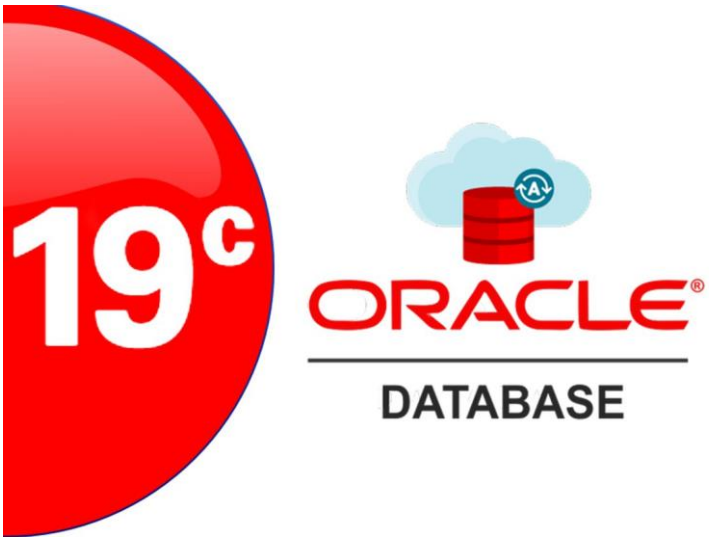
Upgrade to Oracle Database 19c Handson



Chaoyang Han

目录

准备工作

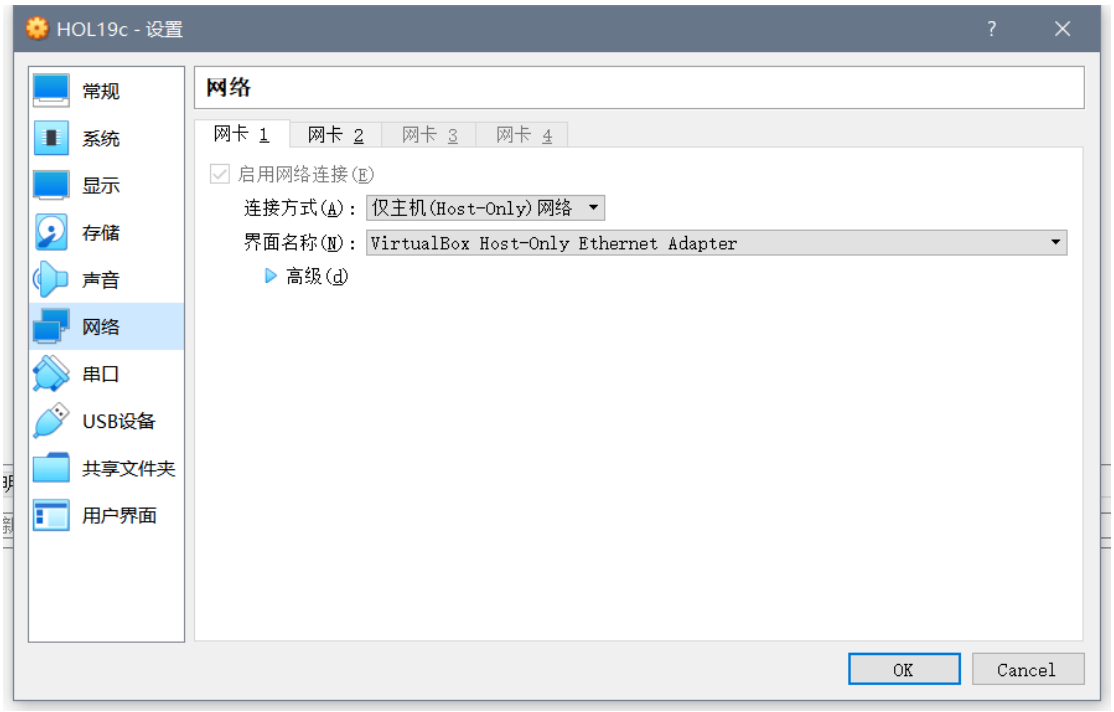
虚拟机环境准备

为顺利升级到 19c，Oracle 推出了一套完整的 Oracle Database 19c (19.3.0) Upgrade and Migration hands-on Lab 学习教程，用户或合作伙伴可。通过这里 https://www.oracle.com/downloads/community/vts-hands-on-labs-downloads.html 阅读试验教程并下载虚拟机。受环境所限，本次大约需要 8 小时。原文介绍来自Mike Dietrich's Blog

- 环境硬件最低要求
  CPU: Dual-core CPU
  Memory: 虚拟机本身需要 8+ GB RAM
  Disk space: 25 GB for download of zip files
  75GB for expanded VM image
- 环境软件要求
  任何能运行 Oracle VirtualBox 的 64-bit 操作系统
  VM VirtualBox 版本要>5.2，建议安装最新的，可以通过 http://www.virtualbox.org 来下载。同时安装 VirtualBox 扩展包。
- 环境用户密码
  Lab 中所有密码都是 oracle，包括操作系统用户 oracle,root，以及数据库用户 sys/system。

调整网卡信息

为了更好的体验，我们可以配置下网络，通过 putty 或 secureCRT 来访问。



```
[root@hol ~]# {
> echo DEVICE=\"enp0s3\"
> echo BOOTPROTO=\"DHCP\"
> echo ONBOOT=\"yes\"
> echo TYPE=\"Ethernet\"
> echo USERCTL=\"no\"
> echo peerdns=\"yes\"
> } > /etc/sysconfig/network-scripts/ifcfg-enp0s3
[root@hol ~]#
[root@hol ~]# cat /etc/sysconfig/network-scripts/ifcfg-enp0s3
DEVICE="enp0s3"
BOOTPROTO="DHCP"
ONBOOT="yes"
TYPE="Ethernet"
USERCTL="no"
peerdns="yes"
[root@hol ~]#
[root@hol ~]# ifdown enp0s3
Device 'enp0s3' successfully disconnected.
[root@hol ~]# ifup enp0s3
Connection successfully activated (D-Bus active path:
/org/freedesktop/NetworkManager/ActiveConnection/5)
[root@hol ~]# ifconfig enp0s3
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.56.101  netmask 255.255.255.0  broadcast 192.168.10.255
        inet6 fe80::a00:27ff:fe88:a0d2  prefixlen 64  scopeid 0x20<link>
        ether 08:00:27:88:a0:d2  txqueuelen 1000  (Ethernet)
        RX packets 33  bytes 8237 (8.0 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 110  bytes 14068 (13.7 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

[root@hol ~]#
```

数据库环境变量

其中$OH19 指向 19c ORACLE_HOME

```
[oracle@hol bin]$ pwd
/usr/local/bin
[oracle@hol bin]$ ls -ltr
total 72
```

```
-rwxr-xr-x. 1 root   root     34 Apr  4  2019 sqldeveloper
-rwxr-xr-x. 1 root   root    816 Apr 27  2019 upgr
-rwxr-xr-x. 1 root   root    784 Apr 27  2019 ftex
-rwxr-xr-x. 1 root   root    867 Apr 27  2019 db19
-rwxr-xr-x. 1 root   root    816 Apr 27  2019 ftex11
-rwxr-xr-x. 1 root   root    873 Apr 27  2019 ftex19
-rwxr-xr-x. 1 root   root    800 Apr 27  2019 sqldev
-rwxr-xr-x. 1 root   root    816 Apr 27  2019 upgr11
-rwxr-xr-x. 1 root   root    879 Apr 27  2019 upgr12
-rwxr-xr-x. 1 root   root    873 Apr 27  2019 upgr19
-rwxr-xr-x. 1 root   root    816 Apr 27  2019 cdb1
-rwxr-xr-x. 1 root   root    867 Apr 27  2019 cdb2
-rwxr-xr-x. 1 oracle root   2445 Apr 27  2019 dbhome
-rwxr-xr-x. 1 oracle root   6823 Apr 27  2019 oraenv
-rwxr-xr-x. 1 oracle root   6404 Apr 27  2019 coraenv
-rwxr-xr-x. 1 root   root    816 Apr 28  2019 db12
[oracle@hol bin]$
```

DB 环境

操作系统版本是 Oracle Linux Server 7.6，包含了一下版本的数据库。

版本	ORACLE_HOME	ORACLE_SID	容器数据库	用途
11.2.0.4	/u01/app/oracle/product/11.2.0.4	UPGR	否	将 11.2.0.4 版本的 UPGR 升级到 19c，最后再变成 CDB2 下面的 PDB1
11.2.0.4	/u01/app/oracle/product/11.2.0.4	FTEX	否	通过传输表空间方式迁移到 CDB2 中作为 PDB2。后面还用于数据库的回退和降级
12.2.0.1	/u01/app/oracle/product/12.2.0.1	DB12	否	用于 AutoUpgrade
12.2.0.1	/u01/app/oracle/product/12.2.0.1	CDB1	是	将其中的 PDB3 迁移到 CDB2 并升级
19.3.0.0.0	/u01/app/oracle/product/19	CDB2	是	

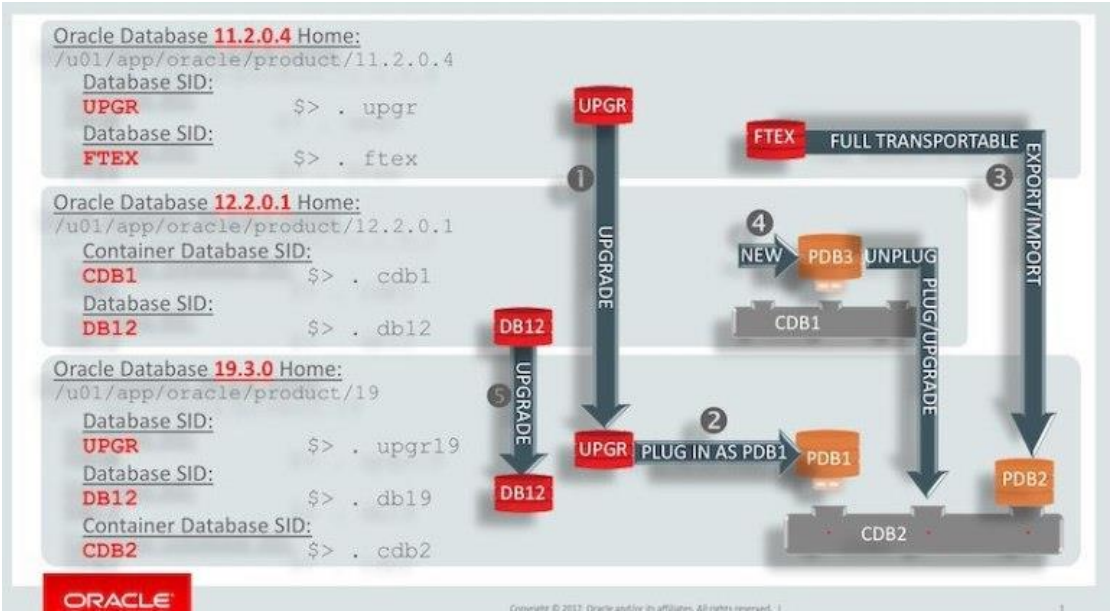
监听信息

```
# listener.ora Network Configuration File:
/u01/app/oracle/product/19/network/admin/listener.ora
# Generated by Oracle configuration tools.
```

```
SID_LIST_LISTENER =
  (SID_LIST =
    (SID_DESC =
      (GLOBAL_DBNAME = UPGR)
      (ORACLE_HOME = /u01/app/oracle/product/11.2.0.4)
      (SID_NAME = UPGR)
    )
    (SID_DESC =
      (GLOBAL_DBNAME = FTEX)
      (ORACLE_HOME = /u01/app/oracle/product/11.2.0.4)
      (SID_NAME = FTEX)
    )
    (SID_DESC =
      (GLOBAL_DBNAME = DB12)
      (ORACLE_HOME = /u01/app/oracle/product/12.2.0.1)
      (SID_NAME = DB12)
    )
    (SID_DESC =
      (GLOBAL_DBNAME = CDB1)
      (ORACLE_HOME = /u01/app/oracle/product/12.2.0.1)
      (SID_NAME = CDB1)
    )
    (SID_DESC =
      (GLOBAL_DBNAME = CDB2)
      (ORACLE_HOME = /u01/app/oracle/product/19)
      (SID_NAME = CDB2)
    )
  )

LISTENER =
  (DESCRIPTION_LIST =
    (DESCRIPTION =
      (ADDRESS = (PROTOCOL = TCP)(HOST = hol)(PORT = 1521))
    )
    (DESCRIPTION =
      (ADDRESS = (PROTOCOL = IPC)(KEY = EXTPROC1521))
    )
  )

ADR_BASE_LISTENER = /u01/app/oracle
```

体验内容

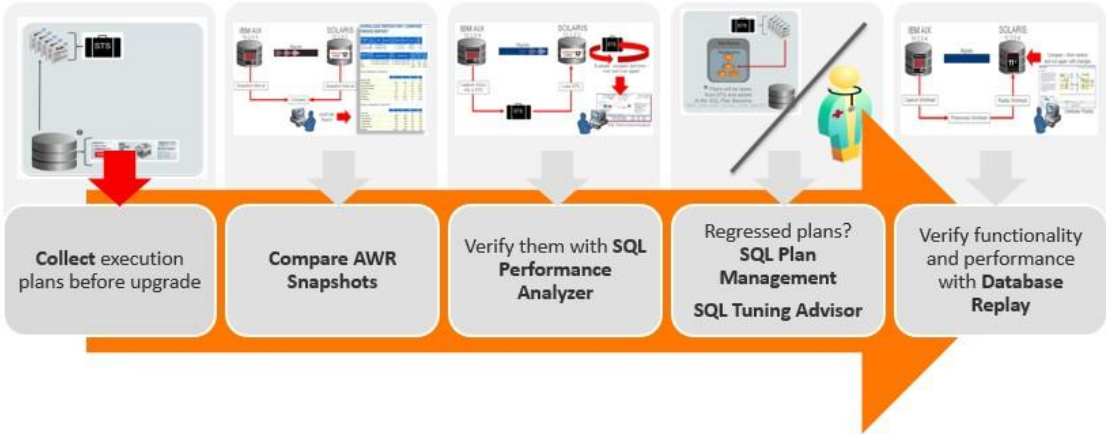
本次体验涉及如下几种场景：



1) 命令行直接升级
   将 11.2.0.4 版本的 UPGR 升级到 19c
2) AutoUpgrade Tool 升级
   将 12.2.0.1 版本的 db12 通过自动升级工具升级到 19c
3) Upgrade non-CDB/Plugin as PDB
   将已经升级的 UPGR 数据库 Plugin 到 19c 版本的 CDB2
4) 通过 full transportable 迁移 FTEX 到 19c PDB2
5) Uplug/Plug/Upgrade
   将 12.2.0.1 下 CDB1 中的 PDB 拔插到 19c 的 CDB2 中，并升级到 19c
6) 升级回退
   将 11.2.0.4 下的 FTEX 数据库升级到 19c 并通过全库导出/导入、Flashback Database 以及降级等进行回退虚拟机内各个数据库的详细说明
7) 创建并升级 12.2.0.1CDB3 到 19c
   创建一个新的 12.2.0.1 的容器数据库，包含两个 PDB，将整个 CDB 升级到 19c

手动升级 UPGR 到 19c

在实际的升级 case 中，往往需要在升级前后进行性能评估，以确保升级后的伸缩性。当然了，验证升级后性能优劣也不完全要按照这个流程，每个运维/支持团队都有自己习惯的方法论。本节是就 Oracle19c 升级 Lab 中涉及到的建议测试工作流一并给大家做个介绍，供以后学习参考使用。



生成负载并捕获 SQL

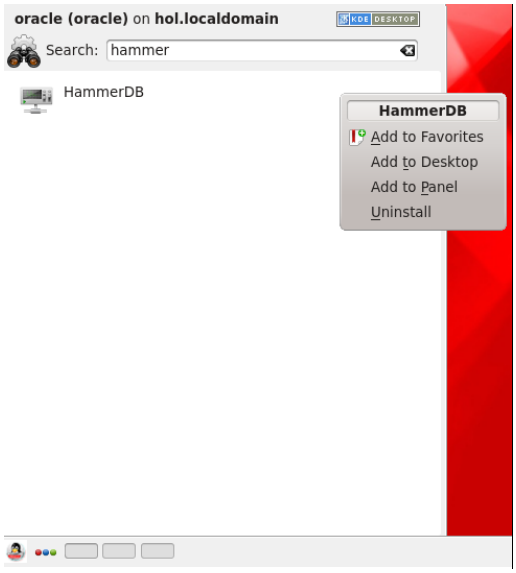
生成开始 AWR 快照

. upgr
cd /home/oracle/scripts
sqlplus / as sysdba
@/home/oracle/scripts/snap.sql

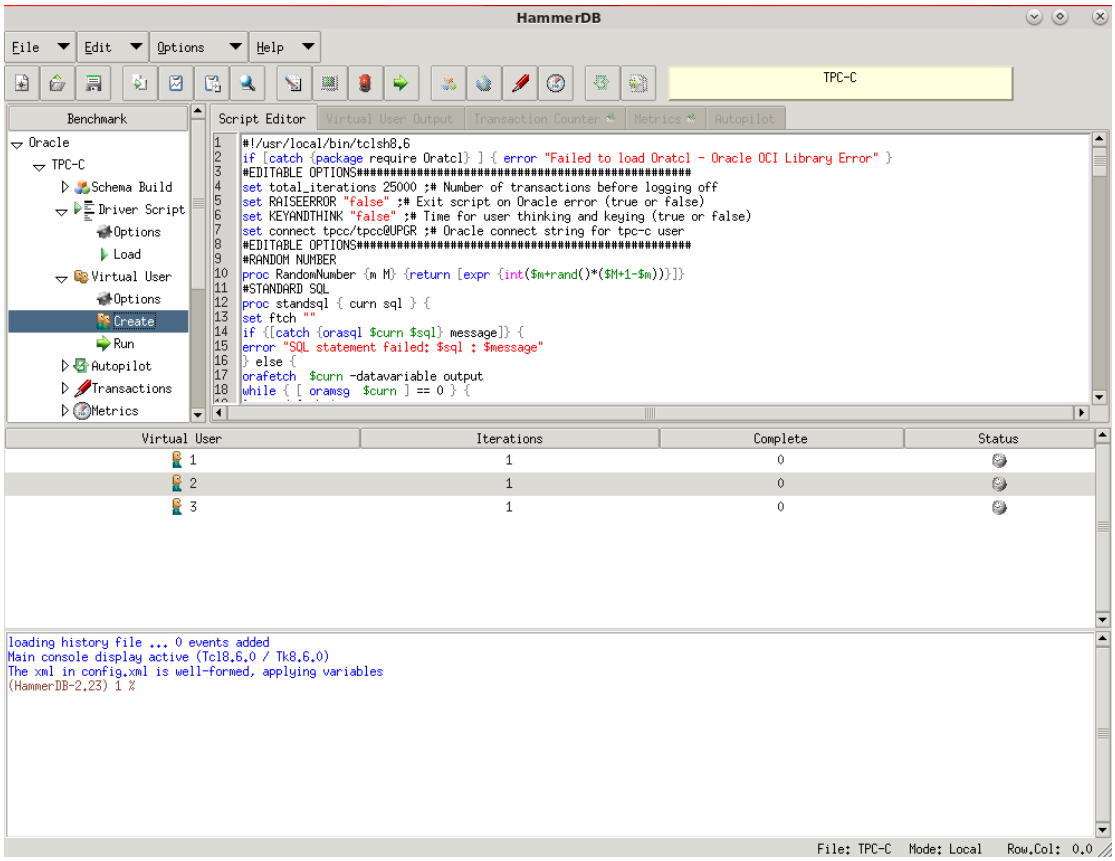
```
[oracle@hol ~]$ . upgr
[UPGR] oracle@hol:~
$ cd /home/oracle/scripts
[UPGR] oracle@hol:~/scripts
$ sqlplus / as sysdba

SQL*Plus: Release 11.2.0.4.0 Production on Tue Dec 17 05:13:09 2019

Copyright (c) 1982, 2013, Oracle.  All rights reserved.


Connected to:
Oracle Database 11g Enterprise Edition Release 11.2.0.4.0 - 64bit Production
With the Partitioning, OLAP, Data Mining and Real Application Testing options

SQL> start /home/oracle/scripts/snap.sql
-----------------------------------------
- AWR Snapshot with Snap-ID: 90 created. -
-----------------------------------------
SQL>
```

打开 HammerDB

在桌面下方的小企鹅，然后在 Search 中输入 Hammer，然后选中点右键添加到桌面，如下图所示：



双击桌面上的 HammerDB 图标，启动 HammerDB，然后点击展开 TPC-C，再点击展开 DriverScript，双击 Load 选项。然后再点击展开 Virtual User，双击 Create，应该会看到 3 个虚拟用户，如下图所示



注：如果要 HammerDB 缺省的标准参数，可以在 GUI 工具界面中，也可以通过编辑 /home/oracle/HammerDB-2.23 下的 config.xml 来完成。

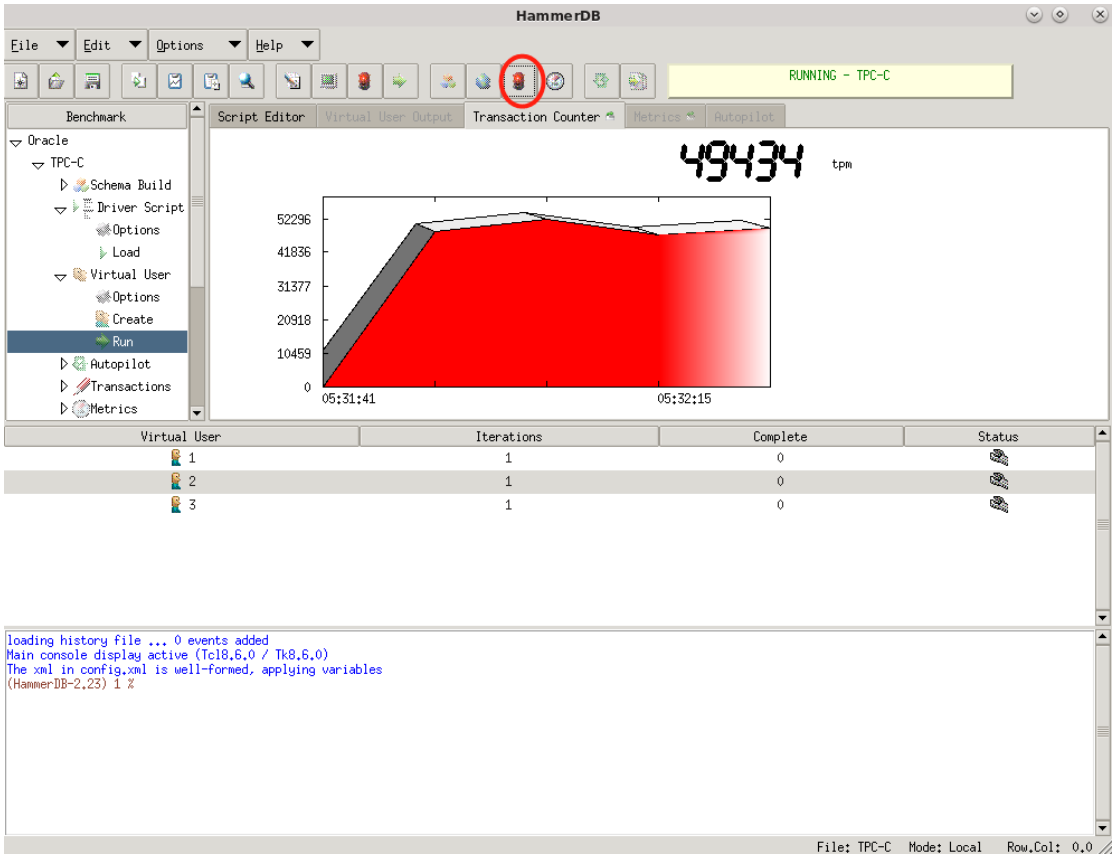
从 Cursor Cache 中捕获 SQL

在 SQL*plus 窗口中以下脚本。
@/home/oracle/scripts/capture_cc.sql
使用这个脚本，将从 CursorCache 中捕获 HammerDB 生成负载时所产生的所有 SQL 语句。这个捕获调度时间为 240 秒，它每 10 秒轮询一次缓存。

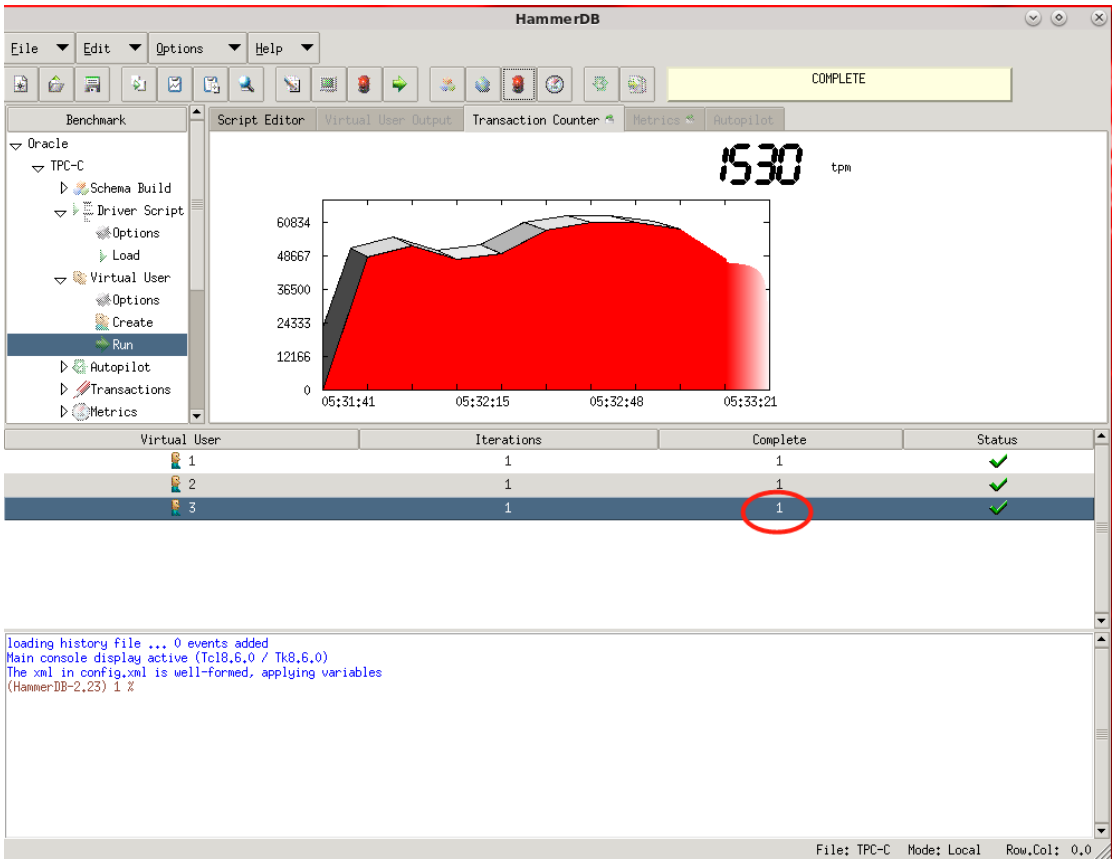
```
SQL> @/home/oracle/scripts/capture_cc.sql
SQL Tuning Set does not exist - will be created ...
Now polling the cursor cache for 240 seconds every 10 seconds ...
You will get back control in 4 minutes.
.
There are now 41 SQL Statements in this STS.
SQL>
```

