

Upgrade Oracle Database from 12.2.0.1 to 19c and convert RAC

Notebook: 19c

Created: 6/25/2020 3:23 PM

Updated: 6/25/2020 3:24 PM

Author: appsdba1981@gmail.com

Upgrade Oracle Database from 12.2.0.1 to 19c

0. Check Compatibility Before Upgrading Oracle Database
1. Environment

PRE-UPGRADE TASKS

2. Backup
3. Run preupgrade script
4. View Preupgrade log
5. Minimum tablespace sizes for upgrade
6. Update INITIALIZATION PARAMETERS
7. Gather DICTIONARY STATS
8. Purge Recyclebin
9. Refresh MVs
10. Run preupgrade_fixups.sql
11. Verify archive log dest size
12. Stop LISTENER
13. Create Flashback Guaranteed Restore Point

UPGRADE TASK

14. Shutdown Database
15. Copy init and password files from 12c to 19c dbs home
16. Startup DB in Upgrade mode
17. Run dbupgrade
18. Startup DB from 19c home

POST-UPGRADE TASKS WHEN DBUA USING

19. Run catcon.pl to start utlrp.sql
20. Run postupgrade_fixups.sql
21. Upgrade Timezone
22. Run utlstmts.sql
23. Run catupst.sql
24. Re-Run postupgrade_fixups.sql
25. Reverify INVALID OBJECTS
26. Drop Restore point
27. Set COMPATIBLE parameter value to 19.0.0
28. Verify DBA_REGISTRY
29. Add TNS Entries in 19c TNS home
30. Password File – orapwCID
31. Edit oratab
32. Back Up the Database

#####

0. Check Compatibility Before Upgrading Oracle Database

 Use oracle support compatibility matrix

1. Environment

#####

Hostname : [RAC1.ORACLE.COM](#)

Database Name : PRODDB

DB VERSION : 12.2.0.1

CDB : NON-CDB, Single Instance

DB Home Path : /u01/app/oracle/product/12.2.0/dbhome_1

Datafile Location : /u01/app/oracle/oradata/PRODDB

Target DB VERSION : 19c (19.3.0.0.0)

Target DB Path : /u01/app/oracle/product/19.0.0/dbhome_1

PRE-UPGRADE TASKS

```
#####
2. Backup
#####
Use RMAN Backup script and take level 0 backup
```

TNS Files

```
[oracle@rac1 ~]$ cd u01/app/oracle/product/12.2.0/dbhome_1/network/admin/
[oracle@rac1 admin]$ cp -p listener.ora sqlnet.ora tnsnames.ora /u01/app/backup/
```

PFILE/SPFILE/PASSWORD (orapwSID) FILES

```
[oracle@rac1 dbs]$ pwd
/u01/app/oracle/product/12.2.0/dbhome_1/dbs
[oracle@rac1 dbs]$ cp -p spfilePRODDB.ora orapwPRODDB /u01/app/backup/
```

INVALID OBJECTS

```
SQL> select count(*) from dba_objects where status='INVALID';
```

```
COUNT(*)
-----
0 <---
```

```
SQL>
```

```
#####
3. Run preupgrade script
#####
```

```
[oracle@rac1 ~]$ mkdir -p /home/oracle/PRODDB/preupgrade
[oracle@rac1 ~]$ . oraenv
    ORACLE_SID = [oracle] ? PRODDB
The Oracle base has been set to /u01/app/oracle
[oracle@rac1 ~]$ /u01/app/oracle/product/12.2.0/dbhome_1/jdk/bin/java -jar
/u01/app/oracle/product/19.0.0/dbhome_1/rdbms/admin/preupgrade.jar FILE DIR /home/oracle/PRODDB/preupgrade
```

```
=====
PREUPGRADE SUMMARY
=====
/home/oracle/PRODDB/preupgrade/preupgrade.log
/home/oracle/PRODDB/preupgrade/preupgrade_fixups.sql
/home/oracle/PRODDB/preupgrade/postupgrade_fixups.sql
```

Execute fixup scripts as indicated below:

Before upgrade:

Log into the database and execute the preupgrade fixups
@/home/oracle/PRODDB/preupgrade/preupgrade_fixups.sql

After the upgrade:

Log into the database and execute the postupgrade fixups
@/home/oracle/PRODDB/preupgrade/postupgrade_fixups.sql

```
#####
4. View Preupgrade log
#####
```

```
[oracle@rac1 ~]$ cat /home/oracle/PRODDB/preupgrade/preupgrade.log
Report generated by Oracle Database Pre-Upgrade Information Tool Version
```

19.0.0.0 Build: 1

Upgrade-To version: 19.0.0.0

```
=====
Status of the database prior to upgrade
=====
```

Database Name: PRODDB

Container Name: PRODDB

Container ID: 0

Version: 12.2.0.1.0

DB Patch Level: No Patch Bundle applied

Compatible: 12.2.0

Blocksize: 8192

Platform: Linux x86 64-bit
Timezone File: 26
Database log mode: ARCHIVELOG
 Readonly: FALSE
 Edition: EE

Oracle Component	Upgrade Action	Current Status
Oracle Server	[to be upgraded]	VALID
JServer JAVA Virtual Machine	[to be upgraded]	VALID
Oracle XDK for Java	[to be upgraded]	VALID
Real Application Clusters	[to be upgraded]	OPTION OFF
Oracle Workspace Manager	[to be upgraded]	VALID
OLAP Analytic Workspace	[to be upgraded]	VALID
Oracle Label Security	[to be upgraded]	VALID
Oracle Database Vault	[to be upgraded]	VALID
Oracle Text	[to be upgraded]	VALID
Oracle XML Database	[to be upgraded]	VALID
Oracle Java Packages	[to be upgraded]	VALID
Oracle Multimedia	[to be upgraded]	VALID
Oracle Spatial	[to be upgraded]	VALID
Oracle OLAP API	[to be upgraded]	VALID

=====

BEFORE UPGRADE

=====

REQUIRED ACTIONS

=====

None

RECOMMENDED ACTIONS

=====

1. (AUTOFIXUP) Gather stale data dictionary statistics prior to database upgrade in off-peak time using:

EXECUTE DBMS_STATS.GATHER_DICTIONARY_STATS;

Dictionary statistics do not exist or are stale (not up-to-date).

Dictionary statistics help the Oracle optimizer find efficient SQL execution plans and are essential for proper upgrade timing. Oracle recommends gathering dictionary statistics in the last 24 hours before database upgrade.

For information on managing optimizer statistics, refer to the 12.2.0.1 Oracle Database SQL Tuning Guide.

2. (AUTOFIXUP) Gather statistics on fixed objects prior the upgrade.

None of the fixed object tables have had stats collected.

Gathering statistics on fixed objects, if none have been gathered yet, is recommended prior to upgrading.

For information on managing optimizer statistics, refer to the 12.2.0.1 Oracle Database SQL Tuning Guide.

INFORMATION ONLY

=====

3. To help you keep track of your tablespace allocations, the following AUTOEXTEND tablespaces are expected to successfully EXTEND during the upgrade process.

Tablespace	Min Size		
	Size	For Upgrade	
SYS AUX	460 MB	500 MB	
SYSTEM	800 MB	912 MB	
TEMP	32 MB	150 MB	
UNDOTBS1	70 MB	439 MB	

Minimum tablespace sizes for upgrade are estimates.

4. Ensure there is additional disk space in LOG_ARCHIVE_DEST_1 for at least 4618 MB of archived logs. Check alert log during the upgrade that there is no write error to the destination due to lack of disk space.

Archiving cannot proceed if the archive log destination is full during upgrade.

Archive Log Destination:

Parameter : LOG_ARCHIVE_DEST_1
Destination : /u01/app/archive/PRODDDB

The database has archiving enabled. The upgrade process will need free disk space in the archive log destination(s) to generate archived logs to.

5. Check the Oracle Backup and Recovery User's Guide for information on how to manage an RMAN recovery catalog schema.

If you are using a version of the recovery catalog schema that is older than that required by the RMAN client version, then you must upgrade the catalog schema.

It is good practice to have the catalog schema the same or higher version than the RMAN client version you are using.

ORACLE GENERATED FIXUP SCRIPT

=====

All of the issues in database PRODDDB which are identified above as BEFORE UPGRADE "(AUTOFIXUP)" can be resolved by executing the following

SQL>@/home/oracle/PRODDDB/preupgrade/preupgrade_fixups.sql

=====

AFTER UPGRADE

=====

REQUIRED ACTIONS

=====

None

RECOMMENDED ACTIONS

=====

6. Upgrade the database time zone file using the DBMS_DST package.

The database is using time zone file version 26 and the target 19 release ships with time zone file version 32.

Oracle recommends upgrading to the desired (latest) version of the time zone file. For more information, refer to "Upgrading the Time Zone File and Timestamp with Time Zone Data" in the 19 Oracle Database Globalization Support Guide.

7. (AUTOFIXUP) Gather dictionary statistics after the upgrade using the command:

EXECUTE DBMS_STATS.GATHER_DICTIONARY_STATS;

Oracle recommends gathering dictionary statistics after upgrade.

Dictionary statistics provide essential information to the Oracle optimizer to help it find efficient SQL execution plans. After a database upgrade, statistics need to be re-gathered as there can now be tables that have significantly changed during the upgrade or new tables that do not have statistics gathered yet.

8. Gather statistics on fixed objects after the upgrade and when there is a representative workload on the system using the command:

EXECUTE DBMS_STATS.GATHER_FIXED_OBJECTS_STATS;

This recommendation is given for all preupgrade runs.

Fixed object statistics provide essential information to the Oracle optimizer to help it find efficient SQL execution plans. Those statistics are specific to the Oracle Database release that generates them, and can be stale upon database upgrade.

