­­

This document describes the procedure to upgrade a 12.1.0.2 database to 18c (12.2.0.2) or 19c (12.2.0.3) on a Linux Oracle Standard Build environment. Steps 1-5 are pre-upgrade steps which can be executed prior to the actual upgrade without any impact to database availability unless the application makes use of APEX. However, if you execute these steps prior to the actual upgrade, you must check and re-empty the recycle bin at upgrade time. The areas of the document in a grey box are the output results of each of the commands. The commands to execute are indicated in **BLUE.** Puppet will deliver the 18c or 19c dbms software to each VM prior to the upgrade.

**Set Environment variables and create work directory**

run => **export UPGVER=19.9.0**  
run => **export SRVNAME=xhepydbm21p**  
run => **mkdir -p /home/oracle/tls/upg19c**

Reference: **Oracle 19c - Complete Checklist for Manual Upgrades to Non-CDB Oracle Database 19c (Doc ID 2539778.1)**

1. **Suspend all applicable OEM and Cron jobs**

## For reference

HEPYPRD\_xhepydbm21p - Primary database

HEPYPRD\_xhepydbw21p - Physical standby database

**Saturday steps**

Once Level0 completes sometime mid day Saturday

Reschedule to Run Level 1 (schedule back to what it was after)

**Disable (uncheck) Sunday Window Job via OEM Scheduler**

**Sunday steps**

**Keep running Archive logs all way to the end.**

**Suspend all OEM on Primary and Standby jobs at 5:50 AM on Sunday. They all should complete by that time.**

**Comment out crontab**

ALTER USER S032249

ACCOUNT LOCK

/

1. **Gather Dictionary Stats, Purge Recyclebin, Confirm no MV’s being refreshed**

run => **. oraenv**

**HEPYPRD**

run => **sqlplus / as sysdba**

sql => **execute dbms\_stats.gather\_dictionary\_stats;**

sql => **purge dba\_recyclebin;**

**Confirm no Materialized Views being refreshed at this time**

**select o.name from sys.obj$ o, sys.user$ u, sys.sum$ s**

**where o.type# = 42**

**and bitand(s.mflags, 8) = 8;**

**exit**

expecting no rows returned

1. **Execute pre-upgrade tool**

run => **$ORACLE\_HOME/jdk/bin/java -jar /orahome/u01/app/oracle/product/${UPGVER}/db\_1/rdbms/admin/preupgrade.jar TERMINAL TEXT**

Log can be found in /orahome/u01/app/oracle/cfgtoollogs/HEPYPRD\_xhepydbm21p/preupgrade

Upgrade-To version: 19.0.0.0.0

=======================================

Status of the database prior to upgrade

=======================================

Database Name: HEPYPRD

Container Name: HEPYPRD

Container ID: 0

Version: 12.1.0.2.0

DB Patch Level: DATABASE PATCH SET UPDATE 12.1.0.2.200414

Compatible: 12.1.0.1

Blocksize: 8192

Platform: Linux x86 64-bit

Timezone File: 18

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

Oracle Workspace Manager [to be upgraded] VALID

OLAP Analytic Workspace [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. Run 12.1.0.2.0 $ORACLE\_HOME/rdbms/admin/utlrp.sql to recompile invalid

objects. You can view the individual invalid objects with

SET SERVEROUTPUT ON;

EXECUTE DBMS\_PREUP.INVALID\_OBJECTS;

131 objects are INVALID.

There should be no INVALID objects in SYS/SYSTEM or user schemas before

database upgrade.

2. Remove OLAP Catalog by running the 12.1.0.2.0 SQL script

$ORACLE\_HOME/olap/admin/catnoamd.sql script.

The OLAP Catalog component, AMD, exists in the database.

Starting with Oracle Database 12c, the OLAP Catalog (OLAP AMD) is

desupported and will be automatically marked as OPTION OFF during the

database upgrade if present. Oracle recommends removing OLAP Catalog

(OLAP AMD) before database upgrade. This step can be manually performed

before the upgrade to reduce downtime.

3. Perform one of the following:

1) Expire user accounts that use only the old 10G password version and

follow the procedure recommended in Oracle Database Upgrade Guide under

the section entitled, "Checking for Accounts Using Case-Insensitive

Password Version".

2) Explicitly set SQLNET.ALLOWED\_LOGON\_VERSION\_SERVER in the 19

SQLNET.ORA to a non-Exclusive Mode value, such as "11". (This is a short

term approach and is not recommended because it will retain known

security risks associated with the 10G password version.)

Your database system has at least one account with only the 10G password

version (see the PASSWORD\_VERSIONS column of DBA\_USERS).

Starting with Oracle Database release 12.2.0.1, Exclusive Mode is the new

default password-based authentication mode. All Exclusive Mode

login/authentication attempts will fail for preexisting user accounts

which only have the 10G password version and neither the 11G or 12C

password version (see DBA\_USERS.PASSWORD\_VERSIONS.) For more information,

refer to "Understanding Password Case Sensitivity and Upgrades" in the

Oracle Database Upgrade Guide.

4. Upgrade Oracle Application Express (APEX) manually before the database

upgrade.

The database contains APEX version 4.2.5.00.08. Upgrade APEX to at least

version 18.2.0.00.12.

Starting with Oracle Database Release 18, APEX is not upgraded

automatically as part of the database upgrade. Refer to My Oracle Support

Note 1088970.1 for information about APEX installation and upgrades.

5. Review and remove any unnecessary HIDDEN/UNDERSCORE parameters.

The database contains the following initialization parameters whose name

begins with an underscore:

\_bct\_public\_dba\_buffer\_size

\_bct\_buffer\_allocation\_max

\_bct\_public\_dba\_buffer\_maxsize

\_sqlmon\_max\_planlines

Remove hidden parameters before database upgrade unless your application

vendors and/or Oracle Support state differently. Changes will need to be

made in the pfile/spfile.

6. Review and remove any unnecessary EVENTS.

The database contains events.

There are events set that should be removed before upgrade, unless your

application vendors and/or Oracle Support state differently. Changes

will need to be made in the pfile/spfile.

INFORMATION ONLY

================

7. To help you keep track of your tablespace allocations, the following

AUTOEXTEND tablespaces are expected to successfully EXTEND during the

upgrade process.

Min Size

Tablespace Size For Upgrade

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

SYSTEM 5090 MB 5357 MB

Minimum tablespace sizes for upgrade are estimates.

8. 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.

===========

TER UPGRADE

===========

REQUIRED ACTIONS

================

None

RECOMMENDED ACTIONS

===================

9. (AUTOFIXUP) If you use the -T option for the database upgrade, then run

$ORACLE\_HOME/rdbms/admin/utluptabdata.sql after the upgrade is complete,

to VALIDATE and UPGRADE any user tables affected by changes to

Oracle-Maintained types.

There are user tables dependent on Oracle-Maintained object types.

If the -T option is used to set user tablespaces to READ ONLY during the

upgrade, user tables in those tablespaces, that are dependent on

Oracle-Maintained types, will not be automatically upgraded. If a type is

evolved during the upgrade, any dependent tables need to be re-validated

and upgraded to the latest type version AFTER the database upgrade

completes.

10. Upgrade the database time zone file using the DBMS\_DST package.

The database is using time zone file version 18 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.

11. Recreate directory objects to remove any symbolic links from directory

paths. To identify paths that contain symbolic links before upgrading,

use OS commands like UNIX file or WINDOWS dir. After upgrading, run

$ORACLE\_HOME/rdbms/admin/utldirsymlink.sql to identify directory objects

with symbolic links in the path.

Found 4 user directory objects to be checked: HEDATAPUMPDIR,

PAYOR\_LOG\_DIR, PREUPG\_OUTPUT\_DIR, UPG\_XT\_LOG\_DIR.

Starting in Release 18c, symbolic links are not allowed in directory

object paths used with BFILE data types, the UTL\_FILE package, or

external tables.

12. (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.

13. 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 12.1.0.2

Oracle Database SQL Tuning Guide.

INFORMATION ONLY

================

14. Check the Oracle documentation for the identified components for their

specific upgrade procedure.

The database upgrade script will not upgrade the following Oracle

components: OLAP Catalog,OWB

The Oracle database upgrade script upgrades most, but not all Oracle

Database components that may be installed. Some components that are not

upgraded may have their own upgrade scripts, or they may be deprecated or

obsolete.

ORACLE GENERATED FIXUP SCRIPT

=============================

All of the issues in database HEPYPRD

which are identified above as AFTER UPGRADE "(AUTOFIXUP)" can be resolved by

executing the following

SQL>@/orahome/u01/app/oracle/cfgtoollogs/HEPYPRD\_xhepydbm21p/preupgrade/post

upgrade\_fixups.sql

1. **Remove hidden parameter as instructed in the pre-upgrade findings**

Below to be run in primary and standby(if exists)

run => **cd $DBS**

run => **sqlplus / as sysdba**

\_bct\_public\_dba\_buffer\_size 100000000

\_bct\_buffer\_allocation\_max 1048576000

\_bct\_public\_dba\_buffer\_maxsize 110000000

alter system reset "\_bct\_public\_dba\_buffer\_size" scope=spfile;

alter system reset "\_bct\_buffer\_allocation\_max" scope=spfile;

alter system reset "\_bct\_public\_dba\_buffer\_maxsize" scope=spfile;

sql => **alter system reset "\_sqlmon\_max\_planlines" scope=spfile;**

sql => **alter system reset "event" scope=spfile;**

1. **Execute pre-fixups**

run => **cd $ORACLE\_BASE/cfgtoollogs/${ORACLE\_SID}\_${SRVNAME}/preupgrade**

run => **sqlplus / as sysdba**

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: 2021-03-14 06:11:47

For Source Database: HEPYPRD

Source Database Version: 12.1.0.2.0

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. amd\_exists NO Manual fixup recommended.

3. exclusive\_mode\_auth NO Manual fixup recommended.

4. apex\_manual\_upgrade NO Manual fixup recommended.

5. hidden\_params NO Informational only.

Further action is optional.

6. underscore\_events NO Informational only.

Further action is optional.

7. tablespaces\_info NO Informational only.

Further action is optional.

8. 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.

1. **Check and recompile invalids**

run => **cd /home/oracle/tls/upg19c**

sql => **sqlplus / as sysdba**

SQL\*Plus: Release 12.1.0.2.0 Production on Mon Jan 25 12:55:00 2021

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

Enter user-name: / as sysdba

Connected to:

Oracle Database 12c Enterprise Edition Release 12.1.0.2.0 - 64bit Production

With the Partitioning, Automatic Storage Management, OLAP, Advanced Analytics

and Real Application Testing options

sql => **@check\_invalids**

run => **sqlplus / as sysdba**

sql => **@?/rdbms/admin/utlrp**

sql => **@check\_invalids**

no rows selected

Note: Extract above invalid objects and drop them

1. **Fix 10g user ID passwords and change SYS password**
   1. **Check for 10g password version**

sql =>  **select USERNAME from DBA\_USERS where ( PASSWORD\_VERSIONS = '10G ' or PASSWORD\_VERSIONS = '10G HTTP ') and USERNAME <> 'ANONYMOUS';**

**USERNAME  
---------**

**ORATIVOLI  
DBCMS  
AEAUDIT**

* 1. **Fix standard users 10g password version**

run => **sqlplus / as sysdba**

sql => **@fix\_standard\_user\_10g\_password.sql**

SQL> set echo on

SQL> ALTER PROFILE "TRUSTED\_ID\_NO\_EXPIRE" LIMIT PASSWORD\_REUSE\_MAX UNLIMITED PASSWORD\_REUSE\_TIME UNLIMITED;

Profile altered.

SQL> alter user dbcms identified by VALUES 'S:8406389178114D80367B21E95B64D166500DFC0F6CB00D8B09A116F8385F;T:D216BD1BE9C8187C66D822B4;

User altered.

SQL> alter user tom identified by VALUES 'S:8EC3851304545F0EE2B4098FE13EF981E7A8D3BB9A50EA2CCEF8790E24C9;T:A7C769D7B2DBE4141F5096BE12;

User altered.

SQL> alter user AEAUDIT identified by VALUES 'S:39AE6CE7DE6B273DADE99C7F3961D043982B2B512D3F2897215616761A9C;T:2769E98E82E537D2C6747F;

User altered.

SQL> alter user orativoli identified by VALUES 'S:472152CDDB4D5CEF042D73FDFC1147771218C86DD8C4945181E02A3A57C7;T:83E9538D576AF11E266C;

User altered.

SQL> ALTER PROFILE trusted\_id\_no\_expire LIMIT PASSWORD\_REUSE\_MAX 6 PASSWORD\_REUSE\_TIME 365;

Profile altered.

* 1. **Change SYS password**

~~If database has~~ **~~no~~** ~~physical standby database~~

~~run =>~~ **~~sqlplus / as sysdba~~**

~~sql =>~~ **~~alter user sys identified by “Locked#99999”;~~**

If database has a physical standby database run below from the primary server

run => **$SCRIPTS/change\_sys\_pwd.sh HEPYPRD**

**Note: Will prompt for current SYS password**

SYS Password = **bmIA?5Ty** (Need to pull from TPAM)

1. **Save Parameter Files**

run => **cd $DBS**

run => **cp -p spfile${ORACLE\_SID}.ora $HOME/tls/upg19c/spfile${ORACLE\_SID}\_b4\_19cupg.ora**

run => **sqlplus / as sysdba**

sql => **create pfile from spfile;**

run => **cp -p init${ORACLE\_SID}.ora $HOME/tls/upg19c/init${ORACLE\_SID}\_b4\_19cupg.ora**

**Comment heartbeat scripts in cron (primary server)**

#\*/5 \* \* \* \* /home/oracle/tls/rman/heartbeat.ksh HEPYPRD AEDBA > /dev/null 2>&1

#0 \* \* \* \* /home/oracle/tls/rman/confirm\_heartbeat\_active\_in\_cron.sh HEPYPRD > /dev/null 2>&1

1. **Run HeartBeat Script (primary server)**

run => **/home/oracle/tls/rman/heartbeat.ksh ${ORACLE\_SID} AEDBA**

**03-14-2021 06:31:08 AM**

1. **Perform a Log Switch on Primary Database**

sql => **alter system switch logfile;**

1. **Expand FRA, Configure flashback\_database, and Create Restore Point**

**Note: FRA was already defined in HEPYPRD (Otherwise would need to create)**

**Standby Database (if exists)**

run => **dgmgrl /** dgmgrl => **edit database 'HEPYPRD\_xhepydbw21p' set state = 'APPLY-OFF';** dgmgrl => **show database verbose 'HEPYPRD\_xhepydbw21p';**

run => **sqlplus / as sysdba**

sql => **alter system set db\_recovery\_file\_dest\_size = 600G scope=both;**

sql => **show parameter db\_recovery\_file\_dest**

sql => **alter database flashback on;**

sql => **select flashback\_on from v$database;**

FLASHBACK\_ON  
 ---------------------  
 YES

run => **dgmgrl /**

dgmgrl => **edit database 'HEPYPRD\_xhepydbw21p' set state = 'APPLY-ON';** dgmgrl => **show database verbose 'HEPYPRD\_xhepydbw21p';**

**Primary Database**

run => **sqlplus / as sysdba**

sql => **alter system set db\_recovery\_file\_dest\_size = 600G scope=both;**

sql => **show parameter db\_recovery\_file\_dest**

sql => **create restore point b4\_db\_upgrade guarantee flashback database;**

**col name for a25  
 col SCN for 9999999999999999  
 col TIME for a35**

sql => **select name,scn,time from v$restore\_point where guarantee\_flashback\_database='YES';**

NAME SCN TIME  
 ------------------------- ---------------------- ------------------------------------------------

B4\_DB\_UPGRADE 13584454309519 14-MAR-21 06.38.21.000000000 AM

**Uncomment heartbeat script in cron (primary server)**

\*/5 \* \* \* \* /home/oracle/tls/rman/heartbeat.ksh HEPYPRD AEDBA > /dev/null 2>&1

0 \* \* \* \* /home/oracle/tls/rman/confirm\_heartbeat\_active\_in\_cron.sh HEPYPRD > /dev/null 2>&1

1. **Upgrade apex**

(Note: Below created x-12 gb flashback logs)

run =>  **cd /orahome/u01/app/oracle/product/${UPGVER}/db\_1/apex**

run => **sqlplus / as sysdba**

sql => **@apexins SYSAUX SYSAUX TEMP /i/**

timing for: Phase 3 (Switch)

Elapsed: 00:00:35.62

timing for: Complete Installation

Elapsed: 00:05:14.47

PL/SQL procedure successfully completed.

sql => **select count(\*) from v$flashback\_database\_logfile;**

**2**

1. **Drop old APEX schema**

Note: Below created x-12 gb flashback log

run => **sqlplus / as sysdba**

sql => **drop user apex\_030200 cascade;**sql => **drop user** **apex\_040200 cascade;**

sql => **select count(\*) from v$flashback\_database\_logfile;**

**2**

1. **Check and Recompile invalids**

run => **cd /home/oracle/tls/upg19c**

run => **sqlplus / as sysdba**

sql => **@check\_invalids**

sql => **@?/rdbms/admin/utlrp**

sql => **@check\_invalids**

1. **(Optional if DG is in use) Disable Broker**

**Primary**

run => **dgmgrl /**

dgmgrl => **disable configuration;**

dgmgrl => **quit**

run => **sqlplus / as sysdba**

sql => **alter system set dg\_broker\_start=false;**

**Standby**

run => **sqlplus / as sysdba**

sql => **alter system set dg\_broker\_start=false;**

1. **(Optional if DG is in use) Upgrade Standby database on standby server**

**Open a new session and logon to the standby database server**

**. oraenv  
HEPYPRD**

run => **export UPGVER=19.9.0**

run => **export SRVNAME=xhepydbw21p**

**cd $SCRIPTS**

run => **upgrade\_standby\_db.sh ${UPGVER} $ORACLE\_SID**

**Log: /orahome/u01/app/oracle/local/logs/upgrade\_standby\_db\_HEPYPRD\_20210314\_1615719170.out**

EST

Logoff standby database server

1. **(Optional if DG is in use) Start recovery on standby database from standby server**

run => **.oraenv**  
 **HEPYPRD**

run => **sqlplus / as sysdba**

sql => **alter database recover managed standby database disconnect from session;**

sql => **quit**

1. **Shutdown database and listener on primary server**

run => **srvctl stop database -d ${ORACLE\_SID}\_${SRVNAME}**

run => **srvctl stop listener -l ${ORACLE\_SID}**

1. **Copy listener.ora and tnsnames.ora to new network/admin directory.**

run => **cd $ORACLE\_HOME/network/admin**

run => **cp -p tnsnames.ora $ORACLE\_BASE/product/${UPGVER}/db\_1/network/admin**

run => **cp -p listener.ora $ORACLE\_BASE/product/${UPGVER}/db\_1/network/admin**

**Make backup copies**

run => **cp -p tnsnames.ora $HOME/tls/upg19c** run => **cp -p listener.ora $HOME/tls/upg19c**

Update the listener.ora to point to the 19c oracle home if static registration is used

run => **vi $ORACLE\_BASE/product/${UPGVER}/db\_1/network/admin/listener.ora**

1. **Remove database and listener HAS configuration for primary**

run => **srvctl remove database -d ${ORACLE\_SID}\_${SRVNAME}**

**Remove the database HEPYPRD\_xhepydbm21p? (y/[n]) y**

run => **srvctl remove listener -l ${ORACLE\_SID}**

1. **Copy password file, spfile and DG config files (if in use) to new dbs directory**

run => **cd $DBS**

run => **cp -p \*${ORACLE\_SID}\* $ORACLE\_BASE/product/${UPGVER}/db\_1/dbs**

**Make backup copies**

run => **cp -p \*${ORACLE\_SID}\* $HOME/tls/upg19c**

1. **Update oratab to add ${ORACLE\_SID}:/orahome/u01/app/oracle/product/${UPGVER}/db\_1:N  
    and source environment**

run => **cat /etc/oratab**

GCAGENT:/orahome/u01/app/oracle/product/agent12c/agent\_13.4.0.0.0:N # gcagent entry  
200114:/orahome/u01/app/oracle/product/12.1.0.2.200114/db\_1:N # dummy entry  
200414:/orahome/u01/app/oracle/product/12.1.0.2.200414/db\_1:N # dummy entry  
+ASM:/orahome/u01/app/oracle/product/19.0.0/grid:N # line added by Agent  
HEPYDEV:/orahome/u01/app/oracle/product/12.1.0.2.200414/db\_1:N # line added by Agent  
1990:/orahome/u01/app/oracle/product/19.9.0/db\_1:N # dummy entry

run =>  **. oraenv  
 HEPYPRD**

ORACLE\_SID = [HEPYPRD] ? HEPYPRD

The Oracle base remains unchanged with value /orahome/u01/app/oracle

=================================================================================

CUSTOM VARIABLES:

NLS\_LANG American\_America.UTF8

NLS\_DATE\_FORMAT DD-MON-YYYY HH24:MI:SS

DB\_UNIQUE\_NAME hepyprd\_xhepydbm21p

LISTENER\_NAME hepyprd

BDUMP /orahome/u01/app/oracle/diag/rdbms/hepyprd\_xhepydbm21p/HEPYPRD/trace

CDUMP /orahome/u01/app/oracle/diag/rdbms/hepyprd\_xhepydbm21p/HEPYPRD/cdump

UDUMP /orahome/u01/app/oracle/diag/rdbms/hepyprd\_xhepydbm21p/HEPYPRD/trace

DBS /orahome/u01/app/oracle/product/19.9.0/db\_1/dbs

LSNRLOG /orahome/u01/app/oracle/diag/tnslsnr/xhepydbm21p/hepyprd/trace

TNS\_ADMIN /orahome/u01/app/oracle/product/19.9.0/db\_1/network/admin =================================================================================

1. **Start database upgrade**

run => **sqlplus / as sysdba**

sql => **startup upgrade;**

1. **Execute upgrade**

run => **sqlplus / as sysdba**

sql => **select count(\*) from v$flashback\_database\_logfile;**

**3**

sql => **exit;**

run => **cd $HOME/tls/upg19c**

run => **nohup upgrade\_db\_to\_19c.ksh ${ORACLE\_SID} &**

**Optional: Open new ssh connection**

run => **cd $HOME/tls/upg19c**

**tail -f nohup.out**

**Optional: Open new ssh connection**

**tail -f $ORACLE\_BASE/local/logs/catupgrd0.log**

**Note: generated x-12 gb flashback logs**

Note: Running with default parallel of 4

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

Phases [0-107] End Time:[2021\_03\_14 07:38:25]

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

Grand Total Time: 1226s

LOG FILES: (/orahome/u01/app/oracle/local/logs/catupgrd\*.log)

Upgrade Summary Report Located in:

/orahome/u01/app/oracle/local/logs/upg\_summary.log

Grand Total Upgrade Time: [0d:0h:20m:26s]

1. **Rename the upgrade summary log to contain the database name.**

run => **mv /orahome/u01/app/oracle/local/logs/upg\_summary.log /orahome/u01/app/oracle/local/logs/${ORACLE\_SID}\_upg\_summary.log**

1. **Add primary database and listener HAS configuration.**

Note: Standby database (if exists) HAS was configured in upgrade script upgrade\_standby\_db.sh

HEPYPRD PORT = 1574

run => **srvctl add listener -l ${ORACLE\_SID} -oraclehome ${ORACLE\_HOME} -endpoints "TCP:1574/IPC:${ORACLE\_SID}\_IPC"**

run => **srvctl add database -db ${ORACLE\_SID}\_${SRVNAME} -oraclehome ${ORACLE\_HOME} -spfile ${ORACLE\_HOME}/dbs/spfile${ORACLE\_SID}.ora -instance ${ORACLE\_SID} -diskgroup "DATA\_01,IND\_01,REDOA\_01,REDOB\_01"**

run => **srvctl setenv  listener -l HEPYPRD -envs "TNS\_ADMIN=$ORACLE\_HOME/network/admin"**

1. **Start database and Listener**

run => **srvctl start database -d ${ORACLE\_SID}\_${SRVNAME}**

**Note: starting the database will start the listener as well.**

1. **Remove parameters per Health Edge vendor and add back removed parameter**

**Primary**

run => **cd $DBS**

run => **cp -p spfile${ORACLE\_SID}.ora spfile${ORACLE\_SID}\_b4\_param\_removal**

run => **sqlplus / as sysdba**

sql => **alter system reset optimizer\_index\_caching scope=spfile;**

sql => **alter system reset optimizer\_index\_cost\_adj scope=spfile;**

sql => **alter system set "\_sqlmon\_max\_planlines"=1000 scope=spfile;**

**alter system set "\_bct\_public\_dba\_buffer\_size"=100000000 scope=spfile;**

**alter system set "\_bct\_buffer\_allocation\_max"=1048576000 scope=spfile;**

**alter system set "\_bct\_public\_dba\_buffer\_maxsize"=110000000 scope=spfile;**

run => **srvctl stop database -d ${ORACLE\_SID}\_${SRVNAME}**

run => **srvctl start database -d ${ORACLE\_SID}\_${SRVNAME}**

**select count(\*) from v$flashback\_database\_logfile;**

**5**

1. **Replace ORACLE\_HOME symbolic Link**

run => **. oraenv  
 HEPYPRD**

run => **cd $ORACLE\_BASE/admin/${ORACLE\_SID}**

run => **ls -l oracle\_home**

lrwxrwxrwx 1 oracle dba 52 Feb 2 16:35 oracle\_home -> /orahome/u01/app/oracle/product/12.1.0.2.200414/db\_1

run => **rm oracle\_home**

run => **ln -s $ORACLE\_HOME oracle\_home**

run => **ls -l oracle\_home**

lrwxrwxrwx 1 oracle dba 43 Feb 25 12:39 oracle\_home -> /orahome/u01/app/oracle/product/19.9.0/db\_1

1. **Execute Post fixup script**

run => **sqlplus / as sysdba**

sql => **@/orahome/u01/app/oracle/cfgtoollogs/${ORACLE\_SID}\_${SRVNAME}/preupgrade/postupgrade\_fixups.sql**

SQL> @/orahome/u01/app/oracle/cfgtoollogs/${ORACLE\_SID}\_${SRVNAME}/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: 19.0.0.0.0 Build: 1

Generated on: 2021-03-14 06:11:50

For Source Database: HEPYPRD

Source Database Version: 12.1.0.2.0

For Upgrade to Version: 19.0.0.0.0

Preup Preupgrade

Action Issue Is

Number Preupgrade Check Name Remedied Further DBA Action

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

9. depend\_usr\_tables YES None.

10. old\_time\_zones\_exist NO Manual fixup recommended.

11. dir\_symlinks YES None.

12. post\_dictionary YES None.

13. post\_fixed\_objects NO Informational only.

Further action is optional.

14. upg\_by\_std\_upgrd 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.

1. **Upgrade RMAN catalog**

run => **rmanc**

rman => **upgrade catalog;**

rman => **upgrade catalog;**

connected to target database: HEPYPRD (DBID=915700946)

connected to recovery catalog database

PL/SQL package HEPYPRD.DBMS\_RCVCAT version 12.01.00.02. in RCVCAT database is too old

RMAN> upgrade catalog;

recovery catalog owner is HEPYPRD

enter UPGRADE CATALOG command again to confirm catalog upgrade

RMAN> upgrade catalog;

recovery catalog upgraded to version 19.09.00.00.00

DBMS\_RCVMAN package upgraded to version 19.09.00.00

DBMS\_RCVCAT package upgraded to version 19.09.00.00.

1. **Upgrade Time zone file**
   1. **Execute utlz\_countstats.sql to get optimizer statistics showing num\_rows of all the tables having TIMESTAMP WITH TIME ZONE (TSTZ) data.**

run => **sqlplus / as sysdba**

sql => **@?/rdbms/admin/utltz\_countstats.sql**

Amount of TSTZ data using num\_rows stats info in DBA\_TABLES.

.

For SYS tables first ...

Note: empty tables are not listed.