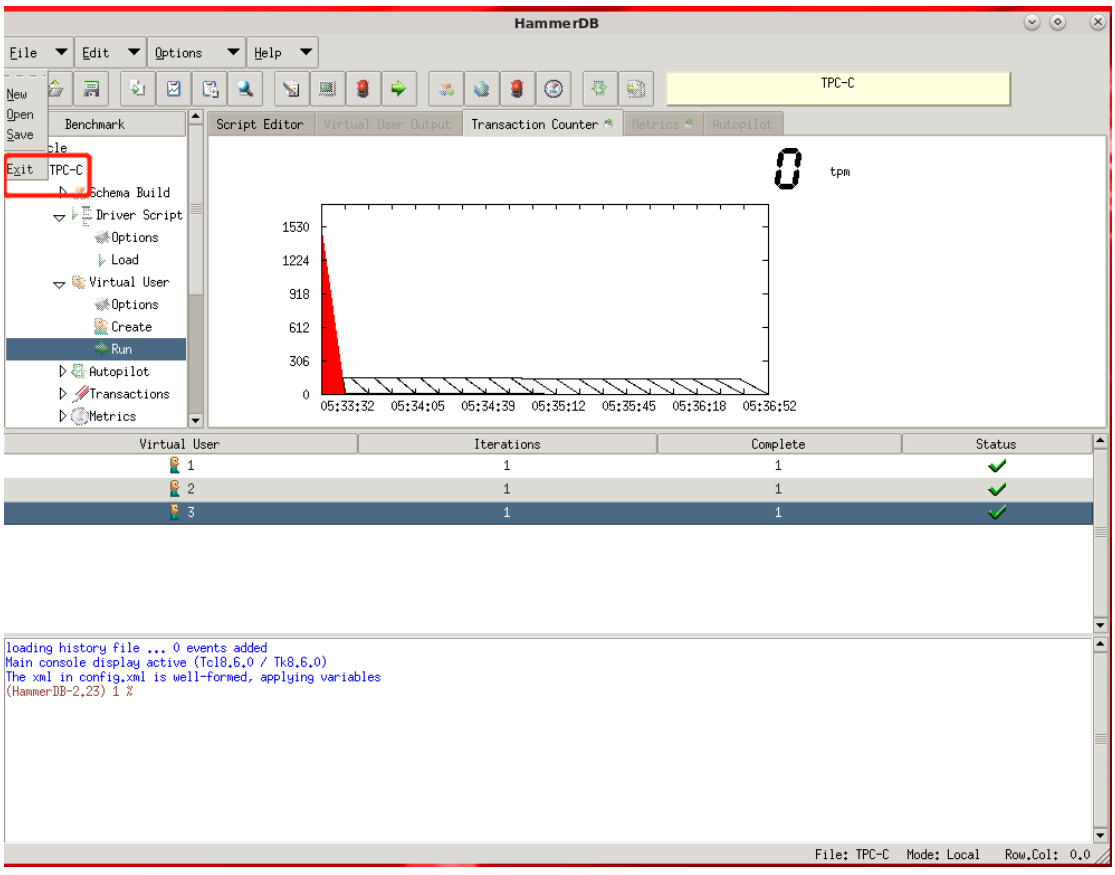
运行 HammerDB 生成 TPC-C 负载

如下图，双击 run，并点击用红色做标记的按钮，能看到 tpm 的信息。



结束时，如下图所示


HammerDB
File
Edit
Options
Help
COMPLETE
Benchmark
Script Editor
Virtual User Output
Transaction Counter
Metrics
Autopilot
Oracle
TPC-C
Schema Build
Driver Script
Options
Load
Virtual User
Options
Create
Run
Autopilot
Transactions
Metrics
1530
tpm
60834
48667
36500
24333
12166
0
05:31:41
05:32:15
05:32:48
05:33:21
Virtual User
Iterations
Complete
Status
1
1
1
2
1
1
3
1
1
loading history file ... 0 events added
Main console display active (Tcl8.6.0 / Tk8.6.0)
The xml in config.xml is well-formed, applying variables
(HammerDB-2.23) 1 %
File: TPC-C
Mode: Local
Row.Col: 0.0


HammerDB
File
Edit
Options
Help
New
Open
Save
Exit
TPC-C
Benchmark
Script Editor
Virtual User Output
Transaction Counter
Metrics
Autopilot
Schema Build
Driver Script
Options
Load
Virtual User
Options
Create
Run
Autopilot
Transactions
Metrics
0
tpm
1530
1224
918
612
306
0
05:33:32
05:34:05
05:34:39
05:35:12
05:35:45
05:36:18
05:36:52
Virtual User
Iterations
Complete
Status
1
1
1
2
1
1
3
1
1
loading history file ... 0 events added
Main console display active (Tcl8.6.0 / Tk8.6.0)
The xml in config.xml is well-formed, applying variables
(HammerDB-2.23) 1 %
File: TPC-C
Mode: Local
Row.Col: 0.0

生成结束 AWR 快照

SQL> @/home/oracle/scripts/snap.sql

- AWR Snapshot with Snap-ID: 91 created. -

SQL>

```
SQL> start /home/oracle/scripts/snap.sql
-----------------------------------------
- AWR Snapshot with Snap-ID: 91 created. -
-----------------------------------------
SQL>
```

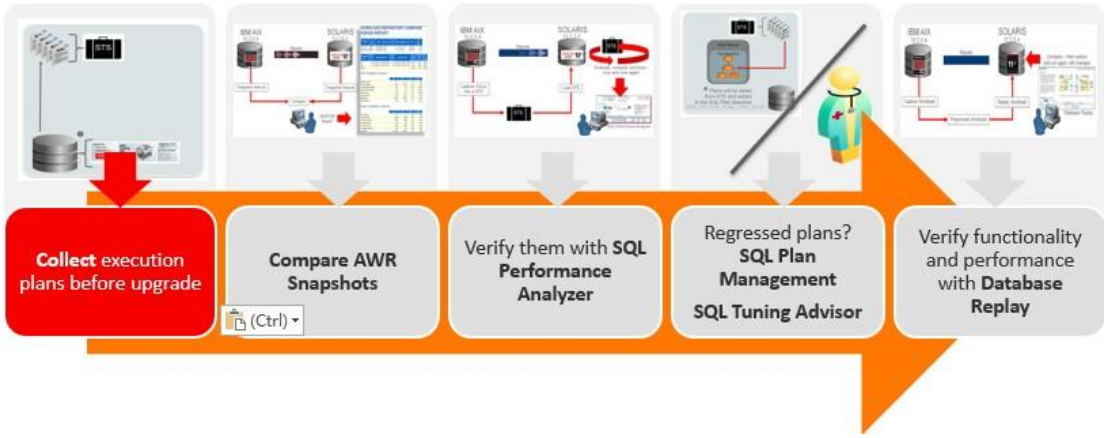
从 AWR 中捕获 SQL

SQL> @/home/oracle/scripts/capture_awr.sql
Snapshot Range between 89 and 92.
There are 30 SQL Statements in STS_CaptureAWR.
SQL>

```
SQL> start /home/oracle/scripts/capture_awr.sql
Snapshot Range between 89 and 92.
There are 30 SQL Statements in STS_CaptureAWR.
SQL>
```

完成升级前信息的捕获

截止到此步骤，我们完成了从 Cursor Cache 和 AWR 中收集相关的信息。



select name, owner, statement_count from dba_sqlset;

```
SQL> select name, owner, statement_count from dba_sqlset;

NAME                           OWNER                          STATEMENT_COUNT
------------------------------ ------------------------------ ---------------
STS_CaptureCursorCache         SYS                                         41
STS_CaptureAWR                 SYS                                         30

SQL>
```

导出 AWR（可选）

在涉及到迁移场景时，导出和保存 AWR 非常重要。升级时，AWR 将保留在数据库中，后面的 AWRDiff 练习会用到。

```
SQL> @?/rdbms/admin/awrextr.sql
~~~~~~~~~~~~~~
AWR EXTRACT
~~~~~~~~~~~~~~
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
~
~   This script will extract the AWR data for a range of snapshots   ~
~   into a dump file.   The script will prompt users for the          ~
~   following information:                                           ~
~       (1) database id                                             ~
~       (2) snapshot range to extract                               ~
~       (3) name of directory object                                ~
~       (4) name of dump file                                       ~
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
~


Databases in this Workload Repository schema
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

   DB Id        DB Name         Host
------------ ------------ ------------
* 72245725     UPGR            hol.localdom
                              ain


The default database id is the local one: '   72245725'.   To use this
database id, press <return> to continue, otherwise enter an alternative.

Enter value for dbid:
```

```
Using     72245725 for Database ID


Specify the number of days of snapshots to choose from
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
Entering the number of days (n) will result in the most recent
(n) days of snapshots being listed.    Pressing <return> without
specifying a number lists all completed snapshots.


Enter value for num_days:

Listing all Completed Snapshots

DB Name            Snap Id      Snap Started
------------ --------- ------------------
UPGR                    89 17 Dec 2019 04:34
                        90 17 Dec 2019 05:14
                        91 17 Dec 2019 05:37
                        92 17 Dec 2019 05:42


Specify the Begin and End Snapshot Ids
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
Enter value for begin_snap: 90
Begin Snapshot Id specified: 90

Enter value for end_snap: 91
End     Snapshot Id specified: 91


Specify the Directory Name
~~~~~~~~~~~~~~~~~~~~~~~~~~~~

Directory Name                 Directory Path
------------------------------ -------------------------------------------------
DATA_PUMP_DIR                    /u01/app/oracle/admin/UPGR/dpdump/
ORACLE_OCM_CONFIG_DIR            /u01/app/oracle/product/11.2.0.4/ccr/hosts/localh
                                 ost.localdomain/state

ORACLE_OCM_CONFIG_DIR2           /u01/app/oracle/product/11.2.0.4/ccr/state
PREUPGRADE_DIR                   /u01/app/oracle/cfgtoollogs/UPGR/preupgrade
XMLDIR                           /u01/app/oracle/product/11.2.0.4/rdbms/xml
```

```
Choose a Directory Name from the above list (case-sensitive).

Enter value for directory_name: DATA_PUMP_DIR

Using the dump directory: DATA_PUMP_DIR

Specify the Name of the Extract Dump File
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
The prefix for the default dump file name is awrdat_90_91.
To use this name, press <return> to continue, otherwise enter
an alternative.

Enter value for file_name: awr90-91

Using the dump file prefix: awr90-91
|
| ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
|   The AWR extract dump file will be located
|   in the following directory/file:
|     /u01/app/oracle/admin/UPGR/dpdump/
|     awr90-91.dmp
| ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
|
|   *** AWR Extract Started ...
|
|   This operation will take a few moments. The
|   progress of the AWR extract operation can be
|   monitored in the following directory/file:
|     /u01/app/oracle/admin/UPGR/dpdump/
|     awr90-91.log
|

End of AWR Extract
SQL>
```

升级前的准备

UPGR 的版本是 11.2.0.4，主要聚焦在升级一个 non-CDB 数据库，而且是可以直接升级到 19c 的版本（适用于 11.2.0.4，12.1.0.2，12.2.0.1，18c)。要完成这样的升级需求，包括如下几个关键步骤

- 操作系统满足安装 Oracle Database 19c 的安装要求
- 安装 Oracle Database 19c 软件

- 运行预升级脚本并修复相关问题，同时做好相关的回退准备
- 升级数据库&升级后的相关动作
- 测试验证，确保升级后应用性能和功能不受影响
- 如果要转换为 CDB 架构（因为从 20c 版本开始 non-CDB 不再支持，所以建议大家转换到 CDB 架构），那么再创建新的 CDB 数据库，并将升级后的 non-CDB 插入到新的 CDB 环境中。
- 后续其他相关的调整
- 在我们这个 workshop 中，19c 数据库环境已经提前建好，所以直接从运行预升级脚本开始。

注意事项

关于目标 19c ORACLE_HOME

(1)检查目标硬件平台/操作系统是否兼容 19c 版本。
(2)下载安装 19c 软件到一个新的 ORACLE_HOME 并确认没有编译错误。
(3)确保设置好相应的 ORACLE_HOME，PATH，LD_LIBRARY_PATH，LIBPATH 等指向到 19c 目标库 HOME。
(4)如果有目前最新的 RU 是 19.5.0，建议下载并安装它们，可参考 2118136.2 来下载这些补丁。
(5)查看文章"Patches to apply before upgrading Oracle GI and DB to 19c(2539751.1)"中给出的补丁建议

关于 19c Schema-Only 特性说明

在开始升级之前，请确认是否要对密码处于 EXPIRED 状态且其账户处于 LOCKED 状态的默认 Oracle 数据库账户使用密码身份验证。

在升级到 Oracle Database 19c 之后，默认的 Oracle 账号（没有设置密码并且处于 EXPIRED 和 LOCKED 状态）会被置为 NOAUTHENTICATION 状态。由于此新功能，这些默认账号会变为 schema-only 账户，并无法使用密码验证。此功能的好处是管理员不再需要定期修改这些 Oracle 默认账号的密码。此功能还可以降低未授权者使用默认密码侵入这些账户的安全风险。

如果要在升级期间阻止将这些 Oracle 账户设置为 schema-only 账户，则必须在开始升级之前为该账户设置有效的强密码，或者在升级后为这些账户设置有效的强密码，或者在升级前解锁账户。

当然了，升级后管理员也是可以为 schema-only 启用密码身份验证的。为了更好的安全性，Oracle 建议您将这些账户保留为 schema-only 账号。

关于升级前清空回收站说明

Oracle 建议升级前在清空回收站。
SQL>PURGEDBA_RECYCLEBIN;
如果版本是基于 12c 的 CDB 环境，也可以通过如下脚本一次清空所有的 PDBs 下的回收站

```
$ORACLE_HOME/perl/bin/perl $ORACLE_HOME/rdbms/admin/catcon.pl -l /tmp -b
purge_recyclebin -- --x "PURGEDBA_RECYCLEBIN"
```

关于升级前收集统计信息说明

Oracle 强烈推荐升级前收集统计信息。推荐使用
DBMS_STATS.GATHER_DICTIONARY_STATS 来收集统计信息，比如，执行下面的 SQL:

对于 non-CDB 环境，直接调用如下命令来收集就好!
SQL>EXECDBMS_STATS.GATHER_DICTIONARY_STATS;

对于 CDB 环境，通过下面的命令就可以。该命令会在所有的 PDB 上执行收集数据字典统计信息。

```
$ORACLE_HOME/perl/bin/perl $ORACLE_HOME/rdbms/admin/catcon.pl -l /tmp -b
gatherstats -- --x "execdbms_stats.gather_dictionary_stats"
```

要在某个特定的 PDB 上收集统计信息，比如 PDB1，可通过下面的命令来收集

```
$ORACLE_HOME/perl/bin/perl $ORACLE_HOME/rdbms/admin/catcon.pl -l /tmp -b
gatherstats_pdb1 -- --x "execdbms_stats.gather_dictionary_stats" -c 'PDB1'
```

升级前备份数据库

对于直接就地升级的环境，强烈建议在运行 Pre-Upgrade Information Tool 之后备份数据库，同时考虑创建 guaranteed flashback restore point。并测试备份的有效性，以确保出现问题后有回退方案。

关于 catcon.pl

请参考 How to execute sql scripts in Multitenant environment (catcon.pl) (Doc ID 1932340.1)

在 UPGR 上运行 preupgrade.jar

java -jar $OH19/rdbms/admin/preupgrade.jar TEXT TERMINAL
注：TEXT TERMINAL 表示以文本形式输出到屏幕
阅读输出并遵循所有建议，打开第二个 terminal，运行 SQL*plus，在 UPGR 数据库中执行必要的步骤，包括 preupgrade_fixups.sql。

其中“AUTOFIXUP”意思是 preupgrade_fixups.sql 脚本将为您完成此任务。在这种情况下不需要手动操作。

```
$ java -jar $OH19/rdbms/admin/preupgrade.jar TEXT TERMINAL
Report generated by Oracle Database Pre-Upgrade Information Tool Version
19.0.0.0.0 Build: 1 on 2019-12-17T08:17:51

Upgrade-To version: 19.0.0.0.0

======================================
Status of the database prior to upgrade
======================================
      Database Name:  UPGR
     Container Name:  Not Applicable in Pre-12.1 database
       Container ID:  Not Applicable in Pre-12.1 database
            Version:  11.2.0.4.0
     DB Patch Level:  PSU 11.2.0.4.190416
         Compatible:  11.2.0.4.0
          Blocksize:  8192
           Platform:  Linux x86 64-bit
      Timezone File:  14
  Database log mode:  NOARCHIVELOG
           Readonly:  FALSE
            Edition:  EE
```


$ java -jar $OH19/rdbms/admin/preupgrade.jar TEXT TERMINAL
Report generated by Oracle Database Pre-Upgrade Information Tool Version
19.0.0.0.0 Build: 1 on 2019-12-17T08:17:51

Upgrade-To version: 19.0.0.0.0

======================================
Status of the database prior to upgrade
======================================
Database Name: UPGR
Container Name: Not Applicable in Pre-12.1 database
Container ID: Not Applicable in Pre-12.1 database
Version: 11.2.0.4.0
DB Patch Level: PSU 11.2.0.4.190416
Compatible: 11.2.0.4.0
Blocksize: 8192
Platform: Linux x86 64-bit
Timezone File: 14

```
  Database log mode:    NOARCHIVELOG
                Readonly:    FALSE
                 Edition:    EE

  Oracle Component                          Upgrade Action       Current Status
  ----------------                          --------------       --------------
  Oracle Server                            [to be upgraded]    VALID
  Oracle Workspace Manager                  [to be upgraded]    VALID
  Oracle Label Security                    [to be upgraded]    VALID
  Oracle XML Database                       [to be upgraded]    VALID

==============
BEFORE UPGRADE
==============

  REQUIRED ACTIONS
  ================
  1.  Logged in AS SYSDBA, run $ORACLE_HOME/rdbms/admin/olspreupgrade.sql from
      the new Oracle Database 19 home.

      olspreupgrade.sql has not been run on this database.    To view the number
      of records that olspreupgrade.sql moves, use the following command:

        SELECT count(*) FROM system.aud$;

      As part of the upgrade to 19, records in the 11.2.0.4.0 audit table
      SYSTEM.AUD$ are moved to SYS.AUD$. This step can be manually performed
      before the upgrade to reduce downtime.    Refer to the 19 Oracle Label
      Security Administrator's Guide, or to Oracle Database Upgrade Guide for
      further details.

  RECOMMENDED ACTIONS
  ===================
  2.  Update NUMERIC INITIALIZATION PARAMETERS to meet estimated minimums.
      This action may be done now or when starting the database in upgrade mode
      using the 19 ORACLE HOME.

       Parameter                                 Currently    19 minimum
       ---------                                 ---------   ------------------
       processes                                        79                  300
      *sga_target                                813694976           1002438656

      The database upgrade process requires certain initialization parameters
      to meet minimum values.    The Oracle upgrade process itself has minimum
```

values which may be higher and are marked with an asterisk. After upgrading, those asterisked parameter values may be reset if needed.

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

   The database contains the following initialization parameters whose name begins with an underscore:

   _cursor_obsolete_threshold

   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.

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

   EXECUTE DBMS_STATS.GATHER_DICTIONARY_STATS;

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

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

   For information on managing optimizer statistics, refer to the 11.2.0.4 Oracle Database Performance Tuning Guide.

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

   None of the fixed object tables have had stats collected.

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

   For information on managing optimizer statistics, refer to the 11.2.0.4 Oracle Database Performance Tuning Guide.

INFORMATION ONLY

==========

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

Tablespace	Size	Min Size For Upgrade
TEMP	20 MB	150 MB
UNDOTBS1	305 MB	408 MB

Minimum tablespace sizes for upgrade are estimates.

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

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

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

ORACLE GENERATED FIXUP SCRIPT

==========

All of the issues in database UPGR

which are identified above as BEFORE UPGRADE "(AUTOFIXUP)" can be resolved by executing the following

SQL>@/u01/app/oracle/cfgtoollogs/UPGR/preupgrade/preupgrade_fixups.sql

==========

AFTER UPGRADE

==========

REQUIRED ACTIONS

==========

None

RECOMMENDED ACTIONS

==========

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

The database is using time zone file version 14 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.

9. To identify directory objects with symbolic links in the path name, run $ORACLE_HOME/rdbms/admin/utldirsymlink.sql AS SYSDBA after upgrade. Recreate any directory objects listed, using path names that contain no symbolic links.

   Some directory object path names may currently contain symbolic links.

   Starting in Release 18c, symbolic links are not allowed in directory object path names used with BFILE data types, the UTL_FILE package, or external tables.

10. (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.

11. 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 11.2.0.4 Oracle Database Performance Tuning Guide.

ORACLE GENERATED FIXUP SCRIPT

==========

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

      SQL>@/u01/app/oracle/cfgtoollogs/UPGR/preupgrade/postupgrade_fixups.sql


==================
PREUPGRADE SUMMARY
==================
   /u01/app/oracle/cfgtoollogs/UPGR/preupgrade/preupgrade.log
   /u01/app/oracle/cfgtoollogs/UPGR/preupgrade/preupgrade_fixups.sql
   /u01/app/oracle/cfgtoollogs/UPGR/preupgrade/postupgrade_fixups.sql

Execute fixup scripts as indicated below:

Before upgrade:

Log into the database and execute the preupgrade fixups
@/u01/app/oracle/cfgtoollogs/UPGR/preupgrade/preupgrade_fixups.sql

After the upgrade:

Log into the database and execute the postupgrade fixups
@/u01/app/oracle/cfgtoollogs/UPGR/preupgrade/postupgrade_fixups.sql

Preupgrade complete: 2019-12-17T08:17:52
```

本试验环境 preupgrade.jar 的日志输出要求运行 olspreupgrade.sql，将 audit table 从 system 用户移动到 SYS，修改必要的参数，并运行 preupgrade_fix.sql 来修复。在实际的升级场景中，按照输出建议依次修复。

执行升级前的 prefixups 相关脚本

在实际的生产环境中，可能和这个不太一样，要详细阅读预升级脚本的输出，然后按照建议完成相应操作。

```
sqlplus / as sysdba
@/u01/app/oracle/product/19/rdbms/admin/olspreupgrade.sql
alter system set processes=300 scope=spfile;
alter system set sga_target=1G scope=spfile;
create pfile from spfile;
```

在后面编辑 pfile 时,去掉类似_cursor_obsolete_threshold 这样的隐藏参数

```
@/u01/app/oracle/cfgtoollogs/UPGR/preupgrade/preupgrade_fixups.sql
```

```
SQL> start /u01/app/oracle/product/19/rdbms/admin/olspreupgrade.sql

Session altered.


Function created.

No errors.

Function created.

No errors.

Function created.

No errors.
2019-12-17 08:33:00 ******* BEGINNING OLS PRE UPGRADE SCRIPT ********
The amount of FREE space required = 131072 Bytes
Free space available  on SYSTEM tablespace= 34044428288 Bytes
2019-12-17 08:33:00 ******** PROCEEDING WITH OLS PRE UPGRADE *******
Audit records successfully moved to SYS.PREUPG_AUD$

PL/SQL procedure successfully completed.

No errors.
Total number of rows in SYS.PREUPG_AUD$: 0
2019-12-17 08:33:00  ******* FINISHING OLS PRE UPGRADE SCRIPT ********

PL/SQL procedure successfully completed.

No errors.
SQL> alter system set processes=300 scope=spfile;

System altered.

SQL> alter system set sga_target=1G scope=spfile;

System altered.

SQL> create pfile from spfile;

File created.
```

当运行 preupgrade_fix.sql 时，可能会看到 4 个"NO"的结果，不用紧张，这些调整 restart 后才会生效，可以安全地忽略。如果在实际的升级环境，那么这些问题要适当的关注一下。

```
Executing Oracle PRE-Upgrade Fixup Script

Auto-Generated by:       Oracle Preupgrade Script
                         Version: 19.0.0.0.0 Build: 1
Generated on:            2019-12-17 08:17:48

For Source Database:     UPGR
Source Database Version: 11.2.0.4.0
For Upgrade to Version:  19.0.0.0.0

Preup                                Preupgrade
```

```
Action                                  Issue Is
Number   Preupgrade Check Name          Remedied      Further DBA Action
------   ------------------------       ----------    ------------------------------
     1.  ols_sys_move                    YES           None.
     2.  parameter_min_val               NO            Manual fixup recommended.
     3.  hidden_params                   NO            Informational only.
                                                       Further action is optional.
     4.  dictionary_stats                YES           None.
     5.  pre_fixed_objects               YES           None.
     6.  tablespaces_info                NO            Informational only.
                                                       Further action is optional.
     7.  rman_recovery_version           NO            Informational only.
                                                       Further action is optional.

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

PL/SQL procedure successfully completed.
```

```
SQL> start /u01/app/oracle/cfgtoollogs/UPGR/preupgrade/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:            2019-12-17 08:17:48

For Source Database:     UPGR
Source Database Version: 11.2.0.4.0
For Upgrade to Version:  19.0.0.0.0

Preup                              Preupgrade
Action                             Issue Is
Number  Preupgrade Check Name      Remedied    Further DBA Action
------  -----------------------    ----------  --------------------------------
    1.  ols_sys_move               YES         None.
    2.  parameter_min_val          NO          Manual fixup recommended.
    3.  hidden_params              NO          Informational only.
                                               Further action is optional.
    4.  dictionary_stats           YES         None.
    5.  pre_fixed_objects          YES         None.
    6.  tablespaces_info           NO          Informational only.
                                               Further action is optional.
    7.  rman_recovery_version      NO          Informational only.
                                               Further action is optional.

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

PL/SQL procedure successfully completed.

SQL>
```

关闭 11.2.0.4UPGR 数据库准备升级

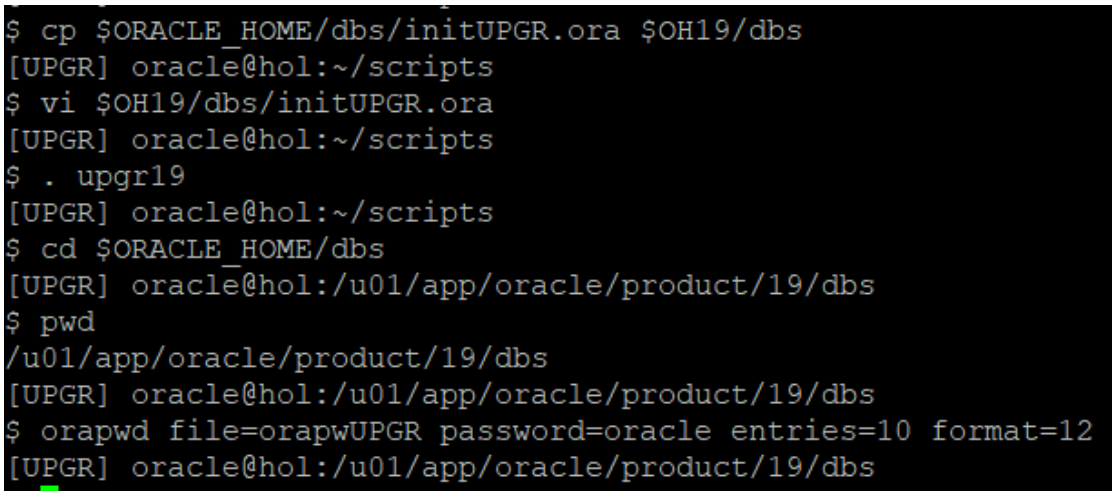
SQL> shut immediate
Database closed.
Database dismounted.
ORACLE instance shut down.
SQL>

准备 19c Oracle Home 下参数文件和密码文件

cp $ORACLE_HOME/dbs/initUPGR.ora $OH19/dbs
编辑$OH19/dbs/initUPGR.ora，删除不支持的初始化参数并调整已弃用的初始化参数。在新版本中，某些参数不再被支持，有些参数已经被废弃。从启动新 Oracle 数据库实例的任何参

数文件中删除所有不支持的参数以及隐藏参数(除非有特定要求)。不受支持的参数可能会导致新的 Oracle 数据库版本出错。
本 Lab 练习中，我们删除_cursor_obsolete_threshold 参数，其他保持不变，保存退出。