For information on managing optimizer statistics, refer to the 12.2.0.1 Oracle Database SQL Tuning Guide.

ORACLE GENERATED FIXUP SCRIPT

```
=====
```

All of the issues in database PRODDB which are identified above as AFTER UPGRADE "(AUTOFIXUP)" can be resolved by executing the following

```
SQL>@/home/oracle/PRODDB/preupgrade/postupgrade_fixups.sql
```

```
#####
5. Verify tablespace sizes for upgrade
#####
** Tablespace Auto extend ON and max size need to be set, hence no action taken.
```

TABLESPACE_NAME	AUT	FILE_NAME	TOTAL_SPACE	FREE_SPACE	Free%	MAX_SPACE
SYSAUX	YES	/u01/app/oracle/oradata/PRODDB/sysaux01.dbf	460	23	5.08	31.9999847
SYSTEM	YES	/u01/app/oracle/oradata/PRODDB/system01.dbf	800	4	.48	31.9999847
UNDOTBS1	YES	/u01/app/oracle/oradata/PRODDB/undotbs01.dbf	70	3	4.2	31.9999847
USERS	YES	/u01/app/oracle/oradata/PRODDB/users01.dbf	5	4	80	31.9999847

			1335	34	127.999939	

```
#####
6. Update INITIALIZATION PARAMETERS
```

```
#####
In this test scenario, noting to update as per preupgrade.log. Hence no action taken. But need to check if there is an action recommended
```

```
#####
7. Gather DICTIONARY STATS
#####
SQL> SET ECHO ON;
SQL> SET SERVEROUTPUT ON;
SQL> EXECUTE DBMS_STATS.GATHER_DICTIONARY_STATS;
```

PL/SQL procedure successfully completed.

```
SQL>
```

```
#####
8. Purge Recyclebin
#####
SQL> PURGE DBA_RECYCLEBIN;
```

DBA Recyclebin purged.

```
SQL>
```

```
#####
9. Refresh MVs
#####
*** Before upgrading Oracle Database, you must wait until all materialized views have completed refreshing.
```

Run the following SQL query:

```
SQL> SELECT o.name FROM sys.obj$ o, sys.user$ u, sys.sum$ s WHERE o.type# = 42 AND bitand(s.mflags, 8) =8;
```

```
no rows selected
```

```
SQL>
```

```
SQL> declare
list_failures integer(3) :=0;
```

```
begin
DBMS_MVIEW.REFRESH_ALL_MVIEWS(list_failures,'C','');
end;
/
```

PL/SQL procedure successfully completed.

SQL>

```
#####
10. Run preupgrade_fixups.sql
#####
```

```
SQL> @/home/oracle/PRODDB/preupgrade/preupgrade_fixups.sql
```

```
SQL> REM
```

```
SQL> REM Oracle PRE-Upgrade Fixup Script
```

```
SQL> REM
```

```
SQL> REM Auto-Generated by: Oracle Preupgrade Script
```

```
SQL> REM Version: 19.0.0.0 Build: 1
```

```
SQL> REM Generated on: 2020-06-20 20:45:02
```

```
SQL> REM
```

```
SQL> REM Source Database: PRODDB
```

```
SQL> REM Source Database Version: 12.2.0.1.0
```

```
SQL> REM For Upgrade to Version: 19.0.0.0
```

```
SQL> REM
```

```
SQL>
```

```
SQL> REM
```

```
SQL> REM Setup Environment
```

```
SQL> REM
```

```
SQL> SET ECHO OFF SERVEROUTPUT ON FORMAT WRAPPED TAB OFF LINESIZE 200;
```

```
Executing Oracle PRE-Upgrade Fixup Script
```

```
Auto-Generated by: Oracle Preupgrade Script
```

```
Version: 19.0.0.0 Build: 1
```

```
Generated on: 2020-06-20 20:45:02
```

```
For Source Database: PRODDB
```

```
Source Database Version: 12.2.0.1.0
```

```
For Upgrade to Version: 19.0.0.0
```

Preup	Preupgrade		
Action	Issue Is		
Number	Preupgrade Check Name	Remedied	Further DBA Action
1.	dictionary_stats	YES	None.
2.	pre_fixed_objects	YES	None.
3.	tablespaces_info	NO	Informational only. Further action is optional.
4.	min_archive_dest_size	NO	Informational only. Further action is optional.
5.	rman_recovery_version	NO	Informational only. Further action is optional.

The fixup scripts have been run and resolved what they can. However, there are still issues originally identified by the preupgrade that have not been remedied and are still present in the database. Depending on the severity of the specific issue, and the nature of the issue itself, that could mean that your database is not ready for upgrade. To resolve the outstanding issues, start by reviewing the preupgrade_fixups.sql and searching it for the name of the failed CHECK NAME or Preupgrade Action Number listed above. There you will find the original corresponding diagnostic message from the preupgrade which explains in more detail what still needs to be done.

PL/SQL procedure successfully completed.

SQL>

```
#####
11. Verify archive log dest size
#####
```

```
*** Please verify free space on ALL LOG_ARCHIVE_DEST_locations including ALL standby destinations
```

```

SQL> archive log list
Database log mode      Archive Mode
Automatic archival     Enabled
Archive destination     /u01/app/archive/PRODDDB
Oldest online log sequence  1
Next log sequence to archive  2
Current log sequence     2
SQL>
SQL> !df -h /u01/app/archive/PRODDDB
Filesystem  Size Used Avail Use% Mounted on
/dev/sda5   67G  35G  33G  52% /u01 <-----

SQL>

#####
12. Stop LISTENER
#####
[oracle@rac1 ~]$ ps -ef | grep tns
root    15  2 0 20:01 ?    00:00:00 [netns]
oracle  3943  1 0 20:08 ?    00:00:00 /u01/app/oracle/product/12.2.0/dbhome_1/bin/tnslsnr LISTENER_PRODDDB -inherit
oracle 16771 3093 0 21:03 pts/1  00:00:00 grep --color=auto tns
[oracle@rac1 ~]$ lsnrctl stop LISTENER_PRODDDB

LSNRCTL for Linux: Version 12.2.0.1.0 - Production on 28-JAN-2020 21:03:14

Copyright (c) 1991, 2016, Oracle. All rights reserved.

Connecting to (DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)(HOST=RAC1.ORACLE.COM)(PORT=1524)))
The command completed successfully
[oracle@rac1 ~]$ ps -ef | grep tns
root    15  2 0 20:01 ?    00:00:00 [netns]
oracle 16812 3093 0 21:03 pts/1  00:00:00 grep --color=auto tns
[oracle@rac1 ~]$


#####
13. Create Flashback Guaranteed Restore Point
#####
*** NO need to enable Flashback Database from 11.2.0.1 onwards
*** Database MUST be in Archive Log mode
*** MUST NOT change the compatible parameter to higher version

SQL> select flashback_on from v$database;

FLASHBACK_ON
-----
NO <-----

SQL> select name,open_mode,log_mode from v$database;

NAME  OPEN_MODE      LOG_MODE
-----+
PRODDDB  READ WRITE      ARCHIVELOG

SQL> show parameter compatible

NAME          TYPE      VALUE
-----
compatible      string    12.2.0 <-----
noncdb_compatible  boolean   FALSE
SQL>
SQL> show parameter recovery

NAME          TYPE      VALUE
-----
db_recovery_file_dest      string
db_recovery_file_dest_size  big integer 0
recovery_parallelism        integer   0
remote_recovery_file_dest  string
SQL>
SQL>!mkdir -p /u01/app/oracle/fast_recovery_area

SQL> alter system set db_recovery_file_dest_size=10G;

```

System altered.

```
SQL> alter system set db_recovery_file_dest='/u01/app/oracle/fast_recovery_area';
```

System altered.

```
SQL> show parameter recovery
```

NAME	TYPE	VALUE
db_recovery_file_dest	string	/u01/app/oracle/fast_recovery_area
db_recovery_file_dest_size	big integer	10G
recovery_parallelism	integer	0
remote_recovery_file_dest	string	

```
SQL>
```

```
SQL> select * from V$restore_point;
```

no rows selected

```
SQL>
```

```
SQL> create restore point pre_upgrade_guarantee flashback database;
```

Restore point created.

```
SQL>
```

```
SQL> col name for a20
col GUARANTEE_FLASHBACK_DATABASE for a10
col TIME for a60
set lines 190
select NAME,GUARANTEE_FLASHBACK_DATABASE,TIME from V$restore_point;
```

NAME	GUARANTEE_TIME
PRE_UPGRADE	YES 20-JUN-20 09.05.50.000000000 PM

```
#####
#UPGRADE TASK
#####
14. Shutdown Database
#####
SQL> SELECT NAME,OPEN_MODE FROM V$DATABASE;
```

NAME	OPEN_MODE
PRODDB	READ WRITE

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

```
#####
15. Copy init and password files from 12c to 19c dbs home
#####
[oracle@rac1 ~]$ cd /u01/app/oracle/product/12.2.0/dbhome_1/dbs
[oracle@rac1 dbs]$ ls -ltr *PRODDB*
-rw-r----. 1 oracle oinstall 24 Jan 28 20:08 lkPRODDB
-rw-r----. 1 oracle oinstall 3584 Jan 28 20:11 orapwPRODDB <---
-rw-r----. 1 oracle oinstall 3584 Jan 28 21:05 spfilePRODDB.ora <---
-rw-rw---. 1 oracle oinstall 1544 Jan 28 21:07 hc_PRODDB.dat
[oracle@rac1 dbs]$
[oracle@rac1 dbs]$ cp orapwPRODDB spfilePRODDB.ora /u01/app/oracle/product/19.0.0/dbhome_1/dbs/
[oracle@rac1 dbs]$ ls -ltr /u01/app/oracle/product/19.0.0/dbhome_1/dbs/*PRODDB*
-rw-r----. 1 oracle oinstall 3584 Jun 20 21:10 /u01/app/oracle/product/19.0.0/dbhome_1/dbs/spfilePRODDB.ora
-rw-r----. 1 oracle oinstall 3584 Jun 20 21:10 /u01/app/oracle/product/19.0.0/dbhome_1/dbs/orapwPRODDB
[oracle@rac1 dbs]$
```

```
#####
16. Startup DB in Upgrade mode from 19c home
#####
[oracle@rac1 ~]$ export ORACLE_HOME=/u01/app/oracle/product/19.0.0/dbhome_1
[oracle@rac1 ~]$ export ORACLE_SID=PRODDB
[oracle@rac1 ~]$ PATH=/u01/app/oracle/product/19.0.0/dbhome_1/bin:$PATH; export PATH
[oracle@rac1 ~]$ which sqlplus
/u01/app/oracle/product/19.0.0/dbhome_1/bin/sqlplus
[oracle@rac1 ~]$ sqlplus / as sysdba
```

SQL*Plus: Release 19.0.0.0.0 - Production on Tue Jun 20 21:13:24 2020
Version 19.4.0.0.0

Copyright (c) 1982, 2019, Oracle. All rights reserved.

Connected to an idle instance.

```
SQL> startup upgrade;
ORACLE instance started.
```

```
Total System Global Area 1560277408 bytes
Fixed Size          8896928 bytes
Variable Size       939524096 bytes
Database Buffers    603979776 bytes
Redo Buffers        7876608 bytes
```

Database mounted.

Database opened.

```
SQL>
```

```
SQL> select name,open_mode,cdb,version,status from v$database,v$instance;
```

NAME	OPEN_MODE	CDB VERSION	STATUS
PRODDB	READ WRITE	NO 19.0.0.0.0	OPEN MIGRATE <---

```
SQL>
```

```
#####
17. Run dbupgrade
#####
You can run the upgrade using either of the following commands. The second is actually just a shorthand for the former.
```

```
[oracle@rac1 ~]$ mkdir -p /home/oracle/whileupgrade
[oracle@rac1 ~]$ cd /u01/app/oracle/product/19.0.0/dbhome_1/rdbms/admin
[oracle@rac1 ~]$ nohup /u01/app/oracle/product/19.0.0/dbhome_1/perl/bin/perl catctl.pl -l /home/oracle/whileupgrade -n 4
catupgrd.sql &
---OR---
[oracle@rac1 ~]$ cd /u01/app/oracle/product/19.0.0/dbhome_1/bin/
[oracle@rac1 bin]$ ls -ltr dbupgrade
-rwxr-x---. 1 oracle oinstall 3136 Apr 17 2019 dbupgrade
[oracle@rac1 bin]$
[oracle@rac1 bin]$ nohup ./dbupgrade &
[oracle@rac1 bin]$ jobs -l
[1]+ 22584 Running           nohup ./dbupgrade & <----
[oracle@rac1 bin]$ disown
[oracle@rac1 bin]$ ps -ef | grep -i catctl.pl
oracle 22589 22584 0 21:33 pts/1 00:00:03 /u01/app/oracle/product/19.0.0/dbhome_1/perl/bin/perl -
/u01/app/oracle/product/19.0.0/dbhome_1/perl/lib /u01/app/oracle/product/19.0.0/dbhome_1/rdbms/admin/catctl.pl
/u01/app/oracle/product/19.0.0/dbhome_1/rdbms/admin/catupgrd.sql
oracle 24991 3093 0 21:40 pts/1 00:00:00 grep --color=auto -i catctl.pl
[oracle@rac1 bin]$
```

Monitor upgrade log under below location

```
[oracle@rac1 upgrade20200128213345]$ cd
/u01/app/oracle/product/19.0.0/dbhome_1/cfgtoollogs/PRODDB/upgrade20200128213345
[oracle@rac1 upgrade20200128213345]$ ls *.log
catupgrd1.log
catupgrd2.log
catupgrd3.log
catupgrd0.log
```

```
[oracle@rac1 upgrade20200128213345]$
```

```
tail -f catupgrd0.log
```

```

tail -f catupgrd1.log
tail -f catupgrd2.log
tail -f catupgrd3.log

[oracle@rac1 ~]$ cd /u01/app/oracle/product/19.0.0/dbhome_1/bin/
[oracle@rac1 bin]$ more nohup.out

Argument list for [/u01/app/oracle/product/19.0.0/dbhome_1/rdbms/admin/catctl.pl]
For Oracle internal use only A = 0
Run in          c = 0
Do not run in   C = 0
Input Directory d = 0
Echo OFF        e = 1
Simulate        E = 0
Forced cleanup  F = 0
Log Id          i = 0
Child Process   I = 0
Log Dir          l = 0
Priority List Name L = 0
Upgrade Mode active M = 0
SQL Process Count n = 0
SQL PDB Process Count N = 0
Open Mode Normal o = 0
Start Phase     p = 0
End Phase       P = 0
Reverse Order   r = 0
AutoUpgrade Resume R = 0
Script           s = 0
Serial Run      S = 0
RO User Tablespaces T = 0
Display Phases   y = 0
Debug catcon.pm  z = 0
Debug catctl.pl  Z = 0

catctl.pl VERSION: [19.0.0.0]
  STATUS: [Production]
  BUILD: [RDBMS_19.4.0.0.0DBRU_LINUX.X64_190626]

/u01/app/oracle/product/19.0.0/dbhome_1/rdbms/admin/orahome = [/u01/app/oracle/product/19.0.0/dbhome_1]
/u01/app/oracle/product/19.0.0/dbhome_1/bin/orabasehome = [/u01/app/oracle/product/19.0.0/dbhome_1]
catctlGetOraBaseLogDir = [/u01/app/oracle/product/19.0.0/dbhome_1]

Analyzing file /u01/app/oracle/product/19.0.0/dbhome_1/rdbms/admin/catupgrd.sql

Log file directory = [/tmp/cfgtoollogs/upgrade20200620213334]

catcon::set_log_file_base_path: ALL catcon-related output will be written to
[/tmp/cfgtoollogs/upgrade20200620213334/catupgrd_catcon_22589.lst]

catcon::set_log_file_base_path: catcon: See [/tmp/cfgtoollogs/upgrade20200620213334/catupgrd*.log] files for output generated by
scripts

catcon::set_log_file_base_path: catcon: See [/tmp/cfgtoollogs/upgrade20200620213334/catupgrd_*lst] files for spool files, if any

Number of Cpus      = 1
Database Name      = PRODDB
DataBase Version   = 12.2.0.1.0
catcon::set_log_file_base_path: ALL catcon-related output will be written to
[/u01/app/oracle/product/19.0.0/dbhome_1/cfgtoollogs/PRODDB/upgrade20200620213345/catupgrd_cat
con_22589.lst]

catcon::set_log_file_base_path: catcon: See
[/u01/app/oracle/product/19.0.0/dbhome_1/cfgtoollogs/PRODDB/upgrade20200620213345/catupgrd*.log] files for output
generated by
scripts

catcon::set_log_file_base_path: catcon: See
[/u01/app/oracle/product/19.0.0/dbhome_1/cfgtoollogs/PRODDB/upgrade20200620213345/catupgrd_*lst] files for spool files, if
any

```

Log file directory = [/u01/app/oracle/product/19.0.0/dbhome_1/cfgtoollogs/PRODDB/upgrade20200620213345]

Parallel SQL Process Count = 4

Components in [PRODDB]

 Installed [APS CATALOG CATJAVA CATPROC CONTEXT DV JAVAVM OLS ORDIM OWM SDO XDB XML XOQ]

Not Installed [APEX EM MGW ODM RAC WK]

Phases [0-107] Start Time:[21:33:56]

***** Executing Change Scripts *****

Serial Phase #:0 [PRODDB] Files:1 Time: 32s

***** Catalog Core SQL *****

Serial Phase #:1 [PRODDB] Files:5 Time: 56s

Restart Phase #:2 [PRODDB] Files:1 Time: 1s

***** Catalog Tables and Views *****

Parallel Phase #:3 [PRODDB] Files:19 Time: 32s

Restart Phase #:4 [PRODDB] Files:1 Time: 1s

***** Catalog Final Scripts *****

Serial Phase #:5 [PRODDB] Files:7 Time: 24s

***** Catproc Start *****

Serial Phase #:6 [PRODDB] Files:1 Time: 21s

***** Catproc Types *****

Serial Phase #:7 [PRODDB] Files:2 Time: 14s

Restart Phase #:8 [PRODDB] Files:1 Time: 1s

***** Catproc Tables *****

Parallel Phase #:9 [PRODDB] Files:67 Time: 47s

Restart Phase #:10 [PRODDB] Files:1 Time: 1s

***** Catproc Package Specs *****

Serial Phase #:11 [PRODDB] Files:1 Time: 94s

Restart Phase #:12 [PRODDB] Files:1 Time: 1s

***** Catproc Procedures *****

Parallel Phase #:13 [PRODDB] Files:94 Time: 19s

Restart Phase #:14 [PRODDB] Files:1 Time: 1s

Parallel Phase #:15 [PRODDB] Files:121 Time: 29s

Restart Phase #:16 [PRODDB] Files:1 Time: 1s

Serial Phase #:17 [PRODDB] Files:22 Time: 7s

Restart Phase #:18 [PRODDB] Files:1 Time: 1s

***** Catproc Views *****

Parallel Phase #:19 [PRODDB] Files:32 Time: 37s

Restart Phase #:20 [PRODDB] Files:1 Time: 1s

Serial Phase #:21 [PRODDB] Files:3 Time: 19s

Restart Phase #:22 [PRODDB] Files:1 Time: 1s

Parallel Phase #:23 [PRODDB] Files:25 Time: 217s

Restart Phase #:24 [PRODDB] Files:1 Time: 2s

Parallel Phase #:25 [PRODDB] Files:12 Time: 120s

Restart Phase #:26 [PRODDB] Files:1 Time: 1s

Serial Phase #:27 [PRODDB] Files:1 Time: 0s

Serial Phase #:28 [PRODDB] Files:3 Time: 6s

Serial Phase #:29 [PRODDB] Files:1 Time: 0s

Restart Phase #:30 [PRODDB] Files:1 Time: 0s

***** Catproc CDB Views *****

Serial Phase #:31 [PRODDB] Files:1 Time: 3s

Restart Phase #:32 [PRODDB] Files:1 Time: 1s

Serial Phase #:34 [PRODDB] Files:1 Time: 0s

***** Catproc PLBs *****

Serial Phase #:35 [PRODDB] Files:294 Time: 48s

Serial Phase #:36 [PRODDB] Files:1 Time: 0s

Restart Phase #:37 [PRODDB] Files:1 Time: 1s

Serial Phase #:38 [PRODDB] Files:6 Time: 8s

Restart Phase #:39 [PRODDB] Files:1 Time: 1s

***** Catproc DataPump *****

Serial Phase #:40 [PRODDB] Files:3 Time: 59s

Restart Phase #:41 [PRODDB] Files:1 Time: 0s

***** Catproc SQL *****

Parallel Phase #:42 [PRODDB] Files:13 Time: 131s

Restart Phase #:43 [PRODDB] Files:1 Time: 1s

Parallel Phase #:44 [PRODDB] Files:11 Time: 20s

Restart Phase #:45 [PRODDB] Files:1 Time: 1s

Parallel Phase #:46 [PRODDB] Files:3 Time: 4s

Restart Phase #:47 [PRODDB] Files:1 Time: 2s

***** Final Catproc scripts *****

Serial Phase #:48 [PRODDB] Files:1 Time: 10s

Restart Phase #:49 [PRODDDB] Files:1 Time: 0s
 ***** Final RDBMS scripts *****
 Serial Phase #:50 [PRODDDB] Files:1 Time: 5s
 ***** Upgrade Component Start *****
 Serial Phase #:51 [PRODDDB] Files:1 Time: 3s
 Restart Phase #:52 [PRODDDB] Files:1 Time: 1s
 ***** Upgrading Java and non-Java *****
 Serial Phase #:53 [PRODDDB] Files:2 Time: 382s
 ***** Upgrading XDB *****
 Restart Phase #:54 [PRODDDB] Files:1 Time: 2s
 Serial Phase #:56 [PRODDDB] Files:3 Time: 10s
 Serial Phase #:57 [PRODDDB] Files:3 Time: 7s
 Parallel Phase #:58 [PRODDDB] Files:10 Time: 6s
 Parallel Phase #:59 [PRODDDB] Files:25 Time: 10s
 Serial Phase #:60 [PRODDDB] Files:4 Time: 12s
 Serial Phase #:61 [PRODDDB] Files:1 Time: 0s
 Serial Phase #:62 [PRODDDB] Files:32 Time: 7s
 Serial Phase #:63 [PRODDDB] Files:1 Time: 0s
 Parallel Phase #:64 [PRODDDB] Files:6 Time: 9s
 Serial Phase #:65 [PRODDDB] Files:2 Time: 22s
 Serial Phase #:66 [PRODDDB] Files:3 Time: 32s
 ***** Upgrading ORDIM *****
 Restart Phase #:67 [PRODDDB] Files:1 Time: 0s
 Serial Phase #:69 [PRODDDB] Files:1 Time: 5s
 Parallel Phase #:70 [PRODDDB] Files:2 Time: 45s
 Restart Phase #:71 [PRODDDB] Files:1 Time: 1s
 Parallel Phase #:72 [PRODDDB] Files:2 Time: 4s
 Serial Phase #:73 [PRODDDB] Files:2 Time: 5s
 ***** Upgrading SDO *****
 Restart Phase #:74 [PRODDDB] Files:1 Time: 1s
 Serial Phase #:76 [PRODDDB] Files:1 Time: 63s
 Serial Phase #:77 [PRODDDB] Files:2 Time: 6s
 Restart Phase #:78 [PRODDDB] Files:1 Time: 1s
 Serial Phase #:79 [PRODDDB] Files:1 Time: 55s
 Restart Phase #:80 [PRODDDB] Files:1 Time: 0s
 Parallel Phase #:81 [PRODDDB] Files:3 Time: 115s
 Restart Phase #:82 [PRODDDB] Files:1 Time: 3s
 Serial Phase #:83 [PRODDDB] Files:1 Time: 13s
 Restart Phase #:84 [PRODDDB] Files:1 Time: 1s
 Serial Phase #:85 [PRODDDB] Files:1 Time: 13s
 Restart Phase #:86 [PRODDDB] Files:1 Time: 1s
 Parallel Phase #:87 [PRODDDB] Files:4 Time: 155s
 Restart Phase #:88 [PRODDDB] Files:1 Time: 1s
 Serial Phase #:89 [PRODDDB] Files:1 Time: 4s
 Restart Phase #:90 [PRODDDB] Files:1 Time: 1s
 Serial Phase #:91 [PRODDDB] Files:2 Time: 12s
 Restart Phase #:92 [PRODDDB] Files:1 Time: 1s
 Serial Phase #:93 [PRODDDB] Files:1 Time: 2s
 Restart Phase #:94 [PRODDDB] Files:1 Time: 1s
 ***** Upgrading ODM, WK, EXF, RUL, XOQ *****
 Serial Phase #:95 [PRODDDB] Files:1 Time: 17s
 Restart Phase #:96 [PRODDDB] Files:1 Time: 0s
 ***** Final Component scripts *****
 Serial Phase #:97 [PRODDDB] Files:1 Time: 5s
 ***** Final Upgrade scripts *****
 Serial Phase #:98 [PRODDDB] Files:1 Time: 520s
 ***** Migration *****
 Serial Phase #:99 [PRODDDB] Files:1 Time: 3s
 *** End PDB Application Upgrade Pre-Shutdown ***
 Serial Phase #:100 [PRODDDB] Files:1 Time: 2s
 Serial Phase #:101 [PRODDDB] Files:1 Time: 0s
 Serial Phase #:102 [PRODDDB] Files:1 Time: 108s
 ***** Post Upgrade *****
 Serial Phase #:103 [PRODDDB] Files:1 Time: 15s
 ***** Summary report *****
 Serial Phase #:104 [PRODDDB] Files:1 Time: 3s
 *** End PDB Application Upgrade Post-Shutdown **
 Serial Phase #:105 [PRODDDB] Files:1 Time: 2s
 Serial Phase #:106 [PRODDDB] Files:1 Time: 0s
 Serial Phase #:107 [PRODDDB] Files:1 Time: 30s

Grand Total Time: 2800s

LOG FILES: (/u01/app/oracle/product/19.0.0/dbhome_1/cfgtoollogs/PRODDB/upgrade20200620213345/catupgrd*.log)

Upgrade Summary Report Located in:
/u01/app/oracle/product/19.0.0/dbhome_1/cfgtoollogs/PRODDB/upgrade20200620213345/upg_summary.log

Grand Total Upgrade Time: [0d:0h:46m:40s]
[oracle@rac1 bin]\$

```
[oracle@rac1 upgrade20200620213345]$ pwd  
/u01/app/oracle/product/19.0.0/dbhome_1/cfgtoollogs/PRODDB/upgrade20200620213345  
[oracle@rac1 upgrade20200620213345]$ cat upg_summary.log
```

Oracle Database Release 19 Post-Upgrade Status Tool
Database Name: PRODDB

Component Name	Current Status	Full Version	Elapsed Time HH:MM:SS
Oracle Server	UPGRADED	19.4.0.0.0	00:18:01
JServer JAVA Virtual Machine	UPGRADED	19.4.0.0.0	00:02:29
Oracle XDK	UPGRADED	19.4.0.0.0	00:01:12
Oracle Database Java Packages	UPGRADED	19.4.0.0.0	00:00:13
OLAP Analytic Workspace	UPGRADED	19.4.0.0.0	00:00:19
Oracle Label Security	UPGRADED	19.4.0.0.0	00:00:11
Oracle Database Vault	UPGRADED	19.4.0.0.0	00:00:32
Oracle Text	UPGRADED	19.4.0.0.0	00:00:43
Oracle Workspace Manager	UPGRADED	19.4.0.0.0	00:00:37
Oracle Real Application Clusters	UPGRADED	19.4.0.0.0	00:00:01
Oracle XML Database	UPGRADED	19.4.0.0.0	00:01:49
Oracle Multimedia	UPGRADED	19.4.0.0.0	00:00:55
Spatial	LOADING	19.4.0.0.0	00:07:19
Oracle OLAP API	UPGRADED	19.4.0.0.0	00:00:14
Datapatch		00:08:34	
Final Actions		00:08:43	
Post Upgrade		00:00:12	

Total Upgrade Time: 00:44:08

Database time zone version is 26. It is older than current release time
zone version 32. Time zone upgrade is needed using the DBMS_DST package.