Stat date - Owner.TableName.ColumnName - num\_rows

10/06/2018 - SYS.AQ$\_ALERT\_QT\_S.CREATION\_TIME - 10

10/06/2018 - SYS.AQ$\_ALERT\_QT\_S.DELETION\_TIME - 10

10/06/2018 - SYS.AQ$\_ALERT\_QT\_S.MODIFICATION\_TIME - 10

09/03/2015 - SYS.AQ$\_AQ$\_MEM\_MC\_S.CREATION\_TIME - 3

09/03/2015 - SYS.AQ$\_AQ$\_MEM\_MC\_S.DELETION\_TIME - 3

09/03/2015 - SYS.AQ$\_AQ$\_MEM\_MC\_S.MODIFICATION\_TIME - 3

17/09/2011 - SYS.AQ$\_AQ\_PROP\_TABLE\_S.CREATION\_TIME - 1

17/09/2011 - SYS.AQ$\_AQ\_PROP\_TABLE\_S.DELETION\_TIME - 1

17/09/2011 - SYS.AQ$\_AQ\_PROP\_TABLE\_S.MODIFICATION\_TIME - 1

14/03/2021 - SYS.AQ$\_ORA$PREPLUGIN\_BACKUP\_QTB\_S.CREATION\_TIME - 1

14/03/2021 - SYS.AQ$\_ORA$PREPLUGIN\_BACKUP\_QTB\_S.DELETION\_TIME - 1

14/03/2021 - SYS.AQ$\_ORA$PREPLUGIN\_BACKUP\_QTB\_S.MODIFICATION\_TIME - 1

14/03/2021 - SYS.AQ$\_PDB\_MON\_EVENT\_QTABLE$\_S.CREATION\_TIME - 1

14/03/2021 - SYS.AQ$\_PDB\_MON\_EVENT\_QTABLE$\_S.DELETION\_TIME - 1

14/03/2021 - SYS.AQ$\_PDB\_MON\_EVENT\_QTABLE$\_S.MODIFICATION\_TIME - 1

09/03/2015 - SYS.AQ$\_SCHEDULER$\_EVENT\_QTAB\_S.CREATION\_TIME - 3

09/03/2015 - SYS.AQ$\_SCHEDULER$\_EVENT\_QTAB\_S.DELETION\_TIME - 3

09/03/2015 - SYS.AQ$\_SCHEDULER$\_EVENT\_QTAB\_S.MODIFICATION\_TIME - 3

17/09/2011 - SYS.AQ$\_SCHEDULER$\_REMDB\_JOBQTAB\_S.CREATION\_TIME - 1

17/09/2011 - SYS.AQ$\_SCHEDULER$\_REMDB\_JOBQTAB\_S.DELETION\_TIME - 1

17/09/2011 - SYS.AQ$\_SCHEDULER$\_REMDB\_JOBQTAB\_S.MODIFICATION\_TIME - 1

09/03/2015 - SYS.AQ$\_SCHEDULER\_FILEWATCHER\_QT\_S.CREATION\_TIME - 1

09/03/2015 - SYS.AQ$\_SCHEDULER\_FILEWATCHER\_QT\_S.DELETION\_TIME - 1

09/03/2015 - SYS.AQ$\_SCHEDULER\_FILEWATCHER\_QT\_S.MODIFICATION\_TIME - 1

14/03/2021 - SYS.AQ$\_SUBSCRIBER\_TABLE.CREATION\_TIME - 1

14/03/2021 - SYS.AQ$\_SUBSCRIBER\_TABLE.DELETION\_TIME - 1

14/03/2021 - SYS.AQ$\_SUBSCRIBER\_TABLE.MODIFICATION\_TIME - 1

14/03/2021 - SYS.AQ$\_SYS$SERVICE\_METRICS\_TAB\_S.CREATION\_TIME - 4

14/03/2021 - SYS.AQ$\_SYS$SERVICE\_METRICS\_TAB\_S.DELETION\_TIME - 4

14/03/2021 - SYS.AQ$\_SYS$SERVICE\_METRICS\_TAB\_S.MODIFICATION\_TIME - 4

14/03/2021 - SYS.ATSK$\_SCHEDULE\_CONTROL.MRCT\_TASK\_TIME\_TZ - 2

14/03/2021 - SYS.KET$\_AUTOTASK\_STATUS.ABA\_START\_TIME - 1

14/03/2021 - SYS.KET$\_AUTOTASK\_STATUS.ABA\_STATE\_TIME - 1

14/03/2021 - SYS.KET$\_AUTOTASK\_STATUS.MW\_RECORD\_TIME - 1

14/03/2021 - SYS.KET$\_AUTOTASK\_STATUS.MW\_START\_TIME - 1

14/03/2021 - SYS.KET$\_AUTOTASK\_STATUS.RECONCILE\_TIME - 1

14/03/2021 - SYS.KET$\_CLIENT\_CONFIG.FIELD\_2 - 7

14/03/2021 - SYS.KET$\_CLIENT\_CONFIG.LAST\_CHANGE - 7

14/03/2021 - SYS.KET$\_CLIENT\_TASKS.CURR\_WIN\_START - 3

14/03/2021 - SYS.KET$\_CLIENT\_TASKS.LG\_DATE - 3

14/03/2021 - SYS.KET$\_CLIENT\_TASKS.LT\_DATE - 3

14/03/2021 - SYS.OPTSTAT\_HIST\_CONTROL$.SPARE6 - 45

14/03/2021 - SYS.OPTSTAT\_HIST\_CONTROL$.SVAL2 - 45

14/03/2021 - SYS.OPTSTAT\_SNAPSHOT$.TIMESTAMP - 118

14/03/2021 - SYS.OPTSTAT\_USER\_PREFS$.CHGTIME - 72

10/03/2015 - SYS.RADM\_FPTM$.TSWTZ\_COL - 1

14/03/2021 - SYS.REG$.NTFN\_GROUPING\_START\_TIME - 2

14/03/2021 - SYS.REG$.REG\_TIME - 2

10/03/2021 - SYS.SCHEDULER$\_EVENT\_LOG.LOG\_DATE - 70160

14/03/2021 - SYS.SCHEDULER$\_GLOBAL\_ATTRIBUTE.ATTR\_TSTAMP - 11

14/03/2021 - SYS.SCHEDULER$\_JOB.END\_DATE - 47

14/03/2021 - SYS.SCHEDULER$\_JOB.LAST\_ENABLED\_TIME - 47

14/03/2021 - SYS.SCHEDULER$\_JOB.LAST\_END\_DATE - 47

14/03/2021 - SYS.SCHEDULER$\_JOB.LAST\_START\_DATE - 47

14/03/2021 - SYS.SCHEDULER$\_JOB.NEXT\_RUN\_DATE - 47

14/03/2021 - SYS.SCHEDULER$\_JOB.START\_DATE - 47

13/03/2021 - SYS.SCHEDULER$\_JOB\_RUN\_DETAILS.LOG\_DATE - 22125

13/03/2021 - SYS.SCHEDULER$\_JOB\_RUN\_DETAILS.REQ\_START\_DATE - 22125

13/03/2021 - SYS.SCHEDULER$\_JOB\_RUN\_DETAILS.START\_DATE - 22125

09/03/2015 - SYS.SCHEDULER$\_SCHEDULE.END\_DATE - 4

09/03/2015 - SYS.SCHEDULER$\_SCHEDULE.REFERENCE\_DATE - 4

14/03/2021 - SYS.SCHEDULER$\_WINDOW.ACTUAL\_START\_DATE - 9

14/03/2021 - SYS.SCHEDULER$\_WINDOW.END\_DATE - 9

14/03/2021 - SYS.SCHEDULER$\_WINDOW.LAST\_START\_DATE - 9

14/03/2021 - SYS.SCHEDULER$\_WINDOW.MANUAL\_OPEN\_TIME - 9

14/03/2021 - SYS.SCHEDULER$\_WINDOW.NEXT\_START\_DATE - 9

14/03/2021 - SYS.SCHEDULER$\_WINDOW.START\_DATE - 9

13/03/2021 - SYS.SCHEDULER$\_WINDOW\_DETAILS.LOG\_DATE - 30

13/03/2021 - SYS.SCHEDULER$\_WINDOW\_DETAILS.REQ\_START\_DATE - 30

13/03/2021 - SYS.SCHEDULER$\_WINDOW\_DETAILS.START\_DATE - 30

14/03/2021 - SYS.STATS\_TARGET$.END\_TIME - 4130

14/03/2021 - SYS.STATS\_TARGET$.START\_TIME - 4130

09/03/2015 - SYS.TAB\_STATS$.SPARE6 - 979

10/03/2021 - SYS.WRI$\_ALERT\_HISTORY.CREATION\_TIME - 344

10/03/2021 - SYS.WRI$\_ALERT\_HISTORY.TIME\_SUGGESTED - 344

14/03/2021 - SYS.WRI$\_OPTSTAT\_HISTGRM\_HISTORY.SAVTIME - 4002882

14/03/2021 - SYS.WRI$\_OPTSTAT\_HISTGRM\_HISTORY.SPARE6 - 4002882

14/03/2021 - SYS.WRI$\_OPTSTAT\_HISTHEAD\_HISTORY.SAVTIME - 193108

14/03/2021 - SYS.WRI$\_OPTSTAT\_HISTHEAD\_HISTORY.SPARE6 - 193108

14/03/2021 - SYS.WRI$\_OPTSTAT\_IND\_HISTORY.SAVTIME - 29919

14/03/2021 - SYS.WRI$\_OPTSTAT\_IND\_HISTORY.SPARE6 - 29919

14/03/2021 - SYS.WRI$\_OPTSTAT\_OPR.END\_TIME - 2140

14/03/2021 - SYS.WRI$\_OPTSTAT\_OPR.SPARE6 - 2140

14/03/2021 - SYS.WRI$\_OPTSTAT\_OPR.START\_TIME - 2140

14/03/2021 - SYS.WRI$\_OPTSTAT\_OPR\_TASKS.END\_TIME - 49376

14/03/2021 - SYS.WRI$\_OPTSTAT\_OPR\_TASKS.SPARE6 - 49376

14/03/2021 - SYS.WRI$\_OPTSTAT\_OPR\_TASKS.START\_TIME - 49376

14/03/2021 - SYS.WRI$\_OPTSTAT\_TAB\_HISTORY.SAVTIME - 10575

14/03/2021 - SYS.WRI$\_OPTSTAT\_TAB\_HISTORY.SPARE6 - 10575

14/03/2021 - SYS.WRM$\_DATABASE\_INSTANCE.STARTUP\_TIME\_TZ - 18

13/03/2021 - SYS.WRM$\_SNAPSHOT.BEGIN\_INTERVAL\_TIME\_TZ - 5466

13/03/2021 - SYS.WRM$\_SNAPSHOT.END\_INTERVAL\_TIME\_TZ - 5466

15/06/2015 - SYS.XS$PRIN.END\_DATE - 14

15/06/2015 - SYS.XS$PRIN.START\_DATE - 14

Total numrows of SYS TSTZ columns is : 8785794

There are in total 165 SYS TSTZ columns.

.

For non-SYS tables ...

Note: empty tables are not listed.

Stat date - Owner.Tablename.Columnname - num\_rows

14/03/2021 - APEX\_180200.WWV\_QS\_RANDOM\_NAMES.TSWTZ - 2001

13/03/2021 - DBSNMP.MGMT\_DB\_FEATURE\_LOG.LAST\_UPDATE\_DATE - 3

09/03/2015 - GSMADMIN\_INTERNAL.AQ$\_CHANGE\_LOG\_QUEUE\_TABLE\_S.CREATION\_TIME - 1

09/03/2015 - GSMADMIN\_INTERNAL.AQ$\_CHANGE\_LOG\_QUEUE\_TABLE\_S.DELETION\_TIME - 1

09/03/2015 - GSMADMIN\_INTERNAL.AQ$\_CHANGE\_LOG\_QUEUE\_TABLE\_S.MODIFICATION\_TIME -

1

09/03/2015 - WMSYS.AQ$\_WM$EVENT\_QUEUE\_TABLE\_S.CREATION\_TIME - 1

09/03/2015 - WMSYS.AQ$\_WM$EVENT\_QUEUE\_TABLE\_S.DELETION\_TIME - 1

09/03/2015 - WMSYS.AQ$\_WM$EVENT\_QUEUE\_TABLE\_S.MODIFICATION\_TIME - 1

15/06/2015 - WMSYS.WM$WORKSPACES\_TABLE$.CREATETIME - 1

15/06/2015 - WMSYS.WM$WORKSPACES\_TABLE$.LAST\_CHANGE - 1

Total numrows of non-SYS TSTZ columns is : 2012

There are in total 28 non-SYS TSTZ columns.

Total Minutes elapsed : 0

Session altered.

* 1. **Optionally purge scheduler logs and stats history, we are choosing to keep both**

sql = > **~~exec dbms\_schedule.purge\_log;~~**

sql = > **~~exec dbms\_stats.alter\_stats\_history\_retention(0);~~**

sql = > **~~exec dbms\_stats.purge\_stats(systimestamp);~~**

sql = > **~~exec dbms\_stats.alter\_stats\_history\_retention(31);~~**

* 1. **Execute Timezone upgrade**

run => **cd $HOME/tls/upg19c**

run => **sqlplus / as sysdba**

sql => **spool /orahome/u01/app/oracle/local/logs/${ORACLE\_SID}\_utltz\_upg\_check.log**

sql => **@?/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 DSTv18 .

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.

SQL> **spool off**

SQL> **spool /orahome/u01/app/oracle/local/logs/${ORACLE\_SID}\_utltz\_upg\_apply.log**

SQL> **@?/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 5.1540E+10 bytes

Fixed Size 19527768 bytes

Variable Size 3355443200 bytes

Database Buffers 4.8050E+10 bytes

Redo Buffers 114688000 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 5.1540E+10 bytes

Fixed Size 19527768 bytes

Variable Size 3355443200 bytes

Database Buffers 4.8050E+10 bytes

Redo Buffers 114688000 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: "MDSYS"."SDO\_DIAG\_MESSAGES\_TABLE"

Number of failures: 0

Table list: "GSMADMIN\_INTERNAL"."AQ$\_CHANGE\_LOG\_QUEUE\_TABLE\_L"

Number of failures: 0

Table list: "GSMADMIN\_INTERNAL"."AQ$\_CHANGE\_LOG\_QUEUE\_TABLE\_S"

Number of failures: 0

Table list: "APEX\_180200"."WWV\_FLOW\_ACTIVITY\_LOG1$"

Number of failures: 0

Table list: "APEX\_180200"."WWV\_FLOW\_DEBUG\_MESSAGES"

Number of failures: 0

Table list: "APEX\_180200"."WWV\_FLOW\_DEBUG\_MESSAGES2"

Number of failures: 0

Table list: "APEX\_180200"."WWV\_FLOW\_FEEDBACK"

Number of failures: 0

Table list: "APEX\_180200"."WWV\_QS\_RANDOM\_NAMES"

Number of failures: 0

Table list: "APEX\_180200"."WWV\_FLOW\_ACTIVITY\_LOG2$"

Number of failures: 0

Table list: "APEX\_180200"."WWV\_FLOW\_FEEDBACK\_FOLLOWUP"

Number of failures: 0

Table list: "APEX\_180200"."WWV\_FLOW\_WORKSHEET\_NOTIFY"

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> **spool off**

1. **Check and Recompile Invalids**

run => **sqlplus / as sysdba**

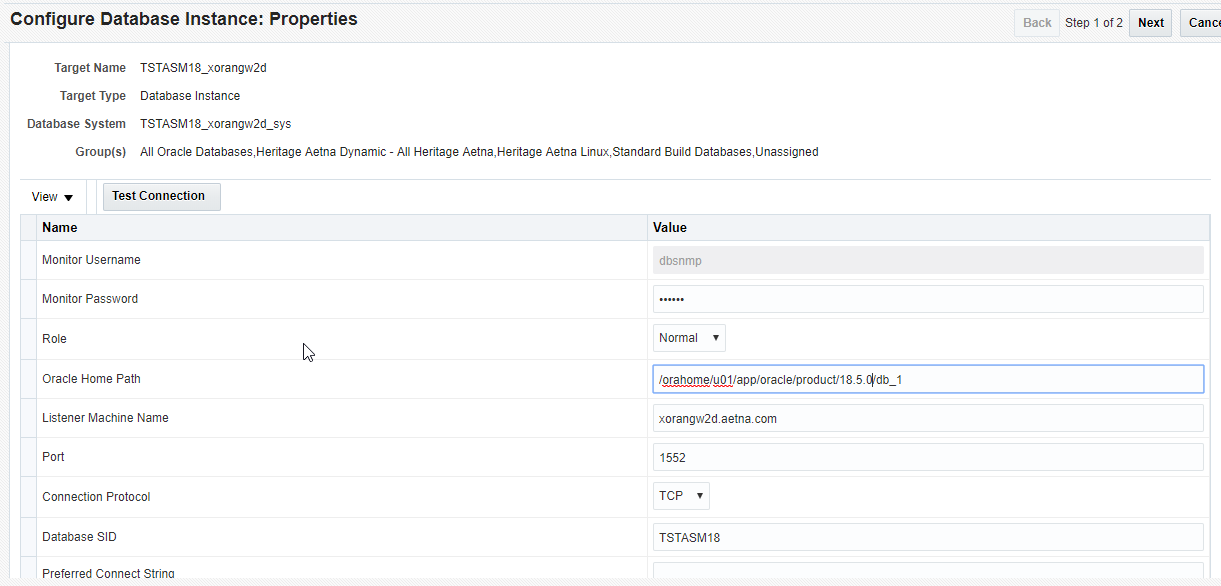
sql => **@?/rdbms/admin/utlrp**

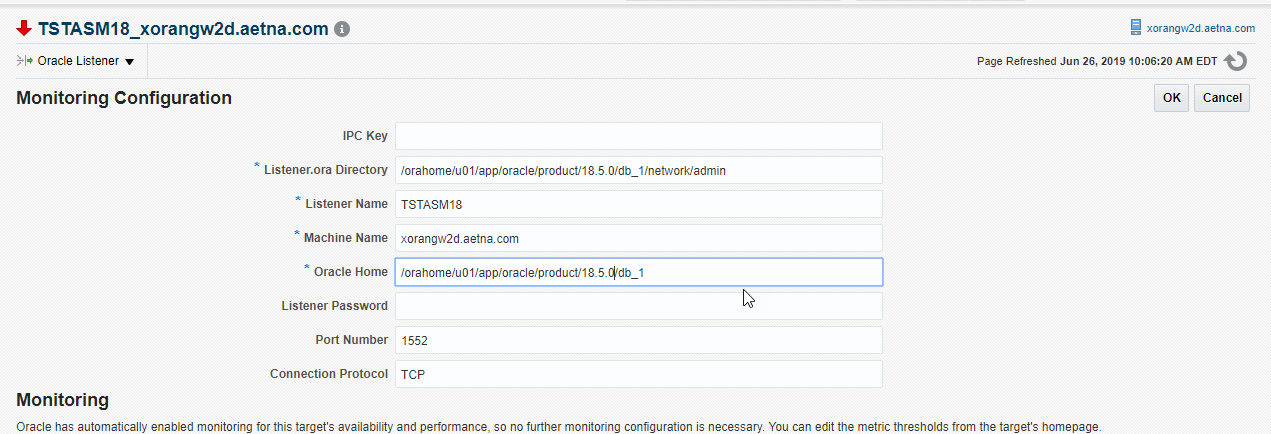
sql => **@check\_invalids**

1. **Update OEM monitoring configuration for database and listener**

**on Primary**

**e.g.**





1. **(optional if standby database) Bounce OEM agent on standby server.**

**. oraenv  
 GCAGENT**

run => **emctl stop agent**

Oracle Enterprise Manager Cloud Control 13c Release 2

Copyright (c) 1996, 2016 Oracle Corporation. All rights reserved.

Stopping agent ... stopped.

run => **emctl start agent**

Oracle Enterprise Manager Cloud Control 13c Release 2

Copyright (c) 1996, 2016 Oracle Corporation. All rights reserved.

Starting agent .............. started.

1. **(optional if ADG is in use) Configure AWR for ADG**

**~~Drop Stats Pack (if applicable)~~**

~~sql =>~~ **~~@${ORACLE\_HOME}/rdbms/admin/sbdrop.sql~~**

~~sql =>~~ **~~@${ORACLE\_HOME}/rdbms/admin/spdrop.sql~~**

**~~How to Generate AWRs in Active Data Guard Standby Databases (Doc ID 2409808.1)~~**

~~run =>~~ **~~. oraenv  
 HEDWPRD~~**

**~~1. unlock user sys$umf and set password on primary database~~**

~~sql =>~~ **~~alter user sys$umf identified by Locked#99999 account unlock;~~**

1. **~~ensure tns entries <$ORACLE\_SID>\_<server\_name> for both primary and standby are in place on both primary and standby servers~~**
2. **~~create database links on primary database~~**~~sql =>~~ **~~create database link primary\_to\_standby connect to "SYS$UMF" identified by "Locked#99999" using '<$ORACLE\_SID>\_<stdby\_server\_name>';~~**~~sql =>~~ **~~create database link standby\_to\_primary connect to "SYS$UMF" identified by "Locked#99999" using '<$ORACLE\_SID\_<prim\_server\_name>';~~**
3. **~~Configure UMF node on primary~~**

**~~Below to be run in primary~~**

~~sql =>~~ **~~alter system set “\_umf\_remote\_enabled”=TRUE scope=BOTH;~~**

~~sql =>~~ **~~exec dbms\_umf.configure\_node('<$ORACLE\_SID>\_<prim\_server\_name>');~~**

1. **~~Configure UMF node on standby~~**

**~~Below to be run in standby~~**

~~sql =>~~ **~~exec dbms\_umf.configure\_node ('<$ORACLE\_SID\_<stdby\_server\_name>','standby\_to\_primary');~~**

1. **~~Create UMF topology on primary  
     
   Below to be run in primary~~**

~~sql =>~~ **~~exec DBMS\_UMF.create\_topology ('Topology\_1');~~**

1. **~~Register standby database with topology on primary~~**~~sql =>~~ **~~exec DBMS\_UMF.register\_node ('Topology\_1', '<$ORACLE\_SID>\_<stdby\_server\_name>', 'primary\_to\_standby', 'standby\_to\_primary', 'FALSE','FALSE');~~**
2. **~~Enable the AWR Service on the node  
     
   Below to be run in primary~~**~~sql =>~~ **~~exec DBMS\_WORKLOAD\_REPOSITORY.register\_remote\_database(node\_name=>'<$ORACLE\_SID>\_<stdby\_server\_name>’);~~**
3. **~~Adjust AWR setting for standby database~~**

**~~Below to be run in primary~~**~~sql =>~~ **~~begin  
dbms\_workload\_repository.modify\_snapshot\_settings(  
retention => 46080,  
interval =>15 ,  
dbid => <standby db\_id>);  
end;  
/~~**

1. **~~To manually create a snap shot from primary database and run a report~~**

~~run =>~~ **~~cd $HOME/tls/upg19c~~**

~~sql =>~~ **~~exec dbms\_workload\_repository.create\_remote\_snapshot('<$ORACLE\_SID>\_<stdby\_server\_name>');~~**

~~sql =>~~ **~~exec dbms\_workload\_repository.create\_remote\_snapshot('<$ORACLE\_SID>\_<stdby\_server\_name>');~~**

~~sql =>~~ **~~@?/rdbms/admin/awrrpti.sql~~**

1. **(optional if DG is in use) Start DG Broker and Enable Broker Configuration**

**Standby**

run => **sqlplus / as sysdba**

sql => **alter system set dg\_broker\_start=true;**

sql => **quit**

**Primary**

run => **sqlplus / as sysdba**

sql => **alter system set dg\_broker\_start=true;**

sql => **quit**

run => **dgmgrl /**

dgmgrl =>  **enable configuration;**

dgmgrl =>  **quit;**

1. **Gather Dictionary Stats (primary)**

run => **sqlplus / as sysdba**

sql => **execute dbms\_stats.gather\_dictionary\_stats;**

1. **(Optional if DG is in use) Remove parameters per Health Edge vendor**

Run on standby server

run => **export SRVNAME=xhepydbw21p**

run => **cd $DBS**

run => **cp -p spfile${ORACLE\_SID}.ora spfile${ORACLE\_SID}\_b4\_param\_removal**

run => **sqlplus / as sysdba**

sql => **alter system reset optimizer\_index\_caching scope=spfile;**

sql => **alter system reset optimizer\_index\_cost\_adj scope=spfile;**

sql => **alter system set "\_sqlmon\_max\_planlines"=1000 scope=spfile;**

**alter system set "\_bct\_public\_dba\_buffer\_size"=100000000 scope=spfile;**

**alter system set "\_bct\_buffer\_allocation\_max"=1048576000 scope=spfile;**

**alter system set "\_bct\_public\_dba\_buffer\_maxsize"=110000000 scope=spfile;**

run => **srvctl stop database -d ${ORACLE\_SID}\_${SRVNAME}**

run => **srvctl start database -d ${ORACLE\_SID}\_${SRVNAME}**

1. **Change user profile to standard for all users created as part of the upgrade process.  
    ~~Change user SYS$UMF to profile trusted\_id\_no\_expire if ADG database involved (primary and standby)~~**
2. **(Optional if DG is in use)  
   Confirm primary and standby functioning properly before dropping restore point**

**and turning of flashback logging**

**SELECT DB\_NAME, HOSTNAME, LOG\_ARCHIVED, LOG\_APPLIED,APPLIED\_TIME,**

**LOG\_ARCHIVED-LOG\_APPLIED LOG\_GAP**

**FROM**

**(**

**SELECT NAME DB\_NAME**

**FROM V$DATABASE**

**),**

**(**

**SELECT UPPER(SUBSTR(HOST\_NAME,1,(DECODE(INSTR(HOST\_NAME,'.'),0,LENGTH(HOST\_NAME),**

**(INSTR(HOST\_NAME,'.')-1))))) HOSTNAME**

**FROM V$INSTANCE**

**),**

**(**

**SELECT MAX(SEQUENCE#) LOG\_ARCHIVED**

**FROM V$ARCHIVED\_LOG WHERE DEST\_ID=1 AND ARCHIVED='YES'**

**),**

**(**

**SELECT MAX(SEQUENCE#) LOG\_APPLIED**

**FROM V$ARCHIVED\_LOG WHERE DEST\_ID=2 AND APPLIED='YES'**

**),**

**(**

**SELECT TO\_CHAR(MAX(COMPLETION\_TIME),'DD-MON/HH24:MI') APPLIED\_TIME**

**FROM V$ARCHIVED\_LOG WHERE DEST\_ID=2 AND APPLIED='YES'**

**);**

**select max(sequence#) as "REDO LOG APPLIED ON STANDBY"**

**from v$archived\_log**

**where applied = 'YES';**

**select thread#,**

**sequence# as "REDO LOG BEING APPLIED",**

**process,**

**status,**

**block#,**

**blocks**

**from v$managed\_standby**

**where sequence# in**

**(select max(sequence#) from v$managed\_standby);**

1. **Get Flashback Log Info on primary**

**sqlplus / as sysdba**

**col name for a60**

**col type for a8**

**col first\_change# for 999999999999999**

**select name,first\_change#,first\_time,type from v$flashback\_database\_logfile;**

Primary: x-12 gb flashback logs were created in support of all the activities in this doc.

NAME FIRST\_CHANGE#

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

FIRST\_TIME TYPE

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

+FLASH\_01/HEPYPRD\_XHEPYDBM21P/FLASHBACK/log\_1.379.1067150301 13584454309519

14-MAR-2021 06:38:34 NORMAL

+FLASH\_01/HEPYPRD\_XHEPYDBM21P/FLASHBACK/log\_2.400.1067150317 13584454566908

14-MAR-2021 07:01:51 NORMAL

+FLASH\_01/HEPYPRD\_XHEPYDBM21P/FLASHBACK/log\_3.606.1067151713 13584455021814

14-MAR-2021 07:20:42 NORMAL

NAME FIRST\_CHANGE#

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

FIRST\_TIME TYPE

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

+FLASH\_01/HEPYPRD\_XHEPYDBM21P/FLASHBACK/log\_4.312.1067152845 13584455579361

14-MAR-2021 07:29:15 NORMAL

+FLASH\_01/HEPYPRD\_XHEPYDBM21P/FLASHBACK/log\_5.429.1067153357 0

RESERVED

1. **Turn off flashback logging**

**Standby Database (if standby exists)**

run => **dgmgrl /**

dgmgrl => **edit database 'HEPYPRD\_xhepydbw21p' set state = 'APPLY-OFF';**

run => **sqlplus / as sysdba**

sql => **alter database flashback off;**

sql = > **alter system set db\_recovery\_file\_dest\_size = 7G scope=both;**

run => **dgmgrl /**

dgmgrl => **edit database 'HEPYPRD\_xhepydbw21p' set state = 'APPLY-ON';**

**Primary Database**

sql => **drop restore point b4\_db\_upgrade;**

sql => **~~alter database flashback off;~~**

sql => **select name, scn, time from v$restore\_point where guarantee\_flashback\_database='YES';**

sql = > **alter system set db\_recovery\_file\_dest\_size = 7G scope=both;**

1. **Resume all applicable OEM and Cron jobs**

**Primary and Standby (if exists)**

Enable (check 3 boxex) Sunday Window Job via OEM Scheduler

Add Index from 19c\_tuning.txt

ALTER USER S032249

ACCOUNT UNLOCK

/

1. **~~Non-prod only - upgrade the database compatibility to 19.0.0~~**

~~Primary and standby (if standby exists)~~

~~Standby (if exists)~~

~~run =>~~ **~~export SRVNAME=xhepydbw21p~~**

~~run =>~~ **~~sqlplus / as sysdba~~**

~~sql =>~~ **~~alter system set compatible='19.0.0' scope=spfile;~~**

~~run =>~~ **~~srvctl stop database -d ${ORACLE\_SID}\_${SRVNAME}~~**

~~run =>~~ **~~srvctl start database -d ${ORACLE\_SID}\_${SRVNAME}~~**

~~Primary~~

~~run =>~~ **~~export SRVNAME=xhepydbm21p~~**

~~run =>~~ **~~sqlplus / as sysdba~~**

~~sql =>~~ **~~alter system set compatible='19.0.0' scope=spfile;~~**

~~run =>~~ **~~srvctl stop database -d ${ORACLE\_SID}\_${SRVNAME}~~**

~~run =>~~ **~~srvctl start database -d ${ORACLE\_SID}\_${SRVNAME}~~**

1. **Take Level 0 backup and adjust level 1 scheduled times**

**Confirm all backup jobs configured with dbsnmp and not SYS user. Will need to recreate if configured with SYS user.**

1. **Prod only - Schedule a change to upgrade the database compatibility to 19.0.0 in 1 week and to gather fixed objects stats**

Standby (if standby exists)

run => **export SRVNAME=xhepydbw21p**

run => **sqlplus / as sysdba**

sql => **alter system set compatible='19.0.0' scope=spfile;**

run => **srvctl stop database -d ${ORACLE\_SID}\_${SRVNAME}**

run => **srvctl start database -d ${ORACLE\_SID}\_${SRVNAME}**

Primary

run => **export SRVNAME=xhepydbm21p**

run => **sqlplus / as sysdba**

sql => **alter system set compatible='19.0.0' scope=spfile;**

run => **srvctl stop database -d ${ORACLE\_SID}\_${SRVNAME}**

run => **srvctl start database -d ${ORACLE\_SID}\_${SRVNAME}**

run => **sqlplus / as sysdba**

sql => **exec dbms\_stats.gather\_fixed\_objects\_stats(no\_invalidate=>false);**

1. **~~Non-prod only - Gather fixed objects stats after business has completed testing.~~**

~~run =>~~ **~~sqlplus / as sysdba~~**

~~sql =>~~ **~~exec dbms\_stats.gather\_fixed\_objects\_stats;~~**

**Rollback Procedures**

**Below are the steps if you are rolling back to the GRP that was set prior to the upgrade process.  
Below requires the GRP to be in place.**

**Do NOT Proceed unless rolling back upgrade**

1. **Comment heartbeat scripts in cron on primary server**

#\*/5 \* \* \* \* /home/oracle/tls/rman/heartbeat.ksh HEPYPRD AEDBA > /dev/null 2>&1

#0 \* \* \* \* /home/oracle/tls/rman/confirm\_heartbeat\_active\_in\_cron.sh HEPYPRD > /dev/null 2>&1

1. **Get current heartbeat timestamp**

run => sqlplus / as sysdba

sql => **select \* from aedba.rman\_heartbeat;**

1. **Disable Broker Configuration in primary (if standby exists)**

run => **dgmgrl /**

dgmgrl => **DISABLE CONFIGURATION;**

dgmgrl => **show configuration**

1. **Stop Broker in Primary (if standby exists)**

sql => **ALTER SYSTEM SET DG\_BROKER\_START=FALSE;**

sql => **show parameter DG\_BROKER\_START**

1. **Stop Broker in Standby (if standby exists)**

sql => **ALTER SYSTEM SET DG\_BROKER\_START=FALSE;**

sql =>  **show parameter DG\_BROKER\_START**

1. **Mount database (Primary database) under 19c**

sql => **shutdown immediate;**

sql => **startup mount;**

1. **Flashback database (Primary database) under 19c**

sql => **flashback database to restore point b4\_db\_upgrade;**

sql => **exit;**

**Shutdown primary database and listener and remove HAS config**

run => **export SRVNAME=xhepydbm21p**

run => **srvctl stop database -d ${ORACLE\_SID}\_${SRVNAME}**

run => **srvctl stop listener -l ${ORACLE\_SID}**

run => **srvctl remove database -d ${ORACLE\_SID}\_${SRVNAME}** run => **srvctl remove listener -l ${ORACLE\_SID}**

1. **Shutdown standby database and listener and remove HAS config (if standby exists)**

run => **export SRVNAME=xhepydbw21p**

run => **srvctl stop database -d ${ORACLE\_SID}\_${SRVNAME}**

run => **srvctl stop listener -l ${ORACLE\_SID}**

run => **srvctl remove database -d ${ORACLE\_SID}\_${SRVNAME}** run => **srvctl remove listener -l ${ORACLE\_SID}**

1. **Set environment to previous Oracle version on primary**

**update oratab to reference previous version**

run => **. oraenv  
 HEPYPRD**

run => **cd $HOME/tls/upg19c**

run => **cp -p spfile${ORACLE\_SID}\_b4\_19cupg.ora $DBS/spfile${ORACLE\_SID}.ora**

1. **Set environment to previous Oracle version on standby (if standby exists)**

**update oratab to reference previous version**

run => **. oraenv  
 HEPYPRD**

1. **Confirm heartbeat in primary**

run => **sqlplus / as sysdba**

sql => **startup mount;**

sql => **alter database open read only;**

sql => **select \* from aedba.rman\_heartbeat;**

1. **Shutdown database (Primary database)**sql => **shutdown immediate;**
2. **Start the Primary Listener**

run => **lsnrctl start HEPYPRD**

1. **Open primary database resetlogs**sql => **startup mount;**

sql => **alter database open resetlogs;**

sql => **col resetlogs\_change# for 99999999999999999999**

sql => **select resetlogs\_change# from v$database;**

**RESETLOGS\_CHANGE#  
 ------------------------------**  
 **xxxxx**

1. **Start the Standby Listener (if standby exists)**

run => **lsnrctl start HEPYPRD**

1. **Mount Standby database (if standby exists)**

run => **startup mount;**

Confirm redo apply if running

sql => **col current\_scn for 99999999999999999999**

sql => **select current\_scn from v$database;**

**CURRENT\_SCN  
 ------------------------------  
 xxxxxx**

\*\*\*\*\*\*\*\*\*\*\* \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**If the standby current\_scn is ahead of the primary resetlogs\_change# then**:

~~Disable Broker Configuration (if enabled)~~ ~~Set Broker to FALSE in primary and standby if set to True:  
 ALTER SYSTEM SET DG\_BROKER\_START=FALSE;~~ **Stop managed recovery manually in standby if running:**  
 alter database recover managed standby database cancel;  
 **shutdown/mount standby database**

shutdown immediately

startup mount   
 flashback standby database to scn xxxxx;

**Confirm current\_scn to flashback scn**  
 **shutdown/mount standby database**

shutdown immediately

startup mount   
 **Start managed recovery manually:**  
 alter database recover managed standby database disconnect from session;

**Perform a log switch on primary**

**alter system switch logfile;**  
 **Confirm incarnation on primary and standby:**  
 select incarnation# from V$DATABASE\_INCARNATION where status = 'CURRENT';

**Stop managed recovery manually in standby:**  
 alter database recover managed standby database cancel;  
 **shutdown/mount standby database**

shutdown immediately

startup mount   
 **Set Broker to TRUE in primary and standby:**

ALTER SYSTEM SET DG\_BROKER\_START=TRUE;

**Enable Broker Configuration**

run => **dgmgrl /**

dgmgrl => **ENABLE CONFIGURATION;**

**Perform a log switch on primary**

**alter system switch logfile;**

**Confirm all is well with Data Guard environment  
 \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

1. **Start the Broker in the primary database (Assuming no issues with original flashback)**

run => **ALTER SYSTEM SET DG\_BROKER\_START=TRUE;**

1. **Start the Broker in the standby database (Assuming no issues with original flashback)**

**(if standby exists)**

run => **ALTER SYSTEM SET DG\_BROKER\_START=TRUE;**

1. **Enable Broker Configuration (Assuming no issues with original flashback)**

**(if standby exists)**

run => **dgmgrl /**

dgmgrl => **ENABLE CONFIGURATION;**

1. **Start redo apply in standby if disabled (if standby exists)**

run => **dgmgrl /**

dgmgrl => **edit database 'HEPYPRD\_xhepydbw21p' set state = 'APPLY-ON';**

1. **Perform a log switch in Primary**

sql => **alter system switch logfile;**

1. **Add dropped Parameter to Primary and Standby (if standby exists)**

sql => **alter system set "\_sqlmon\_max\_planlines"=1000 scope=spfile;**

**alter system set "\_bct\_public\_dba\_buffer\_size"=100000000 scope=spfile;**

**alter system set "\_bct\_buffer\_allocation\_max"=1048576000 scope=spfile;**

**alter system set "\_bct\_public\_dba\_buffer\_maxsize"=110000000 scope=spfile;**

1. **Bounce Primary database**

sql => **shutdown immediate**

sql => **startup**

1. **Bounce Standby database (if standby exists)**

sql => **shutdown immediate**

sql => **startup mount;**

1. **Bounce Agent on Standby Server (if standby exists)**

run => . oraenv  
 GCAGENT

run => **emctl stop agent;**

run => **emctl start agent;**

1. **Add primary database and listener HAS configuration and start**

HEPYPRD PORT = 1574

sql => **shutdown immediate;**

run => **lsnrctl stop HEPYPRD**

run => **srvctl add listener -l ${ORACLE\_SID} -oraclehome ${ORACLE\_HOME} -endpoints "TCP:1574/IPC:${ORACLE\_SID}\_IPC"**

run => **srvctl add database -db ${ORACLE\_SID}\_${SRVNAME} -oraclehome ${ORACLE\_HOME} -spfile ${ORACLE\_HOME}/dbs/spfile${ORACLE\_SID}.ora -instance ${ORACLE\_SID} -diskgroup "DATA\_01,IND\_01,REDOA\_01,REDOB\_01"**

run => **srvctl start database -d ${ORACLE\_SID}\_${SRVNAME}**

**Note: starting the database will start the listener as well.**

run => **srvctl setenv  listener -l ${ORACLE\_SID} -envs "TNS\_ADMIN=$ORACLE\_HOME/network/admin"**

1. **Add standby database and listener HAS configuration and mount (if standby exists)**

sql => **shutdown immediate;**

run => **lsnrctl stop HEPYPRD**

run => **srvctl add listener -l ${ORACLE\_SID} -oraclehome ${ORACLE\_HOME} -endpoints "TCP:1574/IPC:${ORACLE\_SID}\_IPC"**

Config For Non Active Data Guard

run => **srvctl add database -db ${ORACLE\_SID}\_${SRVNAME} -oraclehome ${ORACLE\_HOME} -spfile ${ORACLE\_HOME}/dbs/spfile${ORACLE\_SID}.ora  
-startoption "MOUNT" -role physical\_standby -instance ${ORACLE\_SID} -diskgroup "DATA\_01,IND\_01,REDOA\_01,REDOB\_01"**

Config For Active Data Guard

~~run =>~~ **~~srvctl add database -db ${ORACLE\_SID}\_${SRVNAME} -oraclehome ${ORACLE\_HOME} -spfile ${ORACLE\_HOME}/dbs/spfile${ORACLE\_SID}.ora  
-startoption "READ ONLY" -role physical\_standby -instance ${ORACLE\_SID} -diskgroup "DATA\_01,REDOA\_01,REDOB\_01"~~**

run => **srvctl start database -d ${ORACLE\_SID}\_${SRVNAME}**

**Note: starting the database will start the listener as well.**

run => **srvctl setenv  listener -l ${ORACLE\_SID} -envs "TNS\_ADMIN=$ORACLE\_HOME/network/admin"**

1. **Replace $ORACLE\_HOME Symbolic Link in Primary and Standby (if standby exists)**

run => **cd $ORACLE\_BASE/admin/${ORACLE\_SID}**

run => **ls -l oracle\_home**

run => **rm oracle\_home**

run => **ln -s $ORACLE\_HOME oracle\_home**

run => **ls -l oracle\_home**

1. **Resync RMAN Catalog from primary server**

run => **rmanc**

rman => **resync catalog;**

1. **Disable Flashback Logging and Resize FRA**

**Standby Database (if standby exists)**

run => **dgmgrl /**

dgmgrl => **edit database 'HEPYPRD\_xhepydbw21p' set state = 'APPLY-OFF';**

run => **sqlplus / as sysdba**

sql => **alter database flashback off;**

sql = > **alter system set db\_recovery\_file\_dest\_size = 7G scope=both;**

run => **dgmgrl /**

dgmgrl => **edit database 'HEPYPRD\_xhepydbw21p' set state = 'APPLY-ON';**

**Primary Database**

run => **sqlplus / as sysdba**

sql => **drop restore point b4\_db\_upgrade;**

sql => **select name, scn, time from v$restore\_point where guarantee\_flashback\_database='YES';**

sql = > **alter system set db\_recovery\_file\_dest\_size = 7G scope=both;**

1. **Uncomment heartbeat scripts in cron**

\*/5 \* \* \* \* /home/oracle/tls/rman/heartbeat.ksh HEPYPRD AEDBA > /dev/null 2>&1

0 \* \* \* \* /home/oracle/tls/rman/confirm\_heartbeat\_active\_in\_cron.sh HEPYPRD > /dev/null 2>&1

1. **Update OEM monitoring configuration for databases and listeners $ORACLE\_HOME**
2. **Primary and standby (if applicable)  
     
   Resume all applicable OEM and Cron jobs**