$ cp $ORACLE_HOME/dbs/initUPGR.ora $OH19/dbs
[UPGR] oracle@hol:~/scripts
$ vi $OH19/dbs/initUPGR.ora
[UPGR] oracle@hol:~/scripts
$. upgr19
[UPGR] oracle@hol:~/scripts
$ cd $ORACLE_HOME/dbs
[UPGR] oracle@hol:/u01/app/oracle/product/19/dbs
$ pwd
/u01/app/oracle/product/19/dbs
[UPGR] oracle@hol:/u01/app/oracle/product/19/dbs
$ orapwd file=orapwUPGR password=oracle entries=10 format=12
[UPGR] oracle@hol:/u01/app/oracle/product/19/dbs
$



开始升级

SQL> create spfile from pfile;
SQL> startup upgrade;

```
$ sqlplus / as sysdba

SQL*Plus: Release 19.0.0.0.0 - Production on Tue Dec 17 08:50:53 2019
Version 19.3.0.0.0

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

Connected to an idle instance.

SQL> create spfile from pfile;

File created.

SQL> startup upgrade;
ORACLE instance started.

Total System Global Area 1073737800 bytes
Fixed Size                  8904776 bytes
Variable Size             276824064 bytes
Database Buffers          780140544 bytes
Redo Buffers                7868416 bytes
Database mounted.
Database opened.
SQL> exit
Disconnected from Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
Version 19.3.0.0.0
[UPGR] oracle@hol:/u01/app/oracle/product/19/dbs
```

执行升级脚本

$ cd $ORACLE_HOME/rdbms/admin

$ $ORACLE_HOME/perl/bin/perl catctl.pl -n 4 -l /home/oracle catupgrd.sql

注意：-n 是表示并行度的意思，缺省和 cpu_count 相同，-l 表示输出日志目录

```
Number of Cpus        = 4
Database Name         = UPGR
DataBase Version      = 11.2.0.4.0
Parallel SQL Process Count            = 4
Components in [UPGR]
    Installed [CATALOG CATPROC OLS OWM XDB]
Not Installed [APEX APS CATJAVA CONTEXT DV EM JAVAVM MGW ODM ORDIM RAC SDO WK XML XOQ]

-----------------------------------------------------
Phases [0-107]         Start Time:[2019_12_17 08:54:09]
-----------------------------------------------------
***********   Executing Change Scripts   ***********
Serial   Phase #:0    [UPGR] Files:1    Time: 75s
***************   Catalog Core SQL   ***************
Serial   Phase #:1    [UPGR] Files:5
```

```
$ $ORACLE_HOME/perl/bin/perl catctl.pl -n 4 -l /home/oracle catupgrd.sql

Argument list for [catctl.pl]
For Oracle internal use only A = 0
Run in                       c = 0
Do not run in                C = 0
Input Directory              d = 0
Echo OFF                     e = 1
Simulate                     E = 0
Forced cleanup               F = 0
Log Id                       i = 0
Child Process                I = 0
Log Dir                      l = /home/oracle
Priority List Name           L = 0
Upgrade Mode active          M = 0
SQL Process Count            n = 4
SQL PDB Process Count        N = 0
Open Mode Normal             o = 0
Start Phase                  p = 0
End Phase                    P = 0
Reverse Order                r = 0
AutoUpgrade Resume           R = 0
Script                       s = 0
Serial Run                   S = 0
RO User Tablespaces          T = 0
Display Phases               y = 0
Debug catcon.pm              z = 0
Debug catctl.pl              Z = 0

catctl.pl VERSION: [19.0.0.0.0]
          STATUS: [Production]
           BUILD: [RDBMS_19.3.0.0.0DBRU_LINUX.X64_190417]


/u01/app/oracle/product/19/rdbms/admin/orahome = [/u01/app/oracle/product/19]
/u01/app/oracle/product/19/bin/orabasehome = [/u01/app/oracle/product/19]
catctlGetOraBaseLogDir = [/u01/app/oracle/product/19]

Analyzing file /u01/app/oracle/product/19/rdbms/admin/catupgrd.sql

Log file directory = [/home/oracle]

catcon::set_log_file_base_path: ALL catcon-related output will be written to [/home/oracle/catupgrd_catcon_11422.lst]

catcon::set_log_file_base_path: catcon: See [/home/oracle/catupgrd*.log] files for output generated by scripts

catcon::set_log_file_base_path: catcon: See [/home/oracle/catupgrd_*.lst] files for spool files, if any
```

```
**********************   Post Upgrade   **********************
Serial   Phase #:103  [UPGR] Files:1    Time: 36s
**********************   Summary report   **********************
Serial   Phase #:104  [UPGR] Files:1    Time: 2s
***   End PDB Application Upgrade Post-Shutdown   **
Serial   Phase #:105  [UPGR] Files:1    Time: 2s
Serial   Phase #:106  [UPGR] Files:1    Time: 0s
Serial   Phase #:107  [UPGR] Files:1     Time: 30s

------------------------------------------------------------
Phases [0-107]         End Time:[2019_12_17 09:19:05]
------------------------------------------------------------

Grand Total Time: 1498s

 LOG FILES: (/home/oracle/catupgrd*.log)

Upgrade Summary Report Located in:
/home/oracle/upg_summary.log

Grand Total Upgrade Time:    [0d:0h:24m:58s]
```

升级后续动作

升级动作执行完毕后，数据库将会被关闭，以正常方式打开数据库。然后调用 utlrp.sql 编译失效对象，然后运行 postupgrade_fixups.sql（这个文件是在 preupgrade.jar 执行过程中产生的）执行修复。

编译无效对象

sqlplus / as sysdba
startup;
@?/rdbms/admin/utlrp.sql
select count(*) from dba_objects where status='INVALID';
确保没有无效对象

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

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

SQL>
```

执行 postupgrade 脚本

@/u01/app/oracle/cfgtoollogs/UPGR/preupgrade/postupgrade_fixups.sql

```
Executing Oracle POST-Upgrade Fixup Script

Auto-Generated by:       Oracle Preupgrade Script
                         Version: 19.0.0.0.0 Build: 1
Generated on:            2019-12-17 08:17:51

For Source Database:     UPGR
Source Database Version: 11.2.0.4.0
For Upgrade to Version:  19.0.0.0.0

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

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

PL/SQL procedure successfully completed.


Session altered.
```

修改 timezone

如果您的系统只处理本国的日期，而且本国也没有更改时区或夏令时政策，其实也可以不用进行timezone 的升级。

执行 utltz_upg_check.sql

@?/rdbms/admin/utltz_upg_check.sql

```
SQL> start ?/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 DSTv14 .
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.
```

执行 utltz_upg_apply.sql

@?/rdbms/admin/utltz_upg_apply.sql

```
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 DSTv14 .
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> @?/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 1073737800 bytes
Fixed Size                  8904776 bytes
Variable Size             469762048 bytes
Database Buffers          587202560 bytes
Redo Buffers                7868416 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 1073737800 bytes
Fixed Size                  8904776 bytes
Variable Size             469762048 bytes
Database Buffers          587202560 bytes
Redo Buffers                7868416 bytes
Database mounted.
Database opened.
INFO: Upgrading all non-SYS TSTZ data.
INFO: It might take time before any further output is seen ...
INFO: Do NOT start any application yet that uses TSTZ data!
```

```
INFO: Next is a list of all upgraded tables:
Table list: "GSMADMIN_INTERNAL"."AQ$_CHANGE_LOG_QUEUE_TABLE_S"
Number of failures: 0
Table list: "GSMADMIN_INTERNAL"."AQ$_CHANGE_LOG_QUEUE_TABLE_L"
Number of failures: 0
INFO: Total failures during update of TSTZ data: 0 .
An upgrade window has been successfully ended.
INFO: Your new Server RDBMS DST version is DSTv32 .
INFO: The RDBMS DST update is successfully finished.
INFO: Make sure to exit this SQL*Plus session.
INFO: Do not use it for timezone related selects.

Session altered.

SQL>
```

```
SQL> start ?/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 1073737800 bytes
Fixed Size                  8904776 bytes
Variable Size             469762048 bytes
Database Buffers          587202560 bytes
Redo Buffers                7868416 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 1073737800 bytes
Fixed Size                  8904776 bytes
Variable Size             469762048 bytes
Database Buffers          587202560 bytes
Redo Buffers                7868416 bytes
Database mounted.
Database opened.
INFO: Upgrading all non-SYS TSTZ data.
INFO: It might take time before any further output is seen ...
INFO: Do NOT start any application yet that uses TSTZ data!
INFO: Next is a list of all upgraded tables:
Table list: "GSMADMIN_INTERNAL"."AQ$_CHANGE_LOG_QUEUE_TABLE_S"
Number of failures: 0
Table list: "GSMADMIN_INTERNAL"."AQ$_CHANGE_LOG_QUEUE_TABLE_L"
Number of failures: 0
INFO: Total failures during update of TSTZ data: 0 .
An upgrade window has been successfully ended.
```

```
INFO: Your new Server RDBMS DST version is DSTv32 .
INFO: The RDBMS DST update is successfully finished.
INFO: Make sure to exit this SQL*Plus session.
INFO: Do not use it for timezone related selects.

Session altered.

SQL>
```

升级后查看 timezone

```
SQL> select version from v$timezone_file;

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

1 row selected.

SQL>
```

关于 timezone

在一些 Global 的应用环境中(最典型就是不同时区发生交易)，数据库端的国际化特性的支持也显得尤其重要。Oracle 数据库中也有很多支持国际化的特性，典型的比如字符集、语言、地区特性、时区(timezone)等。如果相关参数设置不当，就会遇到一些麻烦。对于 timezone 来说，Oracle 中的 timezone 可以分为两大类：数据库时区和会话时区

➢ 查看数据库时区

select dbtimezone from dual;

数据库的 timezone 可以在创建数据库的时候指定，也可以在数据库创建之后通过 alter database set time_zone='+8:00';这样的语句来修改，重启后生效。

➢ 查看 session 时区

select sessiontimezone from dual;

session 的 timezone 可以通过 alter session 来修改。

数据库的 timezone 也和 TIMESTAMP WITH LOCAL TIME ZONE 数据类型相关。TIMESTAMP WITH LOCAL TIME ZONE 数据类型从客户端传入数据库以后，将转化为数据库时区存储到数据库。在需要进行相关计算的时候，Oracle 先把事件转换为标准时间（UTC）,完成计算后再把结果转换为数据库时区保存到 数据库。

与 timezone 相关的两种 TIMESTAMP 类型

- TIMESTAMP WITH TIME ZONE
  存储带时区信息的 TIMESTAMP(以和 UTC 时间差或者地区信息的形式保存)
- TIMESTAMP WITH LOCAL TIME ZONE
  是另外一种不同类型的 TIMESTAMP，和 TIMESTAMP WITH TIME ZONE 不同的是：数据库不保存时区相关信息，而是把客户段输入的时间转换为基于 database timezone 的时间后存入数据库（这也是 database timezone 的一个重要意义)。当用户请求此类型信息时，Oracle 把数据转换为用户 session 的时区时间返回给客户端。所以 Oracle 建议把 database timezone 设置为标准时间 UTC，这样可以节省每次转换所需要的时间，提高性能。

time zone 名称有一个国际标准，有些国家偶尔会改变他们的时区，或改变他们处理 DST 的方式，所以在每个 Oracle 数据库版本中都会跟进维护。从 11gR2 开始，针对 Oracle

数据库新的 time zone 文件(Oracle 数据库时区文件包含有效的时区名称，存放在 $ORACLE_HOME/oracore/zoneinfo 目录下)会随着 升级和补丁一起发布，但是他们不会自动应用到数据库中。

数据库版本	缺省的 TimeZone 版本
11.2.0.2 - 11.2.0.4	DST V14
12.1.0.1, 12.1.0.2	DST V18
12.2.0.1	DST V26
18c	DST V31
19c	DST V32

最新的Daylight Saving Time (DST) patch 可参见Note 412160.1。
从 18c 开始这些上面这些?/rdbms/admin 下的脚本随着 RDBMS 软件包一并提供。
下面是升级 timezone 的相关检查和升级脚本

执行顺序	执行脚本	说明
1	SQL> select version from v$timezone_file; 或 SQL> select tz_version FROM registry$database; 或 SQL> SELECT property_name, property_value FROM database_properties WHERE property_name LIKE 'DST_%' ORDER BY property_name;	查看当前 Time Zone 版本，应该是 14
2	SQL> SELECT DBMS_DST.get_latest_timezone_version FROM dual;	查看当前数据库环境下最新可用的 timezone 文件版本，19c 缺省是 32，如果没 有应用 DST Patch，应该返回 32。
3	SQL> **@?**/rdbms/admin/utltz_countstats.sql	通过 dba_tables 中 num_rows 统计信息来提供在数据库中 TIMESTAMP WITH TIME ZONE（简称 TSTZ）数据的数量。
4	SQL> **@?**/rdbms/admin/utltz_countstar.sql	脚本使用 COUNT（*）查询每个具有 TSTZ 列的表来计算数据库中的 TIMESTAMP WITH TIME ZONE 数据的数量。这个脚本在使用 DBMS_DST 包或 utlz_upg_check.sql 和 utlz_upg_apply.sql 脚本时非常有用。
5	SQL> **@?**/rdbms/admin/utltz_upg_check.sql	时区升级检查脚本
6	SQL> **@?**/rdbms/admin/utltz_upg_apply.sql	该脚本将重新启动数据库并调整时区数据。
7	SQL> select version from v$timezone_file;	再次检查当前 Time Zone 版本，应该是 32

兼容性参数调整

当升级到一个新版本时，有些特性必需使用较高的兼容性设置，否则用不了。兼容性也决定了磁盘中的数据格式和结构。compatible 参数就是用来指定数据库磁盘格式必须与之兼容的 Oracle 版本号。兼容性参数也用于确定数据库如何与外部交互（比如应用程序），比如将兼容性参数设置为 12.1.0 就允许数据库的行为和 12.1 版本一样。

所以 Oracle 建议在升级数据库并完成测试以后再调整兼容性参数，这个没有具体的时间上的限制，只是如果你不修改兼容性参数，那么新版本的一些特性将无法使用。

需要注意的是，该参数从 OracleDatabase9i 版本以来只能增加，一旦更改就不能恢复到以前的值。当需要做数据库降级处理的时候，就只能降级为 compatible 参数中指定的版本或任何后续版本。该参数必需在 mount 状态修改，对于 RAC 环境，各个 intance 要保持一致。另外，不论是通过命令行、DBCA 还是 AutoUpgrade 工具来升级，都不会自动修改兼容性参数，都需要我们手动去调整。只有如下 2 种情况下是列外的，会自动推进兼容性设置：

1）升级到的目标版本不再支持更早的兼容性设置
   比如你升级一个兼容性设置为 10.2.0 的 Oracle11.2.0.4 版本的数据库，那么当升级到 19c 时，会强制往上推到 11.2.0，因为 19c 支持的最小兼容性设置是 11.2.0
2）多租户架构下从兼容性低的 CDB 中 Unplug 下然后 Plug 到兼容性高的 CDB
   就会发生隐式兼容性修改，从而阻止你再次插入到源端数据库。

Oracle Database Release	缺省值	最小值	最大值
Oracle Database 19c	19.0.0	11.2.0	The COMPATIBLE parameter should not be changed for an RU or an RUR, either for CDB or Non-CDB instances
Oracle Database 12c Release 2 (12.2)	12.2.0	11.2.0	12.2.0
Oracle Database 12c Release 1 (12.1)	12.0.0	11.0.0	12.1.0
Oracle Database 11g Release 2 (11.2)	11.2.0	10.0.0	11.2.0

参见Oracle Database Compatibility 在线文档。

所以，随着升级成功和测试验证完成，应该调整兼容参数。兼容性参数不会影响优化器的行为，优化器行为是通过 optimizer_features_enable 参数来控制的。

修改 compatible 参数

```
SQL> show parameter compatible
SQL> alter system set compatible='19.0.0' scope=spfile;
SQL> show spparameter compatible
```

SQL> shut immediate
SQL> startup
SQL> show parameter compatible

```
SQL> show parameter compatible

NAME                                 TYPE        VALUE
------------------------------------ ----------- ------------------------------
compatible                           string      11.2.0.4.0
noncdb_compatible                    boolean     FALSE
SQL> alter system set compatible='19.0.0' scope=spfile;

System altered.

SQL> show spparmeter compatible
SP2-0158: unknown SHOW option "spparmeter"
SP2-0158: unknown SHOW option "compatible"
SQL> show spparameter compatible

SID      NAME                          TYPE        VALUE
-------- ----------------------------- ----------- ------------------------------
*        compatible                    string      19.0.0
*        noncdb_compatible             boolean
SQL> shut immediate
Database closed.
Database dismounted.
ORACLE instance shut down.
SQL> startup
ORACLE instance started.

Total System Global Area 1073737800 bytes
Fixed Size                  8904776 bytes
Variable Size             469762048 bytes
Database Buffers          587202560 bytes
Redo Buffers                7868416 bytes
Database mounted.
Database opened.
SQL> show parameter compatible

NAME                                 TYPE        VALUE
------------------------------------ ----------- ------------------------------
compatible                           string      19.0.0
noncdb_compatible                    boolean     FALSE
SQL>
```

查看下升级后的组件和版本信息

set line 200 pages 100
col comp_id format a8
col comp_name format a34
col schema format a12
col status format a10
col version format a12
col con_id format 99
select con_id, comp_id, comp_name, schema, status, version from cdb_registry order by 1,2;

```
SQL> set line 200 pages 100
col comp_id format a8
col comp_name format a34
col schema format a12
col status format a10
col version format a12
col con_id format 99
select con_id, comp_id, comp_name, schema, status, version from cdb_registry order by 1,2;
SQL> SQL> SQL> SQL> SQL> SQL> SQL>
CON_ID COMP_ID  COMP_NAME                      SCHEMA       STATUS     VERSION
------ -------- ------------------------------ ------------ ---------- ------------
     0 CATALOG  Oracle Database Catalog Views  SYS          VALID      19.0.0.0.0
     0 CATPROC  Oracle Database Packages and Types SYS      VALID      19.0.0.0.0
     0 OLS      Oracle Label Security          LBACSYS      VALID      19.0.0.0.0
     0 OWM      Oracle Workspace Manager       WMSYS        VALID      19.0.0.0.0
     0 RAC      Oracle Real Application Clusters SYS        OPTION OFF 19.0.0.0.0
     0 XDB      Oracle XML Database            XDB          VALID      19.0.0.0.0

6 rows selected.
```

修改/etc/oratab

将 UPGR:/u01/app/oracle/product/11.2.0.4:Y
修改为
UPGR:/u01/app/oracle/product/19:Y

```
UPGR:/u01/app/oracle/product/19:Y
```

升级 Recovery Catalog

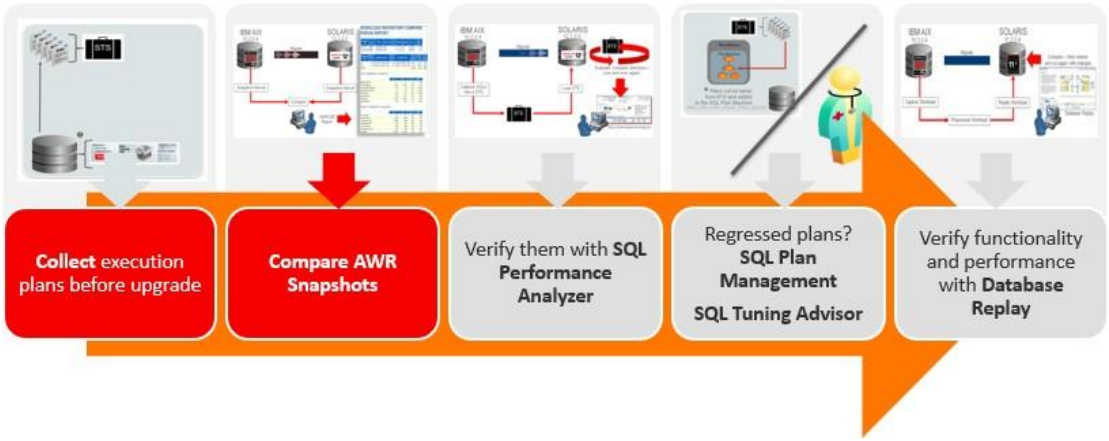
本 workshop 中不涉及这个练习，这里是给大家介绍一下。
如果使用的 Recovery catalog 版本低于 RMAN 客户端的版本，我们必须升级它。可以通过命令 UPGRADE CATALOG 来升级 Recovery Catalog。如果要升级 Recovery catalog 到 Oracle database 12.2 或更高版本，那么必需首先调用 dbmsrmansys.sql 来管理 Recovery catalog 权限，此外，如果你使用了 virtual private catalogs，还必须运行 dbmsrmanvpc.sql 升级 Virtual Private Catalogs。 然后执行如下脚本：
$ rman catalog username/password@alias
RMAN> UPGRADE CATALOG;
RMAN> UPGRADE CATALOG;
RMAN> EXIT;

更多信息，请参照 Oracle 文档中的 "Upgrading the Recovery Catalog" 部分。

升级前后性能对比

性能评估-AWR Diff Report

在这个练习中，我们将创建 AWR 比较报告。这些报告为您提供了可能遇到的问题(或性能改进)的直观指示。通过比较持续时间大致相同的负载情况，可以比较容易看出升级前后的性能变化。



首先，在加载压力测试前创建一个 AWR 快照，然后加载负载，最后在创建一个 AWR 快照，并再次记录下快照编号。这个执行过程和在升级前模拟应用负载测试类似。

生产 AWR 快照

$. upgr19
@/home/oracle/scripts/snap.sql

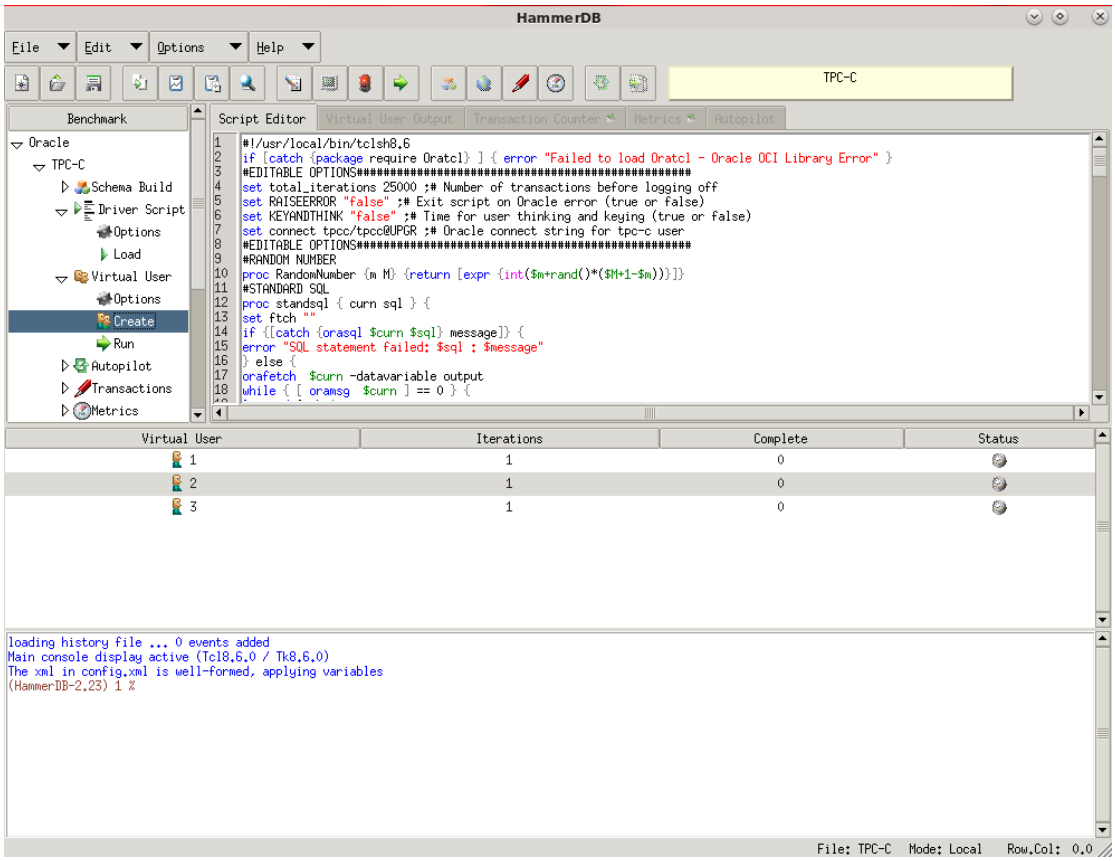
- AWR Snapshot with Snap-ID: 97 created. -

SQL>

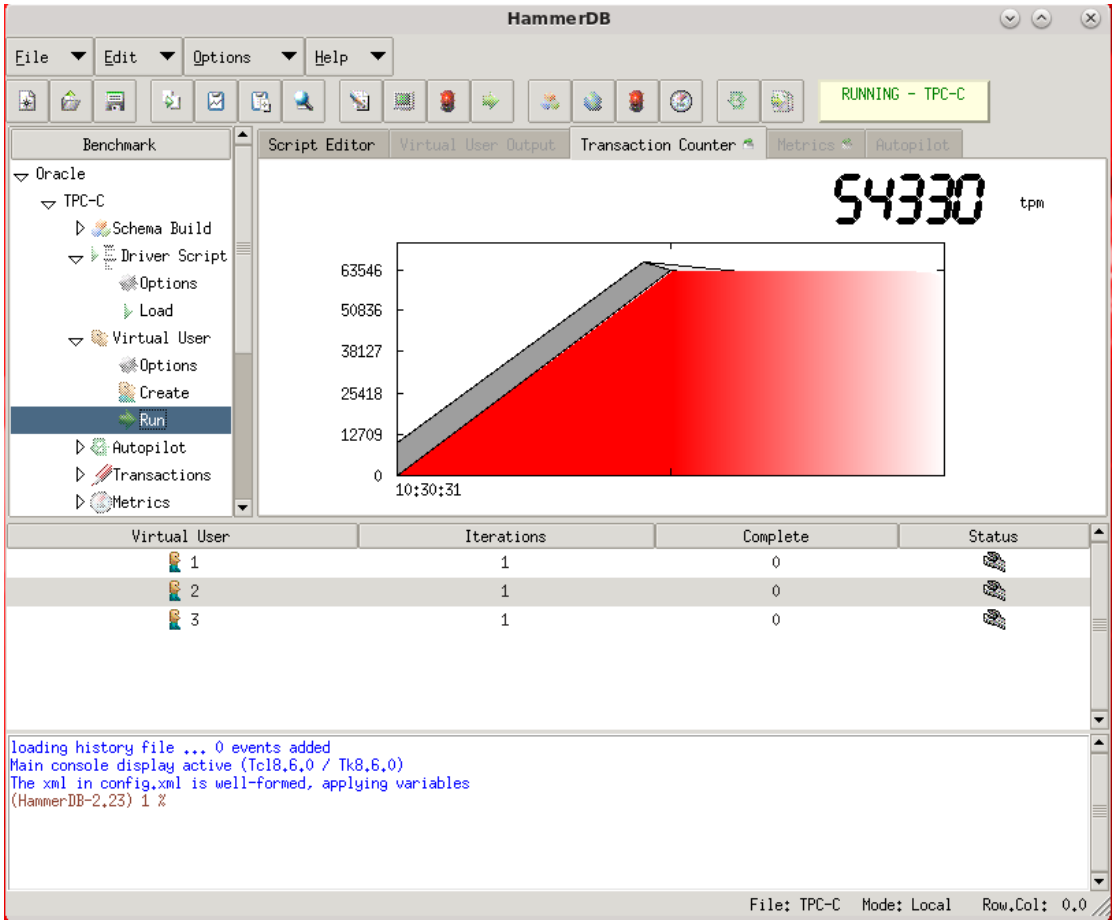
```
SQL> start /home/oracle/scripts/snap.sql
-----------------------------------------
- AWR Snapshot with Snap-ID: 97 created. -
-----------------------------------------
SQL>
```

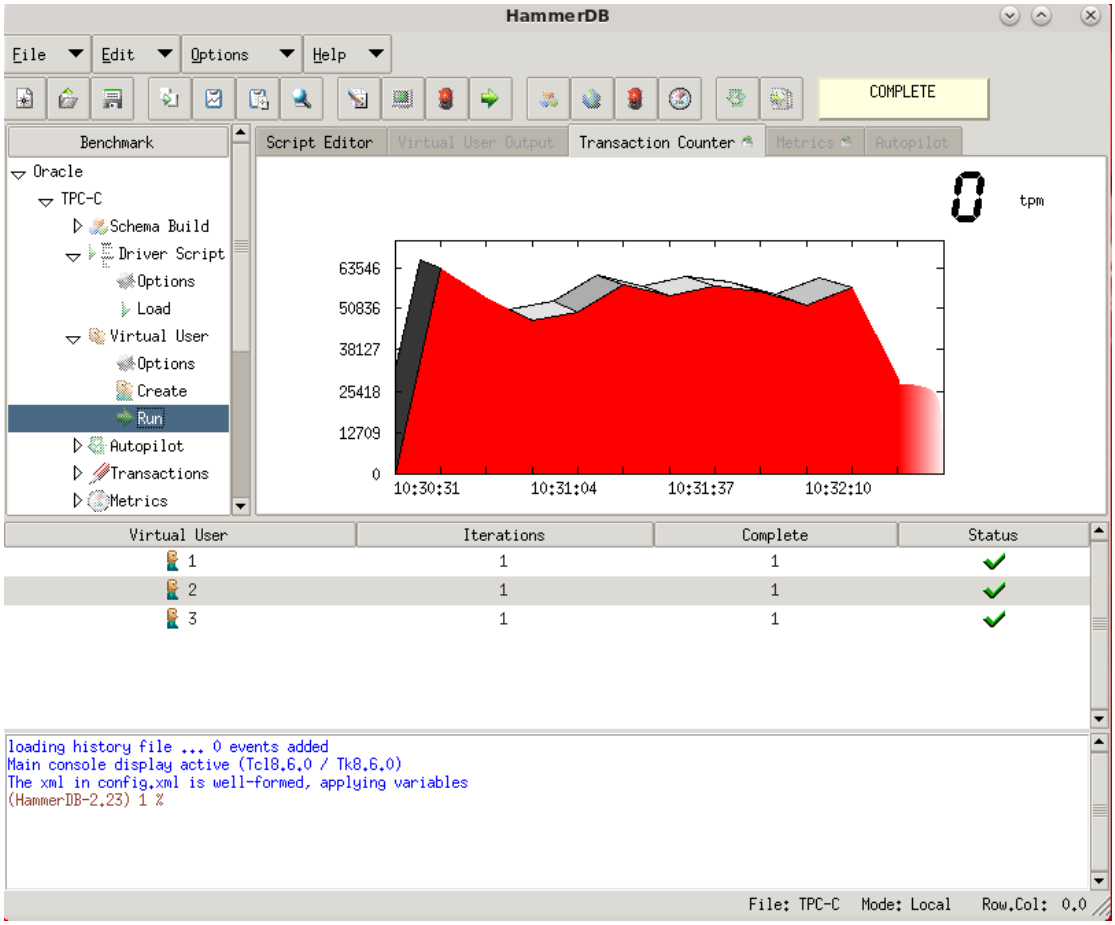
运行 HammerDB 模拟负载

双击桌面上的 HammerDB 图标，启动 HammerDB，然后点击展开 TPC-C，再点击展开 DriverScript，双击 Load 选项。然后再点击展开 Virtual User，双击 Create，应该会看到 3 个虚拟用户，如下图所示



如下图，双击 run，并点击用红色做标记的按钮，能看到 tpm 的信息。


HammerDB
File
Edit
Options
Help
RUNNING - TPC-C
Benchmark
Oracle
TPC-C
Schema Build
Driver Script
Options
Load
Virtual User
Options
Create
Run
Autopilot
Transactions
Metrics
Script Editor
Virtual User Output
Transaction Counter
Metrics
Autopilot
54330
tpm
63546
50836
38127
25418
12709
0
10:30:31
Virtual User
Iterations
Complete
Status
1
1
0
2
1
0
3
1
0
loading history file ... 0 events added
Main console display active (Tcl8.6.0 / Tk8.6.0)
The xml in config.xml is well-formed, applying variables
(HammerDB-2.23) 1 %
File: TPC-C
Mode: Local
Row.Col: 0.0



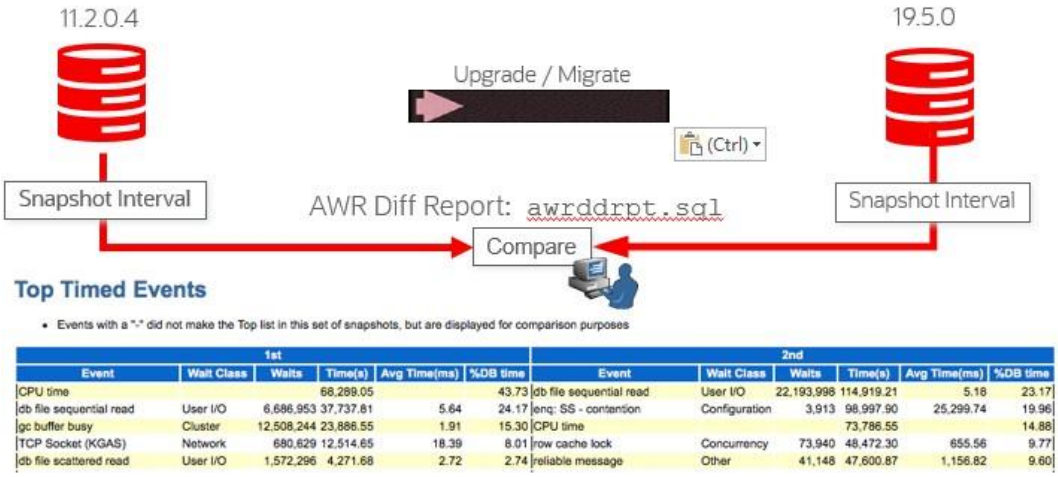
再次生成 AWR

@/home/oracle/scripts/snap.sql

```
SQL> start /home/oracle/scripts/snap.sql
-----------------------------------------
- AWR Snapshot with Snap-ID: 98 created. -
-----------------------------------------
SQL>
```

生成 AWR 比较报告

我们将比较升级前的快照周期与升级后的快照周期



. upgr19

cd /home/oracle/scripts

SQL> **@?**/rdbms/admin/awrddrpt.sql

交互内容	升级前快照周期	升级后快照周期
report_type	用缺省的 html，直接回车	
快照时间范围 num_days	1	1
begin_snap	根据练习时记录的快照编号	根据练习时记录的快照编号
end_snap	根据练习时记录的快照编号	根据练习时记录的快照编号
report_name	awrdiff 报告生成在启动 SQL*Plus 的当前目录下	

```
$ . upgr19
[UPGR] oracle@hol:/u01/app/oracle/product/19/rdbms/admin
$ sqlplus / as sysdba

SQL*Plus: Release 19.0.0.0.0 - Production on Tue Dec 17 10:36:14 2019
Version 19.3.0.0.0

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


Connected to:
Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
Version 19.3.0.0.0

SQL> start ?/rdbms/admin/awrddrpt.sql
```

注意升级前和升级后，相同时段相同负载的显著差异。也可以浏览 SQL 统计数据，看看是否发现这两次运行之间的显著差异。

WORKLOAD REPOSITORY COMPARE PERIOD REPORT

Report Summary

Snapshot Set	DB Name	DB Id	Unique Name	DB Role	Edition	Release	Cluster	CDB	Host	Std Block Size
First (1st)	UPGR	72245725				11.2.0.4.0	NO	NO	hol.localdomain	8192
Second (2nd)	UPGR	72245725	UPGR	PRIMARY	EE	19.0.0.0.0	NO	NO	hol.localdomain	8192

Snapshot Set	Instance	Inst num
First (1st)	UPGR	1
Second (2nd)	UPGR	1

Snapshot Set	Begin Snap Id	Begin Snap Time	End Snap Id	End Snap Time	Avg Active Users	Elapsed Time (min)	DB time (min)
1st	90	17-Dec-19 05:14:21 (Tue)	91	17-Dec-19 05:37:51 (Tue)	0.2	23.5	5.6
2nd	97	17-Dec-19 10:26:48 (Tue)	98	17-Dec-19 10:33:38 (Tue)	0.8	6.8	5.4
%Diff					*229.2*	*-70.9*	*-3.6*

Host Configuration Comparison

	1st	2nd	Diff	%Diff
Number of CPUs:	4	4	0	0.0
Number of CPU Cores:	4	4	0	0.0
Number of CPU Sockets:	1	1	0	0.0
Physical Memory:	7708.6M	7708.6M	0M	0.0
Load at Start Snapshot:	.63	.96	.33	52.4
Load at End Snapshot:	.28	1.12	.84	300.0
%User Time:	3.85	13.52	9.66	251.2
%System Time:	.62	1.46	.84	135.5
%Idle Time:	95.34	84.51	-10.83	-11.4
%IO Wait Time:	.88	2.7	1.82	206.8

Cache Sizes

	1st (M)	2nd (M)	Diff (M)	%Diff
Memory Target				
SGA Target	776.0	1,024.0	248.0	*32.0*
Buffer Cache	548.0	560.0	12.0	*2.2*
Shared Pool	200.0	444.0	244.0	*122.0*
Large Pool	12.0	4.0	-8.0	*-66.7*
Java Pool	4.0		-4.0	*-100.0*
Streams Pool				

Top Timed Events

- Events with a "-" did not make the Top list in this set of snapshots, but are displayed for comparison purposes

1st						2nd					
Event	Wait Class	Waits	Time(s)	Avg Time	%DB time	Event	Wait Class	Waits	Time(s)	Avg Time	%DB time
CPU time			200.81		60.00	CPU time			206.13		63.88
log file sync	Commit	68,478	102.62	1.50ms	30.66	log file sync	Commit	68,356	90.54	1.32ms	28.06
log file parallel write	System I/O	69,329	72.43	1.04ms	21.64	log file parallel write	System I/O	78,745	82.11	1.04ms	25.45
db file async I/O submit	System I/O	458	47.69	104.12ms	14.25	db file async I/O submit	System I/O	260	42.02	161.63ms	13.02
log file switch (checkpoint incomplete)	Configuration	25	14.73	589.26ms	4.40	enq: TX - row lock contention	Application	6,965	12.83	1.84ms	3.98
enq: TX - row lock contention	Application	6,516	13.57	2.08ms	4.05	log file switch (checkpoint incomplete)	Configuration	26	12.70	488.44ms	3.94
control file parallel write	System I/O	693	1.92	2.77ms	0.57	LGWR any worker group	Other	19,057	12.41	651.02us	3.84
db file sequential read	User I/O	5,402	1.84	341.38us	0.55	LGWR worker group ordering	Other	8,425	5.23	620.67us	1.62
db file scattered read	User I/O	3,760	1.50	399.31us	0.45	LGWR all worker groups	Other	4,684	4.17	890.75us	1.29
db file parallel read	User I/O	2,063	1.20	583.07us	0.36	log file switch completion	Configuration	30	1.22	40.66ms	0.38
-log file switch completion	Configuration	23	0.86	37.52ms	0.26	-control file parallel write	System I/O	383	0.78	2.03ms	0.24
-						-db file sequential read	User I/O	7,447	0.32	43.06us	0.10
-						-db file scattered read	User I/O	4,718	0.16	33.59us	0.05
-						-db file parallel read	User I/O	5	0.00	11.40us	0.00

SQL Statistics

- Top SQL Comparison by Elapsed Time
- Top SQL Comparison by CPU Time
- Top SQL Comparison by I/O Time
- Top SQL Comparison by Buffer Gets
- Top SQL Comparison by Physical Reads
- Top SQL Comparison by UnOptimized Read Requests
- Top SQL Comparison by Executions
- Top SQL Comparison by Parse Calls
- Top SQL Comparison by Sharable Memory
- Top SQL Comparison by Version Count
- Top SQL Comparison by Cluster Wait Time

Back to Top

Top SQL Comparison by Elapsed Time

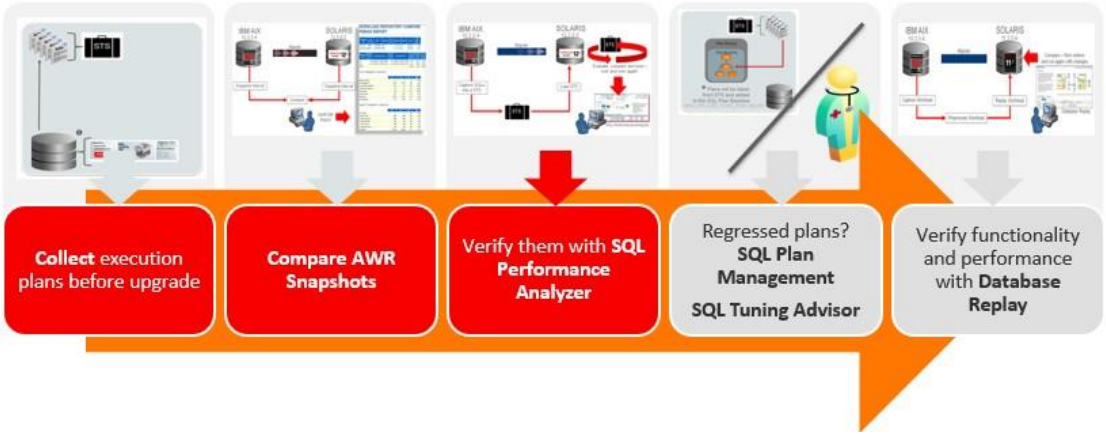
- Ordered by absolute value of 'Diff' column of 'Elapsed Time % of DB time' descending
- '#Plans' column indicates the number of distinct execution plans for the statement in 1st and 2nd periods and in Both periods combined
- '1st Total' and '2nd Total' show respective running totals for '1st' and '2nd' columns of 'Elapsed Time % of DB time'
- DB time First: 334.67 seconds, Second: 322.68 seconds
- Captured SQL Elapsed Time First: 175.40 seconds, Second: 170.77 seconds
- Captured SQL Elapsed Time as a % of DB time First: 52.41%, Second: 52.92%
- Captured PL/SQL Elapsed Time as a % of DB time First: 68.67%, Second: 69.89%
- Common SQL Elapsed Time as a % of DB time First: 118.26%, Second: 121.9%

	Elapsed Time % of DB time					Elapsed Time (ms) per Exec		#Exec/sec (DB time)		CPU Time (ms) per Exec		I/O Time (ms) per Exec		Physical Reads per Exec		#Rows Processed per Exec		#Executions		#Plans	
SQL Id	1st	1st Total	2nd	2nd Total	Diff	1st	2nd	1st	2nd	1st	2nd	1st	2nd	1st	2nd	1st	2nd	1st	2nd	1st/2nd/Both	SQL Text
aw9ttz9acxbc3	28.46	28.46	31.05	31.05	2.60	3	3	96.78	101.47	2	2	0	0	0.22	0.21	1.00	1.00	32,389	32,743		BEGIN payment(:p_w_id, :p_d_id...
5hyh0360hqx2u	2.32	30.78	1.48	32.54	-0.84	2	1	9.55	10.19	2	1	0	0	0.26	0.40	1.00	1.00	3,196	3,288		BEGIN slev(:st_w_id, :st_d_id,...
d4ujh5yqt1fph	5.33	36.11	6.16	38.69	0.83	6	6	9.64	10.01	4	5	0	0	0.80	0.77	1.00	1.00	3,227	3,230		BEGIN delivery(:d_w_id, :d_o_c...
dqvz792jar7w0	1.96	38.07	2.58	41.28	0.62	0	0	983.05	1,003.77	0	0	0	0	0.00	0.00	1.00	1.00	328,995	323,894		INSERT INTO ORDER_LINE (OL_O_I...
4wq725nwpxb1z	8.86	46.94	9.26	50.53	0.39	2	2	58.26	60.47	1	1	0	0	0.00	0.00	7.31	7.17	19,499	19,511	1/ 1/ 2	SELECT C_FIRST, C_MIDDLE, C_ID...
g5u7xuchhfu62	4.90	51.84	5.22	55.75	0.32	1	1	96.76	101.46	0	0	0	0	0.00	0.00	1.00	1.00	32,383	32,738	1/ 1/ 1	UPDATE WAREHOUSE SET W_YTD = W...
16dhat4ta7xs9	20.28	72.11	20.53	76.28	0.25	2	2	98.20	100.41	1	2	0	0	0.44	0.43	1.00	1.00	32,863	32,400		begin neword(:no_w_id, :no_max,...
7m6h0wf6stq6q	10.13	82.24	10.25	86.54	0.13	2	2	64.19	66.70	1	1	0	0	0.34	0.34	1.00	1.00	21,481	21,521	1/ 1/ 2	SELECT COUNT(C_ID) FROM CUSTOM...
csv0xdm9c394t	8.22	90.46	8.30	94.83	0.08	8	8	9.91	10.32	8	8	0	0	0.23	0.29	0.84	0.91	3,317	3,330	1/ 1/ 1	SELECT O_ID, O_CARRIER_ID, O_E...
qcvvtwcxafqp5	10.59	101.05	10.66	105.50	0.07	11	10	9.91	10.32	10	10	0	0	0.94	1.25	1.00	1.00	3,317	3,330		BEGIN ostat(:os_w_id, :os_d_id...
82tfppg8s0dc2	2.99	104.04	3.05	108.55	0.07	0	0	983.05	1,003.75	0	0	0	0	0.00	0.00	1.00	1.00	328,997	323,889	1/ 1/ 1	UPDATE STOCK SET S_QUANTITY = ...

在本试验中，可能不会看到任何明显的区别，因为这个实验的环境、负载都完全一致，我们的目的就是让您了解并记住，将来不论是在测试环境、还是真实的升级/迁移场景中有类似的困惑时，通过负载测试，然后比对 AWR Diff 报告可以快速看出变更前后的性能变化。同时你也会发现，生成 AWR Diff 报告还是非常容易的。

性能分析- SQL Performance Analyzer

在这一部分呢中，我们将熟悉并使用 SQL Performance Analyzer(简称 SPA)再次进行评估，比较和优化，SPA 是 Real Application Testing(简称 RAT)的一部分。通过比较升级前收集的语句与升级后模拟测试的语句，来进行性能分析。
使用 SQL Performance Analyzer 也是我们建议的整个升级过程中测试工作流的第三部分。



关于 Real Application Testing

Oracle 数据库是全球数十万企业以及应用程序开发人员和数据库管理员首选的市场领先数据库。多年来，企业渐已依赖 Oracle 数据库提供无与伦比的性能和可靠性。如今，数据中心环境正在迅速发展和不断变化以便跟上业务之需要，Oracle Database 旨在为此提供助力，它让企业可以迅速采用新技术，同时尽可能减少风险。

当基础设施发生变更的时候，很多企业往往需要为整个应用程序体系投入重复的硬件，以便对生产应用程序进行测试，确保变更后的功能和性能，尤其是性能问题。尽管很多企业为此付出了大量成本，执行了许多测试，但最终在生产系统中实施变更时还是常常会遇到意料之外的问题。这其中的一个主要原因是因为测试负载往往是模拟的，不能准确或完整地表示实际的生产负载。因此，数据中心管理者往往不愿意通过采用新技术和调整业务来适应快速变化的竞争压力，这也是大家不愿意升级的一个关键原因。
Oracle 数据库 Real Application Testing 选件就是针对这样的挑战，迎面解决这些困惑的。
Real Application Testing 由两大部分组成：
（1）SQL Performance Analyzer
（2）Database Replay

SQL Performance Analyzer

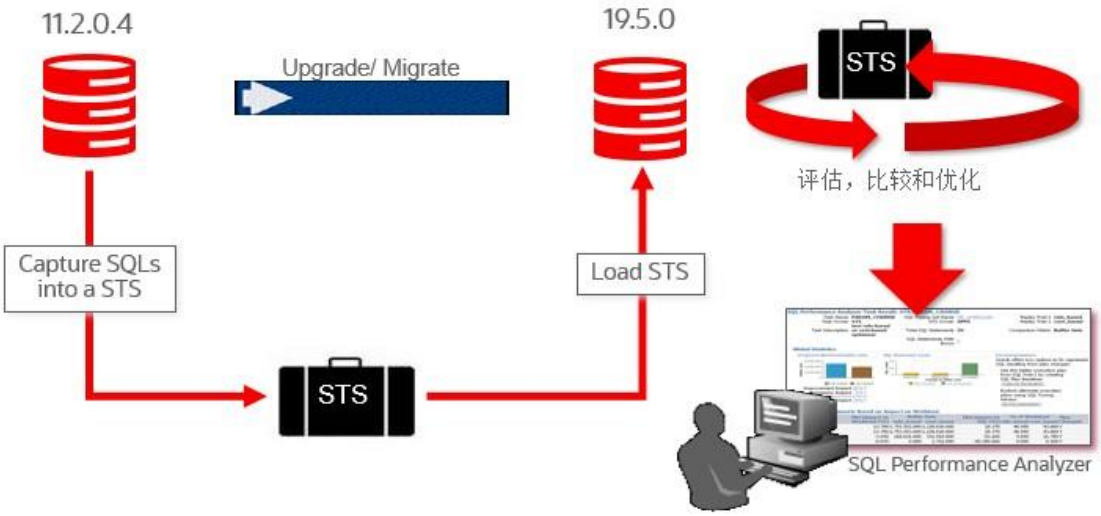
影响应用程序的性能和可用性的一个关键点是变更操作（常见的系统变更操作有：数据库升级、打补丁、参数调整、操作系统/硬件/数据配置变更、数据模型变化（新建表、分区、索引、物化视图等等)、架构变化（比如从 non-CDB 到 CDB 整和）等等。）可能会导致 SQL 执行计划发生变化。因此 DBA 需要付出大量的精力来识别和修复因为系统变更而导致性能下降的 SQL 语句。SQL Performance Analyzer(SPA)可以预测和防止环境更改导致的 SQL 执行性能问题。

SQL Performance Analyzer 通过在系统变更前后顺次运行 SQL 语句，提供环境更改对 SQL 执行计划和统计信息产生的影响的精细视图。生成的 SQL Performance Analyzer 报告会列出系统更改对负载的影响以及一组性能下降的 SQ 语句。对于性能下降的 SQL 语句，会提供相应的执行计划细节和建议的调优方法。

SQL Performance Analyzer 与现有的 SQL Tuning Sets(STS)、SQL Tuning Advisor 和 SQL Plan Management 功能进行了很好的集成。SQL Performance Analyzer 完全自动化和简化了耗时的手动评估更改对特大型 SQL 负载（数千个 SQL 语句）之影响的过程。DBA 可以在测试环境中使用 SQL Tuning Advisor 修复性能下降的 SQL 语句，并生成新计划。这些计划随后被植入到 SQL Plan Management 基准中，然后导出回到生产环境中。这样，通过使用 SQL Performance Analyzer，企业能以极低的成本、极强的信心验证对生产环境的系统更改实际上会产生正面影响。
使用 SQL Performance Analyzer 功能可以通过 DBMS_SQLPA API 或 Oracle Enterprise Manager 来完成。一般涉及到如下 5 个主要步骤

- 捕获要使用 SPA 进行分析的 SQL 负载
  比如前面练习中提到的从 AWR 或 Cursor Cache 中捕获负载到 SQL Tuning Sets (STS) 中这一步在实际环境中通常在生产系统上完成，然后可以将 STS 发送到测试系统中以便执行 SPA 分析。
- 在测试或目标系统中对 STS 执行 SPA
  衡量更改前负载的性能。多次执行运行时间极短的查询，然后对获得的统计信息进行平均，以消除缓冲区缓存状态和其他噪声因素引起的偏差。
- 进行变更操作
  比如数据库升级

- 再次对 STS 执行 SPA
  衡量更改后负载的性能
- 比较两次执行 SQL Tuning Sets 的性能
  以识别出哪些 SQL 语句出现性能下降、性能提高或没有变化。如果遇到任何性能下降，SPA 允许用户使用 SQL Tuning Advisor 或借助 SQL Plan Baselines 对其进行修复

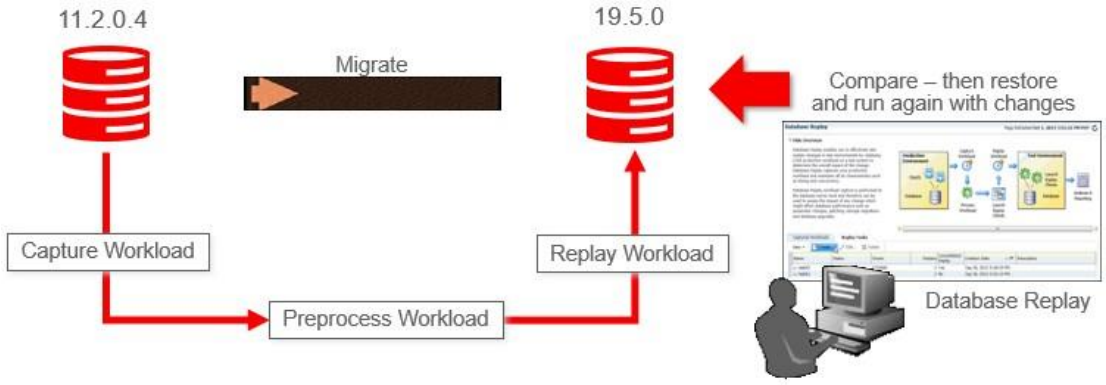


Database Replay

利用数据库重放，DBA 和系统管理员可以在测试环境内如实、准确、逼真地重新运行实际生产负载，其中包括联机用户负载和批处理负载。数据库重放可从生产系统中捕获全部数据库负载（包括所有并发性、相关性和时间性），使您能够从根本上在测试系统中重建生产负载（依赖模拟方法的传统测试工具永远不能重现这样的负载），对系统更改进行逼真的测试。从可以降低测试基础架构的成本，加快部署。

数据库重放包含 4 个主要步骤:
（1）负载捕获
启用负载捕获后，定向到 Oracle 数据库的所有外部客户端请求都被跟踪并存储在数据库服务器主机文件系统中的二进制文件（称为捕获文件）中。Oracle 建议在执行负载捕获之前备份整个数据库。用户指定捕获文件的位置以及负载捕获的开始时间和结束时间。在此过程中，关于外部数据库调用的所有信息都写入捕获文件。
（2）负载处理捕
获负载之后，需要对捕获文件中的信息进行处理。该处理过程将捕获的数据转换为重放文件，并创建重放负载所需的所有必要元数据。通常将捕获文件复制到其他系统进行处理。对于每个捕获的负载，必须进行一次这样的处理，之后才可以进行重放。捕获的负载经过处理后，可以在重放系统上反复重放。
由于负载处理可能既耗时又占用不少资源，因此通常建议在即将重放负载的测试系统上执行该步骤。
（3）负载重放
捕获的负载经过处理之后就可以重放了。一个称为重放客户端的客户端程序处理重放文

件，然后以完全等同于捕获系统中的时间性和并发性提交对数据库的调用。根据捕获的负载，您可能需要一个或多个重放客户端来适当地重放负载。我们提供了一个校准工具来帮助确定负载所需的重放客户端数量。
应注意，由于重放整个负载（包括 DML 和 SQL 查询），因此重放系统中的数据需要与捕获负载的生产系统中的数据完全相同，以便为生成报告而执行可靠的分析。
（4）分析和报告
该步骤提供大量报告以便对捕获和重放进行详细的分析。将会报告重放过程中出现的任何错误。显示 DML 或查询返回的行中的任何差异。提供捕获与重放之间的基本性能对比。并且为进行高级分析而提供重放比较时段报告和其他 AWR 报告，以便于详细比较捕获与重放之间的各项统计信息。



利用 Oracle Database 中的 Real Application Testing 功能，数据库管理员可以轻松应付变化，同时消除任何不想要的负面影响。

通过 SPA 进行性能分析

在前面的升级前性能基线构建 Lab 中，我们已经收集了升级前模拟负载测试 SQL Tuning Sets 到如下 2 个 STS:

- STS_CaptureAWR
- STS_CaptureCursorCache

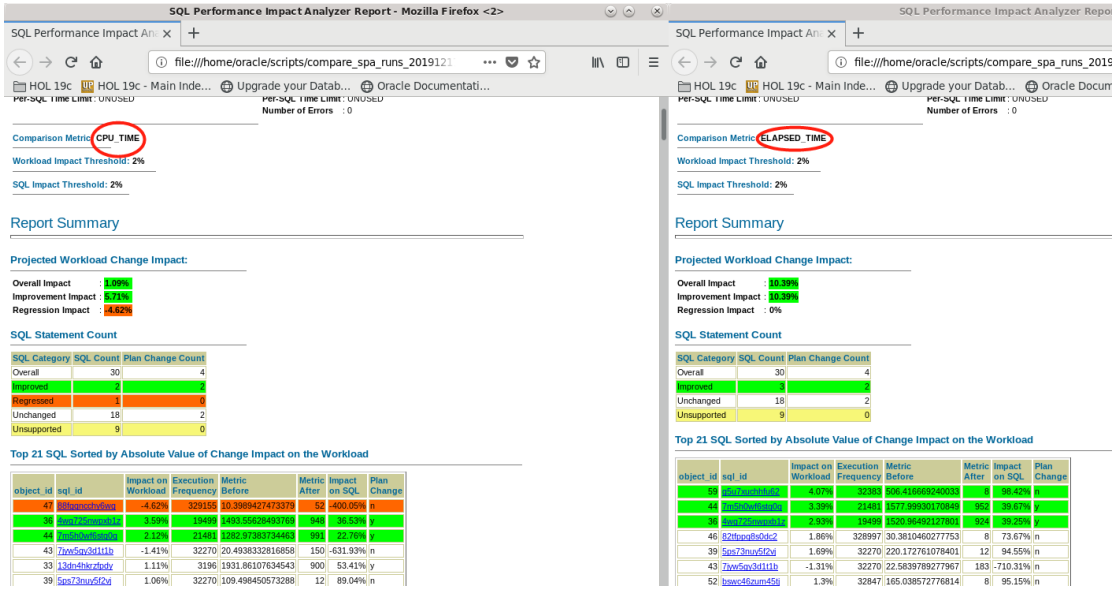
```
SQL> col sqlset_name for a30
SQL> select count(*), sqlset_name from dba_sqlset_statements group by sqlset_name order by 2;

  COUNT(*) SQLSET_NAME
---------- ------------------------------
        30 STS_CaptureAWR
        41 STS_CaptureCursorCache

SQL>
```

本次实验中，我们将通过 CPU_TIME 和 ELAPSED_TIME 这两个不同的比较指标两次模拟。
调用如下脚本启动 CPU_TIME 的初始运行
. upgr19
cd /home/oracle/scripts
@/home/oracle/scripts/spa_cpu.sql

然后生成包含结果的 HTML 报告
@/home/oracle/scripts/spa_report_cpu.sql

调用如下脚本启动 ELAPSED_TIME 的初始运行
@/home/oracle/scripts/spa_elapsed.sql

然后生成包含结果的 HTML 报告
@/home/oracle/scripts/spa_report_elapsed.sql
exit
$ pwd
/home/oracle/scripts
[UPGR] oracle@hol:~/scripts
$ ls -ltr compare*
-rw-r--r--. 1 oracle dba 297737 Dec 17 11:23 compare_spa_runs_20191217112318.html
-rw-r--r--. 1 oracle dba 297908 Dec 17 11:27 compare_spa_runs_20191217112731.html
[UPGR] oracle@hol:~/scripts
$

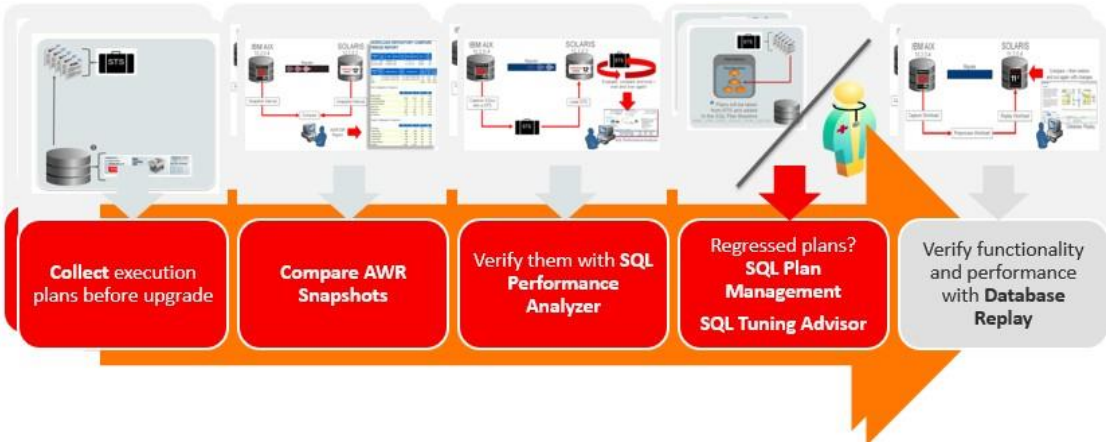


您可能也看到了有些 SQL 语句的执行计划发生了变化，有些没有。有的性能没啥变化，有的有了提升，提升幅度也不尽相同。在实际的生产环境中，可能会看到有些 SQL 的性能会有所回退，也别担心，我们可以通过其他技术对这些 SQL 进行优化，使其具有更好的性能。

这里我们可以重点关注下执行计划发生变化的 SQL_ID,比如我这里是 13dn4hkrzfpdy，点击链接向下查看比较前后的详细执行信息。然后在接下来的 SQL Plan Management 练习中进一步探索。

SQL 执行计划管理

在前一节中，我们在 SQL Performance Analyzer 中发现了有些 SQL 语句的执行计划发生了改变（性能不一定下降）。现在我们可以使用 SQL Plan Management 来修复计划。同时大家也要明白，通过 SPM 来管理执行计划，不是非用不可的。

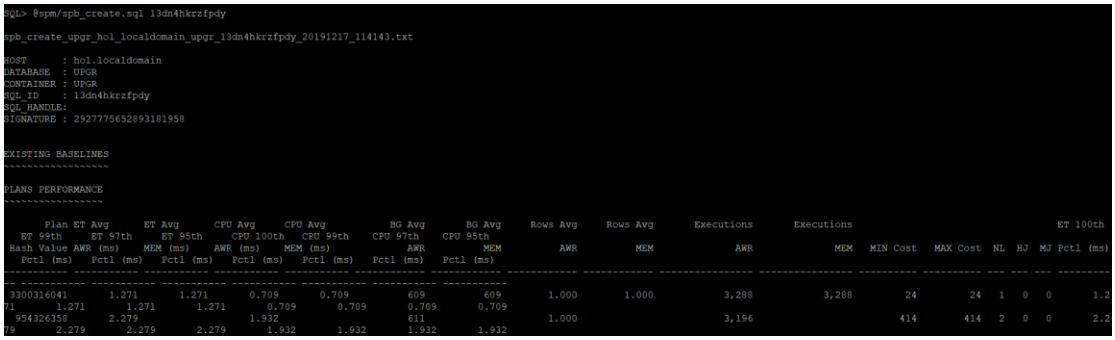


通过 SQL Performance Analyzer 报告可以看出升级到 19c 后，总体运行效果是好的，当然了也有一些 SQL 的执行计划发生了改变（计划改变不一定就效率低），我们本节试验通过尝试修复一个已经改变的具体计划，或将 SQL Tuning Sets 中的所有执行计划都写到 SQL 执行计划基线中，当然了 optimizer 也可能会找到更好的执行路径，来看看效果是否好。本节试验的目的是回顾下 Oracle SPM 针对这种情况的一个应对方案。

修复单个 SQL

```
SQL> start /home/oracle/scripts/spb_create.sql

PL/SQL procedure successfully completed.


PL/SQL procedure successfully completed.


1. Enter SQL_ID (required)
Enter value for 1: 13dn4hkrzfpdy
```

通过比较，我们可能会发现升级以后第一个执行计划效果会更加，所以我们接受它作为我们希望用于将来执行 SQL_ID: 13dn4hkrzfpdy 语句的计划来修复它。按提示依次输入优先顺序的计划的 Plan Hash Value(3300316041)，然后回车确认。

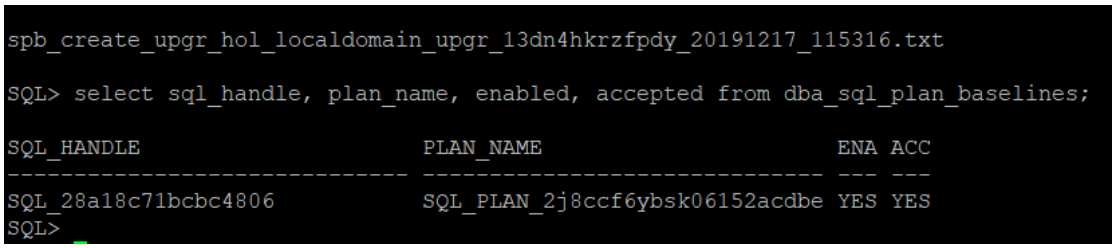


SQL> select sql_handle, plan_name, enabled, accepted from dba_sql_plan_baselines;

SQL_HANDLE PLAN_NAME ENA ACC

SQL_28a18c71bcbc4806 SQL_PLAN_2j8ccf6ybsk06152acdbe YES YES

SQL>

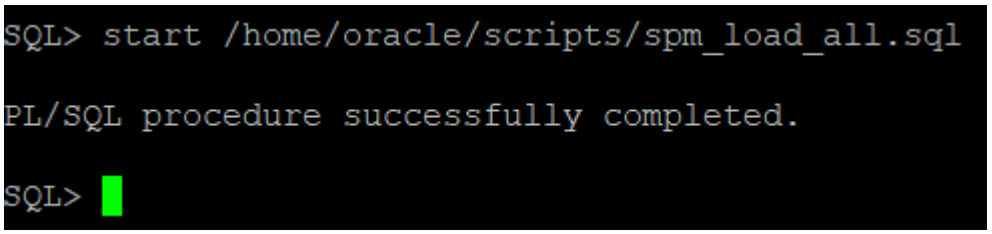


如果你想了解下“为什么这个计划改变了?”，请点击这里阅读Franck Pachot 写的一篇博文。

修复所有 SQL 语句

现在，我们将 SQL Tuning Sets STS_CaptureCursorCache 中收集的所有可能语句写到执行计划基线，并再次使用 SQL Performance Analyzer 验证效果

@/home/oracle/scripts/spm_load_all.sql



修复计划后，使用 SQL Performance Analyzer 验证计划和性能

注：因为 spa_*.sql 之前脚本中是基于 STS_CaptureAWR 调优集的，现在我们修改 spa_cpu.sql、spa_elapsed.sql，将其中 STS_CaptureAWR 替换 STS_CaptureCursorCache。

执行如下 sed 命令进行替换：

sed -i "s/STS_CaptureAWR/STS_CaptureCursorCache/g" spa_cpu.sql

sed -i "s/STS_CaptureAWR/STS_CaptureCursorCache/g" spa_elapsed.sql

sqlplus / as sysdba

@/home/oracle/scripts/spa_cpu.sql

@/home/oracle/scripts/spa_report_cpu.sql
@/home/oracle/scripts/spa_elapsed.sql
@/home/oracle/scripts/spa_report_elapsed.sql
exit

```
[UPGR] oracle@hol:~/scripts
$ ls -ltr compare*
-rw-r--r--. 1 oracle dba 297737 Dec 17 11:23 compare_spa_runs_20191217112318.html
-rw-r--r--. 1 oracle dba 297908 Dec 17 11:27 compare_spa_runs_20191217112731.html
-rw-r--r--. 1 oracle dba 428048 Dec 17 12:02 compare_spa_runs_20191217120228.html
-rw-r--r--. 1 oracle dba 428076 Dec 17 12:02 compare_spa_runs_20191217120244.html
[UPGR] oracle@hol:~/scripts
```

在 19c 中运行时效果更差？那结论时什么呢？
允许新版本环境下让 Oracle 自己找到新的、更好的执行计划。哪怕是非常关键的语句，尽管之前是稳定的，也应该允许从升级后更好的性能中获益。所以，测试的核心思想是识别真正糟糕的语句和计划，并修复它们，但绝不是全部。

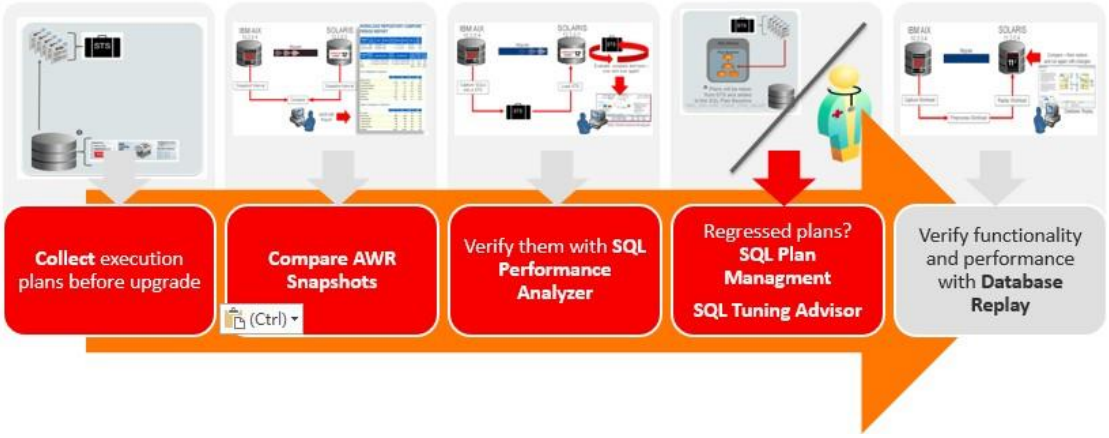
Note：对于 Exadata 平台上的 19c 数据库，SQL Plan Management 是可以完全自动的，不需要 DBA 去管理。

SQL 执行计划的相关资源

为了更好的理解和使用 SPM 特性，我这里提供了一些链接，供大家参考使用。
MOS Note: 789888.1 How to Load SQL Plans into SQL Plan Management (SPM) from the Automatic Workload Repository (AWR)
MOS Note: 456518.1 How to Use SQL Plan Management (SPM) – Plan Stability Worked Example
白皮书：SQL Plan Management with Oracle Database 12c Release 2
博文：Carlos Sierra: Plan Stability

通过 SQL Tuning Advisor 优化

在前一节中，使用 SQL 计划管理修复了计划。接下来让我们通过询问 SQL Tuning Advisor (STA)看看还可以做什么



生成优化建议

我们把在前面从 Cursor Cache 中捕获的 SQL Tuning Set 传递给 SQL Tuning Advisor 执行并检查结果

. upgr19

cd /home/oracle/scripts

sqlplus / as sysdba

@/home/oracle/scripts/sta_cc.sql

该脚本将完成如下工作:

1）生成一个通过 SQL 调优集 STS_CaptureCursorCache 的调优任务

2）运行一个调优 SQL 调优顾问模拟执行任务

3）以文本格式生成结果报告

4）生成语句来实施报告建议

```
SQL> start /home/oracle/scripts/sta_cc.sql
SQL Tuning Task does not exist - will be created ...

PL/SQL procedure successfully completed.


DBMS_SQLTUNE.REPORT_TUNING_TASK(TASK_NAME=>'STA_UPGRADE_TO_19C_CC',SECTION=>'FINDINGS',RESULT_LIMIT=>20)
--------------------------------------------------------------------------------
--------------------------------------------------------------------------------
--------------------------------------------------------------------------------
```

GENERAL INFORMATION SECTION

Tuning Task Name	: STA_UPGRADE_TO_19C_CC
Tuning Task Owner	: SYS
Workload Type	: SQL Tuning Set
Scope	: COMPREHENSIVE
Time Limit(seconds)	: 360
Completion Status	: COMPLETED
Started at	: 12/17/2019 12:13:48
Completed at	: 12/17/2019 12:14:28
SQL Tuning Set (STS) Name	: STS_CaptureCursorCache

SQL Tuning Set Owner : SYS
Number of Statements in the STS : 41

DETAILS SECTION

Statements with Results Ordered by Maximum (Profile/Index) Benefit, Object ID

Object ID : 7
Schema Name: TPCC
SQL ID : csv0xdm9c394t
SQL Text : SELECT O_ID, O_CARRIER_ID, O_ENTRY_D FROM (SELECT O_ID,
O_CARRIER_ID, O_ENTRY_D FROM ORDERS WHERE O_D_ID = :B3 AND
O_W_ID = :B2 AND O_C_ID=:B1 ORDER BY O_ID DESC) WHERE ROWNUM = 1

FINDINGS SECTION (2 findings)

1- Statistics Finding

Optimizer statistics for table "TPCC"."ORDERS" are stale.

Recommendation

- Consider collecting optimizer statistics for this table.
execute dbms_stats.gather_table_stats(ownname => 'TPCC', tabname =>
'ORDERS', estimate_percent => DBMS_STATS.AUTO_SAMPLE_SIZE,
method_opt => 'FOR ALL COLUMNS SIZE AUTO');

Rationale

The optimizer requires up-to-date statistics for the table in order to
select a good execution plan.

2- Index Finding (see explain plans section below)

The execution plan of this statement can be improved by creating one or more
indices.

Recommendation (estimated benefit: 98.06%)

- Consider running the Access Advisor to improve the physical schema design
or creating the recommended index.

create index TPCC.IDX$$_00770003 on TPCC.ORDERS("O_C_ID","O_D_ID","O_W_ID")
;

Rationale

Creating the recommended indices significantly improves the execution plan
of this statement. However, it might be preferable to run "Access Advisor"
using a representative SQL workload as opposed to a single statement. This
will allow to get comprehensive index recommendations which takes into
account index maintenance overhead and additional space consumption.

Object ID : 3
Schema Name: TPCC
SQL ID : 7m5h0wf6stq0q
SQL Text : SELECT COUNT(C_ID) FROM CUSTOMER WHERE C_LAST = :B3 AND C_D_ID =
:B2 AND C_W_ID = :B1

FINDINGS SECTION (2 findings)

1- Index Finding (see explain plans section below)

The execution plan of this statement can be improved by creating one or more
indices.

Recommendation (estimated benefit: 97.96%)

- Consider running the Access Advisor to improve the physical schema design
or creating the recommended index.
create index TPCC.IDX$$_00770001 on TPCC.CUSTOMER("C_LAST","C_D_ID","C_W_ID
");

Rationale

Creating the recommended indices significantly improves the execution plan
of this statement. However, it might be preferable to run "Access Advisor"
using a representative SQL workload as opposed to a single statement. This
will allow to get comprehensive index recommendations which takes into
account index maintenance overhead and additional space consumption.

2- Alternative Plan Finding

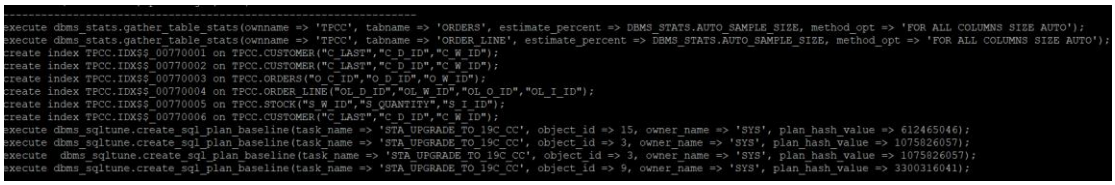
Some alternative execution plans for this statement were found by searching
the system's real-time and historical performance data.

The following table lists these plans ranked by their average elapsed time.
See section "ALTERNATIVE PLANS SECTION" for detailed information on each
plan.

id plan hash last seen elapsed (s) origin note
-- ---------- ------------------- ------------ --------------- ----------
1 1075826057 2019-12-17/10:26:48 0.002 AWR
2 3642382161 2019-12-17/05:14:21 0.002 AWR original plan

Recommendation

- Consider creating a SQL plan baseline for the plan with the best average
elapsed time.
execute dbms_sqltune.create_sql_plan_baseline(task_name =>
'STA_UPGRADE_TO_19C_CC', object_id => 3, owner_name => 'SYS',
plan_hash_value => 1075826057);



保存优化建议并执行

将下面建议的内容，保存成脚本保存到sta_findings_exec.sql，然后执行
execute dbms_stats.gather_table_stats(ownname => 'TPCC', tabname => 'ORDERS',
estimate_percent => DBMS_STATS.AUTO_SAMPLE_SIZE, method_opt => 'FOR ALL
COLUMNS SIZE AUTO');
execute dbms_stats.gather_table_stats(ownname => 'TPCC', tabname => 'ORDER_LINE',
estimate_percent => DBMS_STATS.AUTO_SAMPLE_SIZE, method_opt => 'FOR ALL
COLUMNS SIZE AUTO');
create index TPCC.IDX$$_00770001 on TPCC.CUSTOMER("C_LAST","C_D_ID","C_W_ID");
create index TPCC.IDX$$_00770002 on TPCC.CUSTOMER("C_LAST","C_D_ID","C_W_ID");
create index TPCC.IDX$$_00770003 on TPCC.ORDERS("O_C_ID","O_D_ID","O_W_ID");
create index TPCC.IDX$$_00770004 on
TPCC.ORDER_LINE("OL_D_ID","OL_W_ID","OL_O_ID","OL_I_ID");
create index TPCC.IDX$$_00770005 on TPCC.STOCK("S_W_ID","S_QUANTITY","S_I_ID");

create index TPCC.IDX$$_00770006 on TPCC.CUSTOMER("C_LAST","C_D_ID","C_W_ID");
execute dbms_sqltune.create_sql_plan_baseline(task_name => 'STA_UPGRADE_TO_19C_CC', object_id => 15, owner_name => 'SYS', plan_hash_value => 612465046);
execute dbms_sqltune.create_sql_plan_baseline(task_name => 'STA_UPGRADE_TO_19C_CC', object_id => 3, owner_name => 'SYS', plan_hash_value => 1075826057);
execute dbms_sqltune.create_sql_plan_baseline(task_name => 'STA_UPGRADE_TO_19C_CC', object_id => 3, owner_name => 'SYS', plan_hash_value => 1075826057);
execute dbms_sqltune.create_sql_plan_baseline(task_name => 'STA_UPGRADE_TO_19C_CC', object_id => 9, owner_name => 'SYS', plan_hash_value => 3300316041);

```
Connected to:
Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
Version 19.3.0.0.0

SQL> start /home/oracle/scripts/sta_findings_exec.sql

PL/SQL procedure successfully completed.


PL/SQL procedure successfully completed.


Index created.


Index created.


Index created.


Index created.

create index TPCC.IDX$$_00770006 on TPCC.CUSTOMER("C_LAST","C_D_ID","C_W_ID")
                                                   *
ERROR at line 1:
ORA-01408: such column list already indexed



PL/SQL procedure successfully completed.


PL/SQL procedure successfully completed.


PL/SQL procedure successfully completed.


PL/SQL procedure successfully completed.

SQL>
```

运行 SPA 验证结果

下面这个练习，如果在生产环境中一定要先做好验证测试，在我们这个 Lab 中让我们执行所有的建议，看看会发生什么。

```
@/home/oracle/scripts/spa_cpu.sql
@/home/oracle/scripts/spa_report_cpu.sql
@/home/oracle/scripts/spa_elapsed.sql
@/home/oracle/scripts/spa_report_elapsed.sql
exit

$ ls -ltr *html
-rw-r--r--. 1 oracle dba 297737 Dec 17 11:23 compare_spa_runs_20191217112318.html
-rw-r--r--. 1 oracle dba 297908 Dec 17 11:27 compare_spa_runs_20191217112731.html
-rw-r--r--. 1 oracle dba 428048 Dec 17 12:02 compare_spa_runs_20191217120228.html
-rw-r--r--. 1 oracle dba 428076 Dec 17 12:02 compare_spa_runs_20191217120244.html
-rw-r--r--. 1 oracle dba 427303 Dec 17 12:27 compare_spa_runs_20191217122704.html
-rw-r--r--. 1 oracle dba 428068 Dec 17 12:27 compare_spa_runs_20191217122721.html
[UPGR] oracle@hol:~/scripts
$
```

到这里我们熟悉并回顾了 Oracle 数据库性能优化在变更前后的一些应对方法和过程，本实验就练习到这里，希望能给大家一启发，更进一步的了解，大家就在实践中去不断探索。

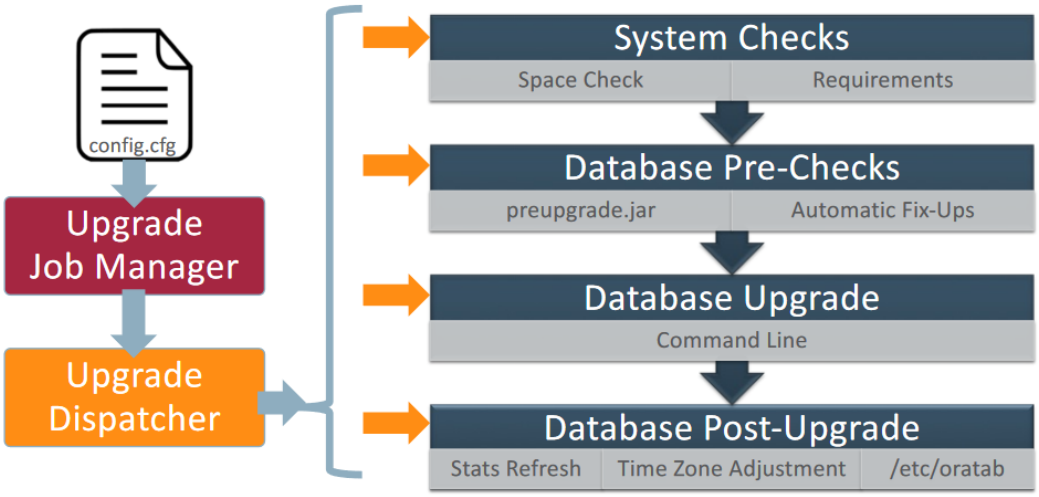
自动升级 DB12 到 19c

我们将使用新的 AutoUpgrade 实用程序将 non-CDB DB12 数据库从 Oracle 12.2.0.1 升级到 Oracle 19c。

AutoUpgrade 工具简介

随着 2019 年 1 月发布的 RU(Release Updates)，针对 12.2 和 18c(18.5)做为升级目标的 AutoUpgrade Tool 在 Support 网站提供，从 19c(19.3)开始，该工具也随着数据库软件一并发布(保存在$ORACLE_HOME/rdbms/admin 目录下)。这个 AutoUpgradeUtility 是基于 Java8 的 JARfile（autoupgrad.jar），是一个轻量级的命令行工具，可以协助我们实现自动化升级数据库。

通过 AutoUpgrade Tool 可以实现一个或多个数据库升级，AutoUpgrade 会自动运行升级前的预检查任务，在需要的地方执行自动修复（修复 99%的潜在问题），可根据配置文件定义一个恢复点以防止出错，接着执行升级数据库，最后自动进行升级后的后续动作（比如编译失效对象，调整时区、更新密码文件、spfile 以及/etc/oratab 等）。整个过程我们无需干预，只需要一个命令和一个配置文件，很显然 AutoUpgrade 显著减少了与数据库升级相关的手工工作，使数据库升级过程更容易完成，从而将低了风险。



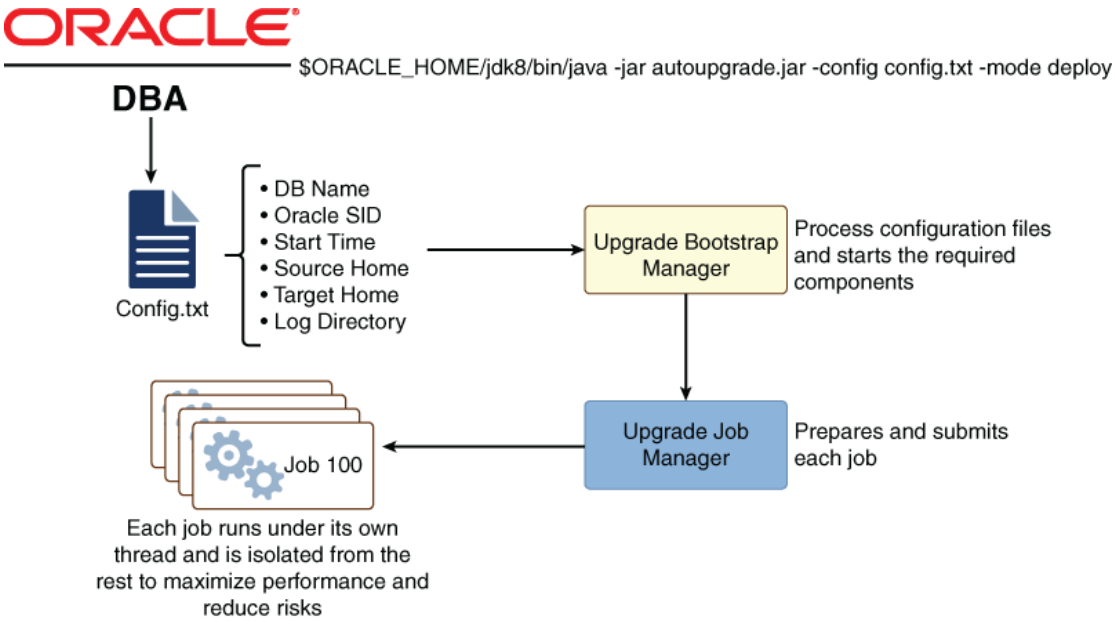
截止到 2019-12 月，AutoUpgrade Tool 的最新上传版本是 Autoupgrade 19.7.2 build version 20191125。
对于 12.2 和 18c(18.5)作为升级目标的环境，需要通过 2485457.1 – AutoUpgradeTool 下载工具并保存到$ORACLE_HOME/rdbms/admin 目录下。虽然说 19.3 开始，该工具随数据库一并提供，但 Oracle 专家建议您到 Support 网站下载最新的版本并替换 $ORACLE_HOME/rdbms/admin 的发行版本。

AutoUpgrade Tool 支持的升级源版本	AutoUpgrade Tool 支持的升级目标版本
11.2.0.4	Oracle Database 19c (19.3 及以后)
	Oracle Database 18c (18.5 及以后)
	Oracle Database 12c Release 2 (12.2 + DBJAN2019RU 及以后)

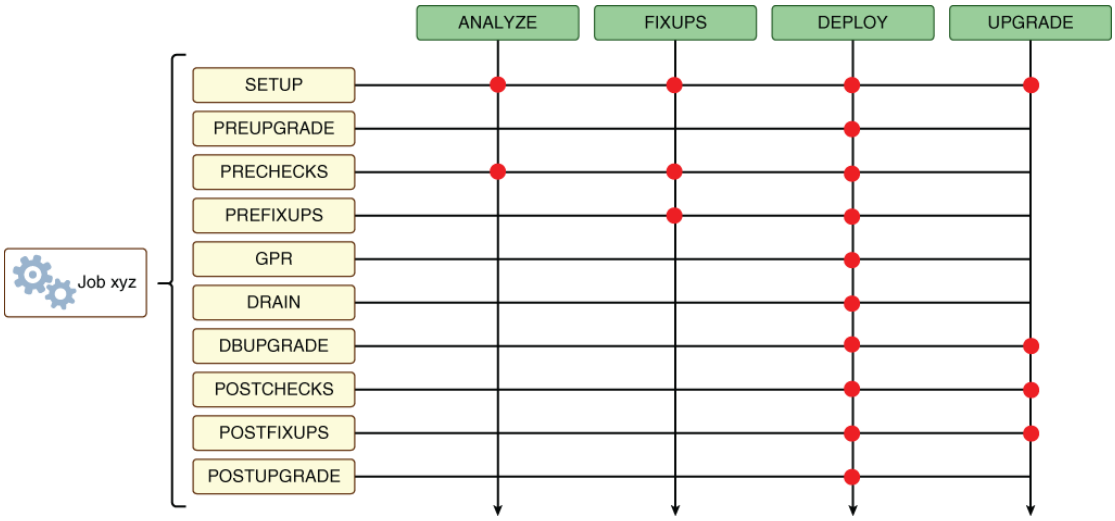
AutoUpgrade 相关的文档：Using AutoUpgrade for Oracle Database Upgrades. AutoUpgrade Tool (Doc ID 2485457.1)

AutoUpgrade 工作流程

https://docs.oracle.com/en/database/oracle/oracle-database/19/upgrd/understanding-autoupgrade-workflows-stages.html#GUID-8C1FC4E1-2C9C-4CBC-AF5B-95F311D73BE4



AutoUpgrade 工作內容



- **Analyze** Mode: Setup, Prechecks.
- **Fixups** Mode: Setup, Prechecks, and Prefixups.
- **Deploy** Mode: Setup, Preupgrade, Prechecks, guaranteed restore point (GRP), Prefixups, Drain, DB (database) Upgrade, Postchecks, and Postupgrade. You can run your own scripts before the upgrade

(Preupgrade stage) or after the upgrade (Postupgrade stage), or both before and after the upgrade.

- **Upgrade** Mode: Setup, DB (database) Upgrade, Postchecks, and Postfixups.

启动 DB12 数据库

. db12
sqlplus / as sysdba
startup
exit

生成并编辑配置文件

$ java -jar $OH19/rdbms/admin/autoupgrade.jar -version
build.version 20190417
build.date 2019/04/17 13:16:25
build.label RDBMS_PT.AUTOUPGRADE_LINUX.X64_190416.1801

[DB12] oracle@hol:~

```
$ java -jar $OH19/rdbms/admin/autoupgrade.jar -help
usage: java -jar autoupgrade.jar [-version | -help | -create_sample_file <create_sample_file>]   [-settings <settings>] [-config <config>]
       [-clear_recovery_data] [-mode <mode>] [-console | -noconsole]  [-restore_on_fail] [-debug]
    -version                                Displays the current build of the jar
    -help                                   Displays the available options
    -create_sample_file <create_sample_file>    Creates a sample configuration file that be used as reference
                                            config   - Creates a sample configuration file
                                            settings - Creates a sample internal configuration file to allow a deeper low level configuration
    -settings <settings>                    Path to config file to overwrite internal settings
    -config <config>                        User config file with the databases to upgrade
    -clear_recovery_data                    Remove the recovery checkpoint to start fresh the next time the tool is launched
    -mode <mode>                            Mode on which the AutoUpgrade will start and behave
                                            analyze - Executes the checks and generates a report of the database status
                                            deploy  - Performs the upgrade of the databases from start to end
                                            fixups  - Executes the checks and pre-upgrade fixups but do not start the upgrade
                                            upgrade - Performs the database upgrade and post-upgrade actions.  The database must already be up and running
                                            with the target home
    -console                                Start the AutoUpgrade with the console enabled (default)
    -noconsole                              Start the AutoUpgrade with the console disabled
    -restore_on_fail                        If present, when a job fails, the database is restored automatically. Errors in PDBs are not considered fatal,
                                            only errors in CDB$ROOT or Non-CDBs
    -debug                                  Debug messages enabled

The config option with sample parameter creates a sample database configuration file with default values.
Or you can use it with a custom database configuration file with an execution mode (deploy, analyze, fixups or upgrade).

The settings parameter lets you use a file with Autoupgrade internal settings, can be default for base settings or you can specify a file path for custom
settings.
```

```
$ java -jar $OH19/rdbms/admin/autoupgrade.jar -create_sample_file config
Created sample configuration file /home/oracle/sample_config.cfg
[DB12] oracle@hol:~
```

在我们这次实验的虚拟机中，已经为大家创建好了一个配置文件 /home/oracle/scripts/DB12.cfg，可以直接用这个

生成的配置文件	编辑后内容
global.autoupg_log_dir=/default/current/location	global.autoupg_log_dir=/home/oracle/logs

# # Database number 1 # upg1.dbname=employee upg1.start_time=NOW upg1.source_home=/u01/app/oracle/product/11.2.0/dbhome_1 upg1.target_home=/u01/app/oracle/product/19.1.0/dbhome_1 upg1.sid=emp upg1.log_dir=/scratch/auto upg1.upgrade_node=node1 upg1.target_version=19.1 #upg1.run_utlrp=yes #upg1.timezone_upg=yes	# # Database number 1 # upg1.dbname=DB12 upg1.start_time=NOW upg1.source_home=/u01/app/oracle/product/12.2.0.1 upg1.target_home=/u01/app/oracle/product/19 upg1.sid=DB12 upg1.log_dir=/home/oracle/logs/DB12 upg1.upgrade_node=localhost upg1.target_version=19 upg1.restoration=no upg1.run_utlrp=yes upg1.timezone_upg=yes

以分析模式运行 AutoUpgrade 工具

您可以直接运行 autoupgrade，建议最好先运行一个分析。一旦分析阶段顺利通过，数据库就可以自动升级

java -jar $OH19/rdbms/admin/autoupgrade.jar -config /home/oracle/scripts/DB12.cfg -mode analyze

```
$ java -jar $OH19/rdbms/admin/autoupgrade.jar -config /home/oracle/scripts/DB12.cfg -mode analyze
Autoupgrade tool launched with default options
+--------------------------------+
| Starting AutoUpgrade execution |
+--------------------------------+
1 databases will be analyzed
Type 'help' to list console commands
upg> lsj
+----+-------+---------+---------+-------+--------------+--------+--------+---------------+
|Job#|DB_NAME|    STAGE|OPERATION| STATUS|    START_TIME|END_TIME| UPDATED|        MESSAGE|
+----+-------+---------+---------+-------+--------------+--------+--------+---------------+
| 100|   DB12|PRECHECKS|PREPARING|RUNNING|19/12/18 03:16|     N/A|03:16:03|Loading DB info|
+----+-------+---------+---------+-------+--------------+--------+--------+---------------+
Total jobs 1
upg> Job 100 completed
------------------ Final Summary --------------------
Number of databases           [ 1 ]

Jobs finished successfully     [1]
Jobs failed                    [0]
Jobs pending                   [0]
------------- JOBS FINISHED SUCCESSFULLY -------------
Job 100 FOR DB12

[DB12] oracle@hol:~/scripts
```

可以看到没有任何问题，接下来就可以实施自动升级了

以部署模式运行 AutoUpgrade 工具

现在我们就正式用 AutoUpgrade 工具来实施升级，当指定-mode deploy 启动升级时，该工具会重复分析阶段，但是添加了修复、升级和升级后的步骤

java -jar $OH19/rdbms/admin/autoupgrade.jar -config /home/oracle/scripts/DB12.cfg -mode deploy

```
[DB12] oracle@hol:~/scripts
$ java -jar $OH19/rdbms/admin/autoupgrade.jar -config /home/oracle/scripts/DB12.cfg -mode deploy
Autoupgrade tool launched with default options
+--------------------------------+
| Starting AutoUpgrade execution |
+--------------------------------+
1 databases will be processed
Type 'help' to list console commands
upg> lsj
+----+-------+---------+---------+-------+--------------+--------+--------+---------------+
|Job#|DB_NAME|    STAGE|OPERATION| STATUS|    START_TIME|END_TIME| UPDATED|        MESSAGE|
+----+-------+---------+---------+-------+--------------+--------+--------+---------------+
| 101|   DB12|PRECHECKS|PREPARING|RUNNING|19/12/18 03:19|     N/A|03:19:20|Loading DB info|
+----+-------+---------+---------+-------+--------------+--------+--------+---------------+
Total jobs 1
upg> status -job 101
Progress
---------------------------------
Start time:      19/12/18 03:19
Elapsed (min):   0
End time:        N/A
Last update:     2019-12-18T03:19:46.285
Stage:           PREFIXUPS
Operation:       EXECUTING
Status:          RUNNING
Pending stages:  6

Job Logs Locations
---------------------------------
Logs Base:    /home/oracle/logs/DB12
Job logs:     /home/oracle/logs/DB12/101
Stage logs:   /home/oracle/logs/DB12/101/prefixups
TimeZone:     /home/oracle/logs/DB12/temp

Additional information
---------------------------------
Details:
+----+---------------+-------+
|Conn|           FixUp| Status|
+----+---------------+-------+
|DB12|DICTIONARY_STATS|STARTED|
+----+---------------+-------+

Error Details:
None

upg>
```

下面是常用的命令

upg>lsj	这将列出作业编号和每个活动作业的概述信息
upg> status -job <number>	提供更多关于指定 Job 的信息，通过多次执行 status -job 可以查看升级进度
upg> resume -job <number>	重新启动一个 Job
upg> abort -job <number>	取消一个 Job
upg> restore -job <number>	从 GRP 恢复数据库

```
upg> status
---------------- Config -------------------
User configuration file     [/home/oracle/scripts/DB12.cfg]
General logs location       [/home/oracle/logs/cfgtoollogs/upgrade/auto]
Mode                        [DEPLOY]
DB upg fatal errors         ORA-00600,ORA-07445
DB Post upgrade abort time  [60] minutes
DB upg abort time           [1440] minutes
DB restore abort time       [120] minutes
DB drop GRP abort time      [3] minutes
------------------------ Jobs ------------------------
Total databases in configuration file [1]
Total Non-CDB being processed         [1]
Total CDB being processed             [0]
Jobs finished successfully            [0]
Jobs finished/aborted                 [0]
jobs in progress                      [1]
------------ Resources ----------------
Threads in use                        [39]
JVM used memory                       [151] MB
CPU in use                            [13%]
Processes in use                      [18]

upg>
```

```
upg> status -job 101
Progress
-----------------------------------
Start time:      19/12/18 03:19
Elapsed (min):   10
End time:        N/A
Last update:     2019-12-18T03:28:25.112
Stage:           DBUPGRADE
Operation:       EXECUTING
Status:          RUNNING
Pending stages:  4

Job Logs Locations
-----------------------------------
Logs Base:    /home/oracle/logs/DB12
Job logs:     /home/oracle/logs/DB12/101
Stage logs:   /home/oracle/logs/DB12/101/dbupgrade
TimeZone:     /home/oracle/logs/DB12/temp

Additional information
-----------------------------------
Details:
[Upgrading] is [21%] completed for [db12]
                +---------+-------------+
                |CONTAINER|   PERCENTAGE|
                +---------+-------------+
                |     DB12|UPGRADE [21%]|
                +---------+-------------+

Error Details:
None

upg>
```

```
Stage:           POSTFIXUPS
Operation:       EXECUTING
Status:          RUNNING
Pending stages:  2

Job Logs Locations
----------------------------------
Logs Base:     /home/oracle/logs/DB12
Job logs:      /home/oracle/logs/DB12/101
Stage logs:    /home/oracle/logs/DB12/101/postfixups
TimeZone:      /home/oracle/logs/DB12/temp

Additional information
----------------------------------
Details:
+----+--------------------+--------+
|Conn|               FixUp|  Status|
+----+--------------------+--------+
|DB12|     POST_DICTIONARY|FINISHED|
|DB12|  POST_FIXED_OBJECTS|FINISHED|
|DB12|OLD_TIME_ZONES_EXIST| STARTED|
+----+--------------------+--------+
```

根据大家实验时所用的硬件环境的不同，这个自动升级过程大约需要 20~45 分钟。在等待这个实验完成的过程中，大家可以继续后面的练习（彼此没有依赖关系）。升级结束后，结果类似如下:

```
upg> Job 101 completed
------------------- Final Summary --------------------
Number of databases           [ 1 ]

Jobs finished successfully    [1]
Jobs failed                   [0]
Jobs pending                  [0]
------------- JOBS FINISHED SUCCESSFULLY -------------
Job 101 FOR DB12

[DB12] oracle@hol:~/scripts
```

到这里，通过 AutoUpgrade 就顺利的完成了数据库升级的全部动作。下面你可以检查一下 AutoUpgrade 执行过程中每个阶段的各种升级目录和日志文件以及 Upgrade Summary report

```
$ pwd
/home/oracle/logs/DB12/101/dbupgrade
[DB12] oracle@hol:~/logs/DB12/101/dbupgrade
$ cat upg_summary.log

Oracle Database Release 19 Post-Upgrade Status Tool    12-18-2019 03:49:3
Database Name: DB12

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

Oracle Server                          UPGRADED      19.3.0.0.0  00:13:01
JServer JAVA Virtual Machine           UPGRADED      19.3.0.0.0  00:01:06
Oracle XDK                             UPGRADED      19.3.0.0.0  00:01:09
Oracle Database Java Packages          UPGRADED      19.3.0.0.0  00:00:12
Oracle Label Security                  UPGRADED      19.3.0.0.0  00:00:06
Oracle Workspace Manager               UPGRADED      19.3.0.0.0  00:00:37
Oracle Real Application Clusters     OPTION OFF      19.3.0.0.0  00:00:00
Oracle XML Database                    UPGRADED      19.3.0.0.0  00:01:41
Oracle Multimedia                      UPGRADED      19.3.0.0.0  00:03:00
Datapatch                                                        00:03:05
Final Actions                                                    00:03:12
Post Upgrade                                                     00:00:13

Total Upgrade Time: 00:26:08

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

Grand Total Upgrade Time:    [0d:0h:32m:7s]
[DB12] oracle@hol:~/logs/DB12/101/dbupgrade
$
```

调整兼容性参数

show parameter compatible
show spparameter compatible
alter system set compatible='19.0.0' scope=spfile;
show spparameter compatible
shut immediate
exit

```
[DB12] oracle@hol:~
$ . db19
[DB12] oracle@hol:~
$ echo $ORACLE_SID
DB12
[DB12] oracle@hol:~
$ sqlplus / as sysdba

SQL*Plus: Release 19.0.0.0.0 - Production on Wed Dec 18 04:06:41 2019
Version 19.3.0.0.0

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


Connected to:
Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
Version 19.3.0.0.0

SQL> show parameter compatible

NAME                                 TYPE        VALUE
------------------------------------ ----------- ------------------------------
compatible                           string      12.2.0
noncdb_compatible                    boolean     FALSE
SQL> show spparameter compatible

SID      NAME                          TYPE        VALUE
-------- ----------------------------- ----------- ------------------------------
*        compatible                    string      12.2.0
*        noncdb_compatible             boolean
SQL> alter system set compatible='19.0.0' scope=spfile;

System altered.

SQL> show spparameter compatible

SID      NAME                          TYPE        VALUE
-------- ----------------------------- ----------- ------------------------------
*        compatible                    string      19.0.0
*        noncdb_compatible             boolean
SQL> shut immediate
Database closed.
Database dismounted.
ORACLE instance shut down.
SQL>
```

恭喜，到这里我们就通过新的 AutoUpgrade 工具成功地将 12.2.0.1 版本的 non-CDB DB12 数据库升级到 Oracle 19c，本试验到这里就结束了。更多关于 AutoUpgrade 的介绍，请大家参阅前面提到的官方文档。

到这里这个试验就练结束了，可以关闭DB12这数据库了，节省点资源!

将升级过的 UPGR 插入到 CDB2

在前面我们完成了 non-CDB UPGR 从 11.2.0.4 到 19c 的升级，现在我们将这个升级后的数据库 plugin 到 19c 下的 CDB2 中，模拟升级并迁移到 19c 多租户架构。CDB2 是一个已经建好的 19c 多租户容器数据库。本练习将把 UPGR 转成 PDB 然后 Plugin 到 CDB2。为了插入 non-CDB 数据库，如本实验中的 UPGR 数据库，必须先将其升级到与所插入的 CDB 相同的版本。

启动 UPGR 到 read only

切换到 19c 下的 UPGR 数据库环境中，将数据库重启到 read only 状态
$. upgr19
$ sqlplus / as sysdba
SQL> shut immediate
SQL> startup open read only;

创建 UPGR 的描述文件

SQL> exec dbms_pdb.describe('/home/oracle/pdb1.xml');

```
SQL> exec dbms_pdb.describe('/home/oracle/pdb1.xml');

PL/SQL procedure successfully completed.

SQL>
```

在 CDB2 中执行兼容性检查

这个步骤不是必需的，但建议大家这么做，通过 compatibility check，可以在 plugin 之前发现并修复问题。

```
SET SERVEROUTPUT ON
DECLARE
compatible BOOLEAN := FALSE;
BEGIN
compatible := DBMS_PDB.CHECK_PLUG_COMPATIBILITY(pdb_descr_file => '/home/oracle/pdb1.xml');
if compatible then
    DBMS_OUTPUT.PUT_LINE('Is the future PDB compatible? ==> YES');
else
    DBMS_OUTPUT.PUT_LINE('Is the future PDB compatible? ==> NO');
end if;
END;
/
```

如果输出结果是“NO”(可能实际环境中会经常遇到“不”)，那么也不要紧张，通过查看 PDB_PLUG_IN_VIOLATIONS 视图

```
set lines 156
column message format a70
column status format a9
column type format a9
```

column con_id format 9
select con_id, type, message, status from pdb_plug_in_violations where status<>'RESOLVED' order by time;

```
SQL>
SQL> column message format a50
column status format a9
column type format a9
column con_id format 9
select con_id, type, message, status from pdb_plug_in_violations where status<>'RESOLVED' order by time;

CON_ID TYPE      MESSAGE                                            STATUS
------ --------- -------------------------------------------------- ---------
     1 WARNING   PDB plugged in is a non-CDB, requires noncdb_to_pd PENDING
                 b.sql be run.

     1 WARNING   CDB parameter sga_target mismatch: Previous 1G Cur PENDING
                 rent 1504M

     1 WARNING   CDB parameter pga_aggregate_target mismatch: Previ PENDING
                 ous 100M Current 200M


SQL>
```

在 CDB2 中执行 plugin 操作

将 UPGR 插入到 CDB2 后，给它取一个新的名字叫 PDB1，也就是说以后不再有 UPGR 这个数据库了。在我们这个练习中，因为在同一个 Server 上，为避免额外的复制时间和磁盘空间消耗所以不再拷贝数据文件。如果需要拷贝数据文件，可以通过 FILE_NAME_CONVERT 参数来进行文件名的映射。

create pluggable database PDB1 using '/home/oracle/pdb1.xml' nocopy tempfile reuse;
show pdbs
alter pluggable database PDB1 open;

```
SQL> show pdbs

    CON_ID CON_NAME                       OPEN MODE  RESTRICTED
---------- ------------------------------ ---------- ----------
         2 PDB$SEED                       READ ONLY  NO
SQL> create pluggable database PDB1 using '/home/oracle/pdb1.xml' nocopy tempfile reuse;

Pluggable database created.

SQL> show pdbs

    CON_ID CON_NAME                       OPEN MODE  RESTRICTED
---------- ------------------------------ ---------- ----------
         2 PDB$SEED                       READ ONLY  NO
         3 PDB1                           MOUNTED
SQL> alter pluggable database PDB1 open;

Warning: PDB altered with errors.

SQL> show pdbs

    CON_ID CON_NAME                       OPEN MODE  RESTRICTED
---------- ------------------------------ ---------- ----------
         2 PDB$SEED                       READ ONLY  NO
         3 PDB1                           READ WRITE YES
SQL>
```

可以看到 PDB 打开时有相应的 Warning 信息，现在也受限模式 Open。我们查看下

PDB_PLUG_IN_VIOLATIONS 视图

```
set lines 156
column message format a80
column status format a9
column type format a9
column con_id format 9
select con_id, type, message, status from pdb_plug_in_violations where
status<>'RESOLVED' order by time;
```

```
SQL> set lines 156
column message format a80
column status format a9
column type format a9
column con_id format 9
select con_id, type, message, status from pdb_plug_in_violations where status<>'RESOLVED' order by time;

CON_ID TYPE      MESSAGE                                                                          STATUS
------ --------- -------------------------------------------------------------------------------- -------
     1 WARNING   PDB plugged in is a non-CDB, requires noncdb_to_pdb.sql be run.                  PENDING
     1 WARNING   CDB parameter sga_target mismatch: Previous 1G Current 1504M                     PENDING
     1 WARNING   CDB parameter pga_aggregate_target mismatch: Previous 100M Current 200M         PENDING
     1 WARNING   CDB parameter sga_target mismatch: Previous 1G Current 1504M                     PENDING
     1 WARNING   CDB parameter pga_aggregate_target mismatch: Previous 100M Current 200M         PENDING
     3 ERROR     PDB plugged in is a non-CDB, requires noncdb_to_pdb.sql be run.                  PENDING

6 rows selected.

SQL>
```

提示的很明确，因为是从 non-CDB 插过来的，所以需要运行 noncdb_to_pdb.sql 脚本。通过调用 noncdb_to_pdb.sql，才能让 UPGR 成为一个真正的 pluggable database，这个脚本运行大约需要 10~20 分钟。

noncdb_to_pdb.sql 脚本用于清理和更改非 CDB 中的内容，并最终将其与 CDB 结合起来，并且是不可逆转的。这个脚本可能运行比较慢，主要取决于需要调整的对象的数量，并映射到 CDB 的字典。以及重新编译所需的时间，因为它是强制重新编译的。
所以，对于停机时间比较紧张的环境，如果采用这种方案，要提前测试一下这个脚本的执行时间，除了增加服务器 CPU 资源外，也没有任何其他可以优化的地方。

从 12.2.0.1 开始，如果运行 noncdb_to_pdb.sql 的过程中异常，可以重新开始，在 12.2.0.1 之前版本，必需从源库重新开始。运行 noncdb_to_pdb.sql 脚本

```
SQL> show pdbs

    CON_ID CON_NAME                       OPEN MODE  RESTRICTED
---------- ------------------------------ ---------- ----------
         2 PDB$SEED                       READ ONLY  NO
         3 PDB1                           READ WRITE YES
SQL> alter session set container=PDB1;

Session altered.

SQL> @?/rdbms/admin/noncdb_to_pdb.sql
SQL> SET FEEDBACK 1
SQL> SET NUMWIDTH 10
SQL> SET LINESIZE 80
SQL> SET TRIMSPOOL ON
SQL> SET TAB OFF
SQL> SET PAGESIZE 100
SQL> SET VERIFY OFF
SQL>
SQL> -- save settings
SQL> STORE SET ncdb2pdb.settings.sql REPLACE
Wrote file ncdb2pdb.settings.sql
```

脚本运行完毕后，重启下 PDB1，警告信息消失

```
SQL> set timing OFF
SQL> set trimout ON
SQL> set trimspool ON
SQL> set underline "-"
SQL> set verify OFF
SQL> set wrap ON
SQL> set xmloptimizationcheck OFF
SQL>
SQL>
SQL> show pdbs

    CON_ID CON_NAME                       OPEN MODE  RESTRICTED
---------- ------------------------------ ---------- ----------
         3 PDB1                           READ WRITE YES
SQL> shut immediate
Pluggable Database closed.
SQL> startup
Pluggable Database opened.
SQL> show pdbs

    CON_ID CON_NAME                       OPEN MODE  RESTRICTED
---------- ------------------------------ ---------- ----------
         3 PDB1                           READ WRITE NO
SQL>
```

到这里我们就顺利的将升级后的UPGR 插入到19c 下的CDB2 了， 名字为 PDB1;

通过 Full Transportable Export/Import 升级到 19c

在 CDB2 中创建 PDB2

. cdb2
sqlplus / as sysdba
create pluggable database PDB2 admin user adm identified by adm
file_name_convert=('pdbseed', 'pdb2');
alter pluggable database PDB2 open;
alter pluggable database PDB2 save state;
alter session set container=PDB2;

create directory mydir as '/u02/oradata/CDB2/mydir';
grant read, write on directory mydir to system;
create public database link SOURCEDB connect to system identified by oracle using 'FTEX';

```
$ . cdb2
[CDB2] oracle@hol:~
$ sqlplus / as sysdba

SQL*Plus: Release 19.0.0.0.0 - Production on Thu Dec 19 13:59:27 2019
Version 19.3.0.0.0

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


Connected to:
Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
Version 19.3.0.0.0

SQL> create pluggable database PDB2 admin user adm identified by adm file_name_convert=('pdbseed', 'pdb2');

Pluggable database created.

SQL> show pdbs

    CON_ID CON_NAME                       OPEN MODE  RESTRICTED
---------- ------------------------------ ---------- ----------
         2 PDB$SEED                       READ ONLY  NO
         3 PDB1                           READ WRITE NO
         4 PDB2                           MOUNTED
SQL> alter pluggable database pdb2 open;

Pluggable database altered.

SQL> alter pluggable database PDB2 save state;

Pluggable database altered.

SQL> alter session set container=PDB2;

Session altered.

SQL> create directory mydir as '/u02/oradata/CDB2/mydir';

Directory created.

SQL> grant read, write on directory mydir to system;

Grant succeeded.

SQL> create public database link SOURCEDB connect to system identified by oracle using 'FTEX';

Database link created.

SQL>
```

准备 FTEX

. ftex
sqlplus / as sysdba

startup
alter tablespace USERS read only;
exit

```
$ . ftex
[FTEX] oracle@hol:~
$ sqlplus / as sysdba

SQL*Plus: Release 11.2.0.4.0 Production on Thu Dec 19 14:01:47 2019

Copyright (c) 1982, 2013, Oracle.  All rights reserved.

Connected to an idle instance.

SQL> startup
ORACLE instance started.

Total System Global Area 1152450560 bytes
Fixed Size                  2252584 bytes
Variable Size             335544536 bytes
Database Buffers          805306368 bytes
Redo Buffers                9347072 bytes
Database mounted.
Database opened.
SQL> alter tablespace USERS read only;

Tablespace altered.

SQL> exit
Disconnected from Oracle Database 11g Enterprise Edition Release 11.2.0.4.0 - 64bit Production
With the Partitioning, OLAP, Data Mining and Real Application Testing options
[FTEX] oracle@hol:~
```

迁移 FTEX 到 PDB2

cp /u02/oradata/FTEX/users01.dbf /u02/oradata/CDB2/pdb2
. cdb2
impdp system/oracle@pdb2 network_link=sourcedb version=12 full=y
transportable=always metrics=y exclude=statistics directory=mydir logfile=pdb2.log
transport_datafiles='/u02/oradata/CDB2/pdb2/users01.dbf'

```
[FTEX] oracle@hol:~
$ cp /u02/oradata/FTEX/users01.dbf /u02/oradata/CDB2/pdb2
[FTEX] oracle@hol:~
$ . cdb2
[CDB2] oracle@hol:~
$ impdp system/oracle@pdb2 network_link=sourcedb version=12 full=y transportable=always metrics=y exclude=statistics
 directory=mydir logfile=pdb2.log transport_datafiles='/u02/oradata/CDB2/pdb2/users01.dbf'

Import: Release 19.0.0.0.0 - Production on Thu Dec 19 14:03:57 2019
Version 19.3.0.0.0

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

UDI-28002: operation generated ORACLE error 28002
ORA-28002: the password will expire within 7 days

Connected to: Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
Starting "SYSTEM"."SYS_IMPORT_FULL_01":  system/********@pdb2 network_link=sourcedb version=12 full=y transportable=
always metrics=y exclude=statistics directory=mydir logfile=pdb2.log transport_datafiles=/u02/oradata/CDB2/pdb2/user
s01.dbf
W-1 Startup took 2 seconds
W-1 Estimate in progress using BLOCKS method...
W-1 Processing object type DATABASE_EXPORT/PLUGTS_FULL/FULL/PLUGTS_TABLESPACE
W-1      Completed 0 PLUGTS_TABLESPACE objects in 6 seconds
```

查看导入日志

```
$ cd /u02/oradata/CDB2/mydir
[FTEX] oracle@hol:/u02/oradata/CDB2/mydir
$ grep ORA- pdb2.log |sort -u
ORA-01119: error in creating database file '/u02/oradata/FTEX/undotbs100.dbf'
ORA-04042: procedure, function, package, or package body does not exist
ORA-21700: object does not exist or is marked for delete
ORA-22303: type "SYS"."JDM_ATTR_NAMES" not found
ORA-27038: created file already exists
ORA-31684: Object type CONTEXT:"DBFS_CONTEXT" already exists
ORA-31684: Object type CONTEXT:"GLOBAL_AQCLNTDB_CTX" already exists
ORA-31684: Object type CONTEXT:"REGISTRY$CTX" already exists
ORA-31684: Object type DIRECTORY:"DATA_PUMP_DIR" already exists
ORA-31684: Object type ROLE:"ADM_PARALLEL_EXECUTE_TASK" already exists
ORA-31684: Object type ROLE:"AQ_ADMINISTRATOR_ROLE" already exists
ORA-31684: Object type ROLE:"AQ_USER_ROLE" already exists
ORA-31684: Object type ROLE:"DBFS_ROLE" already exists
ORA-31684: Object type ROLE:"EXECUTE_CATALOG_ROLE" already exists
ORA-31684: Object type ROLE:"GATHER_SYSTEM_STATISTICS" already exists
ORA-31684: Object type ROLE:"GLOBAL_AQ_USER_ROLE" already exists
ORA-31684: Object type ROLE:"HS_ADMIN_EXECUTE_ROLE" already exists
ORA-31684: Object type ROLE:"HS_ADMIN_ROLE" already exists
ORA-31684: Object type ROLE:"HS_ADMIN_SELECT_ROLE" already exists
ORA-31684: Object type ROLE:"OEM_ADVISOR" already exists
ORA-31684: Object type ROLE:"OEM_MONITOR" already exists
ORA-31684: Object type ROLE:"RECOVERY_CATALOG_OWNER" already exists
ORA-31684: Object type ROLE:"SCHEDULER_ADMIN" already exists
ORA-31684: Object type ROLE:"SELECT_CATALOG_ROLE" already exists
ORA-31684: Object type TABLESPACE:"TEMP" already exists
ORA-31684: Object type USER:"OUTLN" already exists
ORA-31685: Object type USER:"SYS" failed due to insufficient privileges. Failing sql is:
ORA-31693: Table data object "SYSTEM"."SCHEDULER_PROGRAM_ARGS_TMP" failed to load/unload and is being skipped due to error:
ORA-39083: Object type PROCACT_SYSTEM failed to create with error:
ORA-39083: Object type TABLESPACE:"UNDOTBS100" failed to create with error:
[FTEX] oracle@hol:/u02/oradata/CDB2/mydir
```

迁移后后续步骤

将 FTEX 下 users 表空间置于 read write 模式

. ftex

sqlplus / as sysdba

alter tablespace users read write;

shut immediate

exit

```
$ sqlplus / as sysdba

SQL*Plus: Release 11.2.0.4.0 Production on Thu Dec 19 14:08:06 2019

Copyright (c) 1982, 2013, Oracle.  All rights reserved.


Connected to:
Oracle Database 11g Enterprise Edition Release 11.2.0.4.0 - 64bit Production
With the Partitioning, OLAP, Data Mining and Real Application Testing options

SQL> alter tablespace users read write;

Tablespace altered.

SQL> shut immediate
Database closed.
Database dismounted.
ORACLE instance shut down.
SQL> exit
Disconnected from Oracle Database 11g Enterprise Edition Release 11.2.0.4.0 - 64bit Production
With the Partitioning, OLAP, Data Mining and Real Application Testing options
[FTEX] oracle@hol:~
```

Unplug->Plug->Upgrade 升级到 19c

我们将从 12.2.0.1 版本下的 CDB1 中 Unplug 一个 PDB，然后插入到 19c 的 CDB2 中，最后升级到 19c。这个 Lab 模拟实际生产环境中多租户架构下某些特定 PDBs 的升级方法。

. cdb1

sqlplus / as sysdba
startup
show pdbs
alter pluggable database pdb3 open;
show pdbs
exit

```
SQL> startup
ORACLE instance started.

Total System Global Area 1459617792 bytes
Fixed Size                  8792872 bytes
Variable Size             486540504 bytes
Database Buffers          956301312 bytes
Redo Buffers                7983104 bytes
Database mounted.
Database opened.
SQL> show pdbs

    CON_ID CON_NAME                       OPEN MODE  RESTRICTED
---------- ------------------------------ ---------- ----------
         2 PDB$SEED                       READ ONLY  NO
         3 PDB3                           MOUNTED
SQL> alter pluggable database pdb3 open;

Pluggable database altered.

SQL> show pdbs

    CON_ID CON_NAME                       OPEN MODE  RESTRICTED
---------- ------------------------------ ---------- ----------
         2 PDB$SEED                       READ ONLY  NO
         3 PDB3                           READ WRITE NO
SQL>
```

在 CDB1 中运行预升级脚本

java -jar $OH19/rdbms/admin/preupgrade.jar -c 'PDB3' TERMINAL TEXT

```
$ java -jar $OH19/rdbms/admin/preupgrade.jar -c 'PDB3' TERMINAL TEXT
Report generated by Oracle Database Pre-Upgrade Information Tool Version
19.0.0.0.0 Build: 1 on 2019-12-18T04:52:11

Upgrade-To version: 19.0.0.0.0

====================================
```

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

      Database Name:  CDB1
     Container Name:  PDB3
       Container ID:  3
            Version:  12.2.0.1.0
     DB Patch Level:  DATABASE APR 2019 RELEASE UPDATE 12.2.0.1.190416
         Compatible:  12.2.0
          Blocksize:  8192
           Platform:  Linux x86 64-bit
      Timezone File:  26
  Database log mode:  NOARCHIVELOG
           Readonly:  FALSE
            Edition:  EE

  Oracle Component                       Upgrade Action    Current Status
  ----------------                       --------------    --------------
  Oracle Server                          [to be upgraded]  VALID
  Real Application Clusters              [to be upgraded]  OPTION OFF
  Oracle Workspace Manager               [to be upgraded]  VALID
  Oracle XML Database                    [to be upgraded]  VALID

==============
BEFORE UPGRADE
==============

  REQUIRED ACTIONS
  ================
  None

  RECOMMENDED ACTIONS
  ===================
  1.  Review and remove any unnecessary HIDDEN/UNDERSCORE parameters.

      The database contains the following initialization parameters whose name
      begins with an underscore:

      _use_single_log_writer
      _cursor_obsolete_threshold
      _fix_control
      _optimizer_aggr_groupby_elim
      _exclude_seed_cdb_view

      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.

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

```
  EXECUTE DBMS_STATS.GATHER_DICTIONARY_STATS;
```

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

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

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

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

None of the fixed object tables have had stats collected.

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

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

INFORMATION ONLY
==========

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

Tablespace	Size	Min Size For Upgrade
SYSAUX	165 MB	500 MB
SYSTEM	210 MB	244 MB
TEMP	20 MB	150 MB
UNDOTBS1	250 MB	412 MB

Minimum tablespace sizes for upgrade are estimates.

ORACLE GENERATED FIXUP SCRIPT

==========

All of the issues in database CDB1 container PDB3

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

executing the following from within the container

SQL>@/u01/app/oracle/cfgtoollogs/CDB1/preupgrade/preupgrade_fixups.sql

==========

AFTER UPGRADE

==========

REQUIRED ACTIONS

==========

None

RECOMMENDED ACTIONS

==========

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

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

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

6. To identify directory objects with symbolic links in the path name, run $ORACLE_HOME/rdbms/admin/utldirsymlink.sql AS SYSDBA after upgrade. Recreate any directory objects listed, using path names that contain no symbolic links.

   Some directory object path names may currently contain symbolic links.

   Starting in Release 18c, symbolic links are not allowed in directory object path names used with BFILE data types, the UTL_FILE package, or external tables.

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

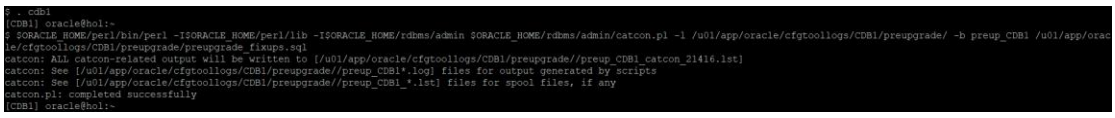
   EXECUTE DBMS_STATS.GATHER_DICTIONARY_STATS;

```
      Oracle recommends gathering dictionary statistics after upgrade.

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

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

        EXECUTE DBMS_STATS.GATHER_FIXED_OBJECTS_STATS;

      This recommendation is given for all preupgrade runs.

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

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

  ORACLE GENERATED FIXUP SCRIPT
  ============================
  All of the issues in database CDB1 container PDB3
  which are identified above as AFTER UPGRADE "(AUTOFIXUP)" can be resolved by
  executing the following from within the container

    SQL>@/u01/app/oracle/cfgtoollogs/CDB1/preupgrade/postupgrade_fixups.sql


==================
PREUPGRADE SUMMARY
==================
  /u01/app/oracle/cfgtoollogs/CDB1/preupgrade/preupgrade.log
  /u01/app/oracle/cfgtoollogs/CDB1/preupgrade/preupgrade_fixups.sql
  /u01/app/oracle/cfgtoollogs/CDB1/preupgrade/postupgrade_fixups.sql

Execute fixup scripts across the entire CDB:

Before upgrade:

1. Execute preupgrade fixups with the below command
```

```
$ORACLE_HOME/perl/bin/perl -I$ORACLE_HOME/perl/lib -I$ORACLE_HOME/rdbms/admin
$ORACLE_HOME/rdbms/admin/catcon.pl -l /u01/app/oracle/cfgtoollogs/CDB1/preupgrade/ -b preup_CDB1
/u01/app/oracle/cfgtoollogs/CDB1/preupgrade/preupgrade_fixups.sql

2. Review logs under /u01/app/oracle/cfgtoollogs/CDB1/preupgrade/

After the upgrade:

1. Execute postupgrade fixups with the below command
$ORACLE_HOME/perl/bin/perl -I$ORACLE_HOME/perl/lib -I$ORACLE_HOME/rdbms/admin
$ORACLE_HOME/rdbms/admin/catcon.pl -l /u01/app/oracle/cfgtoollogs/CDB1/preupgrade/ -b postup_CDB1
/u01/app/oracle/cfgtoollogs/CDB1/preupgrade/postupgrade_fixups.sql

2. Review logs under /u01/app/oracle/cfgtoollogs/CDB1/preupgrade/
```

在 CDB1 中执行 prefixups 脚本

$. cdb1
[CDB1] oracle@hol:~
$ $ORACLE_HOME/perl/bin/perl -I$ORACLE_HOME/perl/lib -I$ORACLE_HOME/rdbms/admin $ORACLE_HOME/rdbms/admin/catcon.pl -l /u01/app/oracle/cfgtoollogs/CDB1/preupgrade/ -b preup_CDB1 /u01/app/oracle/cfgtoollogs/CDB1/preupgrade/preupgrade_fixups.sql



Unplug PDB3 并关闭 CDB1

SQL> show pdbs
SQL> alter pluggable database PDB3 close;
SQL> show pdbs
SQL> alter pluggable database PDB3 unplug into '/home/oracle/pdb3.pdb';
SQL> show pdbs
SQL> !file /home/oracle/pdb3.pdb
SQL> !unzip -l /home/oracle/pdb3.pdb
SQL> drop pluggable database PDB3 including datafiles;
生产环境中，这一步可延迟先不删
从 12.2.0.1 开始，增加了额外的功能可以将.xml 文件和 datafiles 打包压缩在一起，这个特性就是 PDB archive），更加容易传输。这里我们可以看到，除了文件的扩展名外，拔插语法没有任何变化。如果用的文件是.pdb 而不是.xml，那么它就是一个 PDB archive。

```
SQL> !unzip -l /home/oracle/pdb3.pdb
Archive:  /home/oracle/pdb3.pdb
  Length      Date    Time    Name
---------  ---------- -----   ----
220200960  12-18-2019 05:10   system01.dbf
173015040  12-18-2019 05:10   sysaux01.dbf
262144000  12-18-2019 05:10   undotbs01.dbf
  5242880  12-18-2019 05:10   users01.dbf
     7572  12-18-2019 05:10   /home/oracle/pdb3.xml
---------                     -------
660610452                     5 files

SQL> drop pluggable database PDB3 including datafiles;

Pluggable database dropped.

SQL> show pdbs

    CON_ID CON_NAME                       OPEN MODE  RESTRICTED
---------- ------------------------------ ---------- ----------
         2 PDB$SEED                       READ ONLY  NO
SQL>
```

```
Connected to:
Oracle Database 12c Enterprise Edition Release 12.2.0.1.0 - 64bit Production

SQL> show pdbs

    CON_ID CON_NAME                       OPEN MODE  RESTRICTED
---------- ------------------------------ ---------- ----------
         2 PDB$SEED                       READ ONLY  NO
         3 PDB3                           READ WRITE NO
SQL> alter pluggable database PDB3 close;

Pluggable database altered.

SQL> show pdbs

    CON_ID CON_NAME                       OPEN MODE  RESTRICTED
---------- ------------------------------ ---------- ----------
         2 PDB$SEED                       READ ONLY  NO
         3 PDB3                           MOUNTED
SQL> alter pluggable database PDB3 unplug into '/home/oracle/pdb3.pdb';

Pluggable database altered.

SQL> show pdbs

    CON_ID CON_NAME                       OPEN MODE  RESTRICTED
---------- ------------------------------ ---------- ----------
         2 PDB$SEED                       READ ONLY  NO
         3 PDB3                           MOUNTED
SQL> !file /home/oracle/pdb3.pdb
/home/oracle/pdb3.pdb: Zip archive data, at least v2.0 to extract

SQL> !unzip -l /home/oracle/pdb3.pdb
Archive:  /home/oracle/pdb3.pdb
  Length      Date    Time    Name
---------  ---------- -----   ----
220200960  12-18-2019 05:10   system01.dbf
173015040  12-18-2019 05:10   sysaux01.dbf
262144000  12-18-2019 05:10   undotbs01.dbf
  5242880  12-18-2019 05:10   users01.dbf
     7572  12-18-2019 05:10   /home/oracle/pdb3.xml
---------                     -------
660610452                     5 files

SQL>
```

SQL> shut immediate

SQL>

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

在 CDB2 中执行兼容性检查

这个步骤不是必需的，但建议大家这么做，通过 compatibility check，可以在 plugin 之前发现并修复问题。

```
SET SERVEROUTPUT ON
DECLARE
compatible BOOLEAN := FALSE;
BEGIN
compatible := DBMS_PDB.CHECK_PLUG_COMPATIBILITY(pdb_descr_file => '/home/oracle/pdb3.pdb');
if compatible then
    DBMS_OUTPUT.PUT_LINE('Is the future PDB compatible? ==> YES');
else
    DBMS_OUTPUT.PUT_LINE('Is the future PDB compatible? ==> NO');
end if;
END;
/
```

如果输出结果是“NO”(可能实际环境中会经常遇到“不”)，那么也不要紧张，通过查看 PDB_PLUG_IN_VIOLATIONS 视图

```
set lines 156
column message format a80
column status format a9
column type format a9
column con_id format 9
select con_id, type, message, status from pdb_plug_in_violations where
status<>'RESOLVED' and type='ERROR' order by time;
```

```
SQL> SET SERVEROUTPUT ON
DECLARE
compatible BOOLEAN := FALSE;
BEGIN
compatible := DBMS_PDB.CHECK_PLUG_COMPATIBILITY(pdb_descr_file => '/home/oracle/pdb3.pdb');
if compatible then
   DBMS_OUTPUT.PUT_LINE('Is the future PDB compatible? ==> YES');
else
SQL>    DBMS_OUTPUT.PUT_LINE('Is the future PDB compatible? ==> NO');
end if;
  2    3  END;
/  4    5    6    7    8    9   10   11
Is the future PDB compatible? ==> NO

PL/SQL procedure successfully completed.

SQL>
SQL> set lines 156
column message format a80
column status format a9
column type format a9
column con_id format 9
select con_id, type, message, status from pdb_plug_in_violations where status<>'RESOLVED' and type ='ERROR' order by time;

CON_ID TYPE      MESSAGE                                                                          STATUS
------ --------- -------------------------------------------------------------------------------- ---------
     1 ERROR     PDB's version does not match CDB's version: PDB's version 12.2.0.1.0. CDB's vers PENDING
                 ion 19.0.0.0.0.

     1 ERROR     DBRU bundle patch 190416 (DATABASE APR 2019 RELEASE UPDATE 12.2.0.1.190416): Not PENDING
                  installed in the CDB but installed in the PDB

     1 ERROR     '19.3.0.0.0 Release_Update 1904101227' is installed in the CDB but no release up PENDING
                 dates are installed in the PDB


SQL>
```

从第一个可以看出，是因为版本不一致出现的错误。第二、三个是英文补丁集的原因，这两个可以忽略。下面我们来升级 PDB3 到 19c

在 CDB2 中创建 PDB3 并切换到升级模式

create pluggable database pdb3 using '/home/oracle/pdb3.pdb'
file_name_convert=('/home/oracle', '/u02/oradata/CDB2/pdb3');
alter pluggable database pdb3 open upgrade;

```
SQL> show pdbs

    CON_ID CON_NAME                       OPEN MODE  RESTRICTED
---------- ------------------------------ ---------- ----------
         2 PDB$SEED                       READ ONLY  NO
         3 PDB1                           READ WRITE NO
SQL> create pluggable database pdb3 using '/home/oracle/pdb3.pdb' file_name_convert=('/home/oracle', '/u02/oradata/CDB2/pdb3'

Pluggable database created.

SQL> show pdbs

    CON_ID CON_NAME                       OPEN MODE  RESTRICTED
---------- ------------------------------ ---------- ----------
         2 PDB$SEED                       READ ONLY  NO
         3 PDB1                           READ WRITE NO
         4 PDB3                           MOUNTED
SQL> alter pluggable database pdb3 open upgrade;

Pluggable database altered.

SQL> show pdbs

    CON_ID CON_NAME                       OPEN MODE  RESTRICTED
---------- ------------------------------ ---------- ----------
         2 PDB$SEED                       READ ONLY  NO
         3 PDB1                           READ WRITE NO
         4 PDB3                           MIGRATE    YES
SQL>
```

升级 PDB3 到 19c

. cdb2

dbupgrade -c 'PDB3' -l /home/oracle/logs -n 4

```
$ which dbupgrade
/u01/app/oracle/product/19/bin/dbupgrade
[CDB2] oracle@hol:~
$ dbupgrade -c 'PDB3' -l /home/oracle/logs -n 4

Argument list for [/u01/app/oracle/product/19/rdbms/admin/catctl.pl]
For Oracle internal use only A = 0
Run in                       c = PDB3
Do not run in                C = 0
Input Directory              d = 0
Echo OFF                     e = 1
Simulate                     E = 0
Forced cleanup               F = 0
Log Id                       i = 0
Child Process                I = 0
Log Dir                      l = /home/oracle/logs
Priority List Name           L = 0
Upgrade Mode active          M = 0
SQL Process Count            n = 4
SQL PDB Process Count        N = 0
Open Mode Normal             o = 0
Start Phase                  p = 0
End Phase                    P = 0
Reverse Order                r = 0
AutoUpgrade Resume           R = 0
Script                       s = 0
Serial Run                   S = 0
RO User Tablespaces          T = 0
Display Phases               y = 0
Debug catcon.pm              z = 0
Debug catctl.pl              Z = 0
```

```
$ dbupgrade -c 'PDB3' -l /home/oracle/logs -n 4


Argument list for [/u01/app/oracle/product/19/rdbms/admin/catctl.pl]
For Oracle internal use only A = 0
Run in                       c = PDB3
Do not run in                C = 0
Input Directory              d = 0
Echo OFF                     e = 1
Simulate                     E = 0
Forced cleanup               F = 0
Log Id                       i = 0
Child Process                I = 0
Log Dir                      l = /home/oracle/logs
Priority List Name           L = 0
Upgrade Mode active          M = 0
```

```
SQL Process Count              n = 4
SQL PDB Process Count          N = 0
Open Mode Normal               o = 0
Start Phase                    p = 0
End Phase                      P = 0
Reverse Order                  r = 0
AutoUpgrade Resume             R = 0
Script                         s = 0
Serial Run                     S = 0
RO User Tablespaces            T = 0
Display Phases                 y = 0
Debug catcon.pm                z = 0
Debug catctl.pl                Z = 0

catctl.pl VERSION: [19.0.0.0.0]
           STATUS: [Production]
            BUILD: [RDBMS_19.3.0.0.0DBRU_LINUX.X64_190417]



/u01/app/oracle/product/19/rdbms/admin/orahome = [/u01/app/oracle/product/19]
/u01/app/oracle/product/19/bin/orabasehome = [/u01/app/oracle/product/19]
catctlGetOraBaseLogDir = [/u01/app/oracle/product/19]

Analyzing file /u01/app/oracle/product/19/rdbms/admin/catupgrd.sql

Log file directory = [/home/oracle/logs]

catcon::set_log_file_base_path: ALL catcon-related output will be written to
[/home/oracle/logs/catupgrd_catcon_24316.lst]

catcon::set_log_file_base_path: catcon: See [/home/oracle/logs/catupgrd*.log] files for
output generated by scripts

catcon::set_log_file_base_path: catcon: See [/home/oracle/logs/catupgrd_*.lst] files for
spool files, if any



Number of Cpus        = 4
Database Name         = CDB2
DataBase Version      = 19.0.0.0.0
Parallel SQL Process Count (PDB)      = 2
Parallel SQL Process Count (CDB$ROOT) = 4
Concurrent PDB Upgrades               = 2
Generated PDB Inclusion:[PDB3]
```

```
CDB$ROOT  Open Mode = [OPEN]
Concurrent PDB Upgrades Reset           = 1

Start processing of PDBs (PDB3)
[/u01/app/oracle/product/19/perl/bin/perl
/u01/app/oracle/product/19/rdbms/admin/catctl.pl -c 'PDB3' -l /home/oracle/logs -n 2 -I -
i pdb3 /u01/app/oracle/product/19/rdbms/admin/catupgrd.sql]

Argument list for [/u01/app/oracle/product/19/rdbms/admin/catctl.pl]
For Oracle internal use only A = 0
Run in                      c = PDB3
Do not run in               C = 0
Input Directory             d = 0
Echo OFF                    e = 1
Simulate                    E = 0
Forced cleanup              F = 0
Log Id                      i = pdb3
Child Process               I = 1
Log Dir                     l = /home/oracle/logs
Priority List Name          L = 0
Upgrade Mode active         M = 0
SQL Process Count           n = 2
SQL PDB Process Count       N = 0
Open Mode Normal            o = 0
Start Phase                 p = 0
End Phase                   P = 0
Reverse Order               r = 0
AutoUpgrade Resume          R = 0
Script                      s = 0
Serial Run                  S = 0
RO User Tablespaces         T = 0
Display Phases              y = 0
Debug catcon.pm             z = 0
Debug catctl.pl             Z = 0

catctl.pl VERSION: [19.0.0.0.0]
          STATUS: [Production]
           BUILD: [RDBMS_19.3.0.0.0DBRU_LINUX.X64_190417]



/u01/app/oracle/product/19/rdbms/admin/orahome = [/u01/app/oracle/product/19]
/u01/app/oracle/product/19/bin/orabasehome = [/u01/app/oracle/product/19]
catctlGetOraBaseLogDir = [/u01/app/oracle/product/19]
```

```
Analyzing file /u01/app/oracle/product/19/rdbms/admin/catupgrd.sql

Log file directory = [/home/oracle/logs]

catcon::set_log_file_base_path: ALL catcon-related output will be written to
[/home/oracle/logs/catupgrdpdb3_catcon_24561.lst]

catcon::set_log_file_base_path: catcon: See [/home/oracle/logs/catupgrdpdb3*.log] files
for output generated by scripts

catcon::set_log_file_base_path: catcon: See [/home/oracle/logs/catupgrdpdb3_*.lst] files
for spool files, if any



Number of Cpus        = 4
Database Name         = CDB2
DataBase Version      = 19.0.0.0.0
PDB3 Open Mode = [MIGRATE]
Generated PDB Inclusion:[PDB3]
CDB$ROOT  Open Mode = [OPEN]
Components in [PDB3]
    Installed [CATALOG CATPROC OWM XDB]
Not Installed [APEX APS CATJAVA CONTEXT DV EM JAVAVM MGW ODM OLS ORDIM RAC SDO WK XML
XOQ]

------------------------------------------------------
Phases [0-107]         Start Time:[2019_12_18 05:38:26]
Container Lists Inclusion:[PDB3] Exclusion:[NONE]
------------------------------------------------------
**********   Executing Change Scripts   **********
Serial   Phase #:0    [PDB3] Files:1    Time: 23s
***************   Catalog Core SQL   ***************
Serial   Phase #:1    [PDB3] Files:5    Time: 47s
Restart  Phase #:2    [PDB3] Files:1    Time: 2s
**********   Catalog Tables and Views   **********
Parallel Phase #:3    [PDB3] Files:19   Time: 23s
Restart  Phase #:4    [PDB3] Files:1    Time: 2s
*************   Catalog Final Scripts   *************
Serial   Phase #:5    [PDB3] Files:7    Time: 22s
*****************   Catproc Start   ****************
Serial   Phase #:6    [PDB3] Files:1    Time: 15s
*****************   Catproc Types   ****************
Serial   Phase #:7    [PDB3] Files:2    Time: 13s
Restart  Phase #:8    [PDB3] Files:1    Time: 1s
```

```
****************  Catproc Tables  ****************
Parallel Phase #:9    [PDB3] Files:67   Time: 32s
Restart  Phase #:10   [PDB3] Files:1    Time: 2s
*************  Catproc Package Specs  *************
Serial   Phase #:11   [PDB3] Files:1    Time: 55s
Restart  Phase #:12   [PDB3] Files:1    Time: 2s
**************  Catproc Procedures  **************
Parallel Phase #:13   [PDB3] Files:94   Time: 12s
Restart  Phase #:14   [PDB3] Files:1    Time: 1s
Parallel Phase #:15   [PDB3] Files:120  Time: 14s
Restart  Phase #:16   [PDB3] Files:1    Time: 2s
Serial   Phase #:17   [PDB3] Files:22   Time: 8s
Restart  Phase #:18   [PDB3] Files:1    Time: 1s
*****************  Catproc Views  *****************
Parallel Phase #:19   [PDB3] Files:32   Time: 21s
Restart  Phase #:20   [PDB3] Files:1    Time: 2s
Serial   Phase #:21   [PDB3] Files:3    Time: 17s
Restart  Phase #:22   [PDB3] Files:1    Time: 2s
Parallel Phase #:23   [PDB3] Files:25   Time: 151s
Restart  Phase #:24   [PDB3] Files:1    Time: 2s
Parallel Phase #:25   [PDB3] Files:12   Time: 97s
Restart  Phase #:26   [PDB3] Files:1    Time: 1s
Serial   Phase #:27   [PDB3] Files:1    Time: 0s
Serial   Phase #:28   [PDB3] Files:3    Time: 8s
Serial   Phase #:29   [PDB3] Files:1    Time: 0s
Restart  Phase #:30   [PDB3] Files:1    Time: 2s
***************  Catproc CDB Views  ***************
Serial   Phase #:31   [PDB3] Files:1    Time: 6s
Restart  Phase #:32   [PDB3] Files:1    Time: 2s
Serial   Phase #:34   [PDB3] Files:1    Time: 0s
*****************  Catproc PLBs  *****************
Serial   Phase #:35   [PDB3] Files:293  Time: 24s
Serial   Phase #:36   [PDB3] Files:1    Time: 0s
Restart  Phase #:37   [PDB3] Files:1    Time: 2s
Serial   Phase #:38   [PDB3] Files:6    Time: 7s
Restart  Phase #:39   [PDB3] Files:1    Time: 1s
***************  Catproc DataPump  ***************
Serial   Phase #:40   [PDB3] Files:3    Time: 43s
Restart  Phase #:41   [PDB3] Files:1    Time: 2s
******************  Catproc SQL  ******************
Parallel Phase #:42   [PDB3] Files:13   Time: 104s
Restart  Phase #:43   [PDB3] Files:1    Time: 2s
Parallel Phase #:44   [PDB3] Files:11   Time: 10s
Restart  Phase #:45   [PDB3] Files:1    Time: 1s
```

```
Parallel Phase #:46   [PDB3] Files:3    Time: 9s
Restart  Phase #:47   [PDB3] Files:1    Time: 2s
************  Final Catproc scripts   ************
Serial   Phase #:48   [PDB3] Files:1    Time: 12s
Restart  Phase #:49   [PDB3] Files:1    Time: 2s
**************  Final RDBMS scripts   **************
Serial   Phase #:50   [PDB3] Files:1    Time: 9s
************  Upgrade Component Start   ************
Serial   Phase #:51   [PDB3] Files:1    Time: 6s
Restart  Phase #:52   [PDB3] Files:1    Time: 2s
**********  Upgrading Java and non-Java   **********
Serial   Phase #:53   [PDB3] Files:2    Time: 26s
*****************  Upgrading XDB   *****************
Restart  Phase #:54   [PDB3] Files:1    Time: 1s
Serial   Phase #:56   [PDB3] Files:3    Time: 12s
Serial   Phase #:57   [PDB3] Files:3    Time: 9s
Parallel Phase #:58   [PDB3] Files:10   Time: 11s
Parallel Phase #:59   [PDB3] Files:25   Time: 12s
Serial   Phase #:60   [PDB3] Files:4    Time: 15s
Serial   Phase #:61   [PDB3] Files:1    Time: 0s
Serial   Phase #:62   [PDB3] Files:32   Time: 10s
Serial   Phase #:63   [PDB3] Files:1    Time: 0s
Parallel Phase #:64   [PDB3] Files:6    Time: 14s
Serial   Phase #:65   [PDB3] Files:2    Time: 26s
Serial   Phase #:66   [PDB3] Files:3    Time: 29s
****************  Upgrading ORDIM   ****************
Restart  Phase #:67   [PDB3] Files:1    Time: 1s
Serial   Phase #:69   [PDB3] Files:1    Time: 9s
Parallel Phase #:70   [PDB3] Files:2    Time: 9s
Restart  Phase #:71   [PDB3] Files:1    Time: 3s
Parallel Phase #:72   [PDB3] Files:2    Time: 9s
Serial   Phase #:73   [PDB3] Files:2    Time: 8s
*****************  Upgrading SDO   *****************
Restart  Phase #:74   [PDB3] Files:1    Time: 2s
Serial   Phase #:76   [PDB3] Files:1    Time: 8s
Serial   Phase #:77   [PDB3] Files:2    Time: 8s
Restart  Phase #:78   [PDB3] Files:1    Time: 2s
Serial   Phase #:79   [PDB3] Files:1    Time: 8s
Restart  Phase #:80   [PDB3] Files:1    Time: 2s
Parallel Phase #:81   [PDB3] Files:3    Time: 9s
Restart  Phase #:82   [PDB3] Files:1    Time: 2s
Serial   Phase #:83   [PDB3] Files:1    Time: 8s
Restart  Phase #:84   [PDB3] Files:1    Time: 2s
Serial   Phase #:85   [PDB3] Files:1    Time: 9s
```

```
Restart  Phase #:86   [PDB3] Files:1    Time: 1s
Parallel Phase #:87   [PDB3] Files:4    Time: 10s
Restart  Phase #:88   [PDB3] Files:1    Time: 2s
Serial   Phase #:89   [PDB3] Files:1    Time: 9s
Restart  Phase #:90   [PDB3] Files:1    Time: 2s
Serial   Phase #:91   [PDB3] Files:2    Time: 8s
Restart  Phase #:92   [PDB3] Files:1    Time: 2s
Serial   Phase #:93   [PDB3] Files:1    Time: 6s
Restart  Phase #:94   [PDB3] Files:1    Time: 2s
*******   Upgrading ODM, WK, EXF, RUL, XOQ   *******
Serial   Phase #:95   [PDB3] Files:1    Time: 6s
Restart  Phase #:96   [PDB3] Files:1    Time: 2s
***********   Final Component scripts    ***********
Serial   Phase #:97   [PDB3] Files:1    Time: 7s
*************   Final Upgrade scripts   *************
Serial   Phase #:98   [PDB3] Files:1    Time: 69s
*******************   Migration   ******************
Serial   Phase #:99   [PDB3] Files:1    Time: 7s
***   End PDB Application Upgrade Pre-Shutdown   ***
Serial   Phase #:100  [PDB3] Files:1    Time: 6s
Serial   Phase #:101  [PDB3] Files:1    Time: 5s
Serial   Phase #:102  [PDB3] Files:1    Time: 6s
*****************   Post Upgrade   *****************
Serial   Phase #:103  [PDB3] Files:1    Time: 19s
****************   Summary report   ****************
Serial   Phase #:104  [PDB3] Files:1    Time: 6s
***   End PDB Application Upgrade Post-Shutdown   **
Serial   Phase #:105  [PDB3] Files:1    Time: 7s
Serial   Phase #:106  [PDB3] Files:1    Time: 3s
Serial   Phase #:107  [PDB3] Files:1     Time: 0s

------------------------------------------------------
Phases [0-107]         End Time:[2019_12_18 05:59:42]
Container Lists Inclusion:[PDB3] Exclusion:[NONE]
------------------------------------------------------

Grand Total Time: 1278s [PDB3]

 LOG FILES: (/home/oracle/logs/catupgrdpdb3*.log)

Upgrade Summary Report Located in:
/home/oracle/logs/upg_summary.log

     Time: 1354s For PDB(s)
```

```
Grand Total Time: 1354s

 LOG FILES: (/home/oracle/logs/catupgrd*.log)


Grand Total Upgrade Time:    [0d:0h:22m:34s]
[CDB2] oracle@hol:~
$
```

```
-----------------------------------------------------
Phases [0-107]         End Time:[2019_12_18 05:59:42]
Container Lists Inclusion:[PDB3] Exclusion:[NONE]
-----------------------------------------------------

Grand Total Time: 1278s [PDB3]

 LOG FILES: (/home/oracle/logs/catupgrdpdb3*.log)

Upgrade Summary Report Located in:
/home/oracle/logs/upg_summary.log

     Time: 1354s For PDB(s)

Grand Total Time: 1354s

 LOG FILES: (/home/oracle/logs/catupgrd*.log)


Grand Total Upgrade Time:    [0d:0h:22m:34s]
[CDB2] oracle@hol:~
```

在 PDB3 执行后续步骤

sqlplus / as sysdba
alter session set container=PDB3;
startup
select count(*) from dba_objects where status='INVALID';

```
SQL> show pdbs

    CON_ID CON_NAME                       OPEN MODE  RESTRICTED
---------- ------------------------------ ---------- ----------
         2 PDB$SEED                       READ ONLY  NO
         3 PDB1                           READ WRITE NO
         4 PDB3                           MOUNTED
SQL> alter session set container=pdb3;

Session altered.

SQL> startup
Pluggable Database opened.
SQL> show pdbs

    CON_ID CON_NAME                       OPEN MODE  RESTRICTED
---------- ------------------------------ ---------- ----------
         4 PDB3                           READ WRITE NO
SQL> select count(*) from dba_objects where status='INVALID';

  COUNT(*)
----------
       914
```

@?/rdbms/admin/utlrp.sql

```
Function dropped.


PL/SQL procedure successfully completed.

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

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

$ORACLE_HOME/perl/bin/perl -I$ORACLE_HOME/perl/lib -I$ORACLE_HOME/rdbms/admin $ORACLE_HOME/rdbms/admin/catcon.pl -l /u01/app/oracle/cfgtoollogs/CDB1/preupgrade/ -b postup_CDB1 /u01/app/oracle/cfgtoollogs/CDB1/preupgrade/postupgrade_fixups.sql

到这里，我们就完成一个 12.2.0.1 的 PDB 通过拔插的方式升级到 19.3 下 CDB 的练习了，是不是很简单

升级回退

在升级回退部分，我们使用 11.2.0.4 版本的 non-CDB FTEX 数据库。这里我们将数据库升级的回退策略主要分为两个方面：

- 升级过程中遇到问题
- 升级后遇到问题

针对以上 2 个场景中，分别体验 2 种不同的回退方案。

在实际生产环境中，我们必须在升级期间和升级后保护好既有的生产环境，当然了如果遇到问题，我们也必须事先规划好一个可以接受的回退服务水平协议，比如是以秒、分钟、小时或天为单位的回退需求。有一点需要明确：有些回退策略是不允许更改升级后的兼容性参数的，也就说是这种情况下，可能需要额外的停机时间来修改兼容性，因为这涉及到数据库重启。

在回退练习中，我们这个实验中不考虑使用 RMAN 在线备份技术，一般生产系统中实施这样的变更操作，这个是基本要求，大家都很了解，是最经典的回退方案了。受本次实验环境所限，我们也不讨论使用 Oracle Golden Gate 技术实现回退，如果大家对 Golden Gate 感兴趣，可参考其他资料来学习。

升级中遇到问题回退

在本部分中，我们练习两种技术来保护数据库，以防升级期间发生问题需要回退到原来的版本，也适用于需要进行多次测试的场景。我们将练习 Partial Offline Backups 和 Guaranteed Restore Points(GRP)两个方案。

Partial Offline Backup

用于防止升级期间发生故障，或用于测试目的，以避免恢复整个数据库环境。

Partial Offline Backup 适用的场景

- 特别是那种很大的数据库（比如几十 T 什么几百 TB 的），当需要恢复的时候，我们只需要恢复一小部分数据即可，这相对恢复整个数据库来说要快的多。
- 数据库运行在 NOARCHIVELOG 模式下，所以无法实现 online 备份和恢复
- 数据库是 DBSE，无法使用闪回特性（这里是指 Guaranteed Restore Points）

对于采用 Partial Offline Backup 回退策略的情况，必须将所有用户和数据表空间置于 read-only 模式，然后创建数据库"核心"的离线备份。这里的"核心"是指 redolog，控制文件，系统表空间、undo 表空间以及 temp 表空间。

模拟升级故障

切换到 FTEX 数据库

首先将 USERS 表空间设置为 read-only， 然后关闭数据库
. ftex
sqlplus / as sysdba
startup
alter tablespace USERS read only;
insert into SYSTEM.TRACKING_TAB values (1,'partial offline backup');
commit;
注意：这里插入一条记录，用于回退验证
shut immediate

```
SQL> startup
ORACLE instance started.

Total System Global Area 1152450560 bytes
Fixed Size                  2252584 bytes
Variable Size             335544536 bytes
Database Buffers          805306368 bytes
Redo Buffers                9347072 bytes
Database mounted.
Database opened.
SQL> alter tablespace USERS read only;

Tablespace altered.

SQL> insert into SYSTEM.TRACKING_TAB values (1,'partial offline backup');

1 row created.

SQL> commit;

Commit complete.

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

备份 FTEX 核心文件

. /home/oracle/scripts/backupFTEX.sh

```
$ cat /home/oracle/scripts/backupFTEX.sh
#!/bin/sh
cp /u02/oradata/FTEX/*.log /home/oracle/FTEX/bck
cp /u02/oradata/FTEX/*.ctl /home/oracle/FTEX/bck
cp /u02/oradata/FTEX/sys*.dbf /home/oracle/FTEX/bck
cp /u02/oradata/FTEX/undo*.dbf /home/oracle/FTEX/bck
cp /u02/oradata/FTEX/temp*.dbf /home/oracle/FTEX/bck



[FTEX] oracle@hol:~
$ . /home/oracle/scripts/backupFTEX.sh
[FTEX] oracle@hol:~
```

用 19c ORACLE_HOME 打开 FTEX

cd $ORACLE_HOME/dbs
. ftex19
cp spfileFTEX.ora $ORACLE_HOME/dbs
cp orapwFTEX $ORACLE_HOME/dbs
sqlplus / as sysdba
startup upgrade
exit

```
$ . ftex
[FTEX] oracle@hol:~
$ cd $ORACLE_HOME/dbs
[FTEX] oracle@hol:/u01/app/oracle/product/11.2.0.4/dbs
$ . ftex19
[FTEX] oracle@hol:/u01/app/oracle/product/11.2.0.4/dbs
$ cp spfileFTEX.ora $ORACLE_HOME/dbs
[FTEX] oracle@hol:/u01/app/oracle/product/11.2.0.4/dbs
$ cp orapwFTEX $ORACLE_HOME/dbs
[FTEX] oracle@hol:/u01/app/oracle/product/11.2.0.4/dbs
$ sqlplus / as sysdba

SQL*Plus: Release 19.0.0.0.0 - Production on Wed Dec 18 10:37:16 2019
Version 19.3.0.0.0

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

Connected to an idle instance.

SQL> startup upgrade
ORACLE instance started.

Total System Global Area 1157627168 bytes
Fixed Size                  8895776 bytes
Variable Size             335544320 bytes
Database Buffers          805306368 bytes
Redo Buffers                7880704 bytes
Database mounted.
Database opened.
SQL>
```

升级 FTEX 数据库并中断升级

dbupgrade -l /home/oracle/logs

```
$ cd
[FTEX] oracle@hol:~
$ dbupgrade -l /home/oracle/logs

Argument list for [/u01/app/oracle/product/19/rdbms/admin/catctl.pl]
For Oracle internal use only A = 0
Run in                       c = 0
Do not run in                C = 0
Input Directory              d = 0
Echo OFF                     e = 1
Simulate                     E = 0
Forced cleanup               F = 0
Log Id                       i = 0
Child Process                I = 0
Log Dir                      l = /home/oracle/logs
Priority List Name           L = 0
Upgrade Mode active          M = 0
SQL Process Count            n = 0
SQL PDB Process Count        N = 0
Open Mode Normal             o = 0
Start Phase                  p = 0
End Phase                    P = 0
Reverse Order                r = 0
AutoUpgrade Resume           R = 0
Script                       s = 0
Serial Run                   S = 0
RO User Tablespaces          T = 0
Display Phases               y = 0
Debug catcon.pm              z = 0
Debug catctl.pl              Z = 0

catctl.pl VERSION: [19.0.0.0.0]
           STATUS: [Production]
            BUILD: [RDBMS_19.3.0.0.0DBRU_LINUX.X64_190417]


/u01/app/oracle/product/19/rdbms/admin/orahome = [/u01/app/oracle/product/19]
/u01/app/oracle/product/19/bin/orabasehome = [/u01/app/oracle/product/19]
catctlGetOraBaseLogDir = [/u01/app/oracle/product/19]
```

执行 2-3 分钟后，当 Serial Phase #:7 完成后，按下 CTRL-C 中断升级操作，模拟升级故障。

```
Number of Cpus        = 4
Database Name         = FTEX
DataBase Version      = 11.2.0.4.0
Parallel SQL Process Count          = 4
Components in [FTEX]
    Installed [CATALOG CATPROC]
Not Installed [APEX APS CATJAVA CONTEXT DV EM JAVAVM MGW ODM OLS ORDIM OWM RAC SDO WK XDB XML XOQ]

-----------------------------------------------------
Phases [0-107]        Start Time:[2019_12_18 10:39:47]
-----------------------------------------------------
**********   Executing Change Scripts   **********
Serial   Phase #:0    [FTEX] Files:1    Time: 69s
***************   Catalog Core SQL   ***************
Serial   Phase #:1    [FTEX] Files:5    Time: 48s
Restart  Phase #:2    [FTEX] Files:1    Time: 4s
**********   Catalog Tables and Views   **********
Parallel Phase #:3    [FTEX] Files:19   Time: 13s
Restart  Phase #:4    [FTEX] Files:1    Time: 4s
*************   Catalog Final Scripts   *************
Serial   Phase #:5    [FTEX] Files:7    Time: 17s
*****************   Catproc Start   *****************
Serial   Phase #:6    [FTEX] Files:1    Time: 13s
*****************   Catproc Types   *****************
Serial   Phase #:7    [FTEX] Files:2 ^Ccatcon::catcon_HandleSigINT: Signal INT was received.


Unexpected error encountered in catctlMain; Error Stack Below; exiting
Died at /u01/app/oracle/product/19/rdbms/admin/catcon.pm line 18102, <Reader> line 18.
 at /u01/app/oracle/product/19/rdbms/admin/catcon.pm line 18102, <Reader> line 18.
        catcon::catcon_HandleSigINT("INT") called at /u01/app/oracle/product/19/rdbms/admin/catcon.pm line 1176
        eval {...} called at /u01/app/oracle/product/19/rdbms/admin/catcon.pm line 1176
        catcon::exec_DB_script(ARRAY(0x2aad710), "C:A:T:C:O:N", "\x{a}host sqlplus -v >", "/home/oracle/logs/catupgrd_catcon_17903", "/u01/app/
/app/oracle/product/19/rdbms/admin/catcon.pm line 18031
        catcon::catconQuery("SELECT version FROM v\$instance", undef, 0) called at /u01/app/oracle/product/19/rdbms/admin/catctl.pl line 10090
        main::catctlAryQuery("SELECT version FROM v\$instance", undef, 0) called at /u01/app/oracle/product/19/rdbms/admin/catctl.pl line 10057
        main::catctlQuery("SELECT version FROM v\$instance", undef) called at /u01/app/oracle/product/19/rdbms/admin/catctl.pl line 12051
        main::catctlAutoPhaseTrace(7, "I", undef, undef) called at /u01/app/oracle/product/19/rdbms/admin/catctl.pl line 5777
        main::catctlStartPhase(7, undef, undef) called at /u01/app/oracle/product/19/rdbms/admin/catctl.pl line 2020
```

```
----------------------------------------------------
CATCTL FATAL ERROR
----------------------------------------------------

 LOG FILES: (/home/oracle/logs/catupgrd*.log)
TRACE FILE: (/home/oracle/logs/catupgrd_trace.log)

catcon: See [/home/oracle/logs/catupgrd*.log] files for output generated by scripts

catcon: See [/home/oracle/logs/catupgrd_*.lst] files for spool files, if any


!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

catcon version: /st_rdbms_19/10
       catconInit2: start logging catcon output at 2019-12-18 10:39:25

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!


2019-12-18 10:42:36 INFO> /u01/app/oracle/product/19/rdbms/admin/catcon.pm:18092 catcon::catcon_HandleSigINT - Signal INT was received.


Grand Total Time: 169s

 LOG FILES: (/home/oracle/logs/catupgrd*.log)
[FTEX] oracle@hol:~
$
```

重新调度升级程序再次中断升级

dbupgrade -l /home/oracle/logs -R

```
[FTEX] oracle@hol:~
$ dbupgrade -l /home/oracle/logs -R

Argument list for [/u01/app/oracle/product/19/rdbms/admin/catctl.pl]
For Oracle internal use only A = 0
Run in                       c = 0
Do not run in                C = 0
Input Directory              d = 0
Echo OFF                     e = 1
Simulate                     E = 0
Forced cleanup               F = 0
Log Id                       i = 0
Child Process                I = 0
Log Dir                      l = /home/oracle/logs
Priority List Name           L = 0
Upgrade Mode active          M = 0
SQL Process Count            n = 0
SQL PDB Process Count        N = 0
Open Mode Normal             o = 0
Start Phase                  p = 0
End Phase                    P = 0
Reverse Order                r = 0
AutoUpgrade Resume           R = 1
Script                       s = 0
Serial Run                   S = 0
RO User Tablespaces          T = 0
Display Phases               y = 0
Debug catcon.pm              z = 0
Debug catctl.pl              Z = 0

catctl.pl VERSION: [19.0.0.0.0]
          STATUS: [Production]
           BUILD: [RDBMS_19.3.0.0.0DBRU_LINUX.X64_190417]
```

```
Number of Cpus        = 4
Database Name         = FTEX
DataBase Version      = 11.2.0.4.0
Parallel SQL Process Count            = 4
Components in [FTEX]
    Installed [CATALOG CATPROC]
Not Installed [APEX APS CATJAVA CONTEXT DV EM JAVAVM MGW ODM OLS ORDIM OWM RAC SDO WK XDB XML XOQ]

** Database FTEX has already been upgraded successfully. **

------------------------------------------------------
Phases [9-107]         Start Time:[2019_12_18 10:49:54]
------------------------------------------------------
   Time: 24s
****************   Catproc Tables   ****************
Parallel Phase #:9    [FTEX] Files:67    Time: 18s
Restart  Phase #:10   [FTEX] Files:1     Time: 5s
*************   Catproc Package Specs   ************
Serial   Phase #:11   [FTEX] Files:1     Time: 57s
Restart  Phase #:12   [FTEX] Files:1
```

运行一段时间后，再次按下 CTRL-C 中断升级操作，模拟升级故障。到这里我们假设故障无法恢复，要取消本次升级，所以需要执行回退方案。

```
Parallel Phase #:23   [FTEX] Files:25    Time: 139s
Restart  Phase #:24   [FTEX] Files:1     Time: 4s
Parallel Phase #:25   [FTEX] Files:12 ^Ccatcon::catcon_HandleSigINT: Signal INT was received.


Unexpected error encountered in catctlMain; Error Stack Below; exiting
Died at /u01/app/oracle/product/19/rdbms/admin/catcon.pm line 18102.
 at /u01/app/oracle/product/19/rdbms/admin/catcon.pm line 18102.
        catcon::catcon_HandleSigINT("INT") called at /u01/app/oracle/product/19/rdbms/admin/catc
        eval {...} called at /u01/app/oracle/product/19/rdbms/admin/catcon.pm line 9860
        catcon::next_proc(4, 4, 3, ARRAY(0x2538bc0), "/home/oracle/logs/catupgrd", ARRAY(0x25888
e 9939
        catcon::wait_for_completion(4, 4, "/home/oracle/logs/catupgrd", ARRAY(0x2588870), ARRAY(
/product/19/rdbms/admin/catcon.pm line 15535
        catcon::catconExec_int(ARRAY(0x28a5bb8), 0, 0, 0, SCALAR(0x28a02b0), SCALAR(0x28a0388),
.pm line 14480
        catcon::catconExec(ARRAY(0x28a5bb8), 0, 0, 0, SCALAR(0x28a02b0), SCALAR(0x28a0388), SCAL
line 5679
        main::catctlExecutePhaseFiles(25, 12, undef, undef, undef) called at /u01/app/oracle/prc
        main::catctlRunPhase(25, 12, undef, undef, undef) called at /u01/app/oracle/product/19/r
        main::catctlRunPhases(9, 108, 108, undef, undef, undef) called at /u01/app/oracle/produc
        main::catctlRunMainPhases() called at /u01/app/oracle/product/19/rdbms/admin/catctl.pl l
        main::catctlMain() called at /u01/app/oracle/product/19/rdbms/admin/catctl.pl line 1362
        eval {...} called at /u01/app/oracle/product/19/rdbms/admin/catctl.pl line 1360
 at /u01/app/oracle/product/19/rdbms/admin/catcon.pm line 9860.
```

```
CATCTL FATAL ERROR
------------------------------------------------------

 LOG FILES: (/home/oracle/logