­­

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=xhepydbw21s**  
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**
2. **Gather Dictionary Stats, Purge Recyclebin, Confirm no MV’s being refreshed**

run => **. oraenv**

**HEPYSTS**

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;**

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/HEPYSTS\_xhepydbw21s/preupgrade

Report generated by Oracle Database Pre-Upgrade Information Tool Version

19.0.0.0.0 Build: 1 on 2021-02-23T07:24:19

Upgrade-To version: 19.0.0.0.0

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

Status of the database prior to upgrade

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

Database Name: HEPYSTS

Container Name: HEPYSTS

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;

20 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

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 3810 MB 4143 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.

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

AFTER 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 HEPYSTS

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

executing the following

SQL>@/orahome/u01/app/oracle/cfgtoollogs/HEPYSTS\_xhepydbw21s/preupgrade/post

upgrade\_fixups.sql

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

EUPGRADE SUMMARY

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

/orahome/u01/app/oracle/cfgtoollogs/HEPYSTS\_xhepydbw21s/preupgrade/preupgrade.log

/orahome/u01/app/oracle/cfgtoollogs/HEPYSTS\_xhepydbw21s/preupgrade/preupgrade\_fixups.sql

/orahome/u01/app/oracle/cfgtoollogs/HEPYSTS\_xhepydbw21s/preupgrade/postupgrade\_fixups.sql

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 HEPYSTS

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

executing the following

SQL>@/orahome/u01/app/oracle/cfgtoollogs/HEPYSTS\_xhepydbw21s/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**

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

show parameter \_bct\_public\_dba\_buffer\_size

100000000

show parameter \_bct\_buffer\_allocation\_max

1048576000

show parameter \_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;

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: 2021-02-25 10:42:06

For Source Database: HEPYSTS

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 => **@check\_invalids**

SQL>

SQL>

SQL> select

2 owner,

3 substr(object\_name,1,32) "Name",

4 substr(object\_type,1,32) "Type" ,

5 substr(to\_char(last\_ddl\_time, 'DD-MON-YY HH24:MI:SS'),1,25) "Last DDL Time",

6 status

7 from

8 dba\_objects

9 where

10 status <> 'VALID'

11 order by

12 4 desc

13 ;

no rows selected

run => **sqlplus / as sysdba**

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

sql => **@check\_invalids**

SQL>  
SQL> select  
 2 owner,  
 3 substr(object\_name,1,32) "Name",  
 4 substr(object\_type,1,32) "Type" ,  
 5 substr(to\_char(last\_ddl\_time, 'DD-MON-YY HH24:MI:SS'),1,25) "Last DDL Time",  
 6 status  
 7 from  
 8 dba\_objects  
 9 where  
 10 status <> 'VALID'  
 11 order by  
 12 4 desc  
 13 ;

no rows selected

SQL> spool off;

SQL> exit;

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> 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:D216BD1BE9C8187C

66D822B415C81B0A3EA8CCDAFB650FED84604EF2B20A498C0C7A943AF46F4EDD0368ADFEA689586248997C4737B11A6A1F682430DF6F7771065A190AACEA5

6C69CBE084454EC87C9;BBEE1F9B3A94B3CF';

User altered.

SQL> alter user tom identified by VALUES 'S:8EC3851304545F0EE2B4098FE13EF981E7A8D3BB9A50EA2CCEF8790E24C9;T:A7C769D7B2DBE4141F

5096BE1235DFF2C37EDD8AEBB3B3477524496B86D9323936344DE9F4DD8337ECD27939C085FE99CD9A38F5B74F1A6D2E28B62874DF129D5C5109EE51F6D22

51A58BDA9CB853E8F;3141BFFD0D91699D';

User altered.

SQL> alter user AEAUDIT identified by VALUES 'S:39AE6CE7DE6B273DADE99C7F3961D043982B2B512D3F2897215616761A9C;T:2769E98E82E537

D2C6747F5F1F75B39662EF642A0627C2BCBE20FD1D81EB6C54DDF44460A5A0F954A64BDDD7E6F479EA34FA7799A0135B1301F82E72985795A3D96065A349C

15AAF83BD56DE79CCDDC3;D3CF3912523E800E';

User altered.

SQL> alter user orativoli identified by VALUES 'S:472152CDDB4D5CEF042D73FDFC1147771218C86DD8C4945181E02A3A57C7;T:83E9538D576A

F11E266CF2AE62FC668AACBDB7AB86076624C6A210076A77F540B5144F9EAB107BD5396C78B7F686B4FD85D1AFA4551158BBC32C9CA1821E942EC460ABF98

0D18871C44CF91C137A1A83;28CB29E5CDCC920C';

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 HEPYDBA**

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

SYS Password = lfAI?0FT (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)**

#\*/4 \* \* \* \* /home/oracle/tls/rman/heartbeat.ksh HEPYSTS AEDBA > /dev/null 2>&1

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

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

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

**Current TimeStamp**

**02-25-2021 11:01:38 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 HEPYSTS (Otherwise would need to create)**

**Primary Database**

run => **sqlplus / as sysdba**

**show parameter db\_recovery\_file\_dest**

**7000M**

sql => **alter system set db\_recovery\_file\_dest\_size = 100G scope=both;**

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

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

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

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

**Standby Database (if exists)**

run => **dgmgrl** dgmgrl => **connect /** dgmgrl => **edit database 'HEPYDBA\_xsrxdbw1q' set state = 'APPLY-OFF';** dgmgrl => **show database verbose 'HEPYDBA\_xsrxdbw1q';**

run => **sqlplus / as sysdba**

sql => **alter system set db\_recovery\_file\_dest\_size = 15G scope=both;**

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

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

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

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

dgmgrl => **connect /** dgmgrl => **edit database 'HEPYDBA\_xsrxdbw1q' set state = 'APPLY-ON';** dgmgrl => **show database verbose 'HEPYDBA\_xsrxdbw1q';**

**Primary Database**

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 13576958555233 25-FEB-21 11.03.35.000000000 AM

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

\*/4 \* \* \* \* /home/oracle/tls/rman/heartbeat.ksh HEPYSTS AEDBA > /dev/null 2>&1

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

1. **Upgrade apex**

(Note: Below created 4-250 mb flashback logs)

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

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

**…  
 …**

PL/SQL procedure successfully completed

Timing for: Complete Installation

Elapsed: 00:05:46.82  
  
 SYS>

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

2

1. **Drop old APEX schema**

Note: Below created 1-200 mb flashback log

run => **sqlplus / as sysdba**

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

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 from primary database**

run => **dgmgrl**

dgmgrl => **connect /**

dgmgrl => **disable configuration;**

dgmgrl => **quit**

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  
HEPYDBA**

run => **export UPGVER=19.9.0**

run => **export SRVNAME=xsrxdbw1q**

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

**Log: /orahome/u01/app/oracle/local/logs/upgrade\_standby\_db\_HEPYDBA\_20210223\_1614086055.out**

Start standby DB upgrade of HEPYDBA on Linux xsrxdbw1q.aetna.com 3.10.0-1160.11.1.el7.x86\_64 #1 SMP Mon Nov 30 13:05:31 EST

2020 at Tue Feb 23 08:14:15 EST 2021 using /orahome/u01/app/oracle/local/scripts/upgrade\_standby\_db.sh

Review log file /orahome/u01/app/oracle/local/logs/upgrade\_standby\_db\_HEPYDBA\_20210223\_1614086055.out for details

oracle 12286 1 0 Feb22 ? 00:00:01 ora\_pmon\_HEPYDBA

Oracle DBMS software version 19.9.0 is installed, continuing with upgrade

Database exists in ORATAB, continuing with upgrade apply

DBS files for HEPYDBA have been copied to /orahome/u01/app/oracle/product/19.9.0/dbs

Updating srvctl configuration..........

Configuratioin update complete

Setting start option back to READ ONLY.....

Start option has been set.

Switching ORACLE\_HOME link

Oracle Home symbolic link has been created as the following:

lrwxrwxrwx 1 oracle dba 43 Feb 23 08:15 /orahome/u01/app/oracle/admin/HEPYDBA/oracle\_home -> /orahome/u01/app/oracle/product/

19.9.0/db\_1

Starting OEM update.........

emcli exists and is executable, updating OEM

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 hepydba\_xsrxdbw1q

LISTENER\_NAME hepydba

BDUMP /orahome/u01/app/oracle/diag/rdbms/hepydba\_xsrxdbw1q/HEPYDBA/trace

CDUMP /orahome/u01/app/oracle/diag/rdbms/hepydba\_xsrxdbw1q/HEPYDBA/cdump

UDUMP /orahome/u01/app/oracle/diag/rdbms/hepydba\_xsrxdbw1q/HEPYDBA/trace

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

LSNRLOG /orahome/u01/app/oracle/diag/tnslsnr/xsrxdbw1q/hepydba/trace

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

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

Couldn't find matching DB System

OEM system details could not be retrirved, assume the standby OEM target name is the standby database unique name

Oracle Database Home prior to update: /orahome/u01/app/oracle/product/19.9.0/db\_1

Oracle Listener Home prior to update: /orahome/u01/app/oracle/product/19.9.0/db\_1

OEM Database Target has been updated with the ORACLE\_HOME value of /orahome/u01/app/oracle/product/19.9.0/db\_1 for database H

EPYDBA\_xsrxdbw1q

OEM Listener Target has been updated with the ORACLE\_HOME value of /orahome/u01/app/oracle/product/19.9.0/db\_1 for listener

HEPYDBA\_xsrxdbw1q.aetna.com

Oracle Database Home after update: /orahome/u01/app/oracle/product/19.9.0/db\_1

Oracle Listener Home after update: /orahome/u01/app/oracle/product/19.9.0/db\_1

Upgrade of HEPYDBA to 19.9.0 complete on Linux xsrxdbw1q.aetna.com 3.10.0-1160.11.1.el7.x86\_64 #1 SMP Mon Nov 30 13:05:31

EST 2020 at Tue Feb 23 08:15:22 EST 2021 using /orahome/u01/app/oracle/local/scripts/upgrade\_standby\_db.sh

Logoff standby database server

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

run => **.oraenv**  
 **HEPYDBA**

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 HEPYSTS\_xhepydbw21s? (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  
 HEPYSTS**

ORACLE\_SID = [HEPYDBA] ? HEPYDBA

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 hepydba\_xhepydbw21d

LISTENER\_NAME hepydba

BDUMP /orahome/u01/app/oracle/diag/rdbms/hepydba\_xhepydbw21d/HEPYDBA/trace

CDUMP /orahome/u01/app/oracle/diag/rdbms/hepydba\_xhepydbw21d/HEPYDBA/cdump

UDUMP /orahome/u01/app/oracle/diag/rdbms/hepydba\_xhepydbw21d/HEPYDBA/trace

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

LSNRLOG /orahome/u01/app/oracle/diag/tnslsnr/xhepydbw21d/hepydba/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 => **cd $HOME/tls/upg19c**

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

Optional: Open new ssh connection

**tail -f nohup.out**

Optional: Open new ssh connection

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

**Note: generated xx-250 mb flashback logs**

Note: Running with default parallel of 4

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

Phases [0-107] End Time:[2021\_02\_05 10:55:16]

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

Grand Total Time: 1287s

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:21m:27s]

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

HEPYSTS PORT = 1572

run => **srvctl add listener -l ${ORACLE\_SID} -oraclehome ${ORACLE\_HOME} -endpoints "TCP:1572/IPC:${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 HEPYSTS -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}**

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

run => **. oraenv  
 HEPYSTS**

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

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

lrwxrwxrwx 1 oracle dba 52 Feb 22 18:59 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 23 11:28 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: 2021-02-23 07:24:19

For Source Database: HEPYDBA

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.

SQL> quit

Disconnected from Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production

Version 19.9.0.0.0

xhepydbw21d.aetna.com (oracle) HEPYDBA::/home/oracle/tls/upg19c

1. **Upgrade RMAN catalog**

run => **rmanc**

Recovery Manager: Release 19.0.0.0.0 - Production on Wed Jan 27 09:01:15 2021

Version 19.9.0.0.0

Copyright (c) 1982, 2019, Oracle and/or its affiliates. All rights reserved.

connected to target database: HEPYDBA (DBID=1825233320)

connected to recovery catalog database

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

rman => **upgrade catalog;**

rman => **upgrade catalog;**

recovery catalog owner is HEPYDBA

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**

Session altered.

.

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

15/09/2020 - SYS.AQ$\_ALERT\_QT\_S.CREATION\_TIME - 14

15/09/2020 - SYS.AQ$\_ALERT\_QT\_S.DELETION\_TIME - 14

15/09/2020 - SYS.AQ$\_ALERT\_QT\_S.MODIFICATION\_TIME - 14

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

05/02/2021 - SYS.AQ$\_ORA$PREPLUGIN\_BACKUP\_QTB\_S.CREATION\_TIME - 1

05/02/2021 - SYS.AQ$\_ORA$PREPLUGIN\_BACKUP\_QTB\_S.DELETION\_TIME - 1

05/02/2021 - SYS.AQ$\_ORA$PREPLUGIN\_BACKUP\_QTB\_S.MODIFICATION\_TIME - 1

05/02/2021 - SYS.AQ$\_PDB\_MON\_EVENT\_QTABLE$\_S.CREATION\_TIME - 1

05/02/2021 - SYS.AQ$\_PDB\_MON\_EVENT\_QTABLE$\_S.DELETION\_TIME - 1

05/02/2021 - 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

05/02/2021 - SYS.AQ$\_SUBSCRIBER\_TABLE.CREATION\_TIME - 1

05/02/2021 - SYS.AQ$\_SUBSCRIBER\_TABLE.DELETION\_TIME - 1

05/02/2021 - SYS.AQ$\_SUBSCRIBER\_TABLE.MODIFICATION\_TIME - 1

05/02/2021 - SYS.AQ$\_SYS$SERVICE\_METRICS\_TAB\_S.CREATION\_TIME - 4

05/02/2021 - SYS.AQ$\_SYS$SERVICE\_METRICS\_TAB\_S.DELETION\_TIME - 4

05/02/2021 - SYS.AQ$\_SYS$SERVICE\_METRICS\_TAB\_S.MODIFICATION\_TIME - 4

05/02/2021 - SYS.ATSK$\_SCHEDULE\_CONTROL.MRCT\_TASK\_TIME\_TZ - 2

05/02/2021 - SYS.KET$\_AUTOTASK\_STATUS.ABA\_START\_TIME - 1

05/02/2021 - SYS.KET$\_AUTOTASK\_STATUS.ABA\_STATE\_TIME - 1

05/02/2021 - SYS.KET$\_AUTOTASK\_STATUS.MW\_RECORD\_TIME - 1

05/02/2021 - SYS.KET$\_AUTOTASK\_STATUS.MW\_START\_TIME - 1

05/02/2021 - SYS.KET$\_AUTOTASK\_STATUS.RECONCILE\_TIME - 1

05/02/2021 - SYS.KET$\_CLIENT\_CONFIG.FIELD\_2 - 7

05/02/2021 - SYS.KET$\_CLIENT\_CONFIG.LAST\_CHANGE - 7

05/02/2021 - SYS.KET$\_CLIENT\_TASKS.CURR\_WIN\_START - 3

05/02/2021 - SYS.KET$\_CLIENT\_TASKS.LG\_DATE - 3

05/02/2021 - SYS.KET$\_CLIENT\_TASKS.LT\_DATE - 3

05/02/2021 - SYS.OPTSTAT\_HIST\_CONTROL$.SPARE6 - 45

05/02/2021 - SYS.OPTSTAT\_HIST\_CONTROL$.SVAL2 - 45

05/02/2021 - SYS.OPTSTAT\_SNAPSHOT$.TIMESTAMP - 119

05/02/2021 - SYS.OPTSTAT\_USER\_PREFS$.CHGTIME - 72

11/03/2015 - SYS.RADM\_FPTM$.TSWTZ\_COL - 1

05/02/2021 - SYS.REG$.NTFN\_GROUPING\_START\_TIME - 3

05/02/2021 - SYS.REG$.REG\_TIME - 3

04/02/2021 - SYS.SCHEDULER$\_EVENT\_LOG.LOG\_DATE - 67902

05/02/2021 - SYS.SCHEDULER$\_GLOBAL\_ATTRIBUTE.ATTR\_TSTAMP - 11

05/02/2021 - SYS.SCHEDULER$\_JOB.END\_DATE - 72

05/02/2021 - SYS.SCHEDULER$\_JOB.LAST\_ENABLED\_TIME - 72

05/02/2021 - SYS.SCHEDULER$\_JOB.LAST\_END\_DATE - 72

05/02/2021 - SYS.SCHEDULER$\_JOB.LAST\_START\_DATE - 72

05/02/2021 - SYS.SCHEDULER$\_JOB.NEXT\_RUN\_DATE - 72

05/02/2021 - SYS.SCHEDULER$\_JOB.START\_DATE - 72

03/02/2021 - SYS.SCHEDULER$\_JOB\_RUN\_DETAILS.LOG\_DATE - 21887

03/02/2021 - SYS.SCHEDULER$\_JOB\_RUN\_DETAILS.REQ\_START\_DATE - 21887

03/02/2021 - SYS.SCHEDULER$\_JOB\_RUN\_DETAILS.START\_DATE - 21887

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

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

05/02/2021 - SYS.SCHEDULER$\_WINDOW.ACTUAL\_START\_DATE - 9

05/02/2021 - SYS.SCHEDULER$\_WINDOW.END\_DATE - 9

05/02/2021 - SYS.SCHEDULER$\_WINDOW.LAST\_START\_DATE - 9

05/02/2021 - SYS.SCHEDULER$\_WINDOW.MANUAL\_OPEN\_TIME - 9

05/02/2021 - SYS.SCHEDULER$\_WINDOW.NEXT\_START\_DATE - 9

05/02/2021 - SYS.SCHEDULER$\_WINDOW.START\_DATE - 9

04/02/2021 - SYS.SCHEDULER$\_WINDOW\_DETAILS.LOG\_DATE - 30

04/02/2021 - SYS.SCHEDULER$\_WINDOW\_DETAILS.REQ\_START\_DATE - 30

04/02/2021 - SYS.SCHEDULER$\_WINDOW\_DETAILS.START\_DATE - 30

05/02/2021 - SYS.STATS\_TARGET$.END\_TIME - 3645

05/02/2021 - SYS.STATS\_TARGET$.START\_TIME - 3645

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

05/02/2021 - SYS.WRI$\_ALERT\_HISTORY.CREATION\_TIME - 32

05/02/2021 - SYS.WRI$\_ALERT\_HISTORY.TIME\_SUGGESTED - 32

05/02/2021 - SYS.WRI$\_OPTSTAT\_HISTGRM\_HISTORY.SAVTIME - 1689506

05/02/2021 - SYS.WRI$\_OPTSTAT\_HISTGRM\_HISTORY.SPARE6 - 1689506

05/02/2021 - SYS.WRI$\_OPTSTAT\_HISTHEAD\_HISTORY.SAVTIME - 115875

05/02/2021 - SYS.WRI$\_OPTSTAT\_HISTHEAD\_HISTORY.SPARE6 - 115875

05/02/2021 - SYS.WRI$\_OPTSTAT\_IND\_HISTORY.SAVTIME - 9022

05/02/2021 - SYS.WRI$\_OPTSTAT\_IND\_HISTORY.SPARE6 - 9022

05/02/2021 - SYS.WRI$\_OPTSTAT\_OPR.END\_TIME - 532

05/02/2021 - SYS.WRI$\_OPTSTAT\_OPR.SPARE6 - 532

05/02/2021 - SYS.WRI$\_OPTSTAT\_OPR.START\_TIME - 532

05/02/2021 - SYS.WRI$\_OPTSTAT\_OPR\_TASKS.END\_TIME - 22970

05/02/2021 - SYS.WRI$\_OPTSTAT\_OPR\_TASKS.SPARE6 - 22970

05/02/2021 - SYS.WRI$\_OPTSTAT\_OPR\_TASKS.START\_TIME - 22970

05/02/2021 - SYS.WRI$\_OPTSTAT\_TAB\_HISTORY.SAVTIME - 6595

05/02/2021 - SYS.WRI$\_OPTSTAT\_TAB\_HISTORY.SPARE6 - 6595

05/02/2021 - SYS.WRM$\_DATABASE\_INSTANCE.STARTUP\_TIME\_TZ - 7

03/02/2021 - SYS.WRM$\_SNAPSHOT.BEGIN\_INTERVAL\_TIME\_TZ - 5965

03/02/2021 - SYS.WRM$\_SNAPSHOT.END\_INTERVAL\_TIME\_TZ - 5965

10/09/2019 - SYS.XS$PRIN.END\_DATE - 14

10/09/2019 - SYS.XS$PRIN.START\_DATE - 14

Total numrows of SYS TSTZ columns is : 3867366

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

05/02/2021 - APEX\_180200.WWV\_QS\_RANDOM\_NAMES.TSWTZ - 2001

29/01/2021 - 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.

Session altered.

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 8589930600 bytes

Fixed Size 8917096 bytes

Variable Size 2298478592 bytes

Database Buffers 6274678784 bytes

Redo Buffers 7856128 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 8589930600 bytes

Fixed Size 8917096 bytes

Variable Size 2298478592 bytes

Database Buffers 6274678784 bytes

Redo Buffers 7856128 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\_FEEDBACK"

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

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

Number of failures: 0

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

Number of failures: 0

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

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

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**

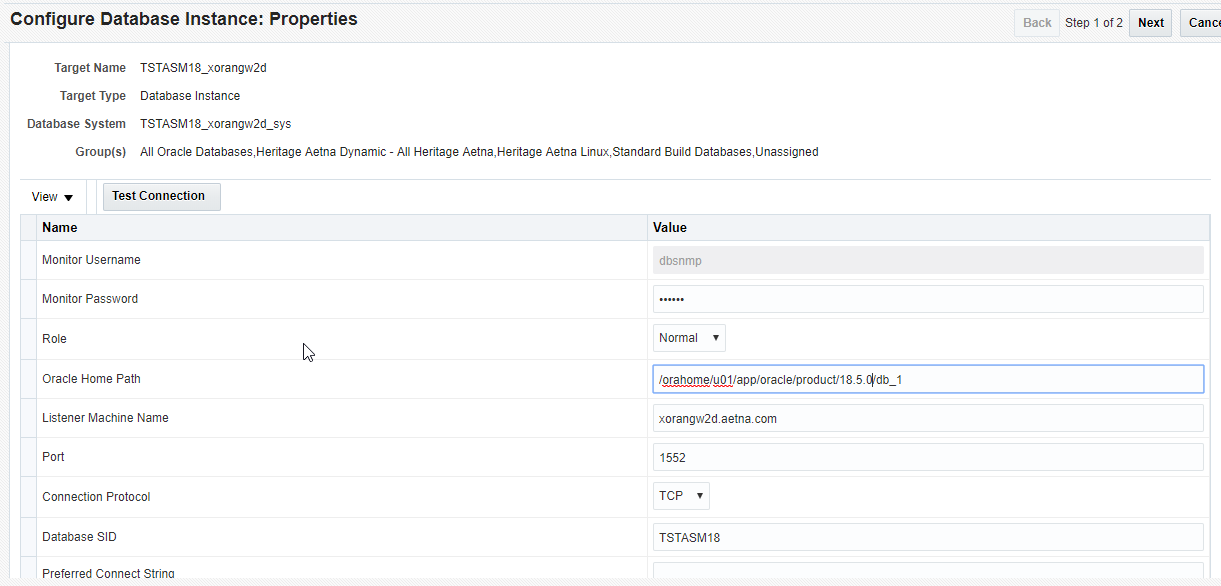
sql => **@?/rdbms/admin/utlrp**

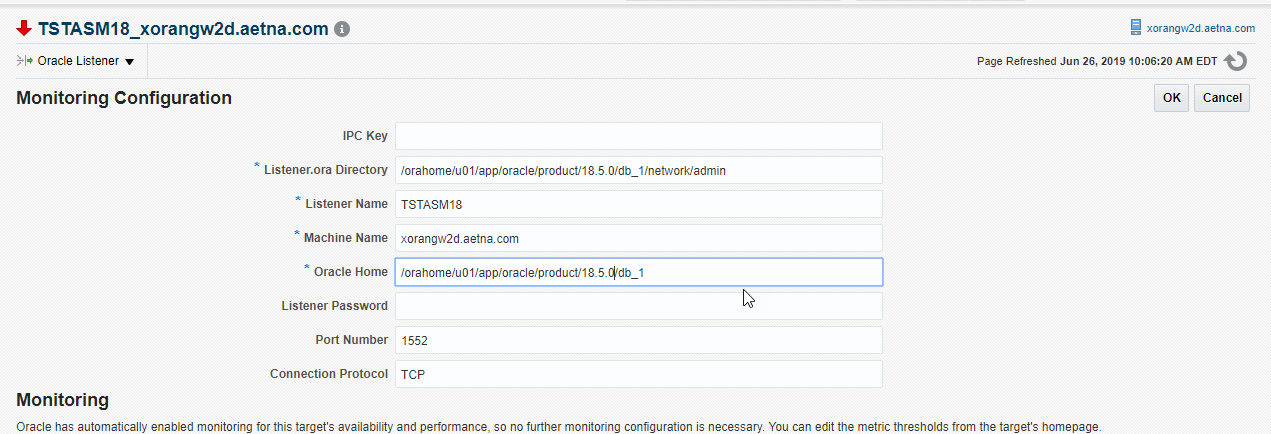
sql => **@check\_invalids**

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

**on Primary**

**e.g.**





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

**Optional bounce OEM agent on Primary server as well**

**. 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  
 HEPYDBA**

**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 from primary database**

run => **sqlplus / as sysdba**

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

sql => **quit**

run => **dgmgrl**

dgmgrl =>  **connect /**

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=xsrxdbw1q**

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;**

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**

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,bytes from v$flashback\_database\_logfile;**

Primary: 20-250 mb flashback logs were created in support of all the activities in this doc.  
**2 – 6GB logs were created for HEPYSTS**

1. **Turn off flashback logging**

**Standby Database (if standby exists)**

run => **dgmgrl**

dgmgrl => **connect /**

dgmgrl => **edit database 'HEPYDBA\_xsrxdbw1q' 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 => **connect /**

dgmgrl => **edit database 'HEPYDBA\_xsrxdbw1q' 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)**

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

Primary and standby (if standby exists)

Standby (if exists)

run => **export SRVNAME=xsrxdbw1q**

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=xhepydbw21s**

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=xhedwdbw21p**

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=xhedwdbm21p**

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 HEPYSTS AEDBA > /dev/null 2>&1

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

1. **Get current heartbeat timestamp**

run => sqlplus / as sysdba

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

1. **Disable Log Shipping in primary (if standby exists)**

run => **$SCRIPTS/DISable\_log\_shipping.sh HEPYSTS**

1. **Stop Redo Apply in Standby (if standby exists)**

run => **dgmgrl**

dgmgrl => **connect /**

dgmgrl => **edit database 'HEPYDBA\_xsrxdbw1q' set state = 'APPLY-OFF';**

dgmgrl => **show database ''HEPYDBA\_xsrxdbw1q'**

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

run => **dgmgrl**

dgmgrl => **connect /**

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

run => **export SRVNAME=xsrxdbw1q**

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 primary database and listener and remove HAS config**

run => **export SRVNAME=xhepydbw21s**

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

sql => **startup mount;**

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

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

1. **Shutdown database (Primary database)**

sql => **shutdown immediate;**

1. **Start the Standby Listener under 19c (if standby exists)**

run =>  **lsnrctl start HEPYDBA**

1. **Flashback database (Standby database) under 19c (if standby exists)**

sql => **startup mount;**

sql => **flashback standby database to scn 13584082401187;**

sql => **shutdown immediate**

1. **Stop the Standby Listener under 19c (if standby exists)**

run =>  **lsnrctl stop HEPYDBA**

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

**update oratab to reference previous version**

run => **. oraenv  
 HEPYDBA**

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  
 HEPYSTS**

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. **Confirm heartbeat in standby (if standby exists)**

run => **sqlplus / as sysdba**

sql => **startup mount;**

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

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

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

run => **lsnrctl start HEPYSTS**

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#  
 ------------------------------

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

run => **lsnrctl start HEPYDBA**

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

run => **startup mount;**

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

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

CURRENT\_SCN  
 ------------------------------

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  
 flashback standby database again  
 Confirm current\_scn to flashback scn  
 shutdown/mount standby database  
 Start managed recovery manually:  
 alter database recover managed standby database disconnect from session;  
 Perform a log switch on primary  
 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  
 Set Broker to TRUE in primary and standby:  
 ALTER SYSTEM SET DG\_BROKER\_START=FALSE;  
 Enable Broker Configuration  
 Perform a log switch on primary

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)**

dgmgrl => **connect /**

dgmgrl => **ENABLE CONFIGURATION;**

1. **Resume log shipping on primary (if standby exists)**

run => **$SCRIPTS/ENable\_log\_shipping.sh HEPYDBA**

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

dgmgrl => **connect /**

dgmgrl => **edit database 'HEPYDBA\_xsrxdbw1q' 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;**

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**

HEPYDBA PORT = 1655

sql => **shutdown immediate;**

run => **lsnrctl stop HEPYDBA**

run => **srvctl add listener -l ${ORACLE\_SID} -oraclehome ${ORACLE\_HOME} -endpoints "TCP:<port>/IPC:${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,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 HEPYDBA**

run => **srvctl add listener -l ${ORACLE\_SID} -oraclehome ${ORACLE\_HOME} -endpoints "TCP:<port>/IPC:${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,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, Drop Restore Point in Primary, and Resize FRA**

**Standby Database (if standby exists)**

run => **dgmgrl**

dgmgrl => **connect /**

dgmgrl => **edit database 'HEPYDBA\_xsrxdbw1q' 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 => **connect /**

dgmgrl => **edit database 'HEPYDBA\_xsrxdbw1q' set state = 'APPLY-ON';**

**Primary Database**

run => **sqlplus / as sysdba**

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. **Uncomment heartbeat scripts in cron**

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

0 \* \* \* \* /home/oracle/tls/rman/confirm\_heartbeat\_active\_in\_cron.sh HEDWDBA > /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**