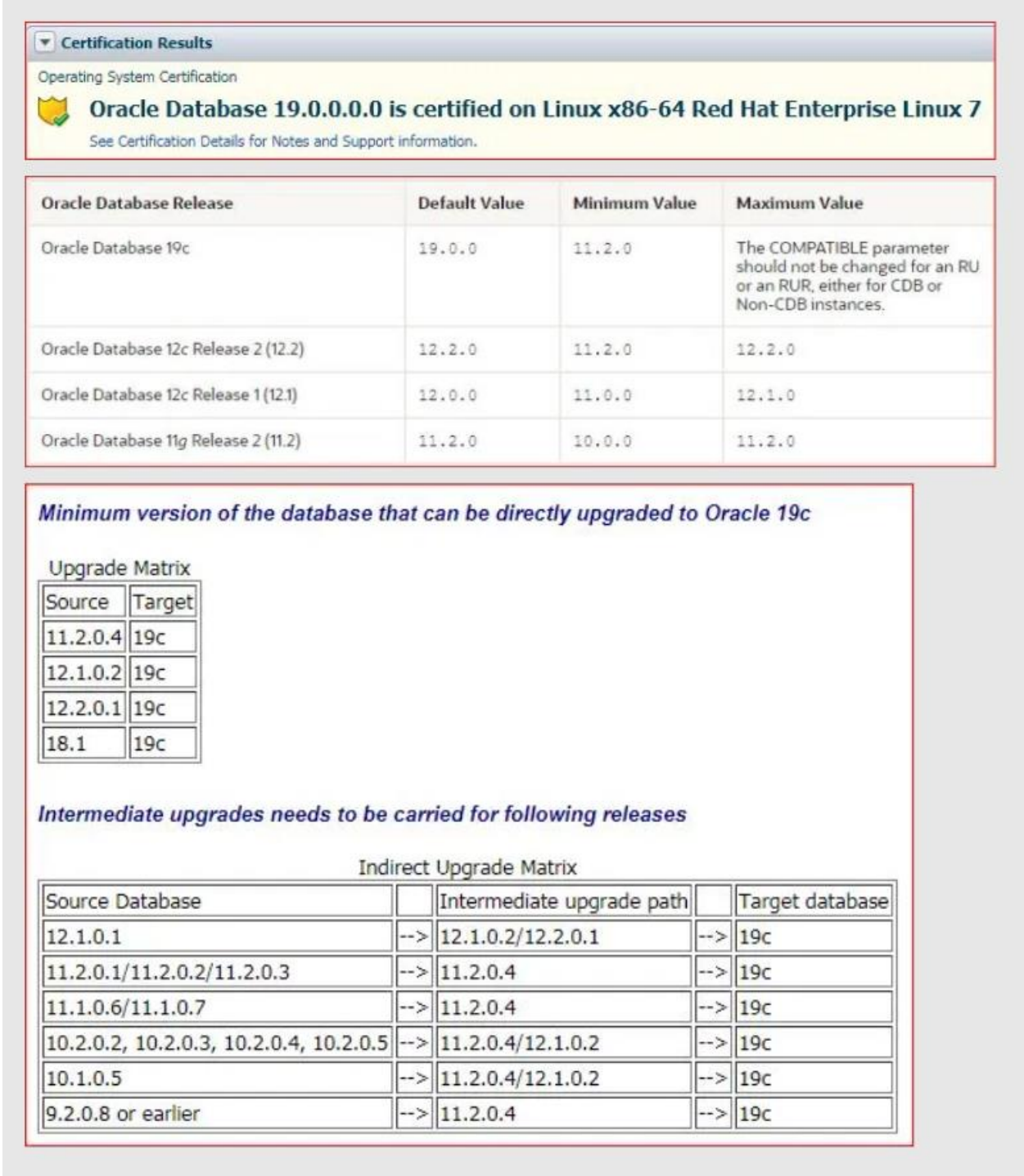
Database upgrade Steps from 12c to 19c

1.0 Document Overview

This document provides detailed steps to manual upgrade Oracle database from 12c to 19c.

1) **Check Compatibility Before Upgrading Oracle Database (Source : Oracle.com)**



Certification Results

Operating System Certification

Oracle Database 19.0.0.0.0 is certified on Linux x86-64 Red Hat Enterprise Linux 7

See Certification Details for Notes and Support information.

Oracle Database Release	Default Value	Minimum Value	Maximum Value
Oracle Database 19c	19.0.0	11.2.0	The COMPATIBLE parameter should not be changed for an RU or an RUR, either for CDB or Non-CDB instances.
Oracle Database 12c Release 2 (12.2)	12.2.0	11.2.0	12.2.0
Oracle Database 12c Release 1 (12.1)	12.0.0	11.0.0	12.1.0
Oracle Database 11g Release 2 (11.2)	11.2.0	10.0.0	11.2.0

Minimum version of the database that can be directly upgraded to Oracle 19c

Upgrade Matrix

Source	Target
11.2.0.4	19c
12.1.0.2	19c
12.2.0.1	19c
18.1	19c

Intermediate upgrades needs to be carried for following releases

Indirect Upgrade Matrix

Source Database		Intermediate upgrade path		Target database
12.1.0.1	-->	12.1.0.2/12.2.0.1	-->	19c
11.2.0.1/11.2.0.2/11.2.0.3	-->	11.2.0.4	-->	19c
11.1.0.6/11.1.0.7	-->	11.2.0.4	-->	19c
10.2.0.2, 10.2.0.3, 10.2.0.4, 10.2.0.5	-->	11.2.0.4/12.1.0.2	-->	19c
10.1.0.5	-->	11.2.0.4/12.1.0.2	-->	19c
9.2.0.8 or earlier	-->	11.2.0.4	-->	19c

2) Environment:

Source

Hostname:ORAHOST1
DB Name: PROD

DB Version: 12.2.0.1.0

Target

Hostname:ORAHOST1
DB Name: PROD
DB Version: 19.3.0.0.0

PRE-UPGRADE TASK

3) Install 19c software on Server

./runInstaller -silent -responsefile
/oracle/product/base19c/19.3.0.0/install/response/db_install.rsp

oracle.install.responseFileVersion=/oracle/install/rspfmt_dbinstall_response_schema_v19.0.0
oracle.install.option=INSTALL_DB_SWONLY
UNIX_GROUP_NAME=dba
INVENTORY_LOCATION=/oracle/oraInventory
ORACLE_HOME=/oracle/product/base19c/19.3.0.0/
ORACLE_BASE=/oracle/product/base19c
oracle.install.db.InstallEdition=EE
oracle.install.db.OSDBA_GROUP=dba
oracle.install.db.OSOPER_GROUP=dba
oracle.install.db.OSBACKUPDBA_GROUP=dba
oracle.install.db.OSDGDBA_GROUP=dba
oracle.install.db.OSKMDBA_GROUP=dba
oracle.install.db.OSRACDBA_GROUP=dba
oracle.install.db.rootconfig.executeRootScript=false

ORAHOST1- CORP-NONEC : PROD /oracle/product/base19c/19.3.0.0/> ./runInstaller -silent -responsefile /oracle/product/base19c/19.3.0.0/install/response/db_install.rsp
Launching Oracle Database Setup Wizard...

[WARNING] [INS-13014] Target environment does not meet some optional requirements.
  CAUSE: Some of the optional prerequisites are not met. See logs for details.
/oracle/oraInventory/logs/InstallActions2021-02-23_05-00-26PM/installActions2021-02-23_05-00-26PM.log
  ACTION: Identify the list of failed prerequisite checks from the log:
/oracle/oraInventory/logs/InstallActions2021-02-23_05-00-26PM/installActions2021-02-23_05-00-26PM.log. Then either from the log file or from installation manual find the appropriate configuration to meet the prerequisites and fix it manually.
The response file for this session can be found at:
 /oracle/product/base19c/19.3.0.0/install/response/db_2021-02-23_05-00-26PM.rsp

You can find the log of this install session at:
/oracle/oraInventory/logs/InstallActions2021-02-23_05-00-26PM/installActions2021-02-23_05-00-26PM.log

As a root user, execute the following script(s):
1. /oracle/product/base19c/19.3.0.0/root.sh

Execute /oracle/product/base19c/19.3.0.0/root.sh on the following nodes:
[HIC015648]
Successfully Setup Software with warning(s).

4) Bring down the listener and take full rman backup of database using below command:

```
RUN

{

ALLOCATE CHANNEL c1 DEVICE TYPE disk;

ALLOCATE CHANNEL c2 DEVICE TYPE disk;

ALLOCATE CHANNEL c3 DEVICE TYPE disk;

ALLOCATE CHANNEL c4 DEVICE TYPE disk;

backup AS COMPRESSED BACKUPSET full database tag ORCL_FULL format
'/oradata/BKP/HRPSPD1/%d_%T_%s_%p_FULL.bkp' ;

sql 'alter system switch logfile';

sql 'alter system switch logfile';

sql 'alter system archive log current';

backup tag ORCL_ARCHIVE format
'/oradata/BKP/HRPSPD1/%d_%T_%s_%p_ARCHIVE.bkp' archivelog all ;

backup tag ORCL_CONTROL current controlfile format
'/oradata/BKP/HRPSPD1/%d_%T_%s_%p_CONTROL.bkp';

release channel c1;

release channel c2;

release channel c3;

release channel c4;

}

nohup rman target =/ cmdfile=rman2.cmd &
```

5) Capture the details of below parameters/commands:

```
spool pre_upgrade_Details.log
set line 200 pages 200
col OWNER for a30
col OBJECT_NAME for a30
select name,open_mode,log_mode,database_role,force_logging from gv$database;
select count(*) from dba_objects where status='INVALID';
select owner,object_name,object_type from dba_objects where status='INVALID' order by 1;
select * from registry$history;
select COMP_ID,COMP_NAME,VERSION,STATUS from dba_registry ;
select patch_id,status,ACTION_TIME,description from dba_registry_sqlpatch;
spool off;


spool obj_count.log
set line 200 pages 200
select owner,object_type,status,count(status) from dba_objects  group by
owner,object_type,status order by owner;
spool off;


spool hidden_param_all.log
col "Parameter" for a70
col "Session Value" for a30
col "Instance Value" for a30
SELECT x.ksppinm "Parameter",
     Y.ksppstvl "Session Value",
     Z.ksppstvl "Instance Value"
FROM   x$ksppi X,
     x$ksppcv Y,
     x$ksppsv Z
WHERE  x.indx = Y.indx
```

```
AND    x.indx = z.indx

AND    x.ksppinm LIKE '/_%' escape '/'

order by x.ksppinm;

spool off;


spool all_param.log

col NAME for a70

col DISPLAY_VALUE for a30

col DEFAULT_VALUE for a30

col ISDEFAULT for a40

col DESCRIPTION for a40

col VALUE for a40

select NAME,VALUE,DISPLAY_VALUE,DEFAULT_VALUE,ISDEFAULT,DESCRIPTION from
v$parameter order by 1;

spool off;
```

6) Run Preupgrade.jar from 12c env and fix all the preupgrade tasks mentioned in recommended action in Preupgrade.log under preupgrade folder.

```
/oracle/product/base/12.2.0.1/jdk/bin/java -jar
/oracle/product/base19c/19.3.0.0/rdbms/admin/preupgrade.jar FILE DIR
/oradata/BKP/HRPSPD1/
```

7) Gather dictionary stats

```
SQL> EXECUTE DBMS_STATS.GATHER_DICTIONARY_STATS;

PL/SQL procedure successfully completed.
```

8) Purge recyclebin

```
SQL> PURGE DBA_RECYCLEBIN;
DBA Recyclebin purged.
```

9) Refresh MVs

```
SQL> select count(1) from sys.sumdelta$;
```

```
 COUNT(1)
----------
     274
SQL> declare
list_failures integer(3) :=0;
begin
DBMS_MVIEW.REFRESH_ALL_MVIEWS(list_failures,'C','', TRUE, FALSE);
end;
/ 2   3   4   5   6
PL/SQL procedure successfully completed.
SQL> SQL> SQL> SQL> SQL> SQL> SQL> SQL> select count(1) from sys.sumdelta$;
 COUNT(1)
----------
       0
```

10) Run preupgrade_fixups.sql from preupgrade directory.

```
SQL> @preupgrade_fixups.sql
Executing Oracle PRE-Upgrade Fixup Script

Auto-Generated by:       Oracle Preupgrade Script
                 Version: 19.0.0.0.0 Build: 1
Generated on:          2021-02-26 07:06:50

For Source Database:     PROD
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

------ ---------- ---------- ----------

1. invalid_objects_exist NO Manual fixup recommended.
2. mv_refresh YES None.
3. hidden_params NO Informational only.
   Further action is optional.
4. dictionary_stats YES None.
5. mv_refresh YES None.
6. min_archive_dest_size NO Informational only.
   Further action is optional.
7. 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.

Verify the output and run postupgrade_fixups.sql

11) Make sure DB is in archivelog mode

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

```
NAME     OPEN_MODE          LOG_MODE
--------- -------------------- ------------
PROD   READ WRITE         ARCHIVELOG
```

12) Check compatible parameter and dont change it for now

```
SQL> show parameter compatible
NAME                                 TYPE      VALUE
------------------------------------ ----------- ------------------------------
compatible                           string     12.2.0.1.0
noncdb_compatible                    boolean    FALSE
```

13) Set below DB_recovery parameters and create a restore point incase you need to downgrade if any failure during upgrade.

```
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> alter system set db_recovery_file_dest_size=50G;

System altered.

SQL> alter system set db_recovery_file_dest='/oraexport/FRA/PROD/';
System altered.
SQL> show parameter recovery
NAME                                 TYPE      VALUE
------------------------------------ ----------- ------------------------------
db_recovery_file_dest                string     /oraexport/FRA/PROD/
```

db_recovery_file_dest_size big integer 50G

recovery_parallelism integer 0

remote_recovery_file_dest string

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;SQL>
SQL> SQL> SQL>

no rows selected

SQL> create restore point pre_upgrade guarantee flashback database;

Restore point created.

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

NAME GUARANTEE_ TIME

---------- ---------- ----------

PRE_UPGRADE YES 26-FEB-21 07.39.34.000000000 AM

UPGRADE TASK

14) Create pfile and shutdown database.

SQL> show parameter pfile

NAME TYPE VALUE

---------- ---------- ----------

spfile string /oracle/product/base/12.2.0.1/
dbs/spfilePROD.ora

SQL> create pfile='/oradata/BKP/PROD/pfile_PROD.ora' from spfile;

File created.

SQL> alter system switch logfile ;

System altered.

SQL> /

System altered.

SQL> /

System altered.

SQL> shut immediate;

Database closed.

Database dismounted.

ORACLE instance shut down.

15) Copy pfile and password file to 19c dbs location.

16) Set environment of 19c and start DB in upgrade mode

ORAHOST1- CORP-NONEC : PROD /oracle/product/base19c/19.3.0.0/dbs/> export ORACLE_SID=PROD
ORAHOST1- CORP-NONEC : PROD /oracle/product/base19c/19.3.0.0/dbs/> export ORACLE_HOME=/oracle/product/base19c/19.3.0.0
ORAHOST1- CORP-NONEC : PROD /oracle/product/base19c/19.3.0.0/dbs/> /usr/local/bin:/usr/bin/X11:/oracle/local/bin:.:/usr/sbin:/sbin:/usr/ucb
<
ORAHOST1- CORP-NONEC : PROD /oracle/product/base19c/19.3.0.0/dbs/> which sqlplus
/oracle/product/base19c/19.3.0.0/bin/sqlplus
ORAHOST1- CORP-NONEC : PROD /oracle/product/base19c/19.3.0.0/dbs/> sdba

SQL*Plus: Release 19.0.0.0.0 - Production on Fri Feb 26 07:59:50 2021
Version 19.3.0.0.0

Connected to an idle instance.

SQL> startup upgrade;
ORACLE instance started.

Total System Global Area 9663673928 bytes
Fixed Size 12445256 bytes
Variable Size 6140461056 bytes
Database Buffers 3422552064 bytes
Redo Buffers 88215552 bytes
Database mounted.
Database opened.
SQL> select name,open_mode,cdb,version,status from v$database,v$instance;

NAME OPEN_MODE CDB VERSION STATUS

--------- ---------- --- ---------- ----------
PROD READ WRITE NO 19.0.0.0.0 OPEN MIGRATE

SQL>
SQL> show parameter pfile

NAME TYPE VALUE
------------------------------------ ----------- ------------------------------
spfile string /oracle/product/base19c/19.3.0
.0/dbs/spfilePROD.ora

17) Run DBupgrade

ORAHOST1- CORP-NONEC : PROD /oracle/product/base19c/19.3.0.0/rdbms/admin/>
pwd
/oracle/product/base19c/19.3.0.0/rdbms/admin
ORAHOST1- CORP-NONEC : PROD /oracle/product/base19c/19.3.0.0/rdbms/admin/>
nohup /oracle/product/base19c/19.3.0.0/perl/bin/perl catctl.pl -l
/oradata/BKP/PROD/UPGRADE -n 4 catupgrd.sql &
[1] 16346
ORAHOST1- CORP-NONEC : PROD /oracle/product/base19c/19.3.0.0/rdbms/admin/>
nohup: ignoring input and appending output to 'nohup.out'
ORAHOST1- CORP-NONEC : PROD /oracle/product/base19c/19.3.0.0/rdbms/admin/>

18) Verify if upgrade is successful and start the database using new 19c environments

ORAHOST1- CORP-NONEC : PROD /home/oracle/> sdba

SQL*Plus: Release 19.0.0.0.0 - Production on Fri Feb 26 08:46:12 2021

Version 19.3.0.0.0

Connected to an idle instance.

SQL> startup

ORACLE instance started.

Total System Global Area 9663673928 bytes

Fixed Size 12445256 bytes

Variable Size 5804916736 bytes

Database Buffers 3758096384 bytes

Redo Buffers 88215552 bytes

Database mounted.

Database opened.

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

NAME      OPEN_MODE          CDB VERSION         STATUS
--------- -------------------- --- ----------------- ------------
PROD    READ WRITE         NO  19.0.0.0.0       OPEN
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;SQL> SQL> SQL> SQL>
SQL>
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
RAC        Oracle Real Application Clusters        19.0.0.0.0      OPTION OFF
OWM        Oracle Workspace Manager               19.0.0.0.0      UPGRADED
XDB        Oracle XML Database                 19.0.0.0.0      UPGRADED


8 rows selected.
```

19) Run Utlrp.sql

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

  COUNT(*)
----------
      1011
```

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

  COUNT(*)

----------

       761

SQL> @/oracle/product/base19c/19.3.0.0/rdbms/admin/utlrp.sql

Session altered.

TIMESTAMP

--------------------------------------------------------------------------------------------------------------------
--------------------------------------------------------------

COMP_TIMESTAMP UTLRP_BGN          2021-02-26 08:48:51


DOC>   The following PL/SQL block invokes UTL_RECOMP to recompile invalid
DOC>   objects in the database. Recompilation time is proportional to the
DOC>   number of invalid objects in the database, so this command may take
DOC>   a long time to execute on a database with a large number of invalid
DOC>   objects.
DOC>
DOC>   Use the following queries to track recompilation progress:
DOC>
DOC>   1. Query returning the number of invalid objects remaining. This
DOC>      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
DOC>      should increase with time.
DOC>         SELECT COUNT(*) FROM UTL_RECOMP_COMPILED;
DOC>
DOC>   This script automatically chooses serial or parallel recompilation
DOC>   based on the number of CPUs available (parameter cpu_count) multiplied
```

```
DOC>   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
DOC>   recompilation. Jobs are created without instance affinity so that they
DOC>   can migrate across RAC nodes. Use the following queries to verify
DOC>   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            2021-02-26 08:50:15
DOC> The following query reports the number of invalid objects.
DOC>
DOC> If the number is higher than expected, please examine the error
DOC> messages reported with each object (using SHOW ERRORS) to see if they
DOC> point to system misconfiguration or resource constraints that must be
DOC> fixed before attempting to recompile these objects.
DOC>#
OBJECTS WITH ERRORS
-------------------
```

```
              10
DOC> The following query reports the number of exceptions caught during
DOC> recompilation. If this number is non-zero, please query the error
DOC> messages in the table UTL_RECOMP_ERRORS to see if any of these errors
DOC> are due to misconfiguration or resource constraints that must be
DOC> fixed before objects can compile successfully.
DOC> Note: Typical compilation errors (due to coding errors) are not
DOC>       logged into this table: they go into DBA_ERRORS instead.
DOC>#

ERRORS DURING RECOMPILATION
---------------------------
                    3
Function created.
PL/SQL procedure successfully completed.
Function dropped.
PL/SQL procedure successfully completed.
SQL>  select count(*) from dba_objects where status='INVALID';
  COUNT(*)
----------
      10
SQL> select count(*) from dba_objects where status='INVALID' and owner in
('SYS','SYSTEM');
  COUNT(*)
----------
       0
```

20) Run postupgrade_fixup.sql

```
SQL> !pwd
/oradata/BKP/PROD/PREUPGRADE1

SQL> @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: 19.0.0.0.0 Build: 1
Generated on: 2021-02-26 07:07:03

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

Preup Action Number	Preupgrade Check Name	Preupgrade Issue Is Remedied	Further DBA Action
8.	old_time_zones_exist	NO	Manual fixup recommended.
9.	dir_symlinks	YES	None.

10. post_dictionary YES None.
11. 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.

21) Upgrade the timezone

SQL> @$ORACLE_HOME/rdbms/admin/utltz_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 utltz_upg_apply.sql to do the actual RDBMS DST update.
INFO: Note that the utltz_upg_apply.sql script will
INFO: restart the database 2 times WITHOUT any confirmation or prompt.

Session altered.

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

  VERSION
----------
        26

1 row selected.

SQL> @/oracle/product/base19c/19.3.0.0/rdbms/admin/utltz_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 9663673928 bytes
Fixed Size              12445256 bytes
Variable Size         5804916736 bytes
Database Buffers      3758096384 bytes
Redo Buffers            88215552 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 9663673928 bytes
Fixed Size              12445256 bytes
Variable Size         5804916736 bytes
Database Buffers      3758096384 bytes
Redo Buffers            88215552 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
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> SELECT version FROM v$timezone_file;

  VERSION
----------
      32

1 row selected.
```

22) Run utlusts.sql

```
SQL>  @/oracle/product/base19c/19.3.0.0/rdbms/admin/utlusts.sql TEXT

Oracle Database Release 19 Post-Upgrade Status Tool    02-26-2021 09:00:5
Database Name: PROD

Component                       Current         Full     Elapsed Time
Name                       Status        Version  HH:MM:SS

Oracle Server                    VALID      19.3.0.0.0  00:13:12
JServer JAVA Virtual Machine     VALID      19.3.0.0.0  00:01:04
Oracle XDK                       VALID      19.3.0.0.0  00:00:56
Oracle Database Java Packages    VALID      19.3.0.0.0  00:00:08
Oracle Workspace Manager         VALID      19.3.0.0.0  00:00:44
Oracle Real Application Clusters OPTION OFF      19.3.0.0.0  00:00:00
Oracle XML Database              VALID      19.3.0.0.0  00:01:31
Datapatch                                       00:00:58
Final Actions                                   00:01:07
```

```
Post Upgrade                              00:00:46
Post Compile                              00:01:23

Total Upgrade Time: 00:22:57

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

23) Run catuppst.sql

```
SQL> @/oracle/product/base19c/19.3.0.0/rdbms/admin/catuppst.sql

TIMESTAMP
--------------------------------------------------------------------------------
COMP_TIMESTAMP DBRESTART          2021-02-26 09:02:37
DBUA_TIMESTAMP DBRESTART    FINISHED 2021-02-26 09:02:37
DBUA_TIMESTAMP DBRESTART       NONE 2021-02-26 09:02:37


TIMESTAMP
--------------------------------------------------------------------------------
DBUA_TIMESTAMP CATUPPST     STARTED 2021-02-26 09:02:37

TIMESTAMP
--------------------------------------------------------------------------------
COMP_TIMESTAMP POSTUP_BGN         2021-02-26 09:02:37
DBUA_TIMESTAMP POSTUP_BGN   FINISHED 2021-02-26 09:02:37
DBUA_TIMESTAMP POSTUP_BGN      NONE 2021-02-26 09:02:37


TIMESTAMP
--------------------------------------------------------------------------------
COMP_TIMESTAMP CATREQ_BGN         2021-02-26 09:02:37
DBUA_TIMESTAMP CATREQ_BGN   FINISHED 2021-02-26 09:02:37
DBUA_TIMESTAMP CATREQ_BGN      NONE 2021-02-26 09:02:37

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

TIMESTAMP
--------------------------------------------------------------------------------
COMP_TIMESTAMP CATREQ_END         2021-02-26 09:02:37
DBUA_TIMESTAMP CATREQ_END   FINISHED 2021-02-26 09:02:37
DBUA_TIMESTAMP CATREQ_END      NONE 2021-02-26 09:02:37
```

```
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          2021-02-26 09:02:37
DBUA_TIMESTAMP POSTUP_END    FINISHED 2021-02-26 09:02:37
DBUA_TIMESTAMP POSTUP_END       NONE 2021-02-26 09:02:37


TIMESTAMP
--------------------------------------------------------------------------------
COMP_TIMESTAMP CATUPPST          2021-02-26 09:02:37
DBUA_TIMESTAMP CATUPPST     FINISHED 2021-02-26 09:02:37
DBUA_TIMESTAMP CATUPPST        NONE 2021-02-26 09:02:37
```

24) Again run postupgrade_fixup.sql

ORAHOST1- CORP-NONEC : PROD /oradata/BKP/PROD/PREUPGRADE1/> sdba

SQL*Plus: Release 19.0.0.0.0 - Production on Fri Feb 26 09:03:36 2021

Version 19.3.0.0.0

Connected to:

Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production

Version 19.3.0.0.0

SQL> @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: 19.0.0.0.0 Build: 1

Generated on: 2021-02-26 07:07:03

For Source Database: PROD

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
------ ----------------------- ---------- --------------------------------
   8.  old_time_zones_exist      YES        None.
   9.  dir_symlinks              YES        None.
  10.  post_dictionary           YES        None.
  11.  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.
```

25) Reverify invalid object count

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

  COUNT(*)
----------
        10
```

26) Drop Restore point

SQL> drop restore point PRE_UPGRADE;

Restore point dropped.

27) Set COMPATIBALE 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.
```

29) Add TNS Entries in 19c TNS home

30) Edit Oratab and modify the Oracle home to 19c home

31) Take Final Backup of Database post upgrade.