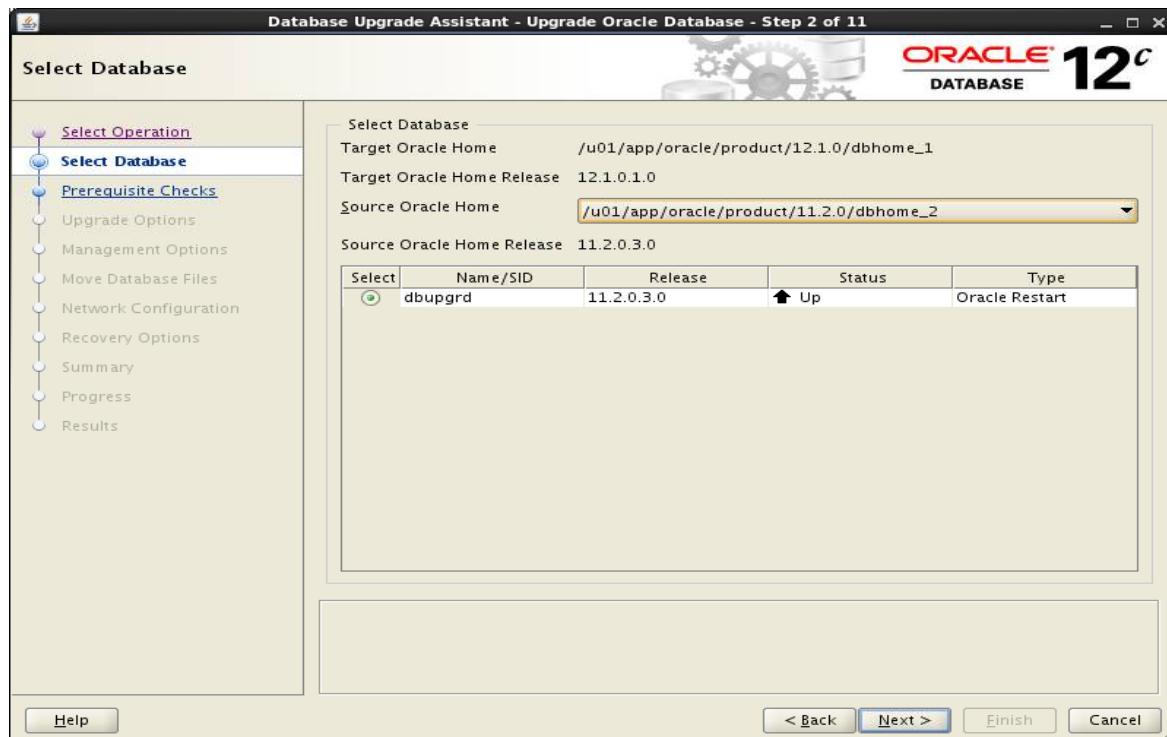
2. Launch DBUA from the Oracle home where the new database software has been installed.

```
$ echo $ORACLE_HOME
/u01/app/oracle/product/12.1.0/dbhome_1
$ dbua
```

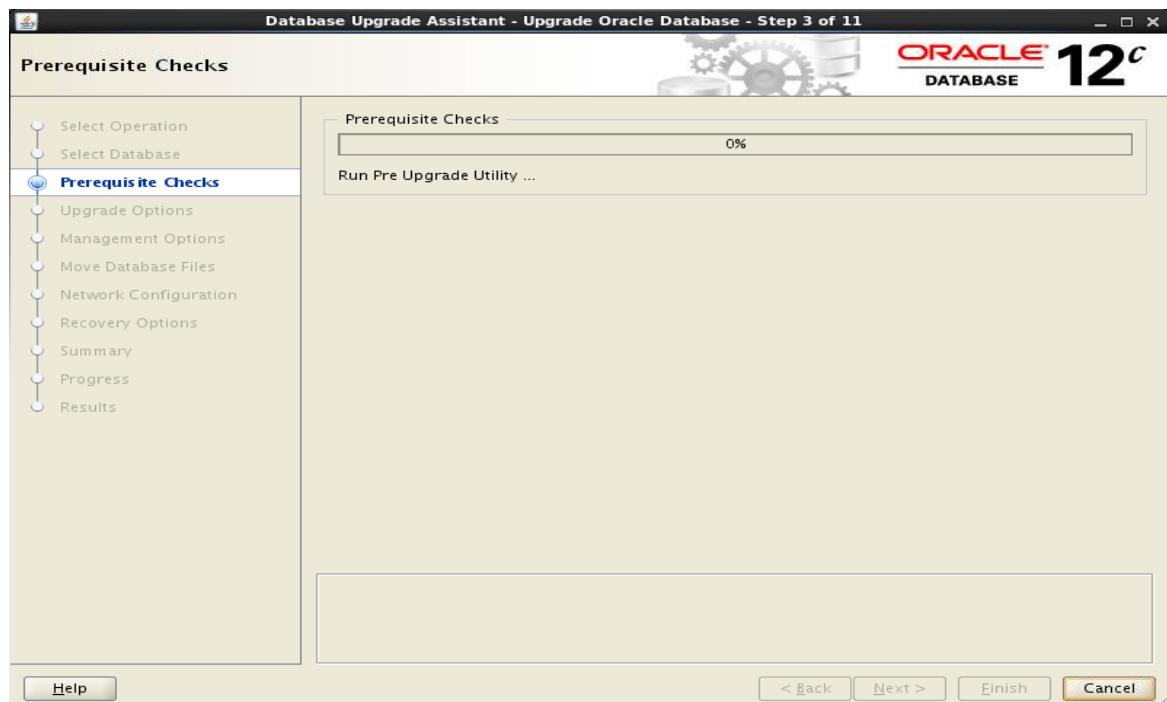
Step	Window/Page Description	Choices or Values
a.	Select Operation page	Verify Upgrade Oracle Database is selected Click Next



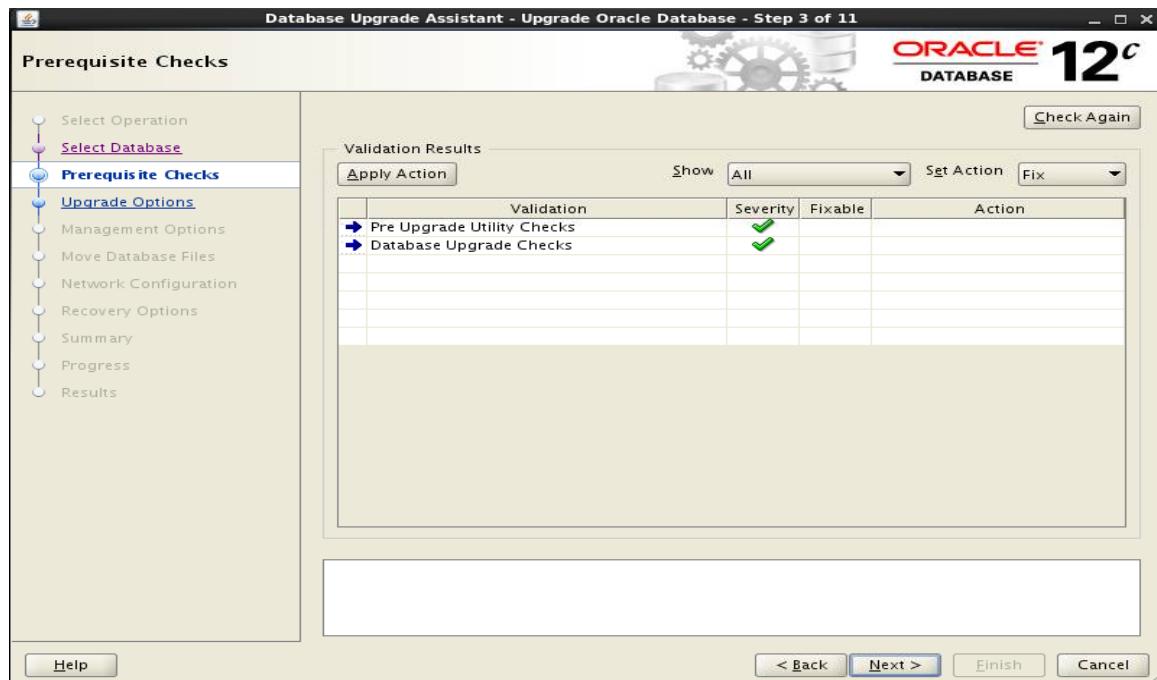
Step	Window/Page Description	Choices or Values
b.	Select Database page	Verify: Target Oracle Home is <code>/u01/app/oracle/product/12.1.0/dbhome_1</code> . Set Source Oracle Home choosing <code>/u01/app/oracle/product/11.2.0/dbhome_2</code> The dbupgrd entry appears. Click Next .



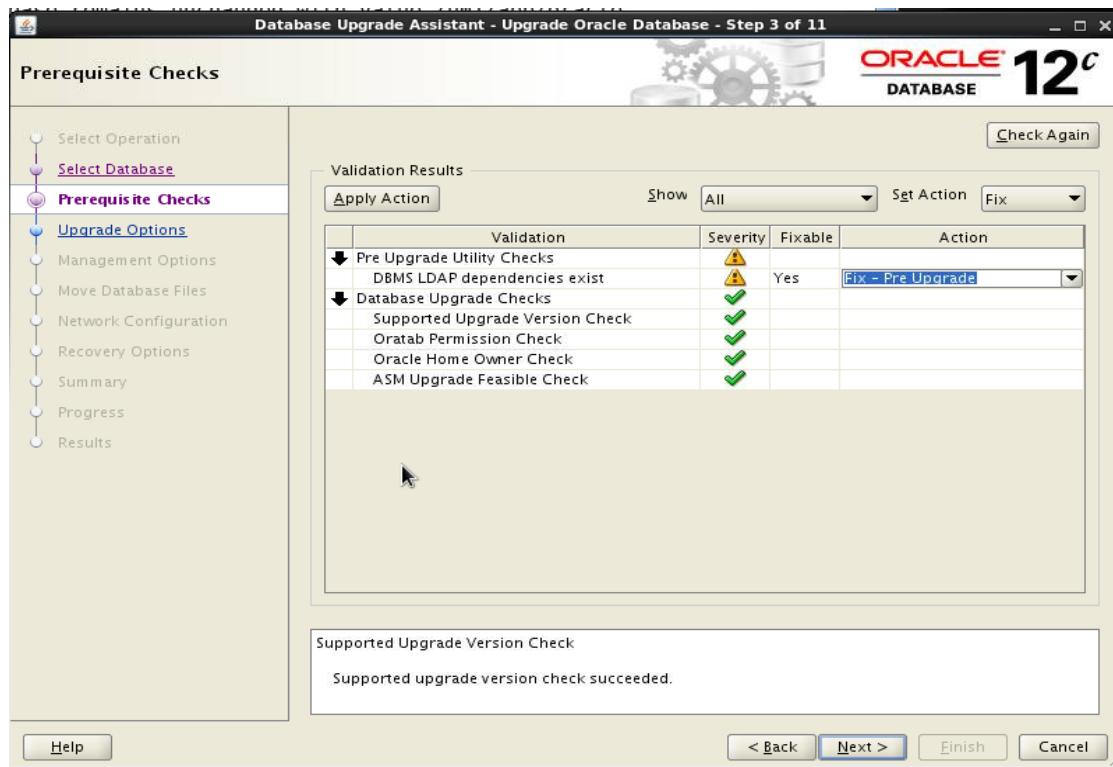
Step	Window/Page Description	Choices or Values
C.	Prerequisite Checks page	Automatic validation starts (using the preupgrd.sql script) and shows progressively the validation result of each prerequisite.



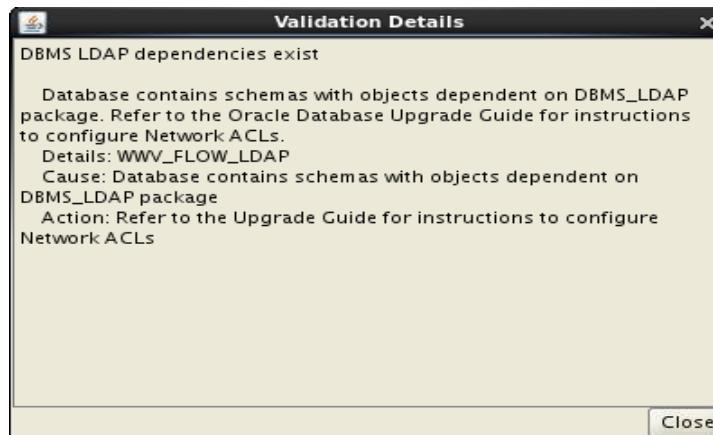
Step	Window/Page Description	Choices or Values
d.	Prerequisite Checks page	<p>Validation results appear</p> <p>The prerequisites are all validated because you already ran the preupgrd.sql script and fixed all prerequisites manually.</p> <p>Click Next.</p>



- If you had prerequisites that were not validated, you would click each Validation entry to get a more detailed list. Below is an example of failed validated prerequisites in a database instance with LDAP dependencies. This is only an example on the procedure you would have to handle if you had to.

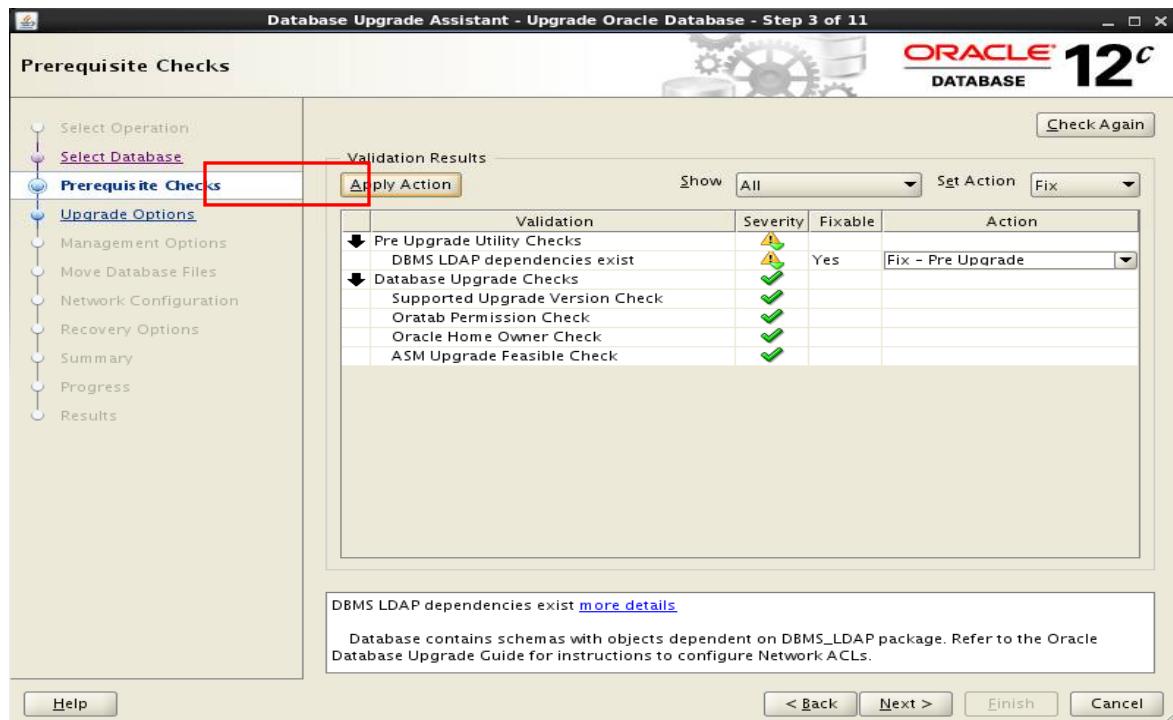


- One check would result in a warning and could be fixed. You would examine in detail the DBMS_LDAP dependencies that could exist. You would select the "DBMS_LDAP dependencies exist" entry and click **more details**.

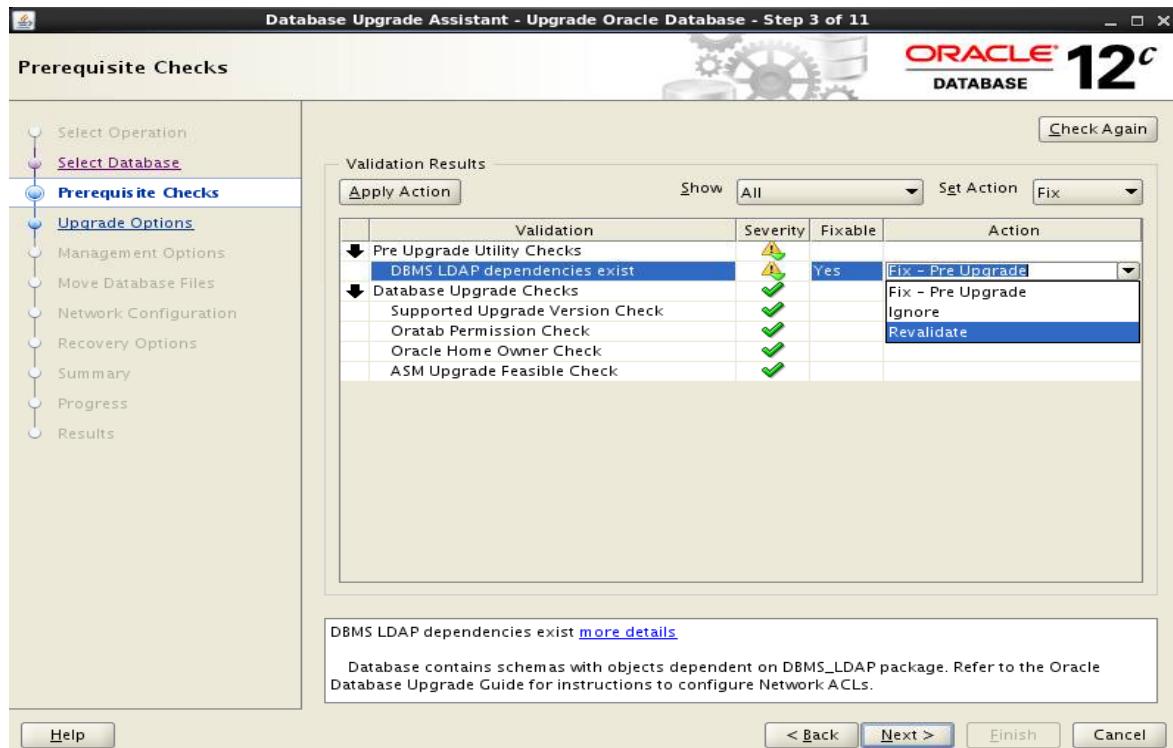


This means that you have not added the object in the ACL privileged list recommended in the preupgrade.log file. You would click **Close**.

- You would click **Apply Action** to fix the issue.



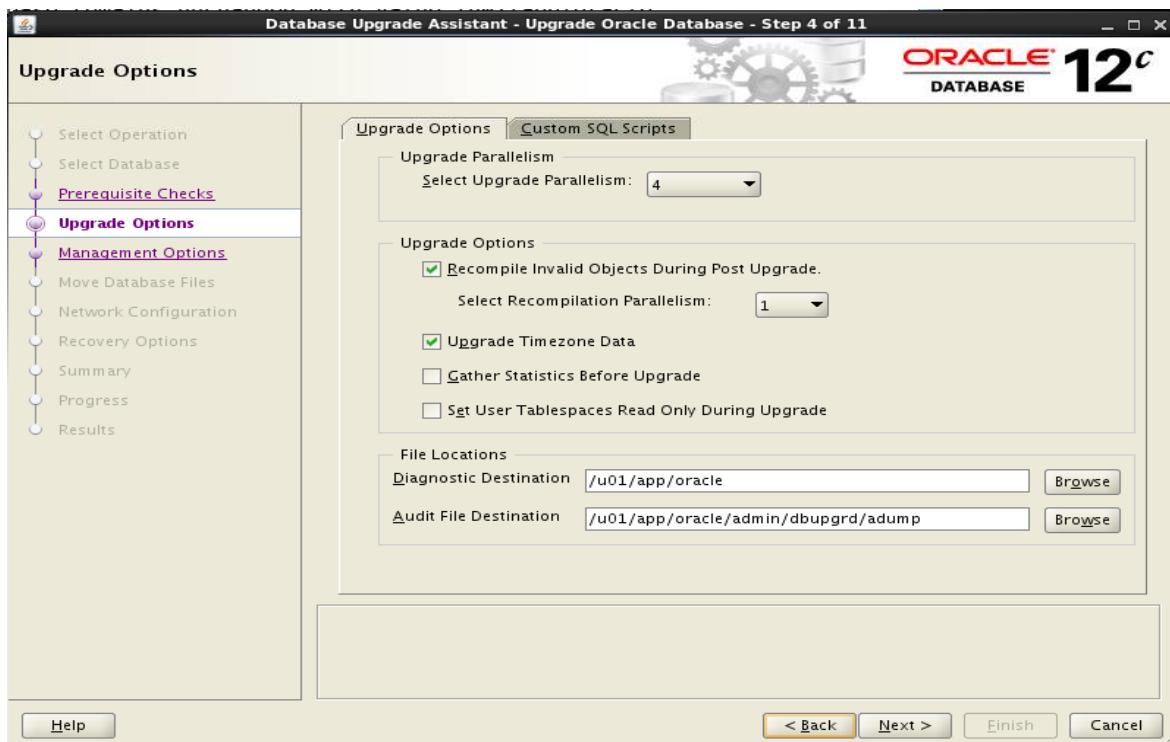
- In Action, you would choose **Revalidate** and click **Apply Action**.



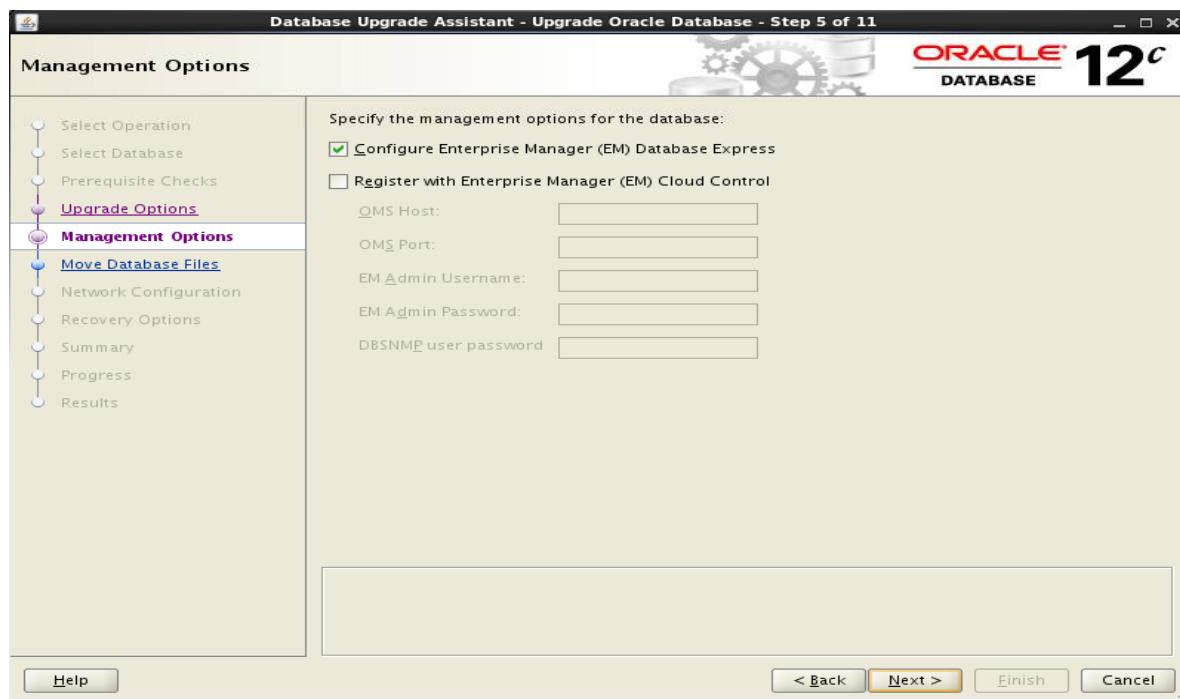
- If the issue persists but would not prevent the upgrade to progress, you could fix this problem after the upgrade. In Action, you would choose **Ignore** and click **Next**.

Step	Window/Page Description	Choices or Values
e.	Upgrade Options page	Verify Upgrade Parallelism is set to the

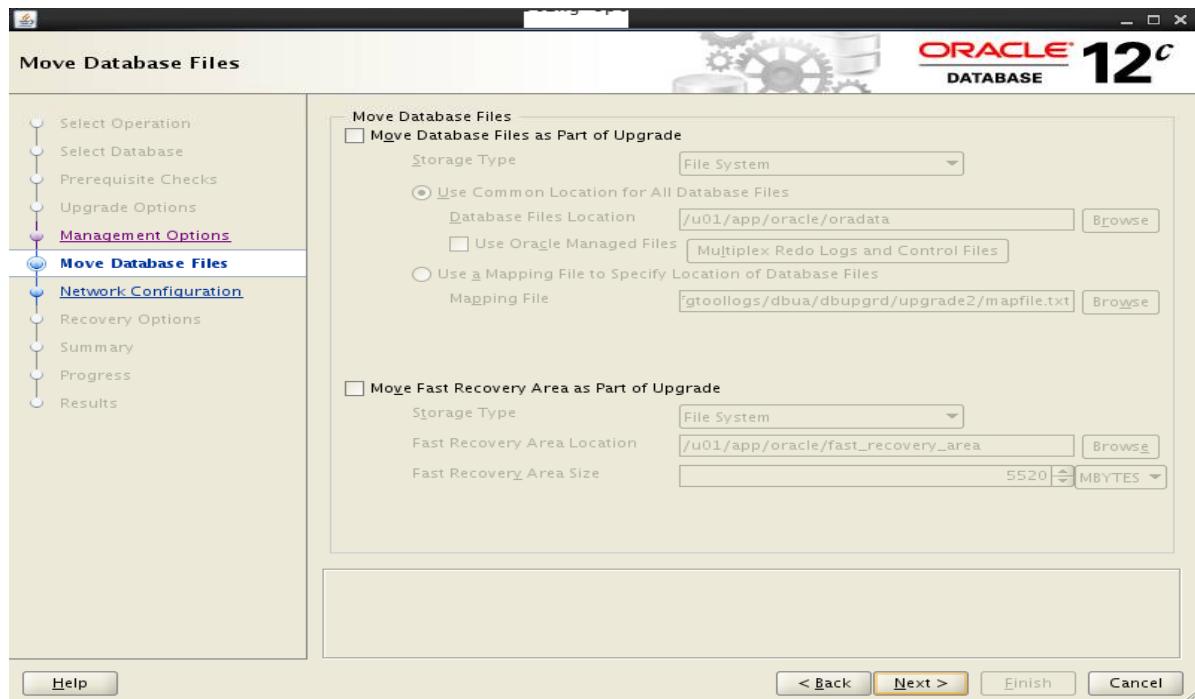
Step	Window/Page Description	Choices or Values
	Upgrade Options tab	<p>number of CPUs or 2 if the number of CPUs is less than 4. You may select a new value from the Degree of Parallelism drop-down list. this value determines the number of parallel processes used to recompile all invalid PL/SQL modules after the upgrade is complete.</p> <p>Select Upgrade Timezone Data</p> <p>Deselect Gather Statistics Before Upgrade. You already gathered the statistics before the upgrade.</p> <p>Click Next.</p>



Step	Window/Page Description	Choices or Values
f.	Management Options page	Click Next

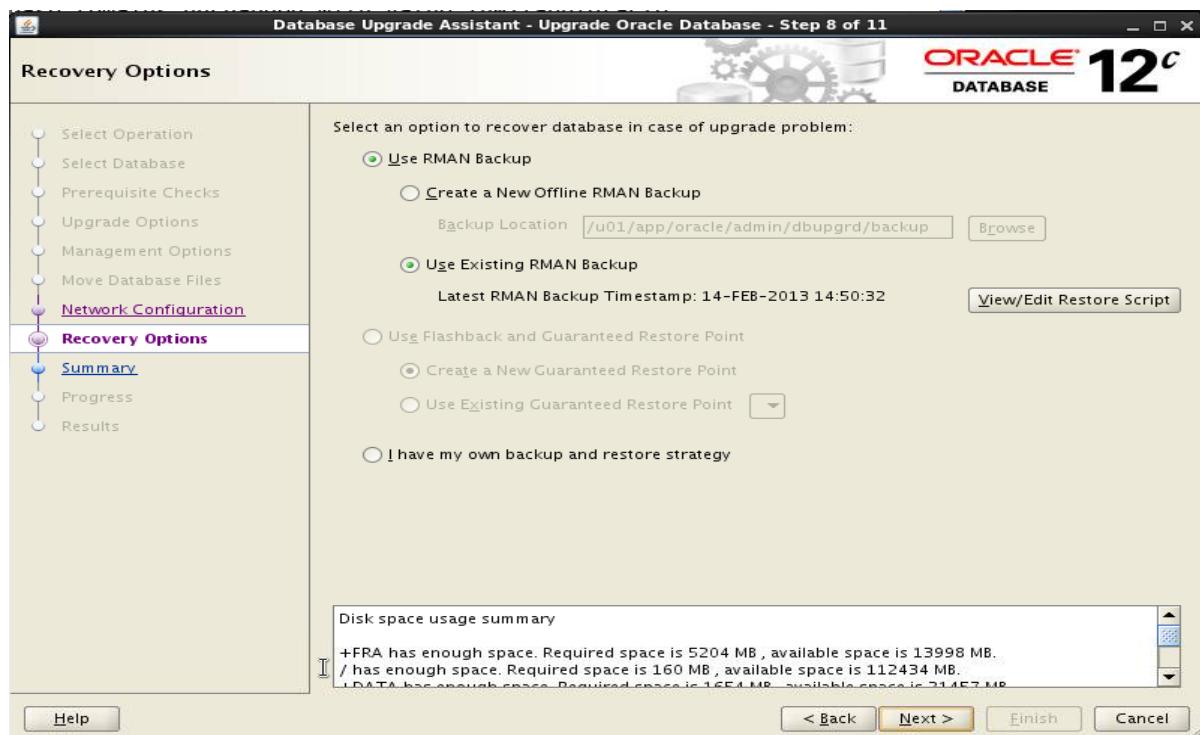


Step	Window/Page Description	Choices or Values
g.	Move Database Files page	<p><i>In this upgrade, do not move the database files to another type of storage such as a file system or other disk groups. You do not want to set the FRA location on another type of storage such as a file system or other disk groups. These operations can be performed after the upgrade.</i></p> <p>Click Next</p>

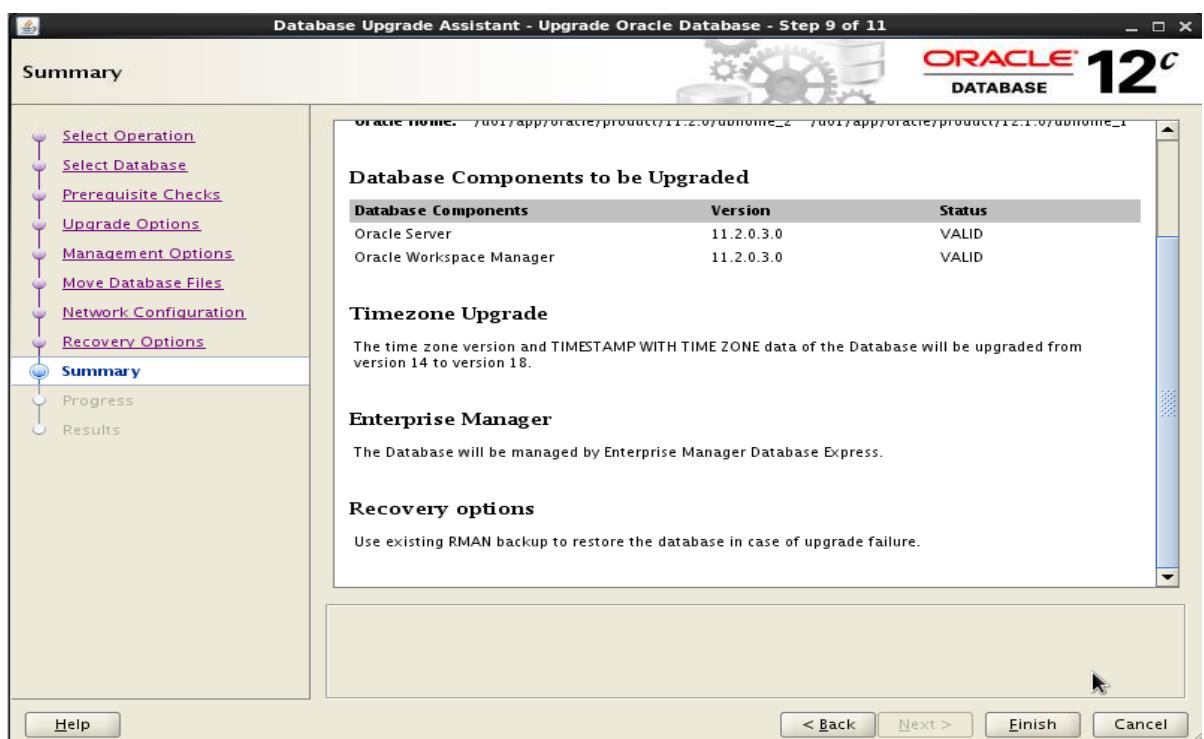
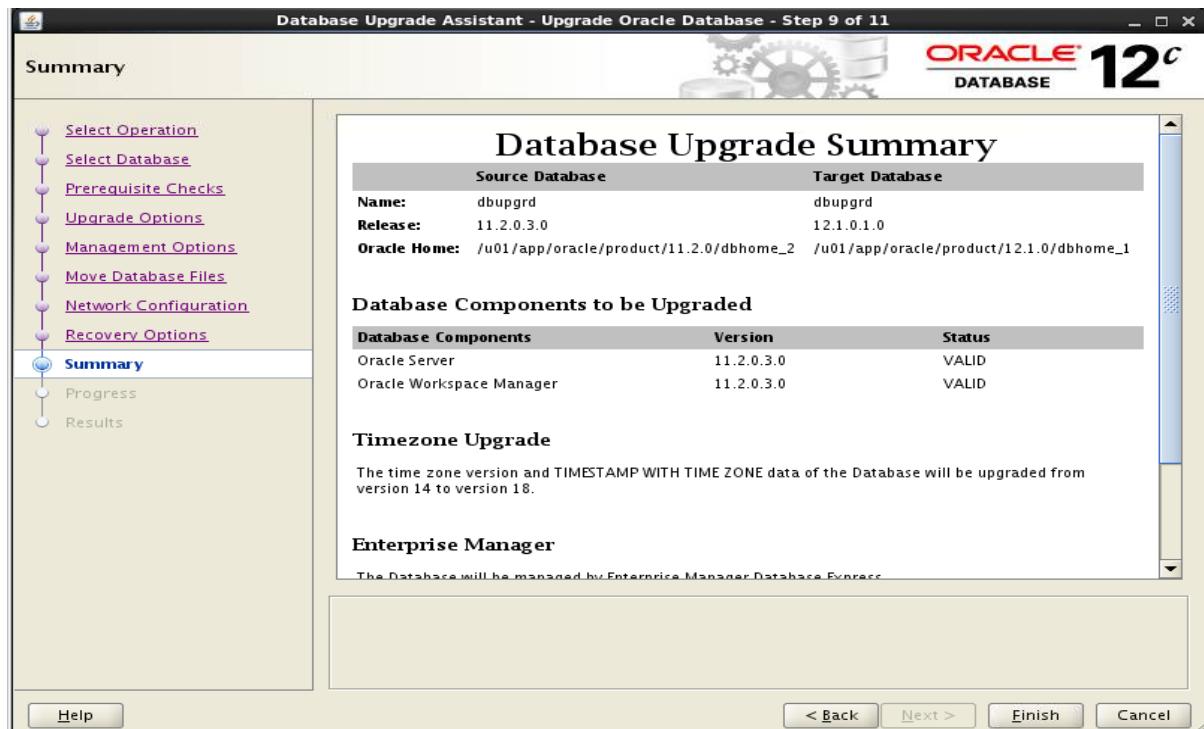


Step	Window/Page Description	Choices or Values
h.	Network Configuration page	Select LISTENER. Click Next
i.	Recovery Options page	Select Use Existing RMAN Backup <i>You backed up the database in the previous practice</i> Click Next

Read carefully the message in the box at the bottom of Recovery Options page if there is any. If it says: "You do not have enough disk space for the archive logs and flashback logs generated during upgrade." you would click **more details**. The Validation Details page would explicitly provide the required disk space for the archive logs and flashback logs generated during upgrade. In another oracle user UNIX session, connected to the dbupgrd instance, you would increase the DB_RECOVERY_FILE_DEST_SIZE to the required size, and click **Close**.



Step	Window/Page Description	Choices or Values
j.	Summary page	<p>Review Database Upgrade Summary Check:</p> <p>Source and target Database Releases, Oracle Homes</p> <p>Warnings ignored if there had been any (as described in task 3.d),</p> <p>Status of the components that will be upgraded,</p> <p>Version of the time zone data.</p> <p>Click Finish. The upgrade may take around 45 minutes. (118 minutes on M82)</p>



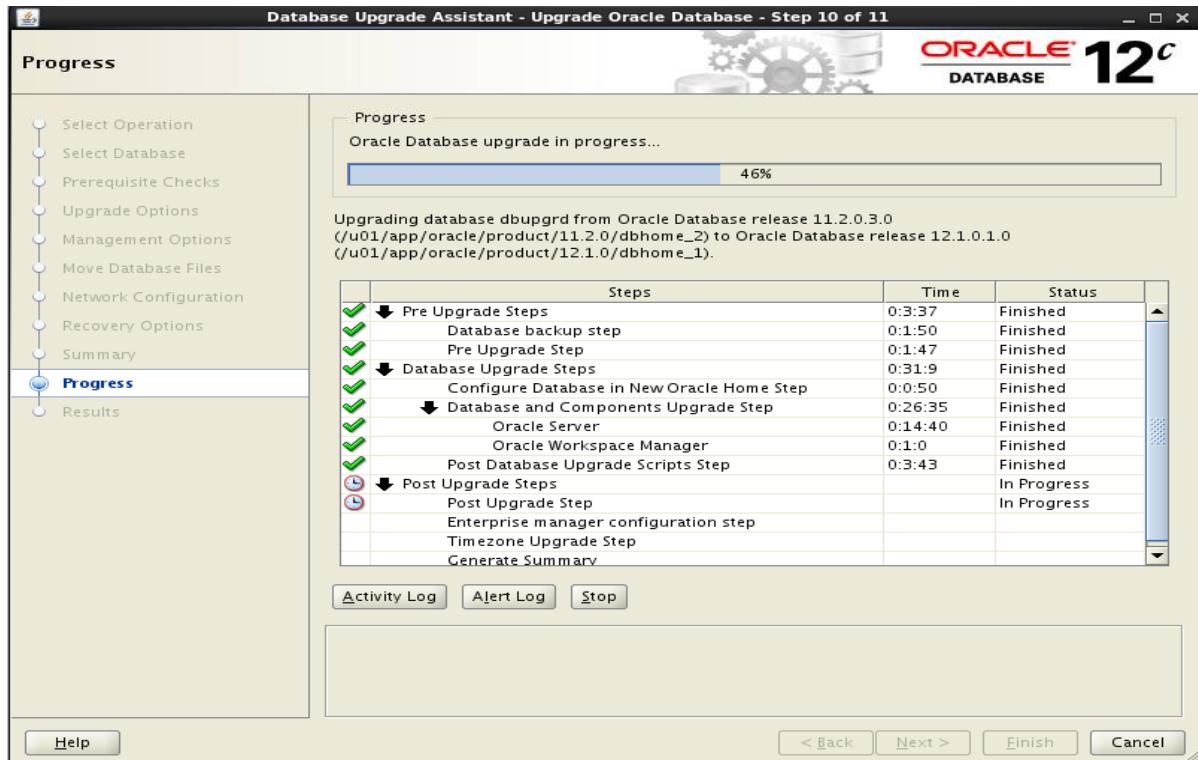
Note: During the upgrade phase, DBUA runs `catctl.pl`, which runs the upgrade processes in parallel instead of serially, optimally taking advantage of CPU capacity to decrease down time as much as possible.

An error message “OLAPSYS does not exist” would appear if the OLAP option had been installed, which is not the case for this database. In this case, you would drop

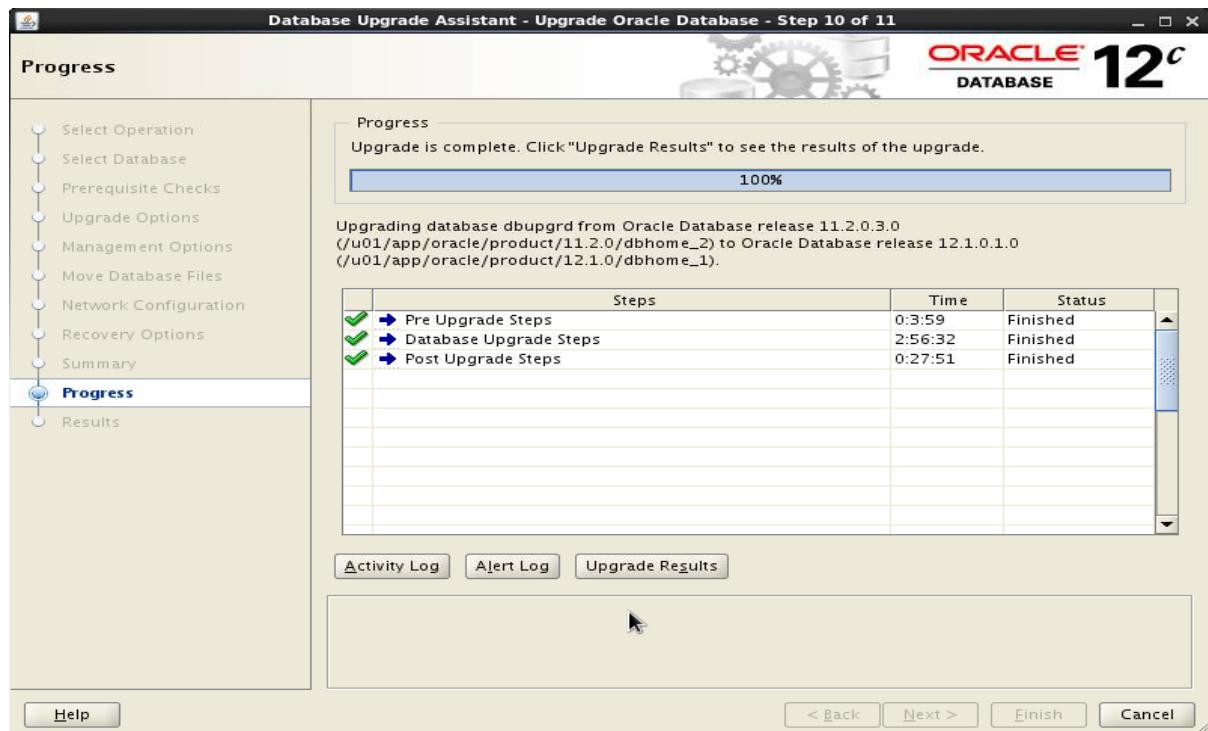
*the OLAP metadata by deleting the OLAPSYS schema (using a recommended script defined in the preupgrd.log). You would then click **Ignore**.*

If you had several options or components installed, there would be other recommendations and actions to take during the pre-upgrade phase. These actions are described in an appendix.

Step	Window/Page Description	Choices or Values
k.	Progress page	You can display the details of the upgrade progress at each step level by expanding each blue arrow.



Step	Window/Page Description	Choices or Values
l.	Progress page	Review the results when the Upgrade Progress is 100% complete. Click Upgrade Results



Step	Window/Page Description	Choices or Values
m.	Results page	<p>Verify the first message is " The database is ready to use."</p> <p><i>The dbupgrd database is now running under version 12.1.0.1.0. All the upgrade steps completed successfully.</i></p> <p>Scroll down to find the Timezone information</p> <p><i>The time zone version and TIMESTAMP WITH TIME ZONE data has been successfully upgraded to version 18.</i></p> <p>Find the SPFILE information</p> <p><i>The SPFILE is created in the DATA disk group.</i></p> <p>Click Close</p>

The screenshot shows the Oracle Database Upgrade Assistant interface at Step 11 of 11. The title bar reads "Database Upgrade Assistant - Upgrade Oracle Database - Step 11 of 11". The main area is titled "Results" and contains a tree view of upgrade steps: Select Operation, Select Database, Prerequisite Checks, Upgrade Options, Management Options, Move Database Files, Network Configuration, Recovery Options, Summary, Progress, and Results. The "Results" node is selected. To the right, there is a section titled "Upgrade Details" which lists the steps performed during the upgrade process, their log file names, and status. A table summarizes the log files and their status:

Step Name	Log File Name	Status
Database Backup	Backup.log	Successful
Pre Upgrade	PreUpgrade.log	Successful
RDBMS Upgrade	Oracle_Server.log	Successful
Post Upgrade	PostUpgrade.log	Successful
Enterprise Manager Configuration	emConfigUpgrade.log	Successful
Timezone Upgrade	UpgradeTimezone.log	Successful
Generate Summary	generateSummary.log	Successful

Below the table, a section titled "Step Execution Information" provides details about the "Database Backup" step, stating it can be restored by running a specific script. A "Restore Database" button is present. At the bottom, there are navigation buttons: Help, Back, Next, Finish, and Close.

The second screenshot shows the same interface after the upgrade has completed. The "Post Upgrade" section indicates that a persistent initialization parameter file (spfile) has been created at a specific location. The "Enterprise Manager Configuration" section states that the database will be managed by Enterprise Manager Database Express. The "Timezone Upgrade" section confirms that the time zone version and TIMESTAMP WITH TIME ZONE data have been successfully upgraded. The "Enterprise Manager" section also states that the database will be managed by Enterprise Manager Database Express. A "Restore Database" button is visible at the bottom.

- n. The upgrade process successfully completed. The log files are located in the /u01/app/oracle/cfgtoollogs/dbua/dbupgrd/upgrade1 directory.

```
$ dbua
```

Database upgrade has been completed successfully, and the database is ready to use.

```
$
```

Practice 9-2: Exporting a Non-CDB Application

Overview

Before you unplug and plug the `orcl` non-CDB into the `cdb1` CDB and therefore make it unusable as a stand-alone database, and in anticipation of another practice, you will export the `SH` schema from the `orcl` non-CDB to later import it into the `cdb1` CDB in another practice.

Assumptions

The `orcl` non-CDB was successfully created in Practice 5-1.

Tasks

- From the oracle user 12c terminal window, export the `SH` schema from the `orcl` non-CDB.

```
$ . oraenv
ORACLE_SID = [orcl] ? orcl
The Oracle base remains unchanged with value /u01/app/oracle
$ sqlplus / as sysdba

Connected to an idle instance.

SQL> startup
ORACLE instance started.

Total System Global Area  668082176 bytes
Fixed Size                  2291328 bytes
Variable Size                499124608 bytes
Database Buffers            159383552 bytes
Redo Buffers                 7282688 bytes
Database mounted.
Database opened.

SQL> exit
$
```

```
$ rm /u01/app/oracle/admin/orcl/dpdump/expSH.dmp
rm: cannot remove `/u01/app/oracle/admin/orcl/dpdump/expSH.dmp':
No such file or directory
$ expdp system/oracle_4U DUMPFILE=expSH.dmp SCHEMAS=SH
LOGFILE=expSH.log

Export: Release 12.1.0.1.0 - Production on Thu Feb 21 02:38:54
2013
```

```
Copyright (c) 1982, 2013, Oracle and/or its affiliates. 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
Starting "SYSTEM"."SYS_EXPORT_SCHEMA_01": system/**********
DUMPFILE=expSH.dmp SCHEMAS=SH LOGFILE=expSH.log
Estimate in progress using BLOCKS method...
Processing object type SCHEMA_EXPORT/TABLE/TABLE_DATA
...
. . . exported "SH"."CUSTOMERS"                                10.27
MB      55500 rows
. . . exported "SH"."COSTS": "COSTS_Q1_1998"                  139.6
KB      4411 rows
...
. . . exported "SH"."SALES": "SALES_Q4_2001"                  2.257
MB      69749 rows
...
. . . exported "SH"."SALES": "SALES_Q4_2003"                  0
KB          0 rows
Master table "SYSTEM"."SYS_EXPORT_SCHEMA_01" successfully
loaded/unloaded
*****
*
Dump file set for SYSTEM.SYS_EXPORT_SCHEMA_01 is:
/u01/app/oracle/admin/orcl/dpdump/expSH.dmp
Job "SYSTEM"."SYS_EXPORT_SCHEMA_01" successfully completed at
Thu Feb 21 02:41:20 2013 elapsed 0 00:02:17

$
```

2. Copy the /u01/app/oracle/admin/orcl/dpdump/expSH.dmp dump file temporarily to the /home/oracle/labs directory.

```
$ cp /u01/app/oracle/admin/orcl/dpdump/expSH.dmp
/home/oracle/labs
$ ls /home/oracle/labs/*SH*
/home/oracle/labs/expSH.dmp
$
```

Practice 9-3: Plugging a Non-CDB into a CDB

Overview

In this practice, you will plug the `orcl` non-CDB into the `cdb1` CDB. You will not use the Data Pump Export/Import method, but instead use the unplugging / plugging method using the `DBMS_PDB` package. This method has the advantages of being fast and easy. When this package is executed in the `orcl` non-CDB, it generates an XML file describing the tablespaces and data files of the `orcl` non-CDB. The XML file is then used to create a new PDB in the `cdb1` CDB. In this exercise, because the `orcl` datafiles are not copied before they are used to plug into a CDB, the `orcl` non-CDB is no longer a valid non-CDB.

Assumptions

The `cdb1` CDB was successfully created in Practice 5-2.

Tasks

1. Unplug the `orcl` non-CDB.

- a. Open the database instance in READ ONLY mode.

```
$ . oraenv
ORACLE_SID = [orcl] ? orcl
The Oracle base remains unchanged with value /u01/app/oracle
$ sqlplus / as sysdba

SQL> shutdown immediate
Database closed.
Database dismounted.
ORACLE instance shut down.
SQL> startup mount
Total System Global Area  668082176 bytes
Fixed Size                  2291328 bytes
```

```
Variable Size            499124608 bytes
Database Buffers        159383552 bytes
Redo Buffers             7282688 bytes
Database mounted.
SQL> alter database open read only;

Database altered.

SQL>
```

- b. Use the `DBMS_PDB.DESCRIBE` package to “unplug” the `orcl` non-CDB.

```
SQL> exec dbms_pdb.describe
('/u01/app/oracle/oradata/orcl/xmlorcl.xml')

PL/SQL procedure successfully completed.
```

```
SQL> shutdown immediate
Database closed.
Database dismounted.
ORACLE instance shut down.
SQL> exit
$
```

2. Create a new PDB, named `pdb_orcl` to plug the `orcl` non-CDB into `cdb1` CDB using the XML file generated. The files described in the XML file are used to create the new PDB. The new PDB will consist of the `SYSTEM` and `SYSAUX` tablespaces, and the user-defined tablespaces of the `orcl` non-CDB. The `NOCOPY` clause enables a very fast operation because the data files of the `orcl` non-CDB are directly used where they are to create the new PDB.

You will have to remove the temp files.

```
$ . oraenv
ORACLE_SID = [orcl] ? cdb1
The Oracle base remains unchanged with value /u01/app/oracle
$ sqlplus / as sysdba

Connected to an idle instance.

SQL> startup
ORACLE instance started.

Total System Global Area  881037312 bytes
Fixed Size                  2293784 bytes
Variable Size                578818024 bytes
Database Buffers            293601280 bytes
Redo Buffers                 6324224 bytes
Database mounted.
Database opened.

SQL>
SQL> create pluggable database PDB_ORCL using
'/u01/app/oracle/oradata/orcl/xmlorcl.xml' NOCOPY;
create pluggable database PDB_ORCL using
'/u01/app/oracle/oradata/orcl/xmlorcl' NOCOPY
*
ERROR at line 1:
ORA-27038: created file already exists
ORA-01119: error in creating database file
'/u01/app/oracle/oradata/orcl/temp01.dbf'

SQL> !rm /u01/app/oracle/oradata/orcl/temp01.dbf
```

```
SQL> create pluggable database PDB_ORCL using
'./u01/app/oracle/oradata/orcl/xmlorcl.xml' NOCOPY;

Pluggable database created.

SQL> exit
$
```

3. Create a net service name, `PDB_ORCL` to connect easily to the new PDB. Use NETMGR to add the `PDB_ORCL` net service name for `pdb_orcl` PDB of `cdb1` to the `tnsnames.ora` file.

```
$ $ORACLE_HOME/bin/netmgr
```

Step	Window/Page Description	Choices or Values
a.	Oracle Network Manager - /u01/app/oracle/product/12.1.0/dbhome_1/network/admin/	Expand Local . Select Service Naming . Click Create (green "+" icon).
b.	Net Service Name Wizard: Welcome	Enter PDB_ORCL as Net Service Name. Click Next .
c.	Net Service Name Wizard: page 2 of 5 -Protocol	Select TCP/IP (Internet Protocol). Click Next .
d.	Net Service Name Wizard: page 3 of 5 –Protocol Settings	Enter the following information: Host : <Your hostname> or LOCALHOST Port: 1521 Click Next .
e.	Net Service Name Wizard: page 4 of 5 –Service	Enter the following information: Service Name: pdb_orcl Click Next .
f.	Net Service Name Wizard: page 5 of 5 –Test	Click Finish
g.	Oracle Network Manager - /u01/app/oracle/product/12.1.0/dbhome_1/network/admin	From the menu, select File > Save Network Configuration . Select File > Exit .

4. To complete the operation, you have to convert the plugged non-CDB to a proper PDB by deleting unnecessary metadata from the SYSTEM tablespace of the PDB.

For this purpose, you execute the \$ORACLE_HOME/rdbms/admin/noncdb_to_pdb.sql script while you are connected to the PDB. The execution may last more than 30 minutes.

- a. Now connect to pdb_orcl as SYSDBA using the net service name.

```
$ sqlplus sys/oracle_4U@pdb_orcl as sysdba

SQL> SELECT con_id, name, open_mode FROM v$pdbs;

  CON_ID NAME          OPEN_MODE
----- -----
        4 PDB_ORCL      MOUNTED

SQL>
```

- b. List the tablespaces and data files of the new PDB.

```
SQL> select name from dba_tablespaces;
select name from dba_tablespaces
*
ERROR at line 1:
ORA-01219: database or pluggable database not open: queries
allowed on fixed
tables or views only

SQL>
```

The PDB is not yet opened.

- c. First execute the \$ORACLE_HOME/rdbms/admin/noncdb_to_pdb.sql script. It will take approximately 30 minutes to complete.

While the conversion is taking place, proceed with the next practice to perform the post-upgrade actions on the dbupgrd database instance.

```
SQL> @$ORACLE_HOME/rdbms/admin/noncdb_to_pdb.sql
...
SQL> DOC
DOC>#####
DOC> The following statement will cause an "ORA-01722: invalid
number"
DOC>   error if we're not in a PDB.
DOC>#####
SQL>
SQL> VARIABLE pdbname VARCHAR2(128)
SQL> BEGIN
 2   SELECT sys_context('USERENV', 'CON_NAME')
 3   INTO :pdbname
```

```
4      FROM dual
5      WHERE sys_context('USERENV', 'CON_NAME') <> 'CDB$ROOT';
6  END;
7 /  
  
PL/SQL procedure successfully completed.  
...  
SQL>  
=====SQL> Rem Run component validation procedure  
SQL> Rem  
=====...  
SQL> alter session set container="&pdbname";  
  
Session altered.  
  
SQL>  
SQL> -- leave the PDB in the same state it was when we started  
SQL> BEGIN
2   execute immediate '&open_sql &restricted_state';
3 EXCEPTION
4   WHEN OTHERS THEN
5   BEGIN
6     IF (sqlcode <> -900) THEN
7       RAISE;
8     END IF;
9   END;
10 END;
11 /  
  
PL/SQL procedure successfully completed.  
  
SQL>  
SQL> WHENEVER SQLERROR CONTINUE;  
SQL>  
SQL>
```

- d. When the conversion is completed, open the PDB and quit the session.

```
SQL> alter pluggable database pdb_orcl open;  
  
Pluggable database altered.  
  
SQL> EXIT
```

5. Connect to PDB_ORCL.

```
$ sqlplus sys/oracle_4U@PDB_ORCL as SYSDBA  
SQL>
```

6. Verify that all application data is in the pdb_orcl PDB.

```
SQL> select tablespace_name from dba tablespaces;  
  
TABLESPACE_NAME  
-----  
SYSTEM  
SYSAUX  
TEMP  
USERS  
EXAMPLE  
  
SQL> select name from v$datafile;  
  
NAME  
-----  
/u01/app/oracle/oradata/cdb1/undotbs01.dbf  
/u01/app/oracle/oradata/orcl/system01.dbf  
/u01/app/oracle/oradata/orcl/sysaux01.dbf  
/u01/app/oracle/oradata/orcl/users01.dbf  
/u01/app/oracle/oradata/orcl/example01.dbf  
  
SQL> select count(empno) from scott.emp;  
  
COUNT(EMPNO)  
-----  
14  
  
SQL> select count(*) from hr.employees;  
  
COUNT(*)  
-----  
107  
  
SQL> EXIT  
$
```

Practices for Lesson 10: Performing Post-Upgrade Tasks

Chapter 10

Practices for Lesson 10

Practices Overview

In the previous practice, you upgraded the Oracle Database 11g dbupgrd instance to Oracle Database12c by using DBUA.

In these practices, you will complete the upgrade process of the dbupgrd database instance currently running in 12c by:

- Performing post-upgrade actions
- Enabling and using Unified Auditing

You are going to work as the `oracle` user for some operations and as the `grid` user for other operations. Therefore, keep a terminal window opened as the `oracle` user and another terminal window opened as the `grid` user.

Practice 10-1: Performing Post-Upgrade Actions

Overview

In this practice, you will:

- Verify that the `/etc/oratab` file sets the value of `ORACLE_HOME` to point to the new Oracle home that is created for the new Oracle Database 12c release.
- Show the current state of the Oracle data dictionary
- Check that there are no invalid objects
- Check that the password file exists
- Verify that the SPFILE is created and create the PFILE
- View the contents of the `alert.log` file
- Launch Enterprise Manager Database Express
- Unlock the application users, if any are locked
- Upgrade optimizer statistics
- Perform a FULL database backup

Tasks

1. From the `oracle user 12c` terminal window, view the `/etc/oratab` file.

```
$ cat /etc/oratab|grep dbupgrd
dbupgrd:/u01/app/oracle/product/12.1.0/dbhome_1:N:          #
line added by Agent
$
```

2. Show the current state of the Oracle data dictionary. The status of all components must be VALID.

```
$ . oraenv
ORACLE_SID = [cdb1] ? dbupgrd
The Oracle base remains unchanged with value /u01/app/oracle

$ sqlplus / as sysdba

SQL> set lines 80 pages 100
SQL> col comp_id format A12
SQL> col comp_name format A30
SQL> col version format A10
SQL> col status format A6
SQL> select substr(comp_id,1,15) comp_id, substr(comp_name,1,30)
       comp_name, substr(version,1,10) version, status
  from dba_registry order by modified;

COMP_ID    COMP_NAME                      VERSION      STATUS
-----  -----
CATALOG    Oracle Database Catalog Views  12.1.0.1.0  VALID
```

```
CATPROC  Oracle Database Packages and T 12.1.0.1.0  VALID
OWM      Oracle Workspace Manager          12.1.0.1.0  VALID
XDB      Oracle XML Database             12.1.0.1.0  VALID

SQL>
```

3. Check that there are no invalid objects.

```
SQL> @$ORACLE_HOME/rdbms/admin/utluiobj.sql
.
.
Oracle Database 12.1 Post-Upgrade Invalid Objects Tool 02-17-
2013 20:44:52
.

This tool lists post-upgrade invalid objects that were not
invalid prior to upgrade (it ignores pre-existing pre-upgrade
invalid objects).

.

Owner           Object Name           Object Type
.

PL/SQL procedure successfully completed.

SQL> select * from REGISTRY$sys_inv_objs;
no rows selected

SQL> select * from REGISTRY$nonsys_inv_objs;
no rows selected

SQL>
```

4. Check that the password file exists.

- a. Check the existence of a traditional password file in the \$ORACLE_HOME/dbs directory.

```
SQL> !ls $ORACLE_HOME/dbs/orapwdupgrd
/u01/app/oracle/product/12.1.0/dbhome_1/dbs/orapwdupgrd

SQL>
```

- b. Check that the password file is used during an authentication.

```
SQL> connect sys/oracle_4U@localhost:1521/dbupgrd as sysdba
Connected.

SQL> connect sys/oracle@localhost:1521/dbupgrd as sysdba
ERROR:
ORA-01017: invalid username/password; logon denied
```

```
Warning: You are no longer connected to ORACLE.  
SQL>
```

- c. Check that after you remove the password file, the only possible authentication is the OS authentication.
 - Remove the file system password file.

```
SQL> !rm $ORACLE_HOME/dbs/orapwdupgrd  
  
SQL> !ls $ORACLE_HOME/dbs/orapwdupgrd  
ls: cannot access  
/u01/app/oracle/product/12.1.0/dbhome_1/dbs/orapwdupgrd: No  
such file or directory  
  
SQL>
```

- Even if you use the password that you used in task b., the authentication cannot be performed.

```
SQL> connect sys/oracle_4U@localhost:1521/dbupgrd as sysdba  
ERROR:  
ORA-01017: invalid username/password; logon denied  
  
SQL> connect / as sysdba  
Connected.  
SQL>
```

- d. If your storage solution is ASM, you can create the password file in the DATA disk group. It is only possible if you advance the COMPATIBLE.ASM disk group attribute to 12.1. Be aware that this is not mandatory but any file stored in a file system can be easily deleted with proper access permissions. Enter oracle_4U when prompted for the password.
 - 1) Update disk group compatibility attributes first.
 - a) The COMPATIBLE.ASM, COMPATIBLE.RDBMS and COMPATIBLE.ADVM disk group attributes determine compatibility. The COMPATIBLE.ASM and COMPATIBLE.RDBMS attribute settings determine the minimum Oracle Database software version numbers that a system can use for Oracle ASM and the database instance types respectively. For example, if the Oracle ASM compatibility setting is 11.2, and RDBMS compatibility is set to 11.1, then the Oracle ASM software version must be at least 11.2, and the Oracle Database client software version must be at least 11.1.
The disk group compatibility settings determine whether your environment can use the latest Oracle ASM features.
The COMPATIBLE.ASM attribute must be advanced before advancing other disk group compatibility attributes and its value must be greater than or equal to the value of other disk group compatibility attributes. This setting also affects the format of the data structures for the Oracle ASM metadata on the disk.

From the grid user terminal window session, use SQL*Plus to advance the COMPATIBLE.ASM attribute of the DATA disk group to 12.1.

```
$ sqlplus / as sysasm

SQL> ALTER DISKGROUP data SET ATTRIBUTE 'compatible.asm'='12.1';

Diskgroup altered.

SQL> col compatibility format A16
SQL> select name, compatibility from v$asm_diskgroup;

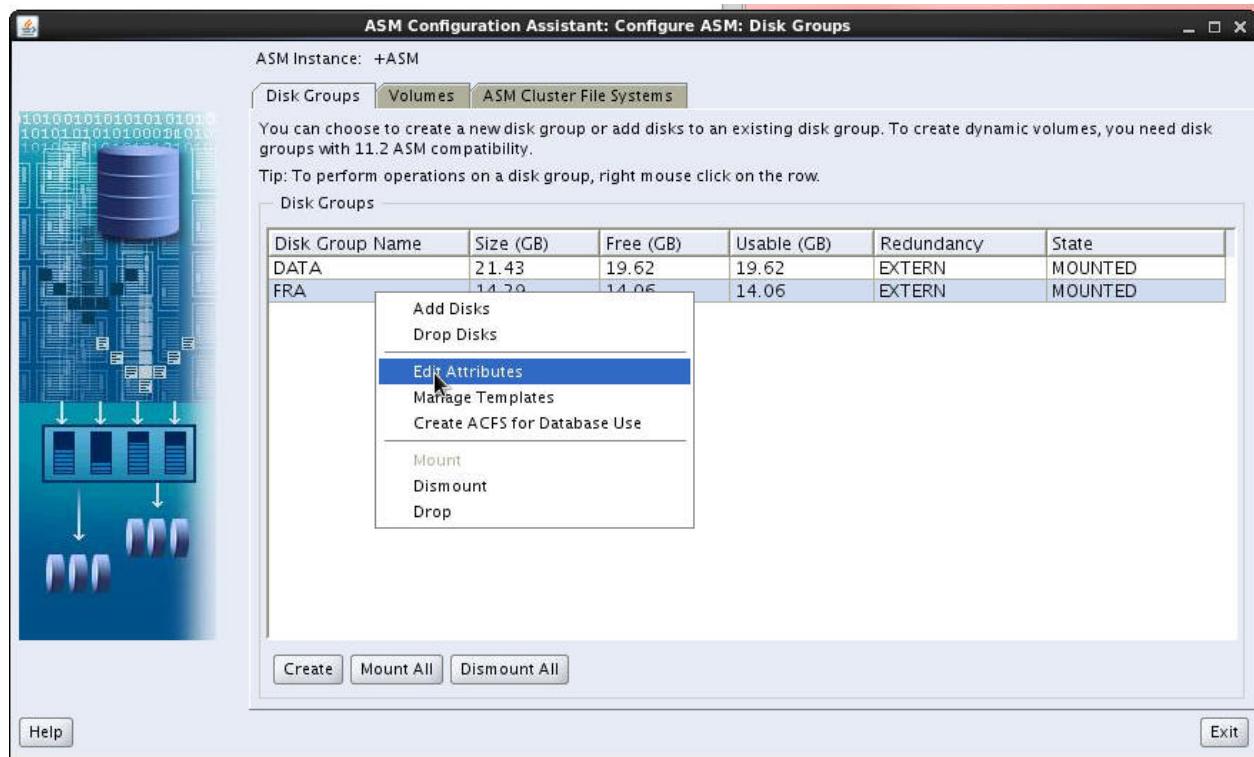
NAME          COMPATIBILITY
-----
FRA           11.2.0.0.0
DATA          12.1.0.0.0

SQL> exit
$
```

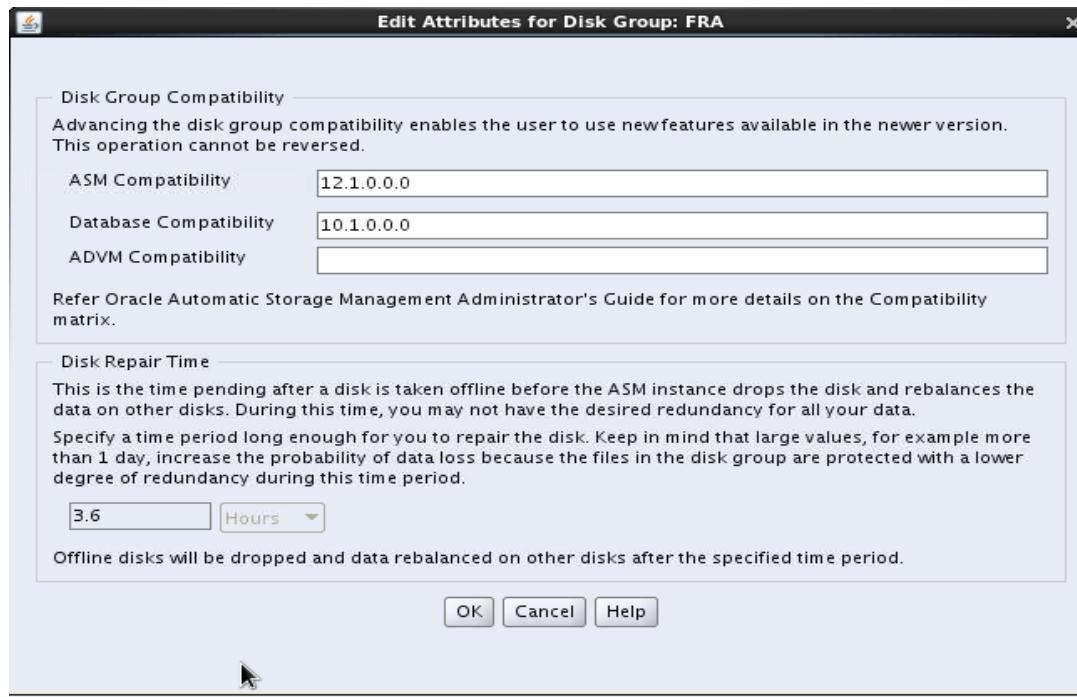
- b) Use ASMCA to advance the COMPATIBLE.ASM attribute of the FRA disk group to 12.1.

```
$ asmca
```

Step	Window/Page Description	Choices or Values
1	Configure ASM: Disk Groups	Right-click the Disk Group Name FRA Click the Edit Attributes



Step	Window/Page Description	Choices or Values
2	Edit Attributes for Disk Group: FRA Disk Group Compatibility section	Set the ASM Compatibility = 12.1.0.0.0 . Click OK



Step	Window/Page Description	Choices or Values
3	Configure ASM: Disk Groups	Click Exit
4	Configure ASM: Disk Groups	Confirm that you want to quit the ASMCA application Click Yes

c) Verify the setting by using the `V$ASM_DISKGROUP` view.

```
$ sqlplus / as sysasm

SQL> col compatibility format A16
SQL> select name, compatibility from v$asm_diskgroup;

NAME          COMPATIBILITY
-----
FRA           12.1.0.0.0
DATA          12.1.0.0.0

SQL> EXIT
$
```

2) From the **oracle 12c user terminal window session**, create the DBUPGRD password file in the DATA disk group.

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

DB_UNIQUE_NAME
-----
dbupgrd

SQL> EXIT
$ 
$ orapwd file='+data/DBUPGRD/orapwdb' dbuniqueusername='dbupgrd'

Enter password for SYS:

$ sqlplus sys/oracle_4U@localhost:1521/dbupgrd as sysdba
Connected.
SQL> connect sys/oracle@localhost:1521/dbupgrd as sysdba
ERROR:
ORA-01017: invalid username/password; logon denied

Warning: You are no longer connected to ORACLE.
SQL> EXIT
```

```
$
```

- 3) From the grid user terminal window, verify that the password file is created in the DATA disk group.

```
$ asmcmd
ASMCMD> ls +DATA/DBUPGRD/PASSWORD
pwddbugrd.269.814286779
ASMCMD>
```

5. Verify that the SPFILE is created.

```
ASMCMD> ls +DATA/DBUPGRD/spfile*
spfiledbupgrd.ora
spfiledbupgrd.ora_1361856137518
ASMCMD>
ASMCMD> ls -l +DATA/DBUPGRD/spfile*
Type          Redund  Striped   Time           Sys  Name
PARAMETERFILE UNPROT  COARSE    FEB 25 20:00:00 N
spfiledbupgrd.ora =>
+DATA/DBUPGRD/PARAMETERFILE/spfile.260.808327751
PARAMETERFILE UNPROT  COARSE    FEB 25 22:00:00 N
spfiledbupgrd.ora_1361856137518 =>
+DATA/DBUPGRD/PARAMETERFILE/spfile.267.808348937
ASMCMD> exit
$
```

6. From the oracle user 12c terminal window, create the PFILE if it does not exist.

```
$ ls $ORACLE_HOME/dbs/initdbupgrd.ora
/u01/app/oracle/product/12.1.0/dbhome_1/dbs/initdbupgrd.ora
$
$ cat $ORACLE_HOME/dbs/initdbupgrd.ora
SPFILE='+DATA/dbupgrd/spfiledbupgrd.ora_1361856137518'
$
```

It already exists. The parameter file references the most recent SPFILE file stored in the DATA disk group.

7. View the contents of the alert log file

```
$ cd $ORACLE_BASE/diag/rdbms/dbupgrd/dbupgrd/trace
$ ls a*
alert_dbupgrd.log
$ gedit alert_dbupgrd.log

...
Control file expanded from 594 to 614 blocks for upgrade.
Successful mount of redo thread 1, with mount id 2071604359
Database mounted in Exclusive Mode
```

```
Lost write protection disabled
Ping without log force is disabled.
Completed: ALTER DATABASE MOUNT
Fri Feb 15 00:55:36 2013
ALTER DATABASE OPEN MIGRATE
...
Completed: ALTER DATABASE OPEN MIGRATE
...
$
```

From now on, if you encounter any disconnection from the database, reconnect and continue with your administrative operations.

8. Launch Enterprise Manager Database Express.
 - a. Configure the port number.

```
$ sqlplus / as sysdba

SQL> select DBMS_XDB_CONFIG.getHTTPPort from dual;

GETHTTPPORT
-----
0

SQL> select DBMS_XDB_CONFIG.getHTTPsPort from dual;

GETHTTPSPORT
-----
5500

SQL>
```

- The port number is already being used by the CDB1 instance.
- b. Verify which port is being used by the CDB1 instance. Switch to the oracle user terminal window where the PDB_ORCL plugging is taking place. Wait until it is completed.

```
$ . oraenv
ORACLE_SID = [oracle] ? cdb1
The oracle base has been set to  /u01/app/oracle
$
```

Connect to the pdb_orcl PDB to retrieve the port number used.

```
$ sqlplus sys/oracle_4U@localhost:1521/pdb_orcl as sysdba

SQL> select DBMS_XDB_CONFIG.getHTTPPort from dual;

GETHTTPPORT
```

```

-----
0

SQL> select DBMS_XDB_CONFIG.getHTTPsPort from dual;

GETHTTPSPORT
-----
5500

SQL> exit
$
```

- c. Configure the port number 5502 for Enterprise Manager Database Express for the dbupgrd instance. Switch back the dbupgrd session.
- 1) Check the value of the DISPATCHERS instance parameter in the dbupgrd instance. There must be at least one dispatcher configured for the XMLDB service with the TCP protocol.

Session 1

SQL> show parameter dispatchers		
NAME	TYPE	VALUE
-		
dispatchers	string	
max_dispatchers	integer	

```
SQL>
```

- 2) Set the DISPATCHERS instance parameter.

```

SQL> alter system set
dispatchers='(protocol=tcp)(service=dbupgrdXDB)' scope=both;

System altered.

SQL>
```

- 3) Set the port number to 5502 for EM Database Express.

```

SQL> exec DBMS_XDB_CONFIG.setHTTPsPort(5502)

PL/SQL procedure successfully completed.

SQL> select DBMS_XDB_CONFIG.getHTTPsPort from dual;

GETHTTPSPORT
-----
```

```
5502
```

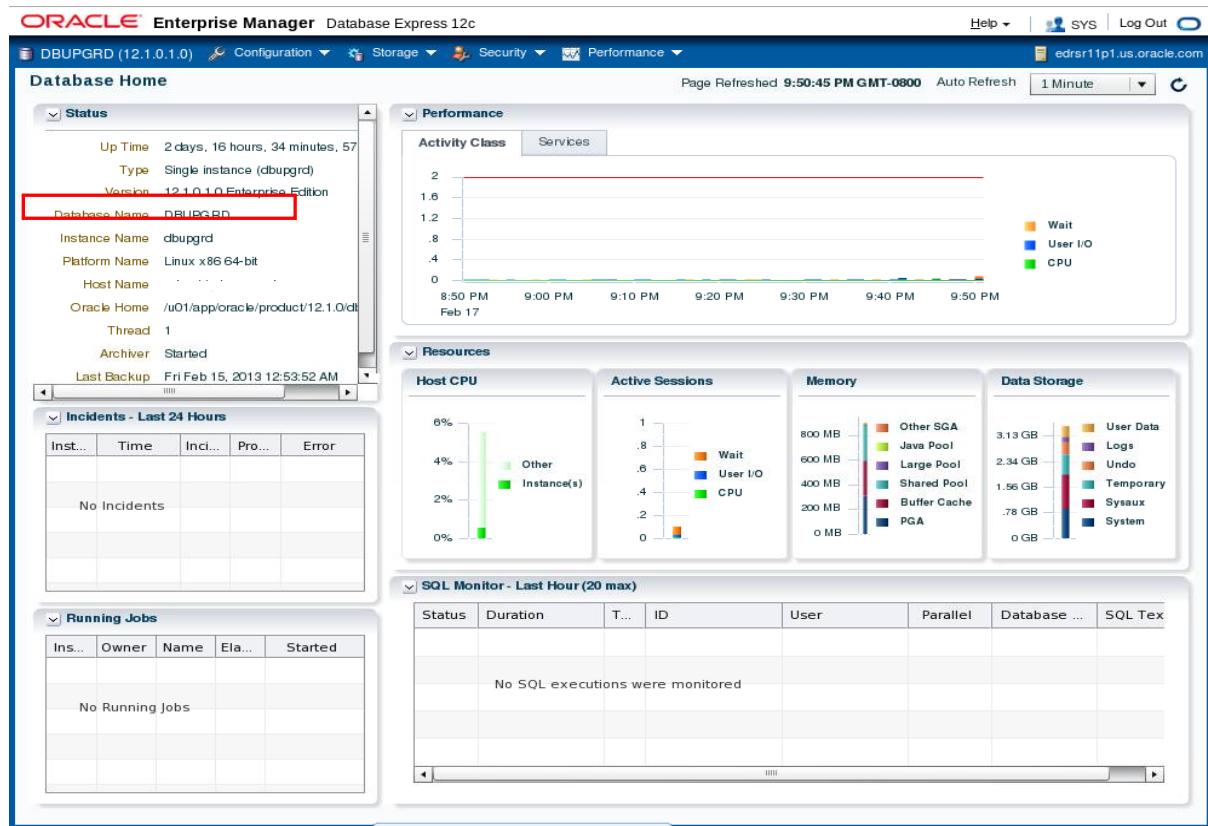
```
SQL>
```

- 4) Restart the instance to register the XMLDB service.

```
SQL> shutdown immediate
Database closed.
Database dismounted.
ORACLE instance shut down.
SQL> startup
ORACLE instance started.

Total System Global Area  663908352 bytes
Fixed Size                  2291280 bytes
Variable Size                285215152 bytes
Database Buffers            369098752 bytes
Redo Buffers                 7303168 bytes
Database mounted.
Database opened.
SQL> exit
$
```

- d. Click the Firefox icon on the top panel (toolbar region) above the desktop to open a browser to access the Enterprise Manager Database Express console.
- e. Enter the URL for Enterprise Manager Database Express:
<https://<hostname>:5502/em>. In the current setup, use
<https://localhost:5502/em>.
- Most likely, a security exception (untrusted connection) is noticed.
 - Enter **sys** in the User Name field and **oracle_4U** in the Password field. Select as SYSDBA. Then click **Login**.



9. From the oracle user 12c terminal window, verify that the SH and HR application users are unlocked.

```
$ sqlplus / as sysdba
...
SQL> set pages 100
SQL> col username format a30
SQL> select username , account_status from dba_users order by 1;

USERNAME                      ACCOUNT_STATUS
-----
ANONYMOUS                     EXPIRED & LOCKED
APPQOSSYS                     EXPIRED & LOCKED
AUDSYS                        EXPIRED & LOCKED
DBSNMP                         OPEN
DIP                            EXPIRED & LOCKED
GSMADMIN_INTERNAL               EXPIRED & LOCKED
GSMCATUSER                     EXPIRED & LOCKED
GSMUSER                        EXPIRED & LOCKED
HR                            OPEN
ORACLE_OCM                     EXPIRED & LOCKED
OUTLN                          EXPIRED & LOCKED
SH                            OPEN
```

```

SYS                      OPEN
SYSBACKUP                EXPIRED & LOCKED
SYSDG                     EXPIRED & LOCKED
SYSKM                     EXPIRED & LOCKED
SYSTEM                    OPEN
WMSYS                     EXPIRED & LOCKED
XDB                       EXPIRED & LOCKED
XS$NULL                   EXPIRED & LOCKED

```

20 rows selected.

SQL>

10. Upgrade optimizer statistics. In Practice 8-2, the last suggestion in /u01/app/oracle/cfgtoollogs/dbupgrd/preupgrade/preupgrade.log was to create statistics on fixed objects two weeks after the upgrade. You can do it now or leave it until the end of the course. In the normal course of database performance management, you will repeat the procedure periodically as well as the `gather_dictionary_stats` procedure to update statistics on data dictionary objects. This procedure takes some time to execute. You can skip it if you are short of time.

```
SQL> exec dbms_stats.gather_fixed_objects_stats
```

PL/SQL procedure successfully completed.

SQL> EXIT

\$

11. After upgrade, perform a FULL database backup using RMAN.

```
$ rman target /
```

```
RMAN> show all;
```

```
using target database control file instead of recovery catalog
RMAN configuration parameters for database with db_unique_name
DBUPGRD are:
```

```
CONFIGURE RETENTION POLICY TO REDUNDANCY 1; # default
CONFIGURE BACKUP OPTIMIZATION OFF; # default
CONFIGURE DEFAULT DEVICE TYPE TO DISK; # default
CONFIGURE CONTROLFILE AUTOBACKUP ON;
```

...

```
RMAN> backup database plus archivelog delete all input;
```

Starting backup at 26-FEB-13

current log archived

```

allocated channel: ORA_DISK_1
channel ORA_DISK_1: SID=263 device type=DISK
channel ORA_DISK_1: starting archived log backup set
channel ORA_DISK_1: specifying archived log(s) in backup set
input archived log thread=1 sequence=35 RECID=1 STAMP=808345285
...
Finished backup at 26-FEB-13

Starting Control File and SPFILE Autobackup at 26-FEB-13
piece
handle=+FRA/DBUPGRD/AUTOBACKUP/2013_02_26/s_808370053.315.808370
057 comment=NONE
Finished Control File and SPFILE Autobackup at 26-FEB-13

RMAN> EXIT
$
```

12. Now that you set the COMPATIBLE.ASM disk group attribute to 12.1 for the disk group DATA, you can create the ASM password file in the disk group if it is not already created. Enter oracle_4U when the password is prompted.

- a. From the grid user terminal window session, check if a password file exists.

```
$ ls $ORACLE_HOME/dbs
ab+_ASM.dat  bkup+ASM.ora  hc+_ASM.dat  init.ora
$
```

- b. Create the same password file in the DATA disk group.

```
$ orapwd file='+data/ASM/orapwdasm' asm=y

Enter password for SYS:
$ asmcmd
ASMCMD> ls -l +DATA/ASM
Type      Redund  Striped  Time          Sys  Name
          Y
ASMPARAMETERFILE/
          Y      PASSWORD/
PASSWORD  UNPROT  COARSE   FEB 17 19:00:00 N  orapwdasm =>
+DATA/ASM/PASSWORD/pwdasm.268.808291301
ASMCMD> exit
$
```


Practices for Lesson 11: Migrating Data by Using Oracle Data Pump

Chapter 11

Practices for Lesson 11

Practices Overview

In these practices, you will use another method for migrating data from an Oracle Database 11g database to an Oracle Database 12c database. Using Oracle Data Pump:

- Perform a transportable tablespace import from the Oracle Database 11g dbupgrd database into another Oracle Database 12c database
- Import the SH schema from the `orcl` non-CDB into the `pdb_sh` PDB of the `cdb1` CDB

You are going to work as the `oracle` user to perform operations in the 12c environment only. Therefore, keep a terminal window opened as the `oracle` user in the 12c environment.

Practice 11-1: Performing a Transportable Tablespace Import

Overview

In this practice, you will perform a transportable tablespace import from the Oracle Database 11g dbupgrd database into another Oracle Database 12c database. In order to avoid the creation of another database, you will use the existing Oracle Database 12c dbupgrd database.

Assumptions

- You successfully installed the Oracle 12c Database software in Practice 5-1.
- In Practice 8-5, the transportable tablespace export from the dbupgrd database before the database was upgraded successfully completed, exporting the EXAMPLE and USERS tablespaces.

Tasks

1. From the oracle user 12c terminal window, configure the target dbupgrd database as if it were a brand new empty database to be the recipient for the migrated data from the source Oracle 11g Database. If you encounter any disconnection from the database, reconnect and continue with your administrative operations.
 - a. Set your environment to the target database dbupgrd.

```
$ . oraenv
ORACLE_SID = [dbupgrd] ? dbupgrd
The Oracle base remains unchanged with value /u01/app/oracle
$ sqlplus / as sysdba

SQL>
```

- b. Drop the application schemas and tablespaces.

```
SQL> SELECT tablespace_name FROM dba_tablespaces;

TABLESPACE_NAME
-----
SYSTEM
SYSAUX
UNDOTBS1
TEMP
USERS
EXAMPLE

6 rows selected.

SQL> drop user hr cascade;

User dropped.
```

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

```
SQL> drop user sh cascade;

User dropped.

SQL> drop tablespace EXAMPLE including contents and datafiles;

Tablespace dropped.

SQL> drop tablespace USERS including contents and datafiles;

drop tablespace USERS including contents and datafiles
*
ERROR at line 1:
ORA-12919: Can not drop the default permanent tablespace

SQL> alter database default tablespace system;

Database altered.

SQL> drop tablespace USERS including contents and datafiles;

Tablespace dropped.

SQL> SELECT tablespace_name FROM dba_tablespaces ORDER BY 1;

TABLESPACE_NAME
-----
SYSAUX
SYSTEM
TEMP
UNDOTBS1

SQL>
```

- c. Create the schemas in the target database instance before importing data.

```
SQL> create user HR identified by oracle_4U account unlock;

User created.

SQL> create user SH identified by oracle_4U account unlock;

User created.
```

```
SQL> grant create session, create table, create materialized
view, unlimited tablespace to HR, SH;
```

Grant succeeded.

```
SQL> EXIT
$
```

- At the end of the Practice 8-5, you moved the dump file to the /home/oracle/labs/dbupgrd12 directory and copied the data files of the 11g dbupgrd database to the /u01/app/grid/exp_files directory.

```
$ ls /home/oracle/labs/dbupgrd12
expTTS.dmp
$
```

- Move the dump file to the default directory for Data Pump in the dbupgrd database instance.

```
$ mv /home/oracle/labs/dbupgrd12/expTTS.dmp
/u01/app/oracle/admin/dbupgrd/dpdump
$
```

- From the grid user terminal window, move the data files from the FRA disk group back to the DATA disk group as stipulated at the end of the Data Pump export (Practice 8-5).

```
$ asmcmd
ASMCMD> ls +DATA/DBUPGRD/DATAFILE
SYSAUX.261.814012959
SYSTEM.260.814012925
UNDOTBS1.262.814012987
ASMCMD> cp +FRA/EXAMPLE +DATA/DBUPGRD/DATAFILE/example01.dbf
copying +FRA/EXAMPLE -> +DATA/DBUPGRD/DATAFILE/example01.dbf
ASMCMD> cp +FRA/USERS +DATA/DBUPGRD/DATAFILE/users01.dbf
copying +FRA/USERS -> +DATA/DBUPGRD/DATAFILE/users01.dbf
ASMCMD> exit
$
```

- From the oracle user terminal window, use the dump file and data files to import the transportable tablespaces from the dbupgrd database into the target database. To make the import quicker, you will import only the HR and SH schemas.

Note: Remove the named log files, if they exist the import will fail.

```
$ rm /u01/app/oracle/admin/dbupgrd/dpdump/import.log
rm: cannot remove
`/u01/app/oracle/admin/dbupgrd/dpdump/import.log': No such file
or directory
$ impdp system/oracle_4U dumpfile=expTTS.dmp
TRANSPORT_DATAFILES='+DATA/DBUPGRD/DATAFILE/users01.dbf',
'+DATA/DBUPGRD/DATAFILE/example01.dbf' logfile=import.log
```

```
Import: Release 12.1.0.1.0 - Production on Thu Feb 21 17:37:21
2013
...
Master table "SYSTEM"."SYS_IMPORT_TRANSPORTABLE_01" successfully
loaded/unloaded
Source TSTZ version is 14 and target TSTZ version is 18.
Starting "SYSTEM"."SYS_IMPORT_TRANSPORTABLE_01":
system/********* dumpfile=expTTS.dmp
TRANSPORT_DATAFILES=+DATA/DBUPGRD/DATAFILE/users01.dbf,
+DATA/DBUPGRD/DATAFILE/example01.dbf logfile=import.log
Processing object type TRANSPORTABLE_EXPORT/PLUGTS_BLK
Processing object type TRANSPORTABLE_EXPORT/TABLE
Processing object type
TRANSPORTABLE_EXPORT/POST_INSTANCE/PLUGTS_BLK
Job "SYSTEM"."SYS_IMPORT_TRANSPORTABLE_01" successfully
completed at Tue Feb 26 18:30:40 2013 elapsed 0 00:00:21
$
```

6. Check in the target database that the EXAMPLE and USERS tablespaces have been plugged, and that the HR.EMPLOYEES table and the SH.SALES table contain the rows as in the source dbupgrd database.

```
$ sqlplus / as sysdba

SQL> SELECT tablespace_name FROM dba_tablespaces;

TABLESPACE_NAME
-----
SYSTEM
SYSAUX
UNDOTBS1
TEMP
USERS
EXAMPLE

6 rows selected.

SQL> SELECT name FROM v$datafile;

NAME
-----
-
+DATA/dbupgrd/datafile/system.263.808326819
+DATA/dbupgrd/datafile/sysaux.264.808326845
+DATA/dbupgrd/datafile/undotbs1.269.808326865
```

```
+DATA/DBUPGRD/DATAFILE/example01.dbf  
+DATA/DBUPGRD/DATAFILE/users01.dbf  
  
SQL> SELECT count(*) FROM HR.employees;  
  
COUNT (*)  
-----  
7  
  
SQL> SELECT count(*) FROM SH.sales;  
  
COUNT (*)  
-----  
5  
  
SQL>
```

7. Set the EXAMPLE and USERS tablespaces to READ WRITE.

```
SQL> alter tablespace example read write;  
  
Tablespace altered.  
  
SQL> alter tablespace users read write;  
  
Tablespace altered.  
  
SQL>
```

8. Reset the default permanent tablespace to the USERS tablespace.

```
SQL> alter database default tablespace users;  
  
Database altered.  
  
SQL>
```

9. Shut down the instance to release resources for the next practice.

```
SQL> shutdown immediate  
Database closed.  
Database dismounted.  
ORACLE instance shut down.  
SQL> exit  
$
```

Practice 11-2: Importing a Non-CDB Application Into a CDB

Overview

In this practice, you will use the export dump file created in Practice 9-2 from the `orcl` non-CDB to import the `SH` schema application in the `cdb1` CDB into a new `pdb_sh` PDB.

Assumption

The export of the `SH` schema performed from the `orcl` non-CDB was successfully completed in Practice 9-2.

The `cdb1` CDB was successfully created in Practice 5-2.

Tasks

1. Still from the `oracle` user 12c terminal window, create a new `pdb_sh` in `cdb1` that will be the container/recipient for the `SH` schema exported from non-CDB `orcl`.
 - a. Create a directory for the new data files of `pdb_sh` of `cdb1`. A recommended pattern is that the `pdb` datafiles reside in a separate folder below the CDB. For example, the CDB datafiles are in `$ORACLE_BASE/oradata/cdb1` and the `pdb` datafiles are in `$ORACLE_BASE/oradata/cdb1/pdb_sh`.

```
$ . oraenv
ORACLE_SID = [dbupgrd] ? cdb1
The Oracle base remains unchanged with value /u01/app/oracle

$ mkdir -p $ORACLE_BASE/oradata/cdb1/pdb_sh
$
```

- b. Connect to the root container of `cdb1` with a user with `CREATE PLUGGABLE DATABASE` privilege. **Note:** The `FILE_NAME_CONVERT` replaces strings, so all the `pdb_seed/*` datafiles will be renamed to `pdb_sh/*` when they are copied for the `pdb_sh` PDB.

```
$ sqlplus / as sysdba

SQL> CREATE PLUGGABLE DATABASE pdb_sh ADMIN USER sh_admin
  IDENTIFIED BY oracle_4U ROLES=(CONNECT)
  FILE_NAME_CONVERT=('/u01/app/oracle/oradata/cdb1/pdbseed'
                    ,'/u01/app/oracle/oradata/cdb1/pdb_sh');
    2      3      4

Pluggable database created.

SQL>
```

- c. Check the open mode of `pdb_sh`. The other pdbs are open read write because you created a trigger to open all of them on startup. `pdb_sh` is NOT open because it was created after the trigger fired.

```
SQL> col con_id format 999
```

```
SQL> col name format A10
SQL> select con_id, NAME, OPEN_MODE, DBID, CON_UID from V$PDBS;

CON_ID NAME          OPEN_MODE        DBID      CON_UID
----- -----
  2 PDB$SEED    READ ONLY   4056847030 4056847030
  3 PDB1        READ WRITE  3321976125 3321976125
  4 PDB_ORCL   READ WRITE  1341873063 1341873063
  5 PDB_SH      MOUNTED     1752541684 1752541684
SQL>
```

- d. Open pdb_sh.

```
SQL> alter pluggable database pdb_sh open;

Pluggable database altered.

SQL> select con_id, NAME, OPEN_MODE, DBID, CON_UID from V$PDBS;

CON_ID NAME          OPEN_MODE        DBID      CON_UID
----- -----
  2 PDB$SEED    READ ONLY   4056847030 4056847030
  3 PDB1        READ WRITE  3321976125 3321976125
  4 PDB_ORCL   READ WRITE  1341873063 1341873063
  5 PDB_SH      READ WRITE 1752541684 1752541684

SQL> exit
$
```

Note: A service name for a pdb is created and registered with the listener when the pdb is created, and EZCONNECT may be used to connect to the PBD without creating the net service name in this step. Net services names are often much more convenient.

2. Use NETCA to add the PDB_SH net service name for pdb_sh pluggable database of cdb1 to the tnsnames.ora file.

```
$ $ORACLE_HOME/bin/netmgr
```

Use netca as you did in Practice 9-3 Task 3 with the following values:

Service Name: pdb_sh

Protocol: TCP

Hostname: <yourservername>, or localhost

Port Number: 1521

3. Test the connection. Connect to pdb_sh AS SYSDBA.

```
$ sqlplus sys/oracle_4U@pdb_sh AS SYSDBA
```

```
SQL> show con_name
```

```
CON_NAME
```

```
-----
```

```
PDB_SH
```

```
SQL> exit
```

4. Connect to the target `pdb_sh` PDB as the `SYSTEM` user.

```
$ sqlplus system/oracle_4U@pdb_sh
```

```
SQL> SELECT name FROM v$pdbs;
```

```
NAME
```

```
-----
```

```
PDB_SH
```

```
SQL>
```

5. Create a Data Pump directory for the dump files stored for any Data Pump operations in `pdb_sh` PDB.

- a. Create the OS directory.

```
SQL> ! mkdir -p /u01/app/oracle/admin/cdb1/pdb_sh/dpdump
```

```
SQL>
```

- b. Create the directory in the PDB.

```
SQL> create directory dp_pdb_sh as  
'/u01/app/oracle/admin/cdb1/pdb_sh/dpdump';
```

```
Directory created.
```

```
SQL>
```

6. Create the `EXAMPLE` tablespace in the PDB.

```
SQL> create tablespace EXAMPLE datafile  
'/u01/app/oracle/oradata/cdb1/pdb_sh/example01.dbf' size 500m;
```

```
Tablespace created.
```

```
SQL>
```

7. Create the users before the import operation.

```
SQL> create user SH identified by oracle_4U account unlock  
      default tablespace EXAMPLE;
```

```
2
```

```
User created.
```

```
SQL> grant create session, create table, create materialized
view, unlimited tablespace to SH;
```

```
SQL> exit
$
```

8. At the end of the Practice 9-2, you moved the

/u01/app/oracle/admin/orcl/dpdump/expSH.dmp export dump file containing the SH schema of the orcl database to the /home/oracle/labs directory. Move the dump file to the directory of the PDB.

```
$ mv /home/oracle/labs/expSH.dmp
/u01/app/oracle/admin/cdb1/pdb_sh/dpdump
$
```

9. Import the SH schema of the orcl database into the pdb_sh PDB. Note that the IMPDP command includes the net service_name in the userid clause.

```
$ rm /u01/app/oracle/admin/cdb1/pdb_sh/dpdump/impSH.log
rm: cannot remove
`/u01/app/oracle/admin/cdb1/pdb_sh/dpdump/impSH.log': No such
file or directory
$
$ impdp system/oracle_4U@pdb_sh FULL=Y dumpfile=expSH.dmp
directory=dp_pdb_sh logfile=impSH.log

Import: Release 12.1.0.1.0 - Production on Thu Feb 21 18:29:00
2013
...
Master table "SYSTEM"."SYS_IMPORT_FULL_01" successfully
loaded/unloaded
Starting "SYSTEM"."SYS_IMPORT_FULL_01": system/********@pdb_sh
FULL=Y dumpfile=expSH.dmp directory=dp_pdb_sh logfile=impSH.log
Processing object type SCHEMA_EXPORT/USER
ORA-31684: Object type USER:"SH" already exists
Processing object type SCHEMA_EXPORT/SYSTEM_GRANT
...
. . imported "SH"."CUSTOMERS" 10.27
MB 55500 rows
. . imported "SH"."COSTS":"COSTS_Q1_1998" 139.6
KB 4411 rows
...
. . imported "SH"."SALES":"SALES_Q1_1998" 1.413
MB 43687 rows
...
. . imported "SH"."SALES":"SALES_Q4_2003" 0
KB 0 rows
...
```

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

```
Processing object type SCHEMA_EXPORT/DIMENSION
Job "SYSTEM"."SYS_IMPORT_FULL_01" completed with 12 error(s) at
Tue Feb 26 18:50:38 2013 elapsed 0 00:01:13

$
```

There are some errors due to missing users in the target PDB. The intention of this practice is to show how to handle the process of exporting and importing a non-CDB schema such as SH into a PDB.

10. Connect as SH to the PDB and check that the SH schema exists and the SH.SALES table is created in the pdb_sh PDB.

```
$ sqlplus sh/oracle_4U@pdb_sh

SQL> SELECT count(*) FROM sh.sales;

COUNT (*)
-----
918843

SQL>
```

If you connect to the root container of the CDB, the SH application data is not known. It belongs only to the PDB.

```
SQL> connect / as sysdba
Connected.
SQL> SELECT count(*) FROM sh.sales;

SELECT count(*) FROM sh.sales
*
ERROR at line 1:
ORA-00942: table or view does not exist

SQL> exit
$
```

Appendix: Upgrading a Database With Options

Chapter 12

Appendix

Overview

In the class setup, most of the options were removed from the dbupgrd database instance so that the upgrade process could complete within a reasonable time. In these practices, we will consider that many options are enabled in the dbupgrd database instance. Therefore you would prepare a dbupgrd database instance currently running in 11g to upgrade to 12c, then upgrade the database and finally perform post-upgrade actions. In these practices, you would:

- Execute the preupgrd.sql script
- Review the information displayed by the pre-upgrade information tool
- Resolve issues if any
- Perform a full database backup before the upgrade process starts
- Upgrade the Oracle Database 11g Release 2 dbupgrd database to Oracle Database 12c by using DBUA
- Perform post-upgrade actions

In the upgrade plan, it is not planned to relocate the data files, control files, redo log files and archive log files in another location.

You are going to work as the oracle user to perform operations in the 11g environment and as the oracle user for other operations in the 12c environment. Therefore, keep a terminal window opened as the oracle user in the 11g environment and another terminal window opened as the oracle user in the 12c environment.

Practice A-1: Executing Preliminary Steps and the Pre-Upgrade Tool

Overview

In this practice, you will prepare your dbupgrd database instance for an upgrade to 12c. You have already installed the Oracle 12c Database software. You need to check that the dbupgrd database instance is compatible with an upgrade from 11.2.0.3.0 to 12.1.0.1.0 before upgrading.

Tasks

1. List the options that require a specific treatment before upgrade.
 - a. Logged in to a Unix session as the `oracle` user, check if Oracle Label Security or/and Oracle Database Vault are installed in the `dbupgrd` database instance.

```
$ . oraenv
ORACLE_SID = [dbupgrd] ? dbupgrd
The Oracle base remains unchanged with value /u01/app/oracle
$ env | grep ORA
ORACLE_SID=dbupgrd
ORACLE_BASE=/u01/app/oracle
ORACLE_HOME=/u01/app/oracle/product/11.2.0/dbhome_2
$ sqlplus / as sysdba

SQL> col parameter format A30
SQL> col value format A10
SQL> select * from v$option
  2 where parameter in
  3      ('Oracle Label Security', 'Oracle Database Vault');

PARAMETER                               VALUE
-----
Oracle Label Security                  FALSE
Oracle Database Vault                  FALSE

SQL>
```

If one of these two options were enabled, you would have to execute the `$ORACLE_HOME/rdbms/admin/olspreupgrade.sql` script copied from the 12c database `ORACLE_HOME` environment to the 11g database `ORACLE_HOME` environment. The script moves the `AUD$` table from the `SYSTEM` schema to the `SYS` schema. You would have to disable Oracle Database Vault if the option is enabled.

- b. Switch to the Oracle Database 12c session. Open an oracle user terminal window and set the environments variables to the 12c Oracle database ORACLE_HOME.

```
$ id
uid=54329(oracle) gid=54330(oinstall)
groups=54330(oinstall),54323(oper),54324(backupdba),54325(dgdba),
,54326(kmdba),54327(asmdba),54331(dba)
context=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
$
```

- c. Set the ORACLE_SID environment variable to orcl. (You know that the orcl database instance runs in the 12c database environment)

```
$ . oraenv
ORACLE_SID = [oracle] ? orcl
The Oracle base remains unchanged with value /u01/app/oracle
$ env | grep ORA
ORACLE_SID=orcl
ORACLE_BASE=/u01/app/oracle
ORACLE_HOME=/u01/app/oracle/product/12.1.0/dbhome_1
$
```

- d. Display the contents of the \$ORACLE_HOME/rdbms/admin/olspreupgrade.sql script.

```
$ vi $ORACLE_HOME/rdbms/admin/olspreupgrade.sql

Rem
Rem $Header: rdbms/admin/olspreupgrade.sql /main/3 2012/09/06
11:27:27 aramappa Exp $
Rem
Rem olspreupgrade.sql
Rem
Rem Copyright (c) 2012, Oracle and/or its affiliates. All rights
reserved.
Rem
Rem      NAME olspreupgrade.sql - OLS Pre processing script prior
to upgrade for 12.1
Rem
Rem      DESCRIPTION
Rem          This is a mandatory OLS preprocess upgrade script that
needs to be run by Label Security and Database Vault customers
when upgrading to 12.1.
Rem
Rem          This script is NOT NEEDED if you DO NOT have Database
Vault or Label Security.
...
```

It is clearly stated that this is a mandatory OLS preprocess upgrade script that needs to be run by Label Security and Database Vault customers when upgrading to

12.1.0.0, but this script is NOT NEEDED if you DO NOT have Database Vault or Label Security.

- e. The script describes how you should proceed:

```
Rem      STEPS TO RUN THIS SCRIPT
Rem -----
Rem      To run the Oracle Label Security preprocess upgrade
script, copy the $ORACLE_HOME/rdbms/admin/olspreupgrade.sql
script to the old ORACLE_HOME. Run the script as SYSDBA in the
old ORACLE_HOME.
```

```
Rem
...
...
```

```
...
Rem      However, if you have Database Vault, the following
steps need to be done after copying the olspreupgrade.sql to the
old ORACLE_HOME:
```

```
...
...
```

```
Rem      To run the OLS preprocess script on a release 10.2.0.5
or 11.2 database before upgrading:
```

```
Rem
```

```
Rem      1. Start SQL*Plus and connect to the database to be
upgraded as DVOWNER.
```

```
Rem      2. Execute the following statement:
```

```
Rem          SQL> GRANT DV_PATCH_ADMIN to SYS;
```

```
Rem      3. Run the OLS preprocess script, at
```

```
Rem          ORACLE_HOME/rdbms/admin/olspreupgrade.sql
```

```
Rem      4. After the olspreupgrade.sql has been successfully
run, start SQL*Plus and connect to the database as DVOWNER.
```

```
Rem      5. Execute the following statement:
```

```
Rem          SQL> REVOKE DV_PATCH_ADMIN from SYS;
```

```
Rem
...
...
```

- f. Quit the vi editor.

- g. How would you disable Oracle Database Vault?

- Log in as the Oracle Database Owner (DV_OWNER) account.

```
sqlplus dvo/password
```

- Execute the following procedure:

```
exec DVSYS.DBMS_MACADM.DISABLE_DV
```

- Restart the instance

```
connect / as sysdba
```

```
shutdown immediate
```

```
startup
```

2. Check that the tablespaces set to READ ONLY or OFFLINE mode are not schema-based tablespaces such as SYSAUX, SYSTEM, XDB, HTMLDB, or CTXSYS. Otherwise the upgrade will fail. In the rare case where queue tables reside in a tablespace that has been set to READ ONLY for the upgrade, then that tablespace should be set back to READ WRITE. Return to your Unix 11g session and run the following statement:

```
SQL> select tablespace_name, status from dba_tablespaces;

TABLESPACE_NAME          STATUS
-----
SYSTEM                  ONLINE
SYSAUX                 ONLINE
UNDOTBS1                ONLINE
TEMP                   ONLINE
USERS                  ONLINE
EXAMPLE                ONLINE

6 rows selected.

SQL>
```

3. Remove the AUDSYS schema, and the AUDIT_ADMIN and AUDIT_VIEWER roles. At this stage, there should be no AUDSYS schema. This user and roles will be created as a predefined user and roles when the database is upgraded to 12c and migrated to Unified Auditing. Unified auditing is the new auditing capability in Oracle Database 12c.

```
SQL> DROP USER AUDSYS CASCADE;
DROP USER AUDSYS CASCADE
*
ERROR at line 1:
ORA-01918: user 'AUDSYS' does not exist

SQL> DROP ROLE AUDIT_ADMIN;
DROP ROLE AUDIT_ADMIN
*
ERROR at line 1:
ORA-01919: role 'AUDIT_ADMIN' does not exist

SQL> DROP ROLE AUDIT_VIEWER;
DROP ROLE AUDIT_VIEWER
*
ERROR at line 1:
ORA-01919: role 'AUDIT_VIEWER' does not exist

SQL>
```

4. Run the Pre-Upgrade Information Tool by using the \$ORACLE_HOME/rdbms/admin/preupgrd.sql script.
 - a. Copy the \$ORACLE_HOME/rdbms/admin/preupgrd.sql and \$ORACLE_HOME/rdbms/admin/utluppkgs.sql scripts from the 12c database ORACLE_HOME environment to the 11g database ORACLE_HOME environment. In the Unix 12c session, run the following statement:

```
$ cp  
/u01/app/oracle/product/12.1.0/dbhome_1/rdbms/admin/preupgrd.sql  
/u01/app/oracle/product/11.2.0/dbhome_2/rdbms/admin/preupgrd.sql  
$ cp  
/u01/app/oracle/product/12.1.0/dbhome_1/rdbms/admin/utluppkgs.sql  
/u01/app/oracle/product/11.2.0/dbhome_2/rdbms/admin/utluppkgs.sql  
$
```

- b. The execution of the preupgrd.sql script will generate a log file and two SQL scripts in the /u01/app/oracle/cfgtoollogs/dbupgrd/preupgrade directory. Create the directory.

```
$ mkdir /u01/app/oracle/cfgtoollogs/dbupgrd  
$ mkdir /u01/app/oracle/cfgtoollogs/dbupgrd/preupgrade  
$
```

- c. In the Unix 11g session, run the preupgrd.sql script:

```
SQL>  
@/u01/app/oracle/product/11.2.0/dbhome_2/rdbms/admin/preupgrd.sql  
Loading Pre-Upgrade Package...  
  
Executing Pre-Upgrade Checks...  
Pre-Upgrade Checks Complete.  
  
*****  
  
Results of the checks are located at:  
/u01/app/oracle/cfgtoollogs/dbupgrd/preupgrade/preupgrade.log  
  
Pre-Upgrade Fixup Script (run in source database environment):  
/u01/app/oracle/cfgtoollogs/dbupgrd/preupgrade/preupgrade_fixups.sql  
  
Post-Upgrade Fixup Script (run shortly after upgrade):  
/u01/app/oracle/cfgtoollogs/dbupgrd/preupgrade/postupgrade_fixups.sql  
  
*****  
Fixup scripts must be reviewed prior to being executed.  
*****
```

```
*****  
====>> USER ACTION REQUIRED <<=====  
*****  
  
The following are *** ERROR LEVEL CONDITIONS *** that must be  
addressed prior to attempting your upgrade.  
Failure to do so will result in a failed upgrade.  
  
You MUST resolve the above errors prior to upgrade  
  
*****  
SQL> exit  
$
```

Practice A-2: Implementing the Pre-Upgrade Information Tool Recommendations

Overview

In this practice, you will examine the Pre-Upgrade Information Tool output log file and implement the recommended actions.

1. Examine the

/u01/app/oracle/cfgtoollogs/dbupgrd/preupgrade/preupgrade.log, output created from the Pre-Upgrade Information Tool.

```
$ cat
/u01/app/oracle/cfgtoollogs/dbupgrd/preupgrade/preupgrade.log
Oracle Database Pre-Upgrade Information Tool 02-14-2013 19:13:10
Script Version: 12.1.0.1.0 Build: 006
*****
*
Database Name: DBUPGRD
Version: 11.2.0.3.0
Compatible: 11.2.0.0.0
Blocksize: 8192
Platform: Linux x86 64-bit
Timezone file: V14
*****
*
[Renamed Parameters]
[No Renamed Parameters in use]
*****
*
*****
*
[Obsolete/Deprecated Parameters]
[No Obsolete or Desupported Parameters in use]
*****
*
[Component List]
*****
*
--> Oracle Catalog Views [upgrade] VALID
--> Oracle Packages and Types [upgrade] VALID
--> JServer JAVA Virtual Machine [upgrade] VALID
--> Oracle XDK for Java [upgrade] VALID
--> Oracle Workspace Manager [upgrade] VALID
--> OLAP Analytic Workspace [upgrade] VALID
--> Oracle Enterprise Manager Repository [upgrade] VALID
--> Oracle Text [upgrade] VALID
```

```
--> Oracle XML Database [upgrade] VALID
--> Oracle Java Packages [upgrade] VALID
--> Oracle Multimedia [upgrade] VALID
--> Oracle Spatial [upgrade] VALID
--> Expression Filter [upgrade] VALID
--> Rule Manager [upgrade] VALID
--> Oracle Application Express [upgrade] VALID
--> Oracle OLAP API [upgrade] VALID
*****
*
[Tablespaces]
*****
*
--> SYSTEM tablespace is adequate for the upgrade.
    minimum required size: 1226 MB
--> SYSAUX tablespace is adequate for the upgrade.
    minimum required size: 1487 MB
--> UNDOTBS1 tablespace is adequate for the upgrade.
    minimum required size: 400 MB
--> TEMP tablespace is adequate for the upgrade.
    minimum required size: 60 MB
--> EXAMPLE tablespace is adequate for the upgrade.
    minimum required size: 310 MB

[No adjustments recommended]

*****
*
*****
*
[Pre-Upgrade Checks]
*****
*
WARNING: --> Process Count may be too low

Database has a maximum process count of 150 which is lower
than the default value of 300 for this release.

You should update your processes value prior to the upgrade
to a value of at least 300.

For example:
    ALTER SYSTEM SET PROCESSES=300 SCOPE=SPFILE
or update your init.ora file.
```

WARNING: --> Enterprise Manager Database Control repository found in the database

In Oracle Database 12c, Database Control is removed during the upgrade. To save time during the Upgrade, this action can be done prior to upgrading using the following steps after copying **rdbms/admin/emremove.sql** from the new Oracle home

- Stop EM Database Control:

```
$> emctl stop dbconsole
```

- Connect to the Database using the SYS account AS SYSDBA:

```
SET ECHO ON;
SET SERVEROUTPUT ON;
@emremove.sql
```

Without the set echo and serveroutput commands you will not be able to follow the progress of the script.

WARNING: --> Existing DBMS_LDAP dependent objects

Database contains schemas with objects dependent on DBMS_LDAP package.

Refer to the Upgrade Guide for instructions to configure Network ACLs.

USER APEX_030200 has dependent objects.

INFORMATION: --> OLAP Catalog (AMD) exists in database

Starting with Oracle Database 12c, OLAP is desupported.

If you are not using the OLAP Catalog component and want to remove it, then execute the

ORACLE_HOME/oraolap/admin/catnoamd.sql script before or after the upgrade.

INFORMATION: --> Older Timezone in use

Database is using a time zone file older than version 18.

After the upgrade, it is recommended that DBMS_DST package be used to upgrade the 11.2.0.3.0 database time zone version

to the latest version which comes with the new release.

Please refer to My Oracle Support note number 977512.1 for details.

```
*****  
*
```

[Pre-Upgrade Recommendations]

```
*****  
*
```

```
*****  
***** Dictionary Statistics  
*****
```

Please gather dictionary statistics 24 hours prior to upgrading the database.

To gather dictionary statistics execute the following command while connected as SYSDBA:

```
EXECUTE dbms_stats.gather_dictionary_stats;
```

^^^ MANUAL ACTION SUGGESTED ^^^

```
*****  
*
```

[Post-Upgrade Recommendations]

```
*****  
*
```

```
*****  
***** Fixed Object Statistics  
*****
```

Please create stats on fixed objects two weeks after the upgrade using the command:

```
EXECUTE DBMS_STATS.GATHER_FIXED_OBJECTS_STATS;
```

^^^ MANUAL ACTION SUGGESTED ^^^

```
*****  
*
```

```
***** Summary *****
```

```
0 ERRORS exist in your database.  
3 WARNINGS that Oracle suggests are addressed to improve  
database performance.  
2 INFORMATIONAL messages that should be reviewed prior to your  
upgrade.
```

After your database is upgraded and open in normal mode you must run **rdbms/admin/catuppst.sql** which executes several required tasks and completes the upgrade process.

You should follow that with the execution of **rdbms/admin/utlrp.sql**, and a comparison of invalid objects before and after the upgrade using **rdbms/admin/utluobj.sql**

If needed you may want to **upgrade your timezone data** using the process described in My Oracle Support note 977512.1

\$

2. You have two ways to address the warnings and recommendations. You can run the \$ORACLE_BASE/cfgtoollogs/dbupgrd/preupgrade/preupgrade_fixups.sql script which will attempt to resolve issues reported. Issues that cannot be resolved automatically by the fixup script are flagged with ** USER ACTION REQUIRED **. You will have to fix them manually one by one.
3. You first execute the /u01/app/oracle/cfgtoollogs/dbupgrd/preupgrade/preupgrade_fixups.sql script.

```
$ sqlplus / as sysdba  
  
SQL>  
@/u01/app/oracle/cfgtoollogs/dbupgrd/preupgrade/preupgrade_fixups.sql  
  
Pre-Upgrade Fixup Script Generated on 2013-02-14 19:13:04  
Version: 12.1.0.1 Build: 006  
Beginning Pre-Upgrade Fixups...  
  
PL/SQL procedure successfully completed.  
  
PL/SQL procedure successfully completed.  
  
*****  
*  
Check Tag:      DEFAULT_PROCESS_COUNT  
Check Summary: Verify min process count is not too low  
Fix Summary:   Review and increase if needed, your PROCESSES  
value.
```

```
*****
*
Fixup Returned Information:
WARNING: --> Process Count may be too low

Database has a maximum process count of 150 which is lower
than the
default value of 300 for this release.
You should update your processes value prior to the upgrade
to a value of at least 300.
For example:
    ALTER SYSTEM SET PROCESSES=300 SCOPE=SPFILE
or update your init.ora file.
*****
*
```

PL/SQL procedure successfully completed.

```
*****
*
Check Tag:      EM_PRESENT
Check Summary: Check if Enterprise Manager is present
Fix Summary:   Execute emremove.sql prior to upgrade.
*****
*
```

Fixup Returned Information:

WARNING: --> Enterprise Manager Database Control repository found in the database

In Oracle Database 12c, Database Control is removed during the upgrade. To save time during the Upgrade, this action can be done prior to upgrading using the following steps after copying rdbms/admin/emremove.sql from the new Oracle home

- Stop EM Database Control:
\$> emctl stop dbconsole
- Connect to the Database using the SYS account AS SYSDBA:

SET ECHO ON;
SET SERVEROUTPUT ON;
@emremove.sql
Without the set echo and serveroutput commands you will not

```
be able to follow the progress of the script.  
*****  
*  
  
PL/SQL procedure successfully completed.  
  
*****  
*  
Check Tag:      DBMS_LDAP_DEPENDENCIES_EXIST  
Check Summary: Check for dependency on DBMS_LDAP package  
Fix Summary:   Network Objects must be reviewed manually.  
*****  
*  
Fixup Returned Information:  
WARNING: --> Existing DBMS_LDAP dependent objects  
  
Database contains schemas with objects dependent on  
DBMS_LDAP package.  
Refer to the Upgrade Guide for instructions to configure  
Network ACLs.  
USER APEX_030200 has dependent objects.  
*****  
*  
  
PL/SQL procedure successfully completed.  
  
*****  
*  
Check Tag:      AMD_EXISTS  
Check Summary: Check to see if AMD is present in the database  
Fix Summary:   Manually execute  
ORACLE_HOME/oraolap/admin/catnoamd.sql script to remove OLAP.  
*****  
*  
Fixup Returned Information:  
INFORMATION: --> OLAP Catalog (AMD) exists in database  
  
Starting with Oracle Database 12c, OLAP is desupported.  
If you are not using the OLAP Catalog component and want  
to remove it, then execute the  
ORACLE_HOME/oraolap/admin/catnoamd.sql script before or  
after the upgrade.  
*****  
*
```

```
PL/SQL procedure successfully completed.

*****
*
[Pre-Upgrade Recommendations]
*****
*

PL/SQL procedure successfully completed.

*****
***** Dictionary Statistics
*****
*****
```

Please gather dictionary statistics 24 hours prior to upgrading the database.

To gather dictionary statistics execute the following command while connected as SYSDBA:

```
EXECUTE dbms_stats.gather_dictionary_stats;
```

^^^ MANUAL ACTION SUGGESTED ^^

```
PL/SQL procedure successfully completed.

*****
***** Fixup Summary *****
```

4 fixup routines generated INFORMATIONAL messages that should be reviewed.

```
PL/SQL procedure successfully completed.

*****
***** Pre-Upgrade Fixup Script Complete
*****
```

```
PL/SQL procedure successfully completed.

SQL>
```

4. The fixup routines did not fix the issues. Perform the actions to address the issues that were not fixed.

- a. First update the PROCESSES value prior to the upgrade to a value of at least 300.

```
SQL> show parameter processes

NAME                           TYPE        VALUE
-----
-
...
processes                      integer     150

SQL> ALTER SYSTEM SET PROCESSES=300 SCOPE=SPFILE;

System altered.

SQL> exit
$
```

- b. Perform the second recommended action. In Oracle Database 12c, Enterprise Manager Database Control is removed during the upgrade. Enterprise Manager Database Control does not exist in Oracle Database 12c. It is replaced by Enterprise Manager Database Express. To save time during the upgrade, this action can be done prior to upgrading using the following steps:
- Copy the \$ORACLE_HOME/rdbms/admin/emremove.sql from the new 12c Oracle home.

```
$ cp
/u01/app/oracle/product/12.1.0/dbhome_1/rdbms/admin/emremove.sql
/u01/app/oracle/product/11.2.0/dbhome_2/rdbms/admin/emremove.sql
$
```

- Stop EM Database Control in case it is running.

```
$ emctl stop dbconsole
Oracle Enterprise Manager 11g Database Control Release
11.2.0.3.0
Copyright (c) 1996, 2011 Oracle Corporation. All rights
reserved.
https://edrsr32p1.us.oracle.com:1158/em/console/aboutApplication
Stopping Oracle Enterprise Manager 11g Database Control ...
... Stopped.
$
```

- Execute the removal script.

```
$ sqlplus / as sysdba

SQL> SET ECHO ON
SQL> SET SERVEROUTPUT ON
SQL> @$ORACLE_HOME/rdbms/admin/emremove.sql
```

```

...
Dropping synonym : MGMT$APPLIED_PATCHSETS ...
Dropping synonym : MGMT$APPLIED_PATCHES ...
Dropping synonym : MGMT$ALERT_NOTIF_LOG ...
Dropping synonym : MGMT$ALERT_HISTORY ...
Dropping synonym : MGMT$ALERT_CURRENT ...
Dropping synonym : MGMT$ALERT_ANNOTATIONS ...
Dropping synonym : EMD_MNTR ...
Dropping synonym : ECM_UTIL ...
Finished phase 5
Starting phase 6 : Dropping Oracle Enterprise Manager related other roles ...
Finished phase 6
The Oracle Enterprise Manager related schemas and objects are dropped.
Do the manual steps to shutdown the DB Control if not done before running this
script and then delete the DB Control configuration files

PL/SQL procedure successfully completed.

SQL>
```

- c. Perform the third recommendation action about network objects.
- Query the `DBA_DEPENDENCIES` view to obtain more information about the dependencies.

```

SQL> col referenced_name format a16
SQL> col owner format a16
SQL> col name format a28
SQL> select owner, name, referenced_name from dba_dependencies
  2  where referenced_name in
  3  ('UTL_TCP','UTL_SMTP','UTL_MAIL','UTL_HTTP',
  4  'UTL_INADDR','DBMS_LDAP')
  5  and owner NOT IN ('SYS','PUBLIC','ORDPLUGINS');
```

OWNER	NAME	REFERENCED_NAME
ORACLE_OCM	MGMT_DB_LL_METRICS	UTL_INADDR
MDSYS	SDO_OLS	UTL_HTTP
APEX_030200	WWV_FLOW_HELP	UTL_HTTP
APEX_030200	WWV_FLOW_MAIL	UTL_SMTP
APEX_030200	WWV_FLOW_PRINT_UTIL	UTL_HTTP
APEX_030200	WWV_FLOW_LDAP	DBMS_LDAP

```

APEX_030200      WWV_FLOW_LDAP          DBMS_LDAP
APEX_030200      WWV_FLOW_WEB_SERVICES UTL_HTTP
APEX_030200      UTL_HTTP              UTL_HTTP
APEX_030200      UTL_SMTP              UTL_SMTP
APEX_030200      WWV_FLOW_CUSTOM_AUTH_LDAP DBMS_LDAP
APEX_030200      WWV_FLOW_DISP_PAGE_PLUGS UTL_HTTP

```

12 rows selected.

```

SQL> exit
$
```

- ii) Add the new access controls. Replace <*your_host_name*> by your own host name.
Use any editor to write the following text and save it in /home/oracle/labs as *acl.sql*.

```

SQL> DECLARE
      acl_path  VARCHAR2(4000);
BEGIN
    SELECT acl INTO acl_path FROM dba_network_acls
    WHERE host = '<your_host_name>' AND lower_port IS NULL
    AND upper_port   IS NULL;

    IF DBMS_NETWORK_ACL_ADMIN.CHECK_PRIVILEGE
        (acl_path, 'APEX_030200','connect') IS NULL
    THEN
        DBMS_NETWORK_ACL_ADMIN.ADD_PRIVILEGE
        (acl_path, 'APEX_030200', TRUE, 'connect');
    END IF;
    IF DBMS_NETWORK_ACL_ADMIN.CHECK_PRIVILEGE
        (acl_path, 'ORACLE_OCM','connect') IS NULL
    THEN
        DBMS_NETWORK_ACL_ADMIN.ADD_PRIVILEGE
        (acl_path, 'ORACLE_OCM', TRUE, 'connect');
    END IF;
    IF DBMS_NETWORK_ACL_ADMIN.CHECK_PRIVILEGE
        (acl_path, 'MDSYS','connect') IS NULL
    THEN
        DBMS_NETWORK_ACL_ADMIN.ADD_PRIVILEGE
        (acl_path, 'MDSYS', TRUE, 'connect');
    END IF;

EXCEPTION
  WHEN no_data_found THEN
```

```
DBMS_NETWORK_ACL_ADMIN.CREATE_ACL('ACL_12c.xml',
    'ACL init after upgrade','APEX_030200',TRUE,'connect');
DBMS_NETWORK_ACL_ADMIN.ADD_PRIVILEGE
    ('ACL_12c.xml', 'ORACLE_OCM', TRUE, 'connect');
DBMS_NETWORK_ACL_ADMIN.ADD_PRIVILEGE
    ('ACL_12c.xml', 'MDSYS', TRUE, 'connect');
DBMS_NETWORK_ACL_ADMIN.ASSIGN_ACL
    ('ACL_12c.xml', '<your_host_name>');
END;
/
PL/SQL procedure successfully completed.

SQL> COMMIT;

Commit complete.

SQL>
```

- d. Perform the fourth recommendation. You are not using the OLAP Catalog component and want to remove OLAP metadata by deleting the OLAPSYS schema.

```
SQL>
@/u01/app/oracle/product/11.2.0/dbhome_2/olap/admin/catnoamd.sql
...
SQL> @@factvdrp.sql
SQL> drop public synonym OlapFactView;

Synonym dropped.

SQL> @@dimvwdrp.sql
SQL> drop public synonym OlapDimView;

Synonym dropped.

SQL> @@olapodrp.sql
SQL> drop public synonym DBMS_ODM;

Synonym dropped.

SQL>
SQL> rem
SQL> rem drop olapsys user
SQL> rem
SQL> drop user olapsys cascade
```

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

```

2 /
SQL>
SQL> rem
SQL> rem drop OLAP_DBA role
SQL> rem
SQL> drop role OLAP_DBA
2 /
Role dropped.

SQL>
```

- e. The last recommendation is to gather dictionary statistics for 24 hours prior to upgrading the database.

```

SQL> exec dbms_stats.gather_dictionary_stats

PL/SQL procedure successfully completed.

SQL> exit
$
```

5. Examine the

`$ORACLE_BASE/cfgtoollogs/dbupgrd/preupgrade/postupgrade_fixups.sql`

script created to address issues that can be fixed after the database has been upgraded.

```

$ cat
$ORACLE_BASE/cfgtoollogs/dbupgrd/preupgrade/postupgrade_fixups.sql
REM Post Upgrade Script Generated on: 2013-02-14 19:13:04
REM Generated by Version: 12.1.0.1 Build: 006
SET ECHO OFF SERVEROUTPUT ON FORMAT WRAPPED TAB OFF LINESIZE
750;
BEGIN
    dbms_output.put_line ('Post Upgrade Fixup Script Generated on
2013-02-14 19:13:04 Version: 12.1.0.1 Build: 006');
    dbms_output.put_line ('Beginning Post-Upgrade Fixups...');

END;
/
BEGIN
dbms_preup.clear_run_flag(FALSE);
END;
/
BEGIN
-- ***** Fixup Details *****
-- Name:          OLD_TIME_ZONES_EXIST
-- Description:  Check for use of older timezone data file
```

```
-- Severity:      Informational
-- Action:        ^^^ MANUAL ACTION REQUIRED ^^^
-- Fix Summary:
--     Update the timezone using the DBMS_DST package after
upgrade is complete.

dbms_preup.run_fixup_and_report('OLD_TIME_ZONES_EXIST');
END;
/
BEGIN
dbms_output.put_line ('*****');
dbms_output.put_line (' [Post-Upgrade Recommendations]');
dbms_output.put_line ('*****');
dbms_output.put_line ('');
END;
/
BEGIN
dbms_output.put_line ('*****');
dbms_output.put_line ('**** Fixed Object Statistics *****');
dbms_output.put_line ('*****');
dbms_output.put_line ('');
dbms_output.put_line ('Please create stats on fixed objects two
weeks');
dbms_output.put_line ('after the upgrade using the command:');
dbms_output.put_line ('    EXECUTE
DBMS_STATS.GATHER_FIXED_OBJECTS_STATS;');
dbms_output.put_line ('');
dbms_output.put_line ('^/^ MANUAL ACTION SUGGESTED ^/^');
dbms_output.put_line ('');
END;
/
BEGIN dbms_preup.fixup_summary(FALSE); END;
/
BEGIN
dbms_output.put_line ('** Post Upgrade Fixup Script Complete
*');
END;
/
REM Post Upgrade Script Closed At: 2013-02-14 19:13:16
$
```

Recommendations are suggested for after upgrade such as updating the time zone using the DBMS_DST package.

Practice A-3: Completing Prerequisites Steps Before Upgrade

Overview

In this practice, you perform the last actions and verifications before the upgrade.

Tasks

1. The Pre-Upgrade Information Tool creates and populates the `registry$sys_inv_objs` and `registry$nonsys_inv_objs` tables. If you did not execute the Pre-Upgrade Information Tool script, you can execute the `utluiobj.sql` script before and after upgrading. The script execution outputs the difference between the invalid objects that existed prior to the upgrade and invalid objects that exist after the upgrade. Consider that you did not execute the Pre-Upgrade Information Tool because you know that DBUA will automatically execute it. You are still logged in the Unix 11g session.

```
$ sqlplus / as sysdba

SQL> @$ORACLE_HOME/rdbms/admin/utluiobj.sql
.
.
Oracle Database 11.1 Post-Upgrade Invalid Objects Tool 02-14-
2013 21:42:13
.

This tool lists post-upgrade invalid objects that were not
invalid prior to upgrade (it ignores pre-existing pre-upgrade
invalid objects).

.

Owner          Object Name           Object Type
.

PL/SQL procedure successfully completed.

SQL> select * from REGISTRY$sys_inv_objs;
no rows selected

SQL> select * from REGISTRY$nonsys_inv_objs;
no rows selected

SQL>
```

2. Verify that materialized view refreshes have completed. If materialized view refreshes have not completed, perform the refresh manually using the `DBMS_MVIEW` package.

```
SQL> select o.name from sys.obj$ o, sys.user$ u, sys.sum$ s
      where o.type# = 42
        and bitand(s.mflags, 8) =8;
2      3
```

```
no rows selected  
SQL>
```

3. Ensure that no files need media recovery.

```
SQL> select * from v$recover_file;  
  
no rows selected  
  
SQL>
```

If there are files that need recovery, perform the required recovery using RMAN.

4. Resolve outstanding distributed transactions.

```
SQL> select * from dba_2pc_pending;  
  
no rows selected  
  
SQL>
```

If the query in the previous step returns any rows, then run the following statements:

```
SELECT local_tran_id FROM dba_2pc_pending;  
EXECUTE dbms_transaction.purge_lost_db_entry('');  
COMMIT;
```

5. Purge the database recycle bin.

```
SQL> purge DBA_RECYCLEBIN;  
  
DBA Recyclebin purged.  
  
SQL>
```

6. If a standby database exists, then you must synchronize it with the primary database.

- a. Check if a standby database exists.

```
SQL> select  
SUBSTR(value,INSTR(value,'=',INSTR(UPPER(value), 'SERVICE'))+1)  
from v$parameter  
where name like 'log_archive_dest%'  
and UPPER(value) like 'SERVICE%';  
  
no rows selected  
  
SQL>
```

- b. If the query in the previous step returns a row, then synchronize the standby database with the primary database.
 - Make sure all the logs are transported to the standby server after a final log switch in the primary.

- Start the recovery of the standby database with the NODELAY option.
7. The upgrade process uses a large number of archived redo log files. Increase the DB_RECOVERY_FILE_DEST_SIZE to 15GB.

```
SQL> alter system set DB_RECOVERY_FILE_DEST_SIZE=15G scope=BOTH;  
System altered.  
  
SQL> exit  
$
```

Practice A-4: Performing a Full Database Backup

Overview

In this practice, you will perform a full database backup of the dbupgrd database before you start the upgrade.

Tasks

1. Verify that the database is in ARCHIVELOG mode.
 - a. Stay in the oracle user terminal window. Connect to the database instance as SYSDBA.

```
$ sqlplus / as sysdba

SQL> select log_mode from v$database;

LOG_MODE
-----
NOARCHIVELOG

SQL>
```

- b. Set the database instance in ARCHIVELOG mode.
 - i) Shut down the database instance.

```
SQL> shutdown immediate
Database closed.
Database dismounted.
ORACLE instance shut down.

SQL>
```

- ii) Start the database instance in MOUNT mode.

```
SQL> startup mount
ORACLE instance started.

Total System Global Area  501059584 bytes
Fixed Size                  2229744 bytes
Variable Size                171969040 bytes
Database Buffers            318767104 bytes
Redo Buffers                 8093696 bytes
Database mounted.

SQL>
```

- iii) Alter the database instance to ARCHIVELOG mode and open the database.

```
SQL> alter database archivelog;
Database altered.
```

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

```
SQL> alter database open;  
  
Database altered.  
  
SQL> exit  
$
```

2. Use RMAN to perform the full database backup.

- a. Verify that the configuration includes the automatic control file and SPFILE backup. If this is not the case, enable the automatic control file and SPFILE backup.

```
$ rman target /  
  
RMAN> show all;  
  
using target database control file instead of recovery catalog  
RMAN configuration parameters for database with db_unique_name  
DBUPGRD are:  
CONFIGURE RETENTION POLICY TO REDUNDANCY 1; # default  
CONFIGURE BACKUP OPTIMIZATION OFF; # default  
CONFIGURE DEFAULT DEVICE TYPE TO DISK; # default  
CONFIGURE CONTROLFILE AUTOBACKUP OFF; # default  
...  
CONFIGURE SNAPSHOT CONTROLFILE NAME TO  
'/u01/app/oracle/product/11.2.0/dbhome_2/dbs/snapcf_dbupgrd.f';  
# default  
  
RMAN> CONFIGURE CONTROLFILE AUTOBACKUP ON;  
  
new RMAN configuration parameters:  
CONFIGURE CONTROLFILE AUTOBACKUP ON;  
new RMAN configuration parameters are successfully stored  
  
RMAN>
```

- b. Perform the full database backup including all data files, control files, SPFILE and archive log files.

```
RMAN> backup database plus archivelog;  
  
Starting backup at 14-FEB-13  
current log archived  
allocated channel: ORA_DISK_1  
channel ORA_DISK_1: SID=147 device type=DISK  
channel ORA_DISK_1: starting archived log backup set  
channel ORA_DISK_1: specifying archived log(s) in backup set
```

```
input archived log thread=1 sequence=12 RECID=1 STAMP=807374917
channel ORA_DISK_1: starting piece 1 at 14-FEB-13
channel ORA_DISK_1: finished piece 1 at 14-FEB-13
piece
handle=+FRA/dbupgrd/backupset/2013_02_14/annnf0_tag20130214t1448
38_0.261.807374919 tag=TAG20130214T144838 comment=NONE
channel ORA_DISK_1: backup set complete, elapsed time: 00:00:03
Finished backup at 14-FEB-13

Starting backup at 14-FEB-13
using channel ORA_DISK_1
channel ORA_DISK_1: starting full datafile backup set
channel ORA_DISK_1: specifying datafile(s) in backup set
input datafile file number=00001
name=+DATA/dbupgrd/datafile/system.256.807214393
input datafile file number=00002
name=+DATA/dbupgrd/datafile/sysaux.257.807214393
input datafile file number=00005
name=+DATA/dbupgrd/datafile/example.265.807214591
input datafile file number=00003
name=+DATA/dbupgrd/datafile/undotbs1.258.807214395
input datafile file number=00004
name=+DATA/dbupgrd/datafile/users.259.807214395
channel ORA_DISK_1: starting piece 1 at 14-FEB-13
channel ORA_DISK_1: finished piece 1 at 14-FEB-13
piece
handle=+FRA/dbupgrd/backupset/2013_02_14/nnndf0_tag20130214t1448
41_0.262.807374923 tag=TAG20130214T144841 comment=NONE
channel ORA_DISK_1: backup set complete, elapsed time: 00:01:45
Finished backup at 14-FEB-13

Starting backup at 14-FEB-13
current log archived
using channel ORA_DISK_1
channel ORA_DISK_1: starting archived log backup set
channel ORA_DISK_1: specifying archived log(s) in backup set
input archived log thread=1 sequence=13 RECID=2 STAMP=807375027
channel ORA_DISK_1: starting piece 1 at 14-FEB-13
channel ORA_DISK_1: finished piece 1 at 14-FEB-13
piece
handle=+FRA/dbupgrd/backupset/2013_02_14/annnf0_tag20130214t1450
27_0.264.807375029 tag=TAG20130214T145027 comment=NONE
channel ORA_DISK_1: backup set complete, elapsed time: 00:00:01
Finished backup at 14-FEB-13
```

```
Starting Control File and SPFILE Autobackup at 14-FEB-13
piece
handle=+FRA/dbupgrd/autobackup/2013_02_14/s_807375029.265.807375
031 comment=NONE
Finished Control File and SPFILE Autobackup at 14-FEB-13

RMAN> exit
$
```

At the end of the full database backup, you have:

- Two backup sets of archived logs: one completed BEFORE the data files backup and one completed AFTER the data files backup because modifications in the data files may have been applied by applications. The backup sets are stored in +FRA/dbupgrd/backupset/20yy_mm_dd
- One backup set of the five data files stored in +FRA/dbupgrd/backupset/20yy_mm_dd
- One backup set of the control file and SPFILE stored in +FRA/dbupgrd/autobackup/20yy_mm_dd

Practice A-5: Upgrading a Database by Using DBUA

Overview

In this practice, you will now upgrade your 11g dbupgrd database instance to 12c.

Tasks

1. Check that you are logged in as the oracle UNIX user.

```
$ id  
uid=54329(oracle) gid=54330(oinstall)  
groups=54330(oinstall),54323(operator),54324(backupdba),54325(dgdba)  
,54326(kmdba),54327(asmdba),54331(dba)  
context=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023  
$
```

2. Release resources by shutting down instances other than the ASM instance and the dbupgrd database instance.

- a. Shut down orcl.

```
$ . oraenv  
ORACLE_SID = [dbupgrd] ? orcl  
The Oracle base remains unchanged with value /u01/app/oracle  
  
$ sqlplus / as sysdba  
  
SQL> SHUTDOWN IMMEDIATE  
Database closed.  
Database dismounted.  
ORACLE instance shut down.  
SQL> exit  
$
```

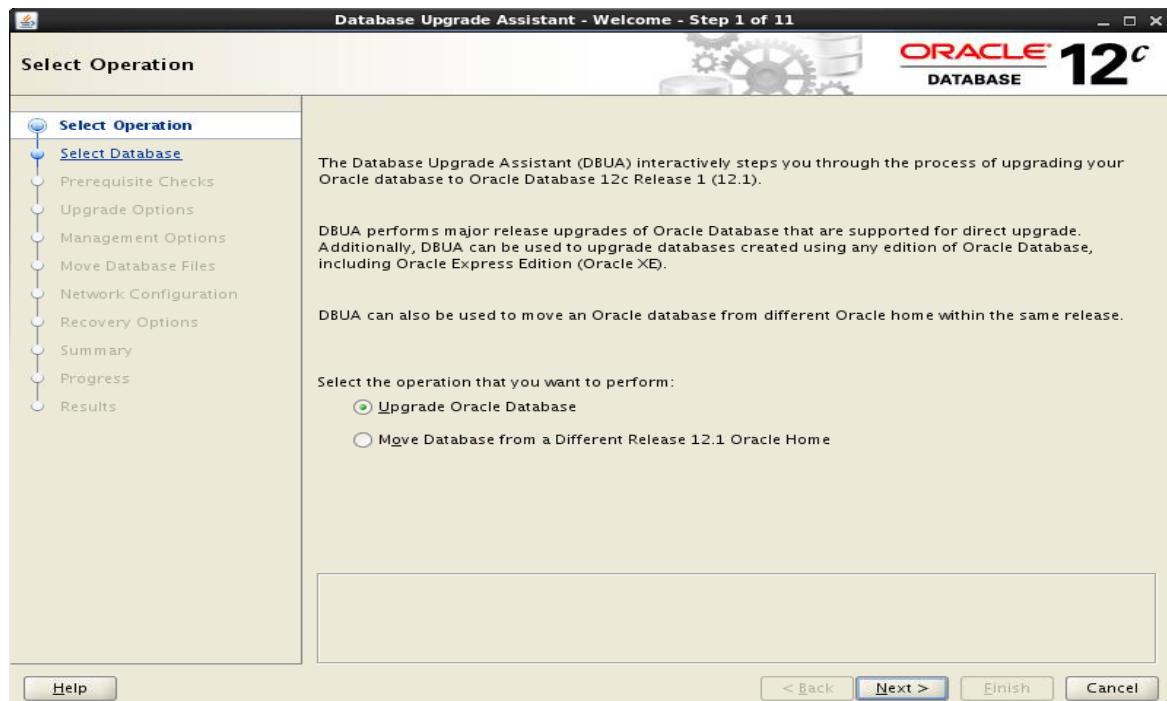
- b. Shut down cdb1.

```
$ . oraenv  
ORACLE_SID = [orcl] ? cdb1  
The Oracle base remains unchanged with value /u01/app/oracle  
  
$ sqlplus / as sysdba  
  
SQL> SHUTDOWN IMMEDIATE  
Database closed.  
Database dismounted.  
ORACLE instance shut down.  
SQL> exit  
$
```

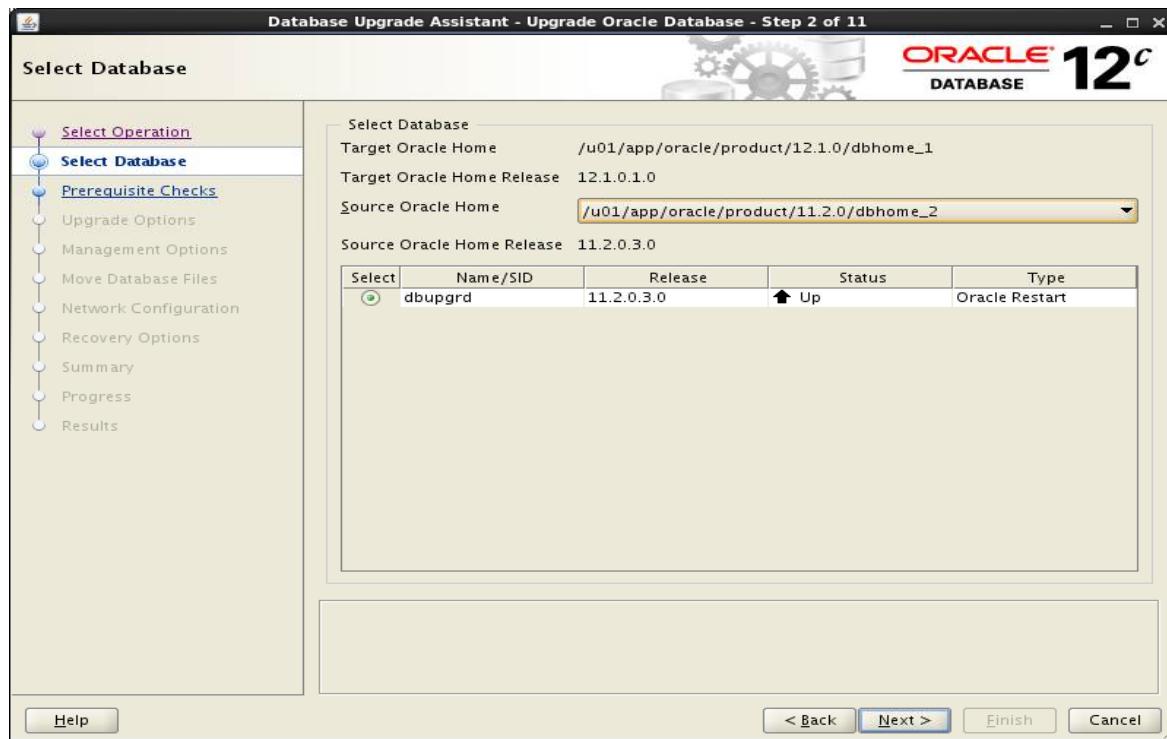
3. Launch DBUA from the Oracle home where the new database software has been installed.

```
$ echo $ORACLE_HOME  
/u01/app/oracle/product/12.1.0/dbhome_1  
$ dbua
```

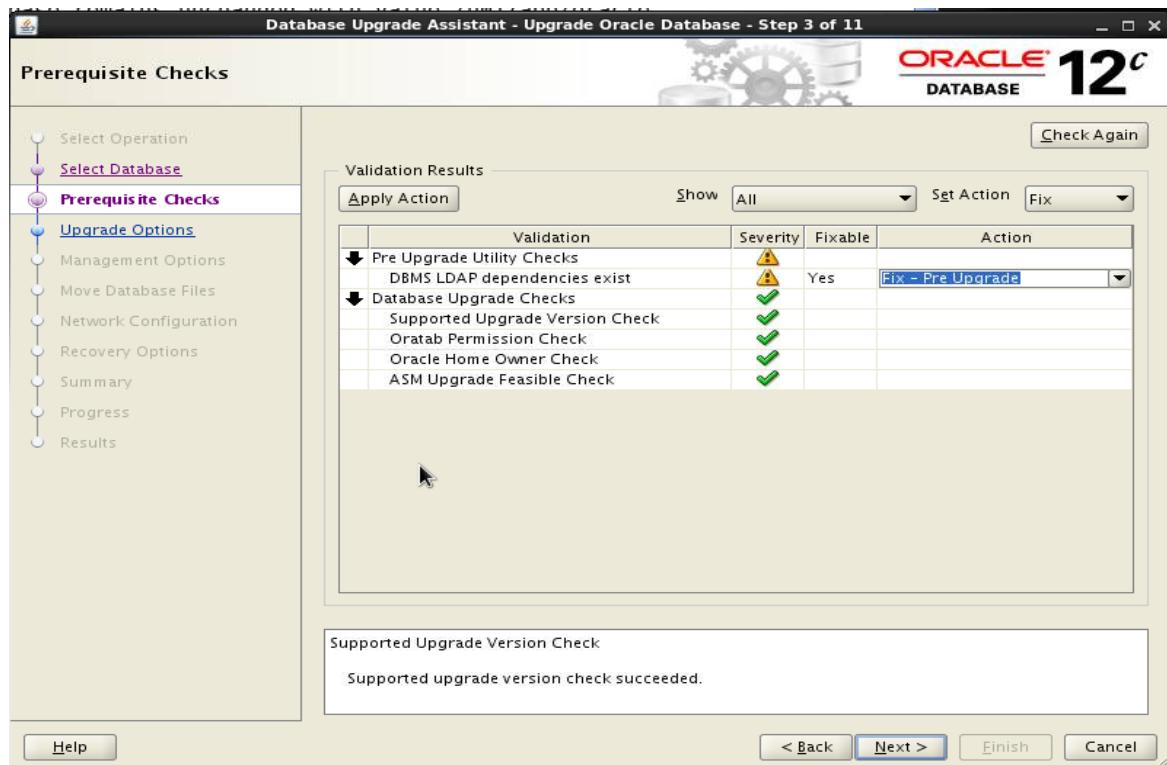
- a. In the Select Operation page, keep the preselected value Upgrade Oracle Database and click **Next**.



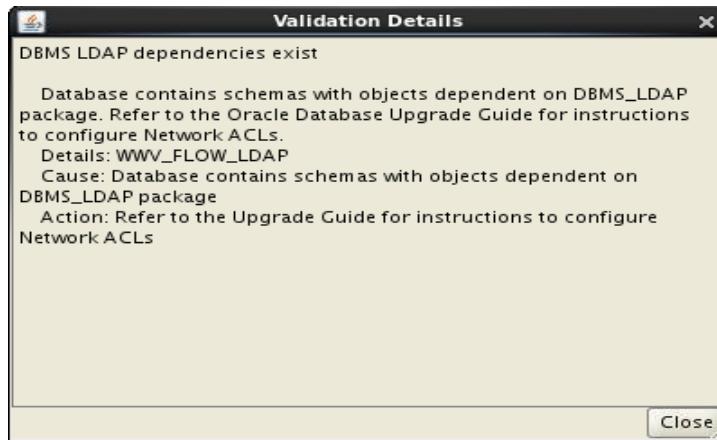
- b. In the Select Database page, verify that the Target Oracle Home is /u01/app/oracle/product/12.1.0/dbhome_1 for the new Oracle Database 12c. From the Source Oracle Home drop-down list, choose the /u01/app/oracle/product/11.2.0/dbhome_2 for the old Oracle Database 11g. The dbupgrd entry appears. Click **Next**.



- c. In the Prerequisite Checks page, the automatic validation starts and shows progressively the validation result of each prerequisite. Click each of the Validation entries to get a more detailed list.

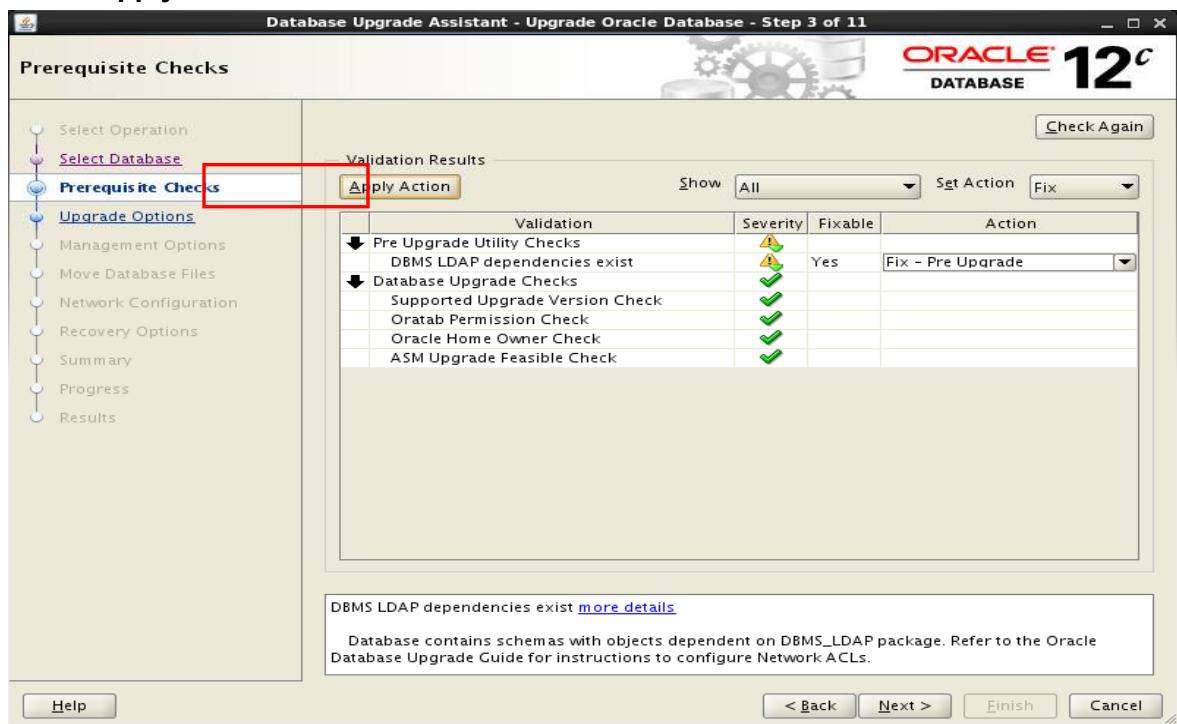


- One check results in a warning and can be fixed. Examine in detail the DBMS_LDAP dependencies that exist. Select the “DBMS_LDAP dependencies exist” entry. Click **more details**.

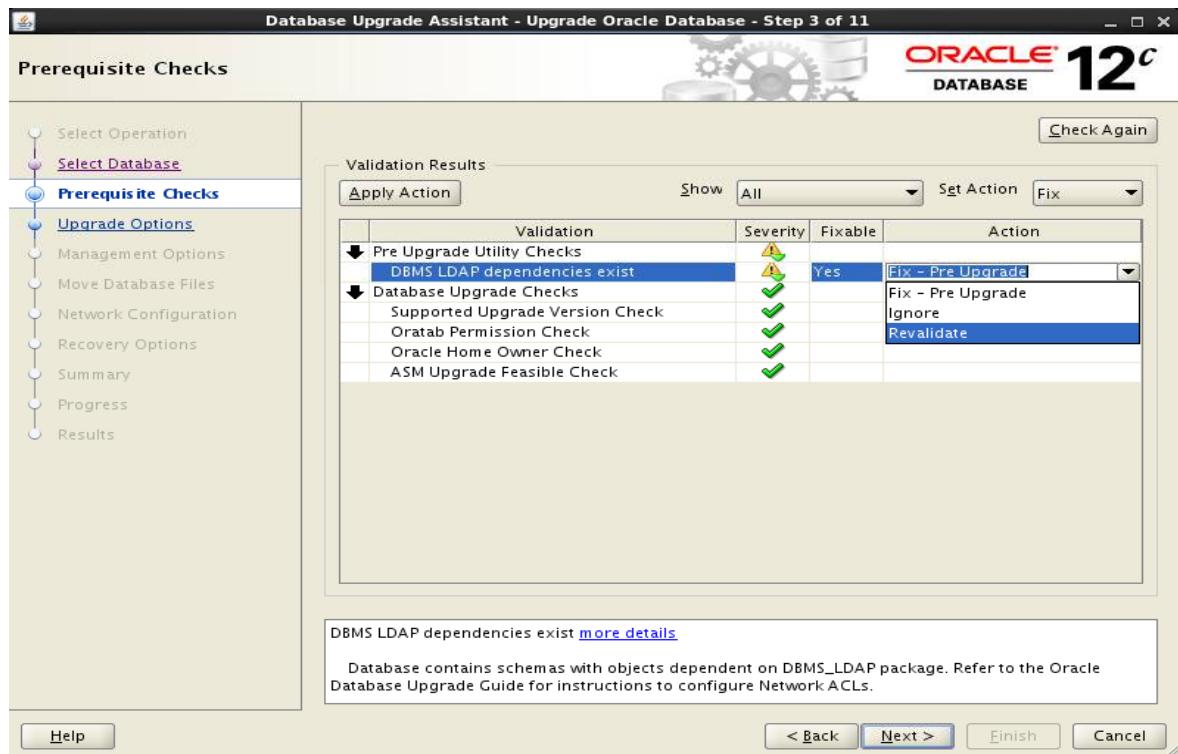


In the previous practice, you did not add this object in the ACL privileged list. Click **Close**.

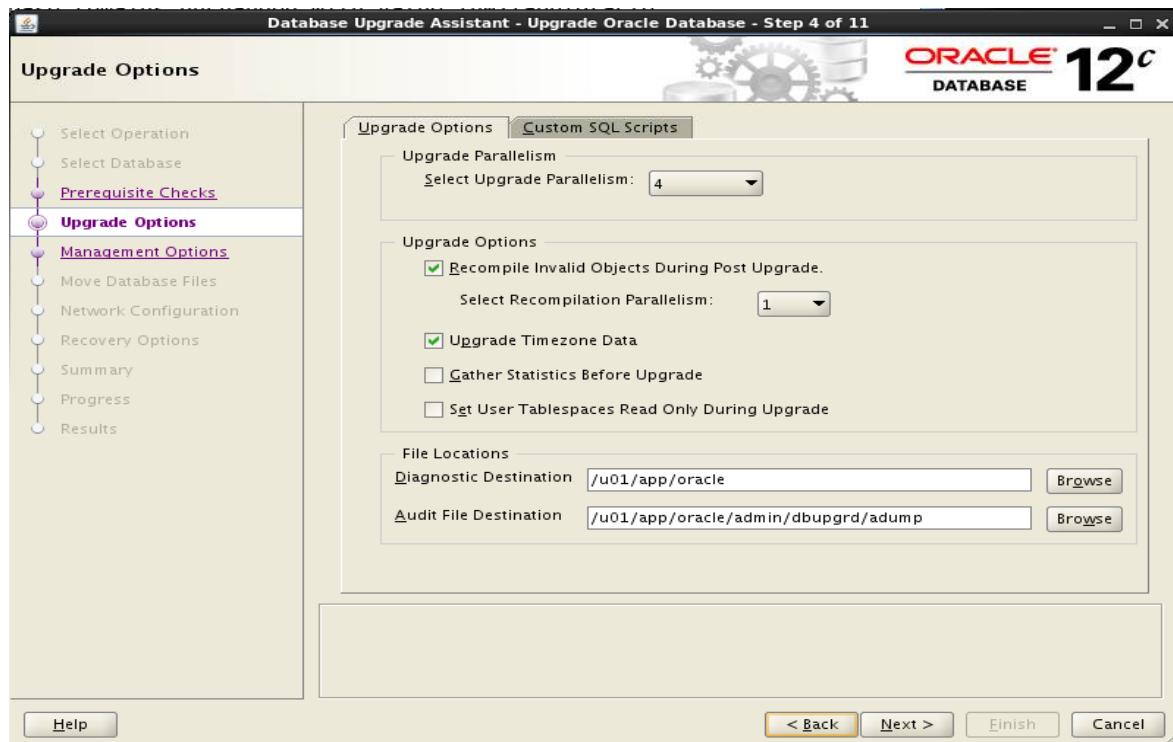
- Click **Apply Action** to fix the issue.



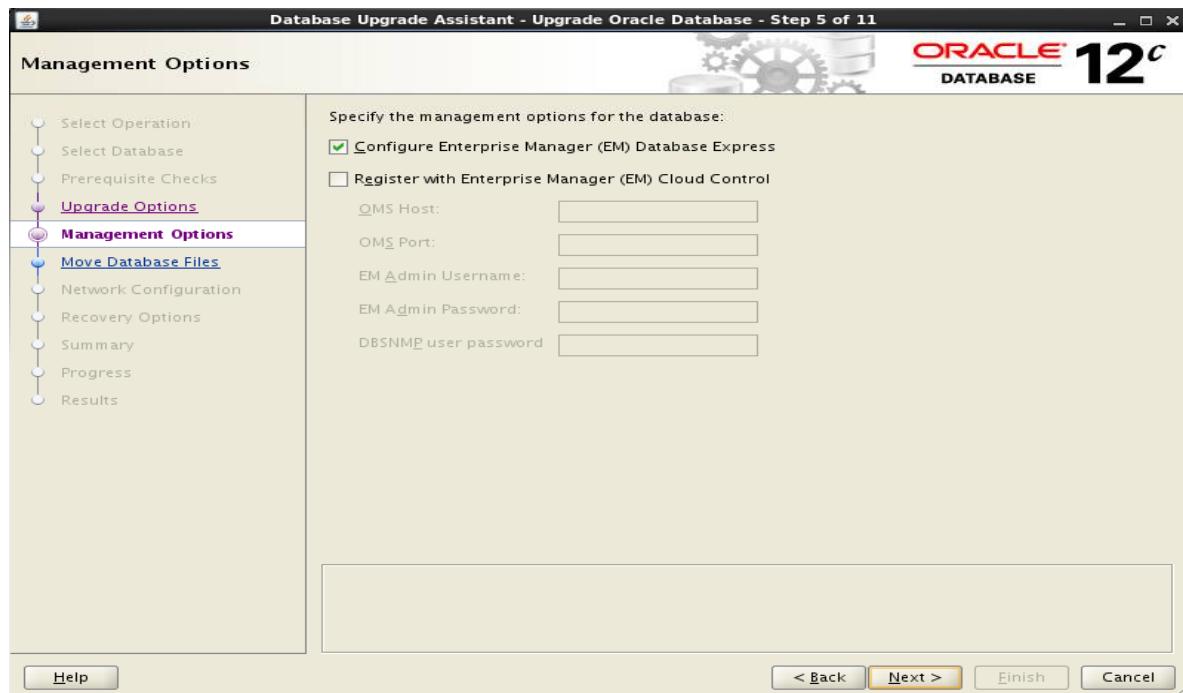
- In Action, choose **Revalidate**. Click **Apply Action**.



- The issue persists but does not prevent the upgrade from progressing. You can fix this issue after the upgrade. In Action, choose **Ignore**. Click **Next**.
- d. In the Upgrade Options, by default, the Upgrade Parallelism is set to the number of CPUs or 2 if the number of CPUs is less than 4. You can adjust this default value by selecting a new value from the Degree of Parallelism drop-down list. DBUA will recompile all invalid PL/SQL modules after the upgrade is complete.
Select **Upgrade Timezone Data**. This is an operation recommended to be executed after the upgrade in the Pre-Upgrade Information Tool output log file. You can choose to update the Time zone data file version during the upgrade. You already gathered the statistics before the upgrade. Click **Next**.

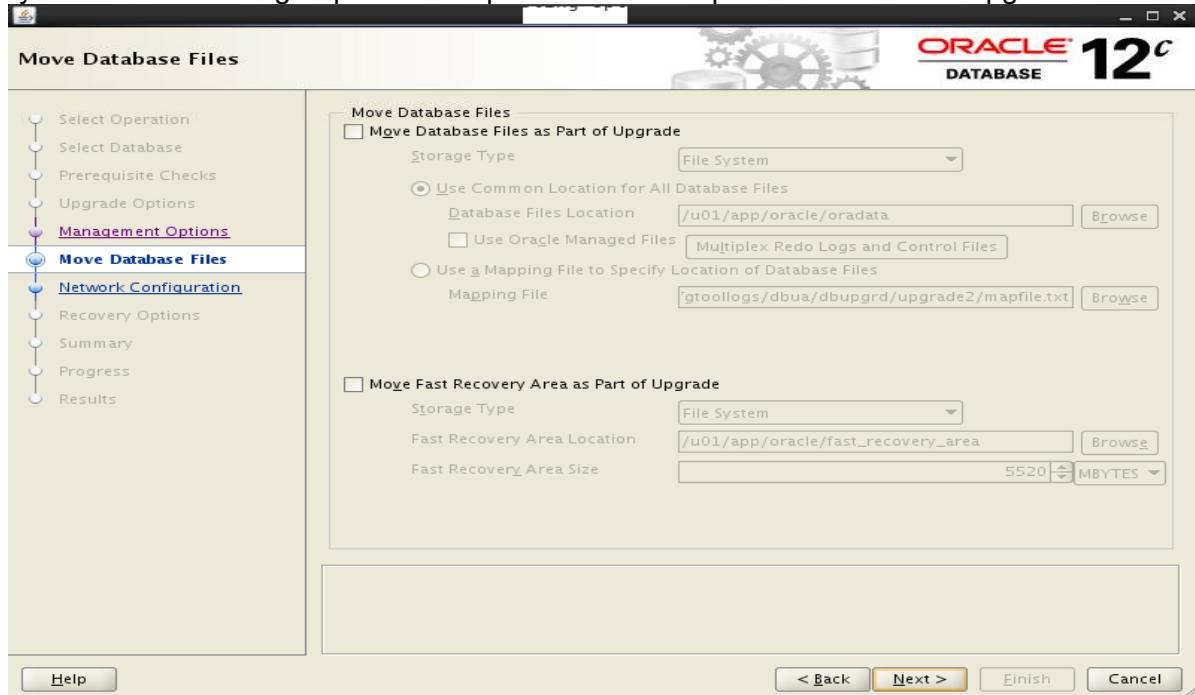


- e. In the Management Options page, click **Next**.

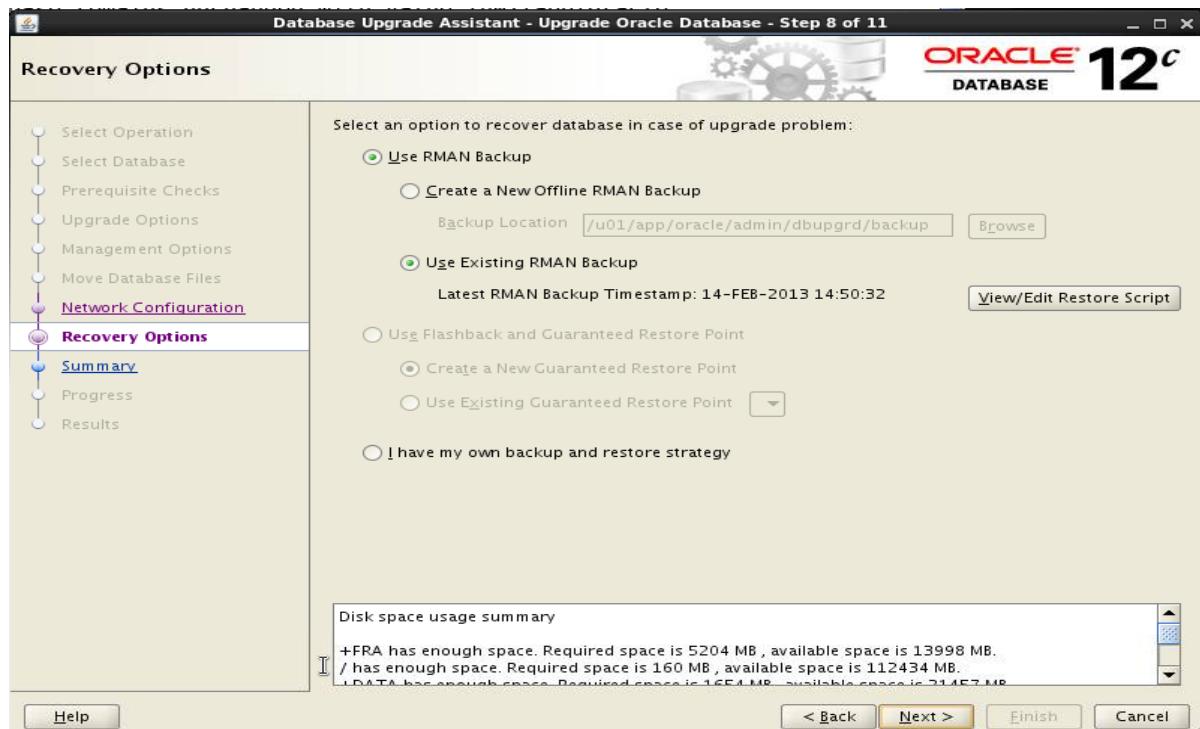


- f. In the Move Database Files page, click **Next** because you do not want to move the database files to another type of storage such as a file system or other disk groups. You do not want to set the FRA location to another type of storage such as a file

system or other disk groups. These operations can be performed after the upgrade.



- g. In the Network Configuration page, select the entry of the LISTENER with port 1521. Click **Next**.
- h. In the Recovery Options, select Use Existing RMAN Backup. You already backed up the database in the previous practice. Click **Next**.



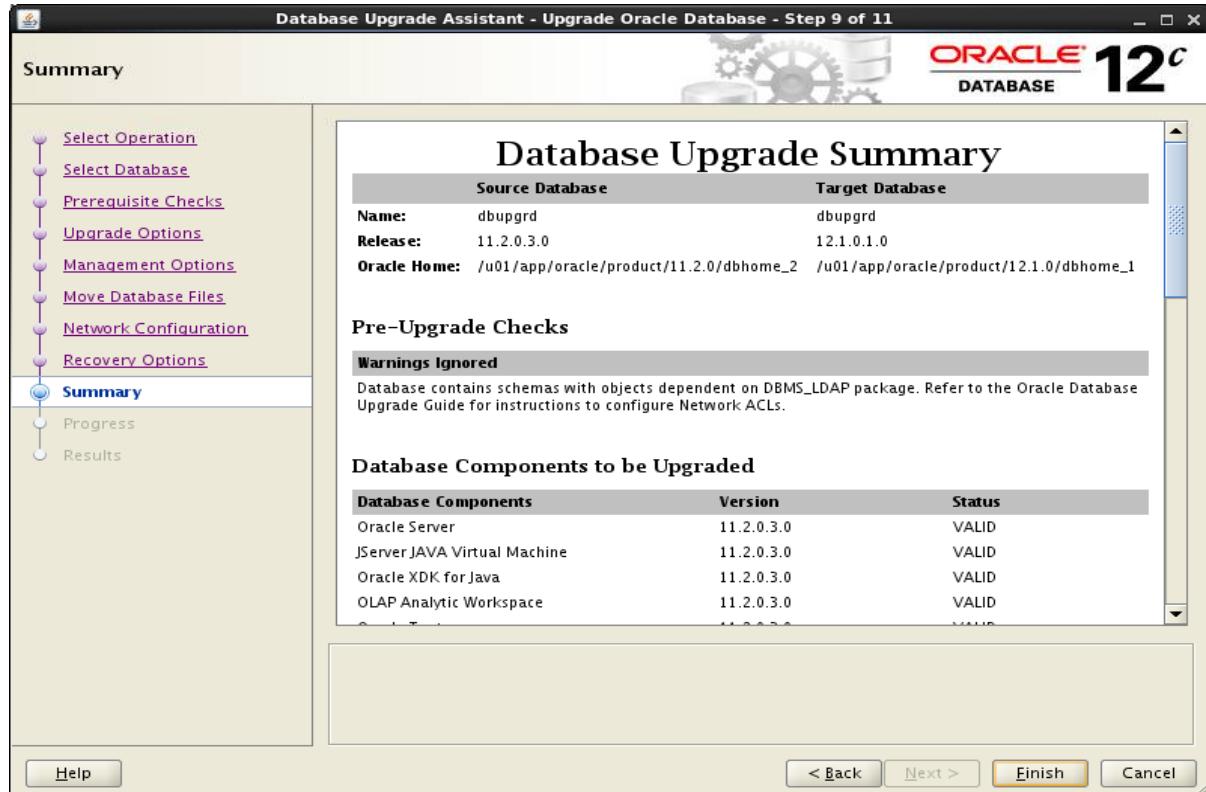
Read carefully the message in the box at the bottom of Recovery Options page. If it says: "You do not have enough disk space for the archive logs and flashback logs generated during upgrade." you would click **more details**. The Validation Details page would explicitly provide the disk space required for the archive logs and

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

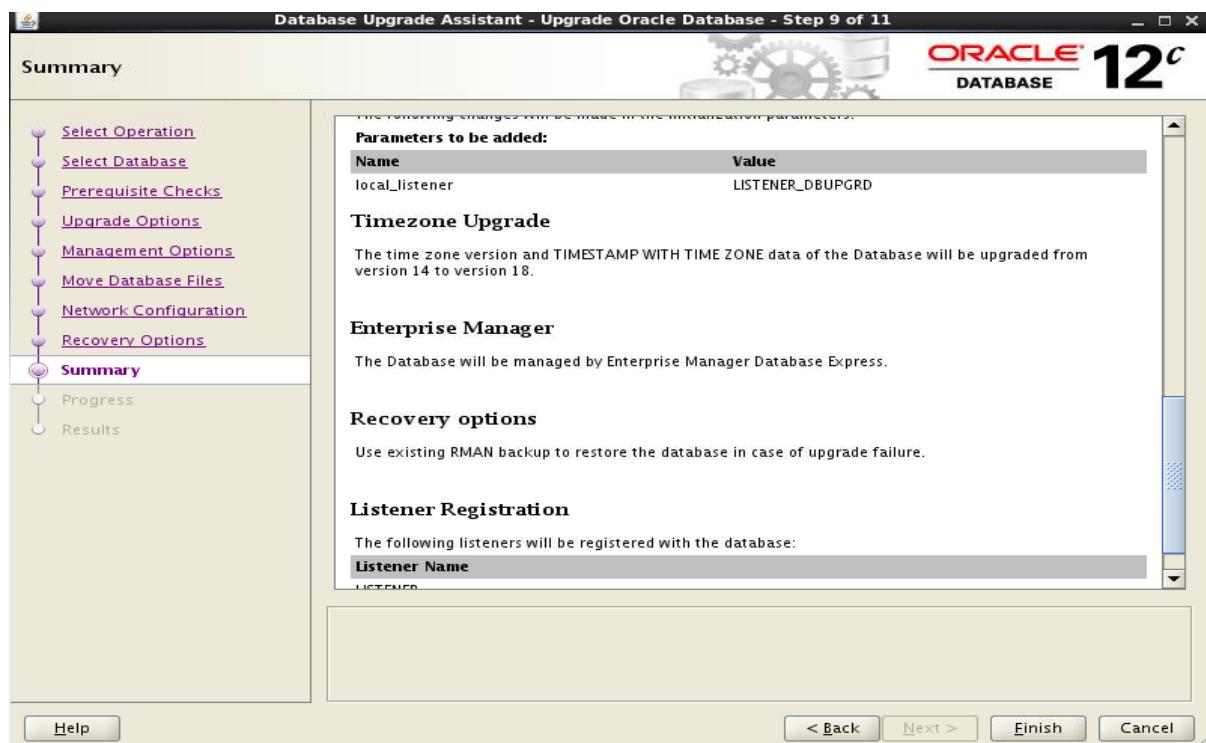
flashback logs generated during upgrade. In another oracle user UNIX session, connected to the dbupgrd instance, you would increase the

DB_RECOVERY_FILE_DEST_SIZE to the required size, and click **Close**. Click **Next**.

- i. In the Summary page, you can view the source and target Database Releases, Oracle Homes, the warnings ignored, the status of the components that will be upgraded, and the version of the time zone data.

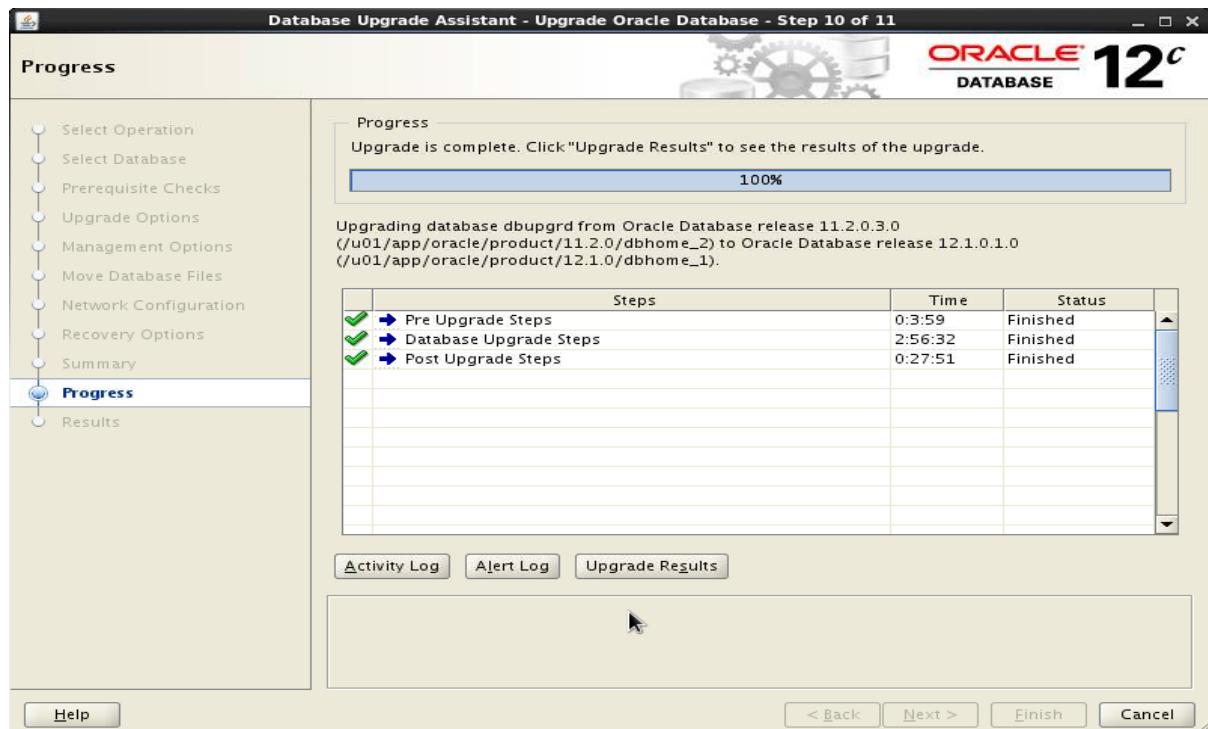


Other types of information are displayed.



Click **Finish**. It can take 2 hours. During the upgrade phase, DBUA runs catctl.pl, which runs the upgrade processes in parallel instead of serially, optimally taking advantage of CPU capacity to decrease down time as much as possible.

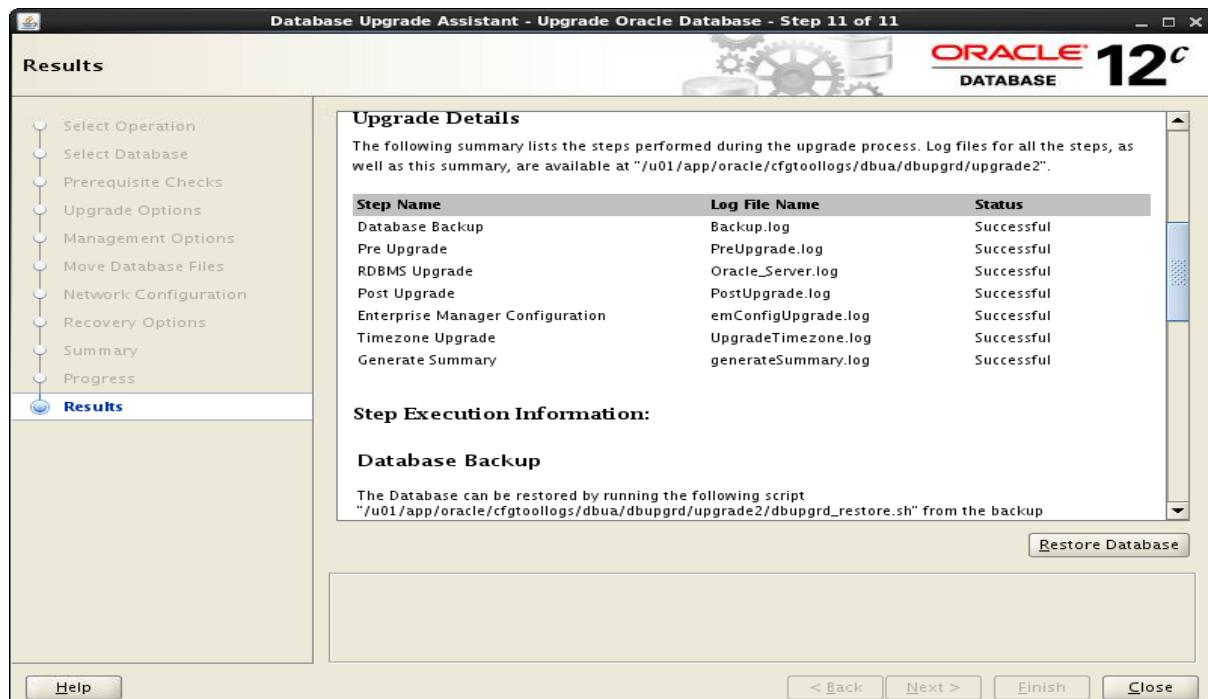
- j. The error message “OLAPSYS does not exist” appears. In the previous, you dropped the OLAP metadata by deleting OLAPSYS schema because you were not using the OLAP Catalog component. Click **Ignore**.
- k. Review the results when the Upgrade Progress is 100% complete. Click **Upgrade Results**.



- In the Results page, the first message is: "The database is ready to use." The dbupgrd database is now running under version 12.1.0.1.0. The Upgrade Results mentions that you ignored the warning related to schemas with objects dependent on DBMS_LDAP package. The operations required to fix this issue require some knowledge about LDAP and are beyond the scope of this course.

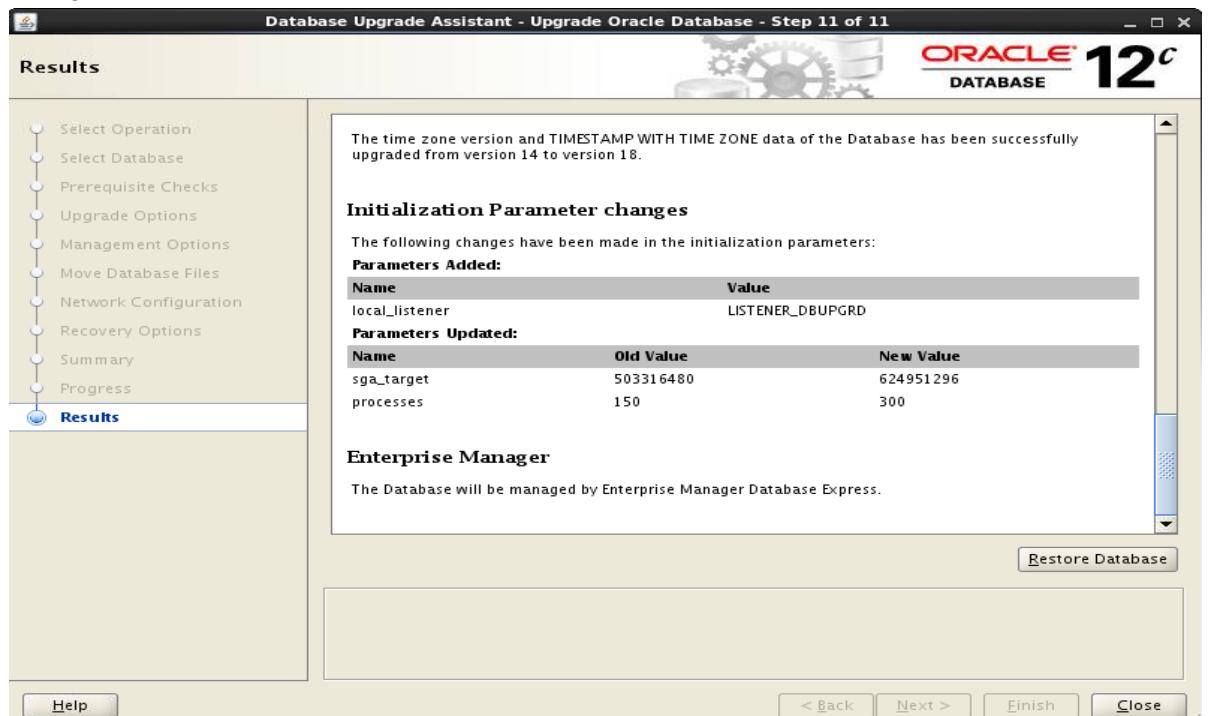


All the upgrade steps completed successfully.



The time zone version and TIMESTAMP WITH TIME ZONE data has been successfully upgraded to version 18.

The SGA_TARGET has been increased and Enterprise Manager Database Express is configured.



Click Close.

- m. The upgrade process successfully completed. The log files are located in the /u01/app/oracle/cfgtoollogs/dbua/dbupgrd/upgrade1 directory.

```
$ dbua
```

```
Database upgrade has been completed successfully, and the  
database is ready to use.
```

```
$
```

Practice A-6: Performing Post-Upgrade Operations

Overview

In this practice, you will:

- Verify that the /etc/oratab file sets the value of ORACLE_HOME to point to the new Oracle home that is created for the new Oracle Database 12c release.
- Show the current state of the Oracle data dictionary
- Check that there are no invalid objects
- Check that the password file exists
- Verify that the SPFILE is created and create the PFILE
- View the content of the alert.log file
- Launch Enterprise Manager Database Express
- Unlock the application users if any are locked
- Upgrade optimizer statistics
- Enable the new extended data type capability
- Enable Database Vault
- Perform a FULL database backup

Tasks

1. Switch to the oracle user Unix session. View the /etc/oratab file.

```
$ cat /etc/oratab|grep dbupgrd
dbupgrd:/u01/app/oracle/product/12.1.0/dbhome_1:N: # 
line added by Agent
$
```

2. Show the current state of the Oracle data dictionary. The status of all components must be VALID.

```
$ . oraenv
ORACLE_SID = [cdb1] ? dbupgrd
The Oracle base remains unchanged with value /u01/app/oracle

$ sqlplus / as sysdba

Connected to:
Oracle Database 12c Enterprise Edition Release 12.1.0.1.0 -
64bit Production
With the Partitioning, Automatic Storage Management, OLAP,
Advanced Analytics and Real Application Testing options

SQL> set lines 80 pages 100
SQL> col comp_id format A12
SQL> col comp_name format A30
SQL> col version format A10
SQL> col status format A6
```

```
SQL> select substr(comp_id,1,15) comp_id, substr(comp_name,1,30)
      comp_name, substr(version,1,10) version, status
    from dba_registry order by modified;

COMP_ID          COMP_NAME                VERSION   STATUS
-----          -----
CATALOG          Oracle Database Catalog Views 12.1.0.1.0 VALID
CATPROC          Oracle Database Packages and T 12.1.0.1.0 VALID
OWM              Oracle Workspace Manager      12.1.0.1.0 VALID
JAVAVM           JServe Java Virtual Machine 12.1.0.1.0 VALID
XML              Oracle XDK                  12.1.0.1.0 VALID
CATJAVA          Oracle Database Java Packages 12.1.0.1.0 VALID
CONTEXT          Oracle Text                 12.1.0.1.0 VALID
XDB              Oracle XML Database        12.1.0.1.0 VALID
ORDIM            Oracle Multimedia          12.1.0.1.0 VALID
XOO              Oracle OLAP API            12.1.0.1.0 VALID
APS              OLAP Analytic Workspace     12.1.0.1.0 VALID
SDO              Spatial                   12.1.0.1.0 VALID
APEX             Oracle Application Express  4.2.0.00.2 VALID
OWB              OWB                      11.2.0.3.0 VALID

14 rows selected.

SQL>
```

3. Check that there are no invalid objects

```
SQL> @$ORACLE_HOME/rdbms/admin/utluobj.sql
.

Oracle Database 12.1 Post-Upgrade Invalid Objects Tool 02-17-
2013 20:44:52

.

This tool lists post-upgrade invalid objects that were not
invalid
prior to upgrade (it ignores pre-existing pre-upgrade invalid
objects).

.

Owner          Object Name          Object Type
.

PL/SQL procedure successfully completed.

SQL> select * from REGISTRY$sys_inv_objs;

no rows selected
```

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

```
SQL> select * from REGISTRY$nonsys_inv_objs;  
  
no rows selected  
  
SQL>
```

4. Check that the password file exists.

- Check the existence of a traditional password file in the \$ORACLE_HOME/dbs directory.

```
SQL> !ls $ORACLE_HOME/dbs/orapwdupgrd  
/u01/app/oracle/product/12.1.0/dbhome_1/dbs/orapwdupgrd  
  
SQL>
```

- Check that the password file is used during an authentication.

```
SQL> connect sys/oracle_4U@localhost:1521/dbupgrd as sysdba  
Connected.  
  
SQL> connect sys/oracle@localhost:1521/dbupgrd as sysdba  
ERROR:  
ORA-01017: invalid username/password; logon denied
```

Warning: You are no longer connected to ORACLE.

```
SQL>
```

- Check that after you remove the password file, the only possible authentication is the OS authentication.

- Remove the file system password file.

```
SQL> !rm $ORACLE_HOME/dbs/orapwdupgrd  
  
SQL> !ls $ORACLE_HOME/dbs/orapwdupgrd  
ls: cannot access  
/u01/app/oracle/product/12.1.0/dbhome_1/dbs/orapwdupgrd: No  
such file or directory  
  
SQL>
```

- Even if you use the password used in task b., the authentication can no longer be performed.

```
SQL> connect sys/oracle_4U@localhost:1521/dbupgrd as sysdba  
ERROR:  
ORA-01017: invalid username/password; logon denied  
  
SQL> connect / as sysdba  
Connected.  
SQL>
```

- d. If your storage solution is ASM, you can create the password file in the DATA disk group. It is possible because you advanced the COMPATIBLE.ASM disk group attribute to 12.1. Be aware that this is not mandatory. Any file stored in a file system can be easily deleted with proper access permissions. Enter oracle_4U when prompted for the password.

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

DB_UNIQUE_NAME
-----
dbupgrd

SQL> !orapwd file='+data/DBUPGRD/orapwdb' dbuniqueusername='dbupgrd'

Enter password for SYS:

SQL> connect sys/oracle_4U@localhost:1521/dbupgrd as sysdba
Connected.

SQL> connect sys/oracle@localhost:1521/dbupgrd as sysdba
ERROR:
ORA-01017: invalid username/password; logon denied

Warning: You are no longer connected to ORACLE.

SQL> exit
$
```

- e. Verify that the password file is created in the DATA disk group. Switch to the grid user terminal window.

```
$ id
uid=54322(grid) gid=54330(oinstall)
groups=54330(oinstall), 54327(asmdba), 54329(asmadmin), 54331(dba)
context=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
$ . oraenv
ORACLE_SID = [grid] ? +ASM
The Oracle base has been changed from /u01/app/oracle to
/u01/app/grid
$ asmcmd
ASMCMD> ls +DATA/DBUPGRD/PASSWORD
pwddbupgrd.267.807938175
ASMCMD>
```

5. Verify that the SPFILE is created.

```
ASMCMD> ls +DATA/DBUPGRD/spfile*
spfileddbupgrd.ora
```

```

spfiledbupgrd.ora_1361440808370
ASMCMD>
ASMCMD> ls -l +DATA/DBUPGRD/spfile*
Type          Redund Striped Time           Sys
NamePARAMETERFILE  UNPROT COARSE   FEB 25 20:00:00  N
spfiledbupgrd.ora =>
+DATA/DBUPGRD/PARAMETERFILE/spfile.260.808327751
PARAMETERFILE  UNPROT COARSE   FEB 25 22:00:00  N
spfiledbupgrd.ora_1361440808370 =>
+DATA/DBUPGRD/PARAMETERFILE/spfile.267.808348937
ASMCMD> exit
$
```

6. Create the PFILE if it does not exist. Return to the oracle user Unix session.

```

$ ls $ORACLE_HOME/dbs/initdbupgrd.ora
/u01/app/oracle/product/12.1.0/dbhome_1/dbs/initdbupgrd.ora
$
```

It already exists. The most current of the parameter file and server parameter file is the SPFILE stored in the DATA disk group.

7. View the contents of the alert log file

```

$ cd $ORACLE_BASE/diag/rdbms/dbupgrd/dbupgrd/trace
$ ls -l a*
-rw-rw---. 1 oracle asmadmin 31009984 Feb 17 18:01
alert_dbupgrd.log
$ vi alert_dbupgrd.log
...
Control file expanded from 594 to 614 blocks for upgrade.
Successful mount of redo thread 1, with mount id 2071604359
Database mounted in Exclusive Mode
Lost write protection disabled
Ping without log force is disabled.
Completed: ALTER DATABASE MOUNT
Fri Feb 15 00:55:36 2013
ALTER DATABASE OPEN MIGRATE
...
Completed: ALTER DATABASE OPEN MIGRATE
...
$
```

8. Launch Enterprise Manager Database Express.

- a. Retrieve the port number.

```

$ sqlplus / as sysdba

SQL> select DBMS_XDB_CONFIG.getHTTPPort from dual;
```

```
GETHTTPPORT
-----
0

SQL> select DBMS_XDB_CONFIG.getHTTPsPort from dual;

GETHTTPSPORT
-----
5500

SQL>
```

- The port number was already used by the ORCL database plugged in CDB1 as PDB_ORCL. Verify. Open another terminal logged in as the oracle UNIX user.

```
$ . oraenv
ORACLE_SID = [oracle] ? cdb1
The Oracle base remains unchanged with value /u01/app/oracle
$
```

- Connect to the pdb_orcl PDB to retrieve the port number used.

```
$ sqlplus sys/oracle_4U@localhost:1521/pdb_orcl as sysdba

SQL> show con_name

CON_NAME
-----
PDB_ORCL
SQL> select DBMS_XDB_CONFIG.getHTTPPort from dual;

GETHTTPPORT
-----
0

SQL> select DBMS_XDB_CONFIG.getHTTPsPort from dual;

GETHTTPSPORT
-----
5500

SQL> exit
$
```

- b. Configure the port 5502 for EM Database Express.

- Check the value of DISPATCHERS parameter in the dbupgrd instance. There must be at least one dispatcher configured for the XMLDB service with the TCP protocol.

```
SQL> show parameter dispatchers
```

NAME	TYPE	VALUE
-	-	-
dispatchers	string	
max_dispatchers	integer	
SQL>		

- Set the DISPATCHERS parameter.

```
SQL> alter system set
dispatchers='(protocol=tcp)(service=dbupgrdXDB)' scope=both;
```

System altered.

```
SQL>
```

- Set the port to 5502 for EM Database Express.

```
SQL> exec DBMS_XDB_CONFIG.setHTTPsPort(5502)
```

PL/SQL procedure successfully completed.

```
SQL> select DBMS_XDB_CONFIG.getHTTPsPort from dual;
```

GETHTTPSPORT

5502

```
SQL>
```

- Restart the instance to register the XMLDB service.

```
SQL> shutdown immediate
```

Database closed.

Database dismounted.

ORACLE instance shut down.

```
SQL> startup
```

ORACLE instance started.

Total System Global Area 663908352 bytes

Fixed Size 2291280 bytes

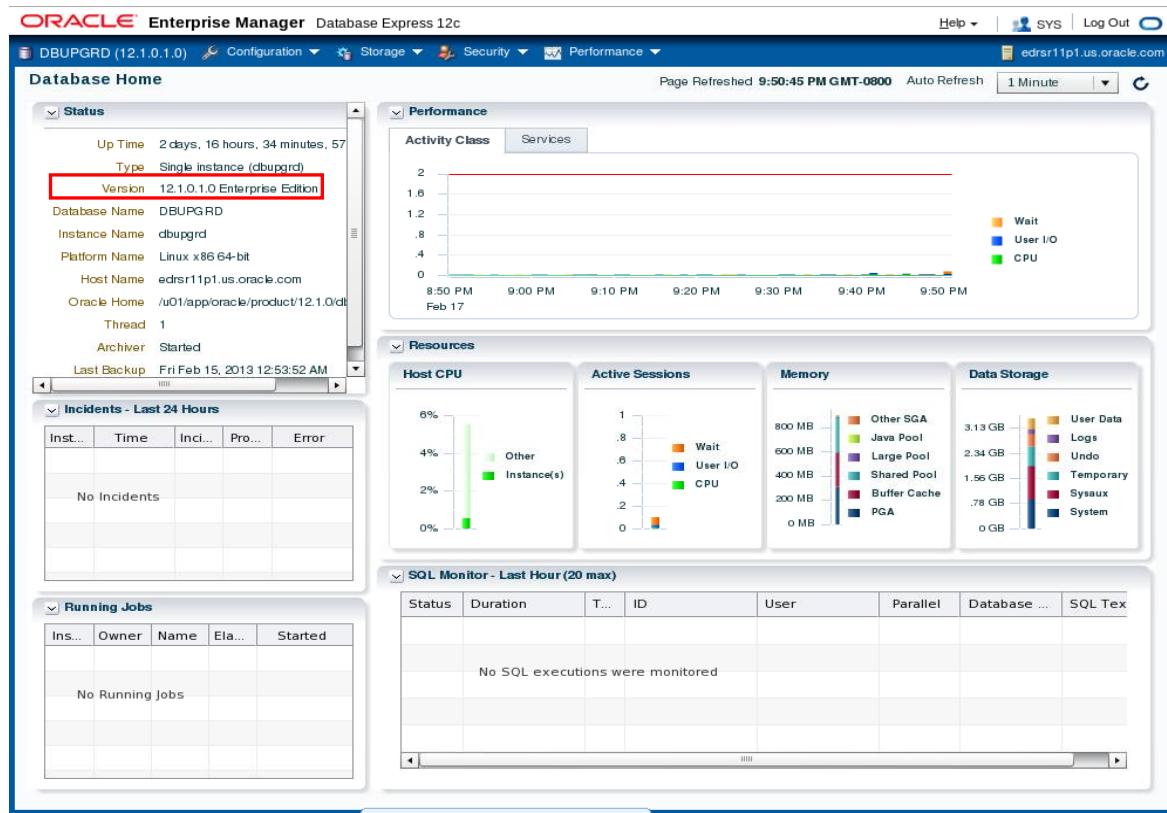
Variable Size 285215152 bytes

Database Buffers 369098752 bytes

Redo Buffers 7303168 bytes

Database mounted.

- ```
Database opened.
SQL> exit
$
```
- c. Check with the listener if the service is registered.
- ```
$ lsnrctl services  
...  
Service "dbupgrd" has 1 instance(s).  
  Instance "dbupgrd", status READY, has 1 handler(s) for this  
  service...  
    Handler(s):  
      "DEDICATED" established:0 refused:0 state:ready  
        LOCAL SERVER  
  
Service "dbupgrdXDB" has 1 instance(s).  
  Instance "dbupgrd", status READY, has 1 handler(s) for this  
  service...  
    Handler(s):  
      "D000" established:0 refused:0 current:0 max:1022  
state:ready  
        DISPATCHER <machine: <your_hostname>, pid: 7436>  
(ADDRESS=(PROTOCOL=tcp) (HOST=<your_hostname>) (PORT=59529))  
The command completed successfully  
$
```
- d. Click the Firefox icon on the top panel (toolbar region) above the desktop to open a browser to access Enterprise Manager Database Express console.
- e. Enter the URL for Enterprise Manager Database Express:
<https://<hostname>:5502/em>. In the current setup, use <https://localhost:5502/em>.
- Most likely, you receive a Secure Connection Failed message and you need to add a security exception. Click **Or you can add an exception**.
 - At the end of the alert box, click **I Understand the Risks**.
 - At the bottom of the page, click **Add Exception**.
 - In the Add Security Exception pop-up window, click **Get Certificate**.
 - Confirm that “Permanently store this exception” is selected in your training environment and click **Confirm Security Exception**.
- f. Enter **sys** in the User Name field and **oracle_4U** in the Password field. Select “as SYSDBA.” Then click **Login**.



9. Unlock the application users if any are locked.

```
SQL> set pages 100
SQL> col username format a30
SQL> select username , account_status from dba_users order by 1;
```

USERNAME	ACCOUNT_STATUS
<hr/>	
ANONYMOUS	EXPIRED & LOCKED
APEX_030200	EXPIRED & LOCKED
APEX_040200	EXPIRED & LOCKED
APEX_PUBLIC_USER	EXPIRED & LOCKED
APPQOSSYS	EXPIRED & LOCKED
AUDSYS	EXPIRED & LOCKED
BI	OPEN
CTXSYS	EXPIRED & LOCKED
DBSNMP	EXPIRED & LOCKED
DIP	EXPIRED & LOCKED
FLOWS_FILES	EXPIRED & LOCKED
GSMADMIN_INTERNAL	EXPIRED & LOCKED
GSMCATUSER	EXPIRED & LOCKED
GSMUSER	EXPIRED & LOCKED
HR	OPEN

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

```

IX                                OPEN
MDDATA                            EXPIRED & LOCKED
MDSYS                            EXPIRED & LOCKED
OE                               OPEN
OJVMSYS                           EXPIRED & LOCKED
ORACLE_OCM                         EXPIRED & LOCKED
ORDDATA                           EXPIRED & LOCKED
ORDPLUGINS                         EXPIRED & LOCKED
ORDSYS                            EXPIRED & LOCKED
OUTLN                             EXPIRED & LOCKED
OWBSYS                            EXPIRED & LOCKED
OWBSYS_AUDIT                      EXPIRED & LOCKED
PM                               OPEN
SCOTT                            EXPIRED & LOCKED
SH                               OPEN
SI_INFORMTN_SCHEMA                OPEN
SPATIAL_CSW_ADMIN_USR              EXPIRED & LOCKED
SPATIAL_WFS_ADMIN_USR              EXPIRED & LOCKED
SYS                              OPEN
SYSBACKUP                          EXPIRED & LOCKED
SYSDG                            EXPIRED & LOCKED
SYSKM                            EXPIRED & LOCKED
SYSTEM                            OPEN
WMSYS                            EXPIRED & LOCKED
XDB                               EXPIRED & LOCKED
XS$NULL                           EXPIRED & LOCKED

41 rows selected.

SQL>

```

10. Upgrade optimizer statistics. When you examined the /u01/app/oracle/cfgtoollogs/dbupgrd/preupgrade/preupgrade.log, the last suggestion was to create statistics on fixed objects two weeks after the upgrade. You can do it now or leave it until the end of the course. You will repeat the procedure regularly as well as the gather_dictionary_stats procedure to update statistics on data dictionary objects. This procedure takes some time.

```

SQL> exec dbms_stats.gather_fixed_objects_stats

PL/SQL procedure successfully completed.

SQL>

```

11. Enable the new extended data type capability. Control the maximum size of VARCHAR2, NVARCHAR2, and RAW data types in SQL. Setting MAX_STRING_SIZE = EXTENDED enables the 32767 byte limit introduced in Oracle Database 12c.

```
SQL> show parameter max_string_size
```

NAME	TYPE	VALUE
max_string_size	string	STANDARD

- a. Attempt to create a table named LONG_VARCHAR with a column defined as VARCHAR2 (32767).

```
SQL> CREATE TABLE long_varchar(id NUMBER,vc VARCHAR2(32767));
CREATE TABLE long_varchar(id NUMBER,vc VARCHAR2(32767))
*
ERROR at line 1:
ORA-00910: specified length too long for its datatype

SQL>
```

- b. Set the MAX_STRING_SIZE parameter to EXTENDED.

```
SQL> alter system set MAX_STRING_SIZE =EXTENDED;
alter system set MAX_STRING_SIZE =EXTENDED
*
ERROR at line 1:
ORA-02097: parameter cannot be modified because specified value
is invalid
ORA-14694: database must be in UPGRADE mode to begin
MAX_STRING_SIZE migration
SQL>
```

- c. Enabling this new capability requires setting the COMPATIBLE instance parameter to 12.0.0.0.0. Even if the database instance was upgraded, it was not required to set the COMPATIBLE instance parameter to 12.0.0.0.0. This allows us to downgrade the database instance if it had been necessary. When the COMPATIBLE instance parameter is set to 12.0.0.0.0, it will become impossible to downgrade the database instance.

```
SQL> alter system set COMPATIBLE=12.0.0.0.0 scope=both;

System altered.

SQL>
```

- d. Restart the database instance in UPGRADE mode.

```
SQL> shutdown immediate
Database closed.
Database dismounted.
```

```
ORACLE instance shut down.  
SQL>  
.   
SQL> startup upgrade  
ORACLE instance started.  
  
Total System Global Area 622149632 bytes  
Fixed Size 2290800 bytes  
Variable Size 276827024 bytes  
Database Buffers 335544320 bytes  
Redo Buffers 7487488 bytes  
Database mounted.  
Database opened.  
SQL>
```

- e. Set the MAX_STRING_SIZE parameter to EXTENDED.

```
SQL> ALTER SYSTEM SET MAX_STRING_SIZE = EXTENDED;  
  
System altered.  
  
SQL>
```

- f. Execute the \$ORACLE_HOME/rdbms/admin/utl32k.sql script as SYSDBA. Be aware that the script may take around one hour to execute. Although, at the very last step when the function is being dropped and the script seems stuck, you can start another session as SYSDBA and perform task 5 to check if the creation of the table is possible with a VARCHAR2 (32767) data type column.

```
SQL> @$ORACLE_HOME/rdbms/admin/utl32k.sql  
  
Session altered.  
  
DOC>#####
DOC> The following statement will cause an "ORA-01722: invalid number" error if the database has not been opened for UPGRADE.  
DOC> Perform a "SHUTDOWN ABORT" and restart using UPGRADE.  
DOC>#####
#  
...  
TIMESTAMP  
-----  
COMP_TIMESTAMP UTLRP_BGN 2013-02-17 22:30:04  
...  
ERRORS DURING RECOMPILE  
-----
```

...
SQL>

- g. Find the invalid objects.

```
SQL> @$ORACLE_HOME/rdbms/admin/utluobj.sql
.
.
Oracle Database 12.1 Post-Upgrade Invalid Objects Tool 02-17-
2013 22:43:48
.
.
This tool lists post-upgrade invalid objects that were not
invalid
prior to upgrade (it ignores pre-existing pre-upgrade invalid
objects).
.
Owner                         Object Name                 Object Type
.
SYS                         DBMS_INTERNAL_LOGSTDBY             PACKAGE
BODY
SYS                         DBMS_SQLTUNE_INTERNAL         PACKAGE BODY
SYS                         PRVT_ADVISOR                 PACKAGE BODY

PL/SQL procedure successfully completed.
```

SQL>

- h. Recompile the package bodies and the view.

```
SQL> alter package DBMS_INTERNAL_LOGSTDBY compile;
.
.
Package altered.
```

```
SQL> alter package DBMS_SQLTUNE_INTERNAL compile;
```

Package altered.

```
SQL> alter package PRVT_ADVISOR compile;
```

Package altered.

SQL>

- i. Restart the database in normal mode.

```
SQL> shutdown immediate
Database closed.
Database dismounted.
```

```
ORACLE instance shut down.

SQL> startup
ORACLE instance started.

Total System Global Area  622149632 bytes
Fixed Size                  2290800  bytes
Variable Size                360713104  bytes
Database Buffers              251658240  bytes
Redo Buffers                  7487488  bytes
Database mounted.
Database opened.
SQL>
```

- j. Verify that MAX_STRING_SIZE is set to EXTENDED.

```
SQL> show parameter MAX_STRING_SIZE

NAME                           TYPE        VALUE
-----
max_string_size                 string      EXTENDED
SQL>
```

- k. Create a table with an extended data type column of 32767 bytes.

```
SQL> CREATE TABLE long_varchar(id NUMBER,vc VARCHAR2(32767));

Table created.

SQL> DESC long_varchar
Name          Null?    Type
-----
ID           NUMBER
VC            VARCHAR2(32767)

SQL> DROP TABLE long_varchar PURGE;

Table dropped.

SQL> exit
$
```

12. How would you reenable Oracle Database Vault if you were using Oracle Database Vault? You would have been instructed to disable it before upgrading your database. You would have to reenable Database Vault using the DVSYS.DBMS_MACADM.ENABLE_DV procedure.

- a. Log in as the Oracle Database Owner (DV_OWNER) account.

```
sqlplus dvo/password
```

- b. Execute the following procedure:
exec DVSYS.DBMS_MACADM.ENABLE_DV
 - c. Check if Oracle Label Security is enabled.
SELECT VALUE FROM V\$OPTION WHERE PARAMETER = 'Oracle Label Security';
 - d. Oracle Label security must be enabled before you can use Database Vault. If it is not enabled, then this query returns FALSE. Then enable it.
CONNECT / AS SYSDBA
exec LBACSYS.CONFIGURE_OLS
exec LBACSYS.OLS_ENFORCEMENT.ENABLE_OLS
 - e. Restart the instance.
connect / as sysdba
shutdown immediate
startup
13. Perform a FULL database backup using RMAN.

```
$ rman target /  
  
RMAN> show all;  
  
using target database control file instead of recovery catalog  
RMAN configuration parameters for database with db_unique_name  
DBUPGRD are:  
CONFIGURE RETENTION POLICY TO REDUNDANCY 1; # default  
CONFIGURE BACKUP OPTIMIZATION OFF; # default  
CONFIGURE DEFAULT DEVICE TYPE TO DISK; # default  
CONFIGURE CONTROLFILE AUTOBACKUP ON;  
...  
  
RMAN> backup database plus archivelog delete all input;  
  
Starting backup at 18-FEB-13  
current log archived  
using target database control file instead of recovery catalog  
allocated channel: ORA_DISK_1  
channel ORA_DISK_1: SID=267 device type=DISK  
channel ORA_DISK_1: starting archived log backup set  
channel ORA_DISK_1: specifying archived log(s) in backup set  
input archived log thread=1 sequence=13 RECID=2 STAMP=807375027  
...  
Finished backup at 18-FEB-13  
  
Starting Control File and SPFILE Autobackup at 18-FEB-13
```

```
piece
handle=+FRA/DBUPGRD/AUTOBACKUP/2013_02_18/s_807675362.412.807675
365 comment=NONE
Finished Control File and SPFILE Autobackup at 18-FEB-13

RMAN> exit
$
```