Grand Total Upgrade Time: [0d:0h:46m:40s]
[oracle@rac1 upgrade20200620213345]\$

```
#####
18. Starup DB from 19c home#####
#####  
[oracle@rac1 ~]$ export ORACLE_HOME=/u01/app/oracle/product/19.0.0/dbhome_1  
[oracle@rac1 ~]$ export ORACLE_SID=PRODDB  
[oracle@rac1 ~]$ PATH=/u01/app/oracle/product/19.0.0/dbhome_1/bin:$PATH; export PATH  
[oracle@rac1 ~]$ which sqlplus  
/u01/app/oracle/product/19.0.0/dbhome_1/bin/sqlplus  
[oracle@rac1 ~]$ sqlplus / as sysdba
```

SQL*Plus: Release 19.0.0.0 - Production
Version 19.4.0.0.0

Copyright (c) 1982, 2019, Oracle. All rights reserved.

Connected to an idle instance.

```
SQL> startup;  
ORACLE instance started.
```

```
Total System Global Area 1560277408 bytes  
Fixed Size          8896928 bytes  
Variable Size       1174405120 bytes  
Database Buffers    369098752 bytes  
Redo Buffers        7876608 bytes
```

```
Database mounted.  
Database opened.  
SQL> select name,open_mode,cdb,version,status from v$database,v$instance;
```

NAME	OPEN_MODE	CDB VERSION	STATUS
PRODDB	READ WRITE	NO 19.0.0.0.0	OPEN <-----

```
SQL>  
SQL> col COMP_ID for a10  
col COMP_NAME for a40  
col VERSION for a15  
set lines 180  
set pages 999  
select COMP_ID,COMP_NAME,VERSION,STATUS from dba_registry;
```

COMP_ID	COMP_NAME	VERSION	STATUS
CATALOG	Oracle Database Catalog Views	19.0.0.0.0	UPGRADED
CATPROC	Oracle Database Packages and Types	19.0.0.0.0	UPGRADED
JAVAVM	JServer JAVA Virtual Machine	19.0.0.0.0	UPGRADED
XML	Oracle XDK	19.0.0.0.0	UPGRADED
CATJAVA	Oracle Database Java Packages	19.0.0.0.0	UPGRADED
APS	OLAP Analytic Workspace	19.0.0.0.0	UPGRADED
RAC	Oracle Real Application Clusters	19.0.0.0.0	UPGRADED
XDB	Oracle XML Database	19.0.0.0.0	UPGRADED
OWM	Oracle Workspace Manager	19.0.0.0.0	UPGRADED
CONTEXT	Oracle Text	19.0.0.0.0	UPGRADED
ORDIM	Oracle Multimedia	19.0.0.0.0	UPGRADED
SDO	Spatial	19.0.0.0.0	LOADING
XOQ	Oracle OLAP API	19.0.0.0.0	UPGRADED
OLS	Oracle Label Security	19.0.0.0.0	UPGRADED
DV	Oracle Database Vault	19.0.0.0.0	UPGRADED

15 rows selected.

```
SQL>
```

POST-UPGRADE TASKS WHEN DBUA USING

```
#####
19. Run utlrp.sql
#####
Run catcon.pl to start utlrp.sql, and to recompile any remaining invalid objects.
```

```
$ORACLE_HOME/perl/bin/perl catcon.pl -n 1 -e -b utlrp -d "" utlrp.sql
```

-- OR --

```
cd /u01/app/oracle/product/19.0.0/dbhome_1/rdbms/admin  
nohup sqlplus "/ as sysdba" @utlrp.sql > /home/oracle/utlrp.out 2>&1 &
```

```
SQL> select count(*) from dba_objects where status='INVALID';
```

COUNT(*)
1413

```
SQL> select count(*) from dba_objects where status='INVALID' and owner in ('SYS','SYSTEM');
```

COUNT(*)
655

```
SQL>
```

```
SQL> @/u01/app/oracle/product/19.0.0/dbhome_1/rdbms/admin/utlrp.sql
```

Session altered.

TIMESTAMP

COMP_TIMESTAMP UTLRP_BGN 2020-06-20 00:31:04

DOC> The following PL/SQL block invokes UTL_RECOMP to recompile invalid objects in the database. Recompilation time is proportional to the number of invalid objects in the database, so this command may take a long time to execute on a database with a large number of invalid objects.
DOC>
DOC> Use the following queries to track recompilation progress:
DOC>
DOC> 1. Query returning the number of invalid objects remaining. This number should decrease with time.
DOC> SELECT COUNT(*) FROM obj\$ WHERE status IN (4, 5, 6);
DOC>
DOC> 2. Query returning the number of objects compiled so far. This number should increase with time.
DOC> SELECT COUNT(*) FROM UTL_RECOMP_COMPILED;
DOC>
DOC> This script automatically chooses serial or parallel recompilation based on the number of CPUs available (parameter cpu_count) multiplied by the number of threads per CPU (parameter parallel_threads_per_cpu).
DOC> On RAC, this number is added across all RAC nodes.
DOC>
DOC> UTL_RECOMP uses DBMS_SCHEDULER to create jobs for parallel recompilation. Jobs are created without instance affinity so that they can migrate across RAC nodes. Use the following queries to verify whether UTL_RECOMP jobs are being created and run correctly:
DOC>
DOC> 1. Query showing jobs created by UTL_RECOMP
DOC> SELECT job_name FROM dba_scheduler_jobs
DOC> WHERE job_name like 'UTL_RECOMP_SLAVE_%';
DOC>
DOC> 2. Query showing UTL_RECOMP jobs that are running
DOC> SELECT job_name FROM dba_scheduler_running_jobs
DOC> WHERE job_name like 'UTL_RECOMP_SLAVE_%';
DOC>#

PL/SQL procedure successfully completed.

TIMESTAMP

COMP_TIMESTAMP UTLRP_END 2020-06-20 00:36:03

DOC> The following query reports the number of invalid objects.
DOC>
DOC> If the number is higher than expected, please examine the error messages reported with each object (using SHOW ERRORS) to see if they point to system misconfiguration or resource constraints that must be fixed before attempting to recompile these objects.
DOC>#

OBJECTS WITH ERRORS

0

DOC> The following query reports the number of exceptions caught during recompilation. If this number is non-zero, please query the error messages in the table UTL_RECOMP_ERRORS to see if any of these errors are due to misconfiguration or resource constraints that must be fixed before objects can compile successfully.
DOC> Note: Typical compilation errors (due to coding errors) are not logged into this table: they go into DBA_ERRORS instead.
DOC>#

ERRORS DURING RECOMPILEATION

0

Function created.

PL/SQL procedure successfully completed.

Function dropped.

PL/SQL procedure successfully completed.

SQL>

```
SQL> select count(*) from dba_objects where status='INVALID';
```

COUNT(*)

0

SQL>

```
#####
#20. Run postupgrade_fixups.sql
#####
SQL> @/home/oracle/PRODDB/preupgrade/postupgrade_fixups.sql
```

Session altered.

PL/SQL procedure successfully completed.

PL/SQL procedure successfully completed.

PL/SQL procedure successfully completed.

Package created.

No errors.

Package body created.

PL/SQL procedure successfully completed.

No errors.

Package created.

No errors.

Package body created.

No errors.

Executing Oracle POST-Upgrade Fixup Script

Auto-Generated by: Oracle Preupgrade Script

Version: 19.0.0.0 Build: 1

Generated on: 2020-06-20 20:45:05

For Source Database: PRODDB

Source Database Version: 12.2.0.1.0

For Upgrade to Version: 19.0.0.0

Preup	Preupgrade	Action	Issue Is	Number	Preupgrade	Check	Name	Remedied	Further DBA Action
-----	-----	-----	-----	-----	-----	-----	-----	-----	-----
6.	old_time_zones_exist	NO	Manual fixup recommended.	6.	old_time_zones_exist	NO	Manual fixup recommended.		
7.	post_dictionary	YES	None.	7.	post_dictionary	YES	None.		
8.	post_fixed_objects	NO	Informational only.	8.	post_fixed_objects	NO	Informational only.		
			Further action is optional.						

The fixup scripts have been run and resolved what they can. However, there are still issues originally identified by the preupgrade that have not been remedied and are still present in the database.

Depending on the severity of the specific issue, and the nature of the issue itself, that could mean that your database upgrade is not fully complete. To resolve the outstanding issues, start by reviewing the postupgrade_fixups.sql and searching it for the name of the failed CHECK NAME or Preupgrade Action Number listed above. There you will find the original corresponding diagnostic message from the preupgrade which explains in more detail what still needs to be done.

PL/SQL procedure successfully completed.

Session altered.

SQL>

```
#####
21. Upgrade Timezone
#####
For releases (18c, 19c), the timezone upgrade scripts are included in the target ORACLE_HOME under rdbms/admin directory
```

The following scripts get delivered with Oracle Database 18c onward

\$ORACLE_HOME/rdbms/admin/utlz_countstats.sql

Script to gives how much TIMESTAMP WITH TIME ZONE data there is in a database using stats info. No restart required.

\$ORACLE_HOME/rdbms/admin/utlz_countstar.sql

Script to approximate how much TIMESTAMP WITH TIME ZONE data there is in a database using a COUNT(*) for each table that has a TSTZ column. This script is useful when using DBMS_DST package or the scripts of utlz_upg_check.sql and utlz_upg_apply.sql scripts.

\$ORACLE_HOME/rdbms/admin/utlz_upg_check.sql

Time zone upgrade check script

\$ORACLE_HOME/rdbms/admin/utlz_upg_apply.sql

Time zone apply script. Warning: This script will restart the database and adjust time zone data.

```
[oracle@rac1 ~]$ cd /u01/app/oracle/product/19.0.0/dbhome_1/rdbms/admin/
[oracle@rac1 admin]$ ls -ltr utlz_countstats.sql utlz_countstar.sql utlz_upg_check.sql utlz_upg_apply.sql
-rw-r--r--. 1 oracle oinstall 8317 Feb 25 2017 utlz_countstats.sql
-rw-r--r--. 1 oracle oinstall 7423 Feb 25 2017 utlz_countstar.sql
-rw-r--r--. 1 oracle oinstall 33684 Sep  9 2017 utlz_upg_check.sql
-rw-r--r--. 1 oracle oinstall 21526 Sep  9 2017 utlz_upg_apply.sql
[oracle@rac1 admin]$
```

SQL> SELECT version FROM v\$timezone_file;

VERSION

26 <-----

SQL>

SQL> @/u01/app/oracle/product/19.0.0/dbhome_1/rdbms/admin/utlz_upg_check.sql

Session altered.

INFO: Starting with RDBMS DST update preparation.

INFO: NO actual RDBMS DST update will be done by this script.

INFO: If an ERROR occurs the script will EXIT sqlplus.

INFO: Doing checks for known issues ...

INFO: Database version is 19.0.0.0 .

INFO: Database RDBMS DST version is DSTv26 .

INFO: No known issues detected.

INFO: Now detecting new RDBMS DST version.

A prepare window has been successfully started.

INFO: Newest RDBMS DST version detected is DSTv32 .

INFO: Next step is checking all TSTZ data.

INFO: It might take a while before any further output is seen ...

A prepare window has been successfully ended.

INFO: A newer RDBMS DST version than the one currently used is found.

INFO: Note that NO DST update was yet done.

INFO: Now run utlz_upg_apply.sql to do the actual RDBMS DST update.

INFO: Note that the utlz_upg_apply.sql script will

INFO: restart the database 2 times WITHOUT any confirmation or prompt.

Session altered.

SQL>

SQL> @/u01/app/oracle/product/19.0.0/dbhome_1/rdbms/admin/utlz_upg_apply.sql

Session altered.

INFO: If an ERROR occurs, the script will EXIT SQL*Plus.
INFO: The database RDBMS DST version will be updated to DSTv32 .
WARNING: This script will restart the database 2 times
WARNING: WITHOUT asking ANY confirmation.
WARNING: Hit control-c NOW if this is not intended.
INFO: Restarting the database in UPGRADE mode to start the DST upgrade.
Database closed.
Database dismounted.
ORACLE instance shut down.
ORACLE instance started.

Total System Global Area 1560277408 bytes

Fixed Size 8896928 bytes
Variable Size 1191182336 bytes
Database Buffers 352321536 bytes
Redo Buffers 7876608 bytes

Database mounted.

Database opened.

INFO: Starting the RDBMS DST upgrade.

INFO: Upgrading all SYS owned TSTZ data.

INFO: It might take time before any further output is seen ...

An upgrade window has been successfully started.

INFO: Restarting the database in NORMAL mode to upgrade non-SYS TSTZ data.

Database closed.

Database dismounted.

ORACLE instance shut down.

ORACLE instance started.

Total System Global Area 1560277408 bytes

Fixed Size 8896928 bytes
Variable Size 1191182336 bytes
Database Buffers 352321536 bytes
Redo Buffers 7876608 bytes

Database mounted.

Database opened.

INFO: Upgrading all non-SYS TSTZ data.

INFO: It might take time before any further output is seen ...

INFO: Do NOT start any application yet that uses TSTZ data!

INFO: Next is a list of all upgraded tables:

Table list: "GSMADMIN_INTERNAL"."AQ\$_CHANGE_LOG_QUEUE_TABLE_S"

Number of failures: 0

Table list: "GSMADMIN_INTERNAL"."AQ\$_CHANGE_LOG_QUEUE_TABLE_L"

Number of failures: 0

Table list: "MDSYS"."SDO_DIAG_MESSAGES_TABLE"

Number of failures: 0

Table list: "DVSYS"."SIMULATION_LOG\$"

Number of failures: 0

Table list: "DVSYS"."AUDIT_TRAIL\$"

Number of failures: 0

INFO: Total failures during update of TSTZ data: 0 .

An upgrade window has been successfully ended.

INFO: Your new Server RDBMS DST version is DSTv32 .

INFO: The RDBMS DST update is successfully finished.

INFO: Make sure to exit this SQL*Plus session.

INFO: Do not use it for timezone related selects.

Session altered.

```
SQL>
SQL> SELECT version FROM v$timezone_file;
```

VERSION

32 <----

```
SQL>
```

```
#####
22. Run utlsts.sql
#####
*** Note: utluNNNs.sql is replaced by utlsts.sql in 19c version
```

*** Note: In 19c Earlier version utluNNNs.sql is replaced by utlsts.sql
 *** Run utlsts.sql as many times as you want, at any time after the upgrade is completed.
 *** utlsts.sql reads the view called dba_registry_log and displays the upgrade results for the database components.

SQL> @/u01/app/oracle/product/19.0.0/dbhome_1/rdbms/admin/utlsts.sql TEXT

Oracle Database Release 19 Post-Upgrade Status Tool 01-29-2020 01:41:3
 Database Name: PROddb

Component Name	Current Status	Full Version	Elapsed Time HH:MM:SS
Oracle Server	VALID	19.4.0.0.0	00:18:01
JServer JAVA Virtual Machine	VALID	19.4.0.0.0	00:02:29
Oracle XDK	VALID	19.4.0.0.0	00:01:12
Oracle Database Java Packages	VALID	19.4.0.0.0	00:00:13
OLAP Analytic Workspace	VALID	19.4.0.0.0	00:00:19
Oracle Label Security	VALID	19.4.0.0.0	00:00:11
Oracle Database Vault	VALID	19.4.0.0.0	00:00:32
Oracle Text	VALID	19.4.0.0.0	00:00:43
Oracle Workspace Manager	VALID	19.4.0.0.0	00:00:37
Oracle Real Application Clusters	OPTION OFF	19.4.0.0.0	00:00:01
Oracle XML Database	VALID	19.4.0.0.0	00:01:49
Oracle Multimedia	VALID	19.4.0.0.0	00:00:55
Spatial	VALID	19.4.0.0.0	00:07:19
Oracle OLAP API	VALID	19.4.0.0.0	00:00:14
Datapatch		00:08:34	
Final Actions		00:08:43	
Post Upgrade		00:00:12	
Post Compile		00:04:58	

Total Upgrade Time: 00:49:07

Database time zone version is 32. It meets current release needs.

SQL>

```
#####
23. Run catuppst.sql
#####
/*
In 12c:
https://docs.oracle.com/en/database/oracle/oracle-database/12.2/upgrd/oracle-database-upgrade-utilities.html#GUID-408F22C3-2AD6-4DA4-8015-F5C6149508F0
```

You must run this script, either through DBUA or manually, if you perform a manual upgrade.

DBUA automatically runs catuppst.sql. You only must run this script separately for manual upgrades.

Do not run this in UPGRADE mode. Run catuppst.sql, located in the ORACLE_HOME/rdbms/admin directory, to perform remaining upgrade actions that do not require the database to be in UPGRADE mode. If an Oracle bundle patch or patch set update (PSU or BP) is installed in the Oracle home, then this script automatically applies that patch set update to the database.

Caution: If you perform a manual upgrade, and you do not run catuppst.sql, then your database suffers performance degradation over time.

*/

*** Actually it will run as part of upgrade. We have reviewed catupgrd0.log and below is the output... found catuppst.sql ran and don't see errors.

```
/*
Rem catuppst.sql
Rem
Rem Copyright (c) 2006, 2018, Oracle and/or its affiliates.
Rem All rights reserved.
Rem
Rem NAME
Rem   catuppst.sql - CATalog UPgrade PoST-upgrade actions
Rem
Rem DESCRIPTION
Rem   This post-upgrade script performs remaining upgrade actions that
Rem   do not require that the database be open in UPGRADE mode.
Rem   Automatically apply the latest PSU.
Rem
```

```

Rem  NOTES
Rem   You must be connected AS SYSDBA to run this script.
..
..
22:19:59 SQL> -- DBUA_TIMESTAMP: catuppst.sql finished
22:19:59 SQL> SELECT dbms_registry_sys.time_stamp('CATUPPST') as timestamp from dual;

TIMESTAMP
-----
COMP_TIMESTAMP CATUPPST      2020-06-20 22:19:59
DBUA_TIMESTAMP CATUPPST     FINISHED 2020-06-20 22:19:59
DBUA_TIMESTAMP CATUPPST     NONE 2020-06-20 22:19:59
*/

```

*** If we had no errors, the "catuppst.sql" script would have been run as part of the upgrade. we need to run it manually if did have errors.

*** However can run one more time make sure no errors during execution.

```

SQL> @/u01/app/oracle/product/19.0.0/dbhome_1/rdbms/admin/catuppst.sql

TIMESTAMP
-----
COMP_TIMESTAMP DBRESTART      2020-06-20 02:24:23
DBUA_TIMESTAMP DBRESTART     FINISHED 2020-06-20 02:24:23
DBUA_TIMESTAMP DBRESTART     NONE 2020-06-20 02:24:23

TIMESTAMP
-----
DBUA_TIMESTAMP CATUPPST      STARTED 2020-06-20 02:24:23

TIMESTAMP
-----
COMP_TIMESTAMP POSTUP_BGN      2020-06-20 02:24:23
DBUA_TIMESTAMP POSTUP_BGN     FINISHED 2020-06-20 02:24:23
DBUA_TIMESTAMP POSTUP_BGN     NONE 2020-06-20 02:24:23

TIMESTAMP
-----
COMP_TIMESTAMP CATREQ_BGN      2020-06-20 02:24:23
DBUA_TIMESTAMP CATREQ_BGN     FINISHED 2020-06-20 02:24:23
DBUA_TIMESTAMP CATREQ_BGN     NONE 2020-06-20 02:24:23

catrequtlmg: b_StatEvt = TRUE
catrequtlmg: b_SelProps = FALSE
catrequtlmg: b_UpgradeMode = FALSE
catrequtlmg: b_InUtlMig = FALSE

TIMESTAMP
-----
COMP_TIMESTAMP CATREQ_END      2020-06-20 02:24:23
DBUA_TIMESTAMP CATREQ_END     FINISHED 2020-06-20 02:24:23
DBUA_TIMESTAMP CATREQ_END     NONE 2020-06-20 02:24:23

catuppst: Dropping library DBMS_DDL_INTERNAL_LIB
catuppst: Dropping view _CURRENT_EDITION_OBJ_MIG
catuppst: Dropping view _ACTUAL_EDITION_OBJ_MIG
catuppst: Dropping view DBA_PART_KEY_COLUMNS_V$_.MIG
catuppst: Dropping view DBA_SUBPART_KEY_COLUMNS_V$_.MIG
catuppst: Dropping table OBJ$MIG
catuppst: Dropping table USER$MIG
catuppst: Dropping table COL$MIG
catuppst: Dropping table CLU$MIG
catuppst: Dropping table CON$MIG
catuppst: Dropping table BOOTSTRAP$MIG
catuppst: Dropping table TAB$MIG
catuppst: Dropping table TS$MIG
catuppst: Dropping table IND$MIG
catuppst: Dropping table ICOL$MIG
catuppst: Dropping table LOB$MIG
catuppst: Dropping table COLTYPE$MIG
catuppst: Dropping table SUBCOLTYPE$MIG

```

```
catuppst: Dropping table NTAB$MIG
catuppst: Dropping table REFCON$MIG
catuppst: Dropping table OPQTYPE$MIG
catuppst: Dropping table ICOLDEP$MIG
catuppst: Dropping table VIEWTRCOL$MIG
catuppst: Dropping table ATTRCOL$MIG
catuppst: Dropping table TYPE_MISC$MIG
catuppst: Dropping table LIBRARY$MIG
catuppst: Dropping table ASSEMBLY$MIG
catuppst: Dropping table TSQ$MIG
catuppst: Dropping table FET$MIG
```

TIMESTAMP

```
-----  
COMP_TIMESTAMP POSTUP_END      2020-06-20 02:24:24  
DBUA_TIMESTAMP POSTUP_END     FINISHED 2020-06-20 02:24:24  
DBUA_TIMESTAMP POSTUP_END     NONE 2020-06-20 02:24:24
```

TIMESTAMP

```
-----  
COMP_TIMESTAMP CATUPPST      2020-06-20 02:24:24  
DBUA_TIMESTAMP CATUPPST     FINISHED 2020-06-20 02:24:24  
DBUA_TIMESTAMP CATUPPST     NONE 2020-06-20 02:24:24
```

SQL>

```
#####
24. Re-Run postupgrade_fixups.sql
#####
SQL> @/home/oracle/PRODDB/preupgrade/postupgrade_fixups.sql
No errors.
No errors.
```

No errors.

No errors.

Executing Oracle POST-Upgrade Fixup Script

Auto-Generated by: Oracle Preupgrade Script
Version: 19.0.0.0 Build: 1
Generated on: 2020-06-20 20:45:05

For Source Database: PRODDB
Source Database Version: 12.2.0.1.0
For Upgrade to Version: 19.0.0.0.0

Preup	Preupgrade		
Action	Issue Is		
Number	Preupgrade Check Name	Remedied	Further DBA Action
6.	old_time_zones_exist	YES	None.
7.	post_dictionary	YES	None.
8.	post_fixed_objects	NO	Informational only. Further action is optional.

The fixup scripts have been run and resolved what they can. However, there are still issues originally identified by the preupgrade that have not been remedied and are still present in the database. Depending on the severity of the specific issue, and the nature of the issue itself, that could mean that your database upgrade is not fully complete. To resolve the outstanding issues, start by reviewing the postupgrade_fixups.sql and searching it for the name of the failed CHECK NAME or Preupgrade Action Number listed above. There you will find the original corresponding diagnostic message from the preupgrade which explains in more detail what still needs to be done.

SQL>

```
#####
25. Reverify INVALID OBJECTS
#####
```

```

SQL> select count(*) from dba_objects where status='INVALID';

COUNT(*)
-----
0 <-----

SQL>

#####
26. Drop Restore point
#####

SQL> col name for a20
col GUARANTEE_FLASHBACK_DATABASE for a10
col TIME for a60
set lines 190
select NAME,GUARANTEE_FLASHBACK_DATABASE,TIME from V$restore_point;

NAME      GUARANTEE_TIME
-----
PRE_UPGRADE    YES    28-JAN-20 09.05.50.000000000 PM

SQL>
SQL> !ls -ltr /u01/app/oracle/fast_recovery_area/PRODDB/flashback
total 1433680
-rw-r----. 1 oracle oinstall 209723392 Jan 28 21:39 o1_mf_h30dfg5q_.flb
-rw-r----. 1 oracle oinstall 209723392 Jan 28 21:52 o1_mf_h30dfkos_.flb
-rw-r----. 1 oracle oinstall 209723392 Jan 28 21:56 o1_mf_h30gdpcm_.flb
-rw-r----. 1 oracle oinstall 209723392 Jan 28 22:12 o1_mf_h30h4zjr_.flb
-rw-r----. 1 oracle oinstall 209723392 Jan 28 22:15 o1_mf_h30hd tcb_.flb
-rw-r----. 1 oracle oinstall 209723392 Jan 28 22:15 o1_mf_h30jhoxw_.flb
-rw-r----. 1 oracle oinstall 209723392 Jan 29 02:35 o1_mf_h30jc0ct_.flb

SQL>
SQL> drop restore point PRE_UPGRADE;

Restore point dropped.

SQL> select NAME,GUARANTEE_FLASHBACK_DATABASE,TIME from V$restore_point;

no rows selected

SQL>
SQL> !ls -ltr /u01/app/oracle/fast_recovery_area/PRODDB/flashback
total 0 <---

SQL>

#####
27. Set COMPATIBLE parameter value to 19.0.0
#####
Warning: If the value of COMPATIBLE parameter is changed to 19.0.0 then if for some reasons database needs to be downgraded to 12.2.0.1 the DBA would not have any option other than export/import to downgrade the database. But if this parameter is left unchanged for sometime to see how the database performs after upgrade then it is very easy and fast to downgrade the database if for some reason it is required to be downgraded.

If you change COMPATIBLE you can directly drop your restore points as they are useless. You can't use Flashback Database to restore point back across a compatibility change of your database.

SQL> show parameter COMPATIBLE

NAME      TYPE      VALUE
-----
compatible      string    12.2.0
noncdb_compatible      boolean   FALSE
SQL>
SQL> ALTER SYSTEM SET COMPATIBLE = '19.0.0' SCOPE=SPFILE;

System altered.

SQL> shut immediate;
Database closed.
Database dismounted.
ORACLE instance shut down.
SQL>
SQL> startup;

```

ORACLE instance started.

Total System Global Area 1560277408 bytes
Fixed Size 8896928 bytes
Variable Size 1191182336 bytes
Database Buffers 352321536 bytes
Redo Buffers 7876608 bytes

Database mounted.
Database opened.

SQL>

SQL> show parameter COMPATIBLE

NAME	TYPE	VALUE
compatible	string	19.0.0 <----
noncdb_compatible	boolean	FALSE

28. Verify DBA_REGISTRY

SQL> col COMP_ID for a10
col COMP_NAME for a40
col VERSION for a15
set lines 180
set pages 999
select COMP_ID,COMP_NAME,VERSION,STATUS from dba_registry;

COMP_ID	COMP_NAME	VERSION	STATUS
CATALOG	Oracle Database Catalog Views	19.0.0.0.0	VALID
CATPROC	Oracle Database Packages and Types	19.0.0.0.0	VALID
JAVAVM	JServer JAVA Virtual Machine	19.0.0.0.0	VALID
XML	Oracle XDK	19.0.0.0.0	VALID
CATJAVA	Oracle Database Java Packages	19.0.0.0.0	VALID
APS	OLAP Analytic Workspace	19.0.0.0.0	VALID
RAC	Oracle Real Application Clusters	19.0.0.0.0	OPTION OFF
XDB	Oracle XML Database	19.0.0.0.0	VALID
OWM	Oracle Workspace Manager	19.0.0.0.0	VALID
CONTEXT	Oracle Text	19.0.0.0.0	VALID
ORDIM	Oracle Multimedia	19.0.0.0.0	VALID
SDO	Spatial	19.0.0.0.0	VALID
XOQ	Oracle OLAP API	19.0.0.0.0	VALID
OLS	Oracle Label Security	19.0.0.0.0	VALID
DV	Oracle Database Vault	19.0.0.0.0	VALID

15 rows selected.

SQL>

29. Add TNS Entries in 19c TNS home

[oracle@rac1 admin]\$ cat listener.ora
listener.ora Network Configuration File: /u01/app/oracle/product/19.0.0/dbhome_1/network/admin/listener.ora
Generated by Oracle configuration tools.

```
LISTENER_PRODDB =  
(DESCRIPTION_LIST =  
(DESCRIPTION =  
(ADDRESS = (PROTOCOL = TCP)(HOST = RAC1.ORACLE.COM)(PORT = 1524))  
)  
)
```

```
SID_LIST_LISTENER_PRODDB =  
(SID_LIST =  
(SID_DESC =  
(GLOBAL_DBNAME = PRODDB.oracle.com)  
(ORACLE_HOME = /u01/app/oracle/product/19.0.0/dbhome_1)  
(SID_NAME = PRODDB)  
)  
)  
[oracle@rac1 admin]$
```

```
[oracle@rac1 admin]$ cat tnsnames.ora
# tnsnames.ora Network Configuration File: /u01/app/oracle/product/19.0.0/dbhome_1/network/admin/tnsnames.ora
# Generated by Oracle configuration tools.

LISTENER_PRODDB =
  (ADDRESS = (PROTOCOL = TCP)(HOST = RAC1.ORACLE.COM)(PORT = 1524))

PRODDB =
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL = TCP)(HOST = RAC1.ORACLE.COM)(PORT = 1524))
    (CONNECT_DATA =
      (SERVER = DEDICATED)
      (SERVICE_NAME = PRODDB.oracle.com)
    )
  )
```

```
[oracle@rac1 admin]$
```

```
[oracle@rac1 admin]$ cat sqlnet.ora
# sqlnet.ora Network Configuration File: /u01/app/oracle/product/19.0.0/dbhome_1/network/admin/sqlnet.ora
# Generated by Oracle configuration tools.
```

```
NAMES DIRECTORY PATH= (TNSNAMES, ONAMES, HOSTNAME)
```

```
[oracle@rac1 admin]$
```

```
[oracle@rac1 admin]$ lsnrctl start LISTENER_PRODDB
```

```
LSNRCTL for Linux: Version 19.0.0.0 - Production on 20-JUN-2020 03:16:28
```

```
Copyright (c) 1991, 2019, Oracle. All rights reserved.
```

```
Starting /u01/app/oracle/product/19.0.0/dbhome_1/bin/tnslsnr: please wait...
```

```
TNSLSNR for Linux: Version 19.0.0.0 - Production
```

```
System parameter file is /u01/app/oracle/product/19.0.0/dbhome_1/network/admin/listener.ora
```

```
Log messages written to /u01/app/oracle/diag/tnslsnr/rac1/listener_proddb/alert/log.xml
```

```
Listening on: (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=RAC1.ORACLE.COM)(PORT=1524)))
```

```
Connecting to (DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)(HOST=RAC1.ORACLE.COM)(PORT=1524)))
STATUS of the LISTENER
```

```
-----
Alias          LISTENER_PRODDB
Version        TNSLSNR for Linux: Version 19.0.0.0 - Production
Start Date     20-JUN-2020 03:16:28
Uptime         0 days 0 hr. 0 min. 10 sec
Trace Level   off
Security       ON: Local OS Authentication
SNMP           OFF
Listener Parameter File  /u01/app/oracle/product/19.0.0/dbhome_1/network/admin/listener.ora
Listener Log File   /u01/app/oracle/diag/tnslsnr/rac1/listener_proddb/alert/log.xml
Listening Endpoints Summary...
  (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=RAC1.ORACLE.COM)(PORT=1524)))
Services Summary...
Service "PRODDB.ORACLEAE.com" has 1 instance(s).
  Instance "PRODDB", status UNKNOWN, has 1 handler(s) for this service...
The command completed successfully
[oracle@rac1 admin]$
```

```
[oracle@rac1 admin]$ tnsping PRODDB
```

```
TNS Ping Utility for Linux: Version 19.0.0.0 - Production on 20-JUN-2020 03:16:50
```

```
Copyright (c) 1997, 2019, Oracle. All rights reserved.
```

```
Used parameter files:
/u01/app/oracle/product/19.0.0/dbhome_1/network/admin/sqlnet.ora
```

```
Used TNSNAMES adapter to resolve the alias
```

```
Attempting to contact (DESCRIPTION = (ADDRESS = (PROTOCOL = TCP)(HOST = RAC1.ORACLE.COM)(PORT = 1524))
(CONNECT_DATA = (SERVER = DEDICATED) (SERVICE_NAME = PRODDB.oracle.com)))
OK (0 msec)
```

```
[oracle@rac1 admin]$ #####
30. Password File – orapwPRODDB
#####
*** REMOTE_LOGIN_PASSWORDFILE=EXCLUSIVE
*** Password file orapwPRODDB copied automatically during upgrade process.
No action taken.

[oracle@rac1 dbs]$ pwd
/u01/app/oracle/product/19.0.0/dbhome_1/dbs
[oracle@rac1 dbs]$ ls -ltr orapwPRODDB
-rw-r----. 1 oracle oinstall 3584 Jun 20 22:26 orapwPRODDB <-----
[oracle@rac1 dbs]$
SQL> show parameter password

NAME          TYPE    VALUE
-----
remote_login_passwordfile    string   EXCLUSIVE
SQL>
SQL>

#####
31. Edit oratab
#####
[oracle@rac1 ~]$ cat /etc/oratab | grep -i PRODDB
#PRODDB:/u01/app/oracle/product/12.2.0/dbhome_1:N
PRODDB:/u01/app/oracle/product/19.0.0/dbhome_1:N
[oracle@rac1 ~]$ #####
32. Back Up the Database using rman level 0
#####

[oracle@rac1 AFTER_UPGRADE_19C]$
```

\$
Steps to be done post Database installation on all nodes and Grid with all setups \$
\$
Reference : <https://docs.oracle.com/en/database/oracle/oracle-database/12.2/racad/converting-single-instance-oracle-databases-to-oracle-rac-and-oracle-rac-one-node.html#GUID-B7E960A0-A4E8-47FE-AB6D-4172AD45944A>

You can use rconfig, a non-interactive command-line utility, to convert a single-instance database to a RAC database.

The utility reads the values provided under the ConvertToRAC.xml file.

The \$ORACLE_HOME/assistants/rconfig/sampleXMLs directory has two templates,
ConvertToRAC_AdminManaged.xml and
ConvertToRAC_PolicyManaged.xml

which you can use to convert a single-instance database to RAC admin or policy-managed database, respectively.

Prerequisites for using rconfig

Before you use rconfig, consider the following prerequisites:

- Use Oracle Database version 10g R2 or later.
- Configure Clusterware and have it running on all the nodes.
- Install Oracle RAC RDBMS on all the nodes.
- Make sure that shared storage, either Oracle Cluster File System or Automatic Storage Management (ASM), is available and accessible from all the nodes.
- Verify that the standalone database and RAC are the same database version.
- You need an active ASM instance across the nodes and the database running on one of the local nodes.

Steps for conversion to RAC

```
#####
STEP 1: SET THE PARAMETERS
#####
You need to set the following parameters in ConvertToRAC_AdminManaged.xml:
Convert verify      YES          ##First keep the value to ONLY for check##
source DB          ORACLE_HOME PATH
Target DB          ORACLE_HOME PATH
Source DB name, SYS Password  SID, PASSWORD
RAC Node List      Node1,Node2
```

RAC Node Prefix	SID
Shared storage type	ASM
Target DB area	DATA
Target DB Recovery	FRA

Please note that The convert verify option in ConvertToRAC.xml has three options:

Convert verify="YES": rconfig performs prerequisite checks followed by RAC conversion.

Convert verify="NO": rconfig performs RAC conversion without the prerequisite checks.

Convert verify="ONLY": rconfig performs only the prerequisite checks and takes no other action.

```
#####
STEP 2: PERFORM THE PREREQUISITE CHECKS
#####
```

```
#####
Run the following command to run the convert_verify="ONLY" option to fix any errors before executing final run
#####
```

```
$ cd $ORACLE_HOME/assistants/rconfig/sampleXMLs
$ ORACLE_HOME/bin/rconfig ConvertToRAC_racdb.xml

#####
STEP 3: EXECUTE RCONFIG FOR RAC CONVERSION
#####
When you run the following commands to execute rconfig to convert the database, make sure you update the parameter to convert_verify="YES". rconfig performs the conversion and the verification.
```

```
$ cd $ORACLE_HOME/assistants/rconfig/sampleXMLs
$ ORACLE_HOME/bin/rconfig ConvertToRAC_racdb.xml
```

Run the following command to monitor alerts in the rconfig log on the standalone database and the new RAC database:

```
$ tail -f /ora/app/oracle/cfgtoollogs/rconfig/rconfig*.log
```

Example:

```
[oracle@racnode1 sampleXMLs]$ rconfig ConvertToRAC_racdb.xml
Converting Database "PRODDDB" to Cluster Database.
Target Oracle Home: /u01/app/oracle/product/11.2.0/dbhome_1. Database Role: PRIMARY.
Setting Data Files and Control Files
Adding Database Instances
Adding Redo Logs
Enabling threads for all Database Instances
Setting TEMP tablespace
Adding UNDO tablespaces
Adding Trace files
Setting Fast Recovery Area
Updating Oratab
Creating Password file(s)
Configuring Listeners
Configuring related CRS resources
Starting Cluster Database
<?xml version="1.0" ?>
<RConfig version="1.1" >
<ConvertToRAC>
<Convert>
<Response>
<Result code="0" >
  Operation Succeeded
</Result>
</Response>
<ReturnValue type="object">
<Oracle_Home>
  /u01/app/oracle/product/11.2.0/dbhome_1
</Oracle_Home>
<Database type="ADMIN_MANAGED" >
  <InstanceList>
    <Instance SID="PRODDDB1" Node="racnode1" >
      </Instance>
    <Instance SID="PRODDDB2" Node="racnode2" >
      </Instance>
  </InstanceList>
</Database>
</ReturnValue>
</Convert>
</ConvertToRAC>
</RConfig>
#####
STEP 4: VERIFY THE CONVERSION
```

```
#####
Run the following commands to verify the log, check the RAC database status, and check the datafile status.
```

Note: If the standalone database was non-ASM, it is now ASM.

```
$ srvctl status database -d racdb
Check by login to sqlplus
SQLPLUS> select * from gv$instance;
SQLPLUS> select file_name from dba_data_files;
```

```
#####
STEP 5: MODIFY TNSENTRY
#####
```

Modify tnsentry on the local node with scan-name and copy it to all other nodes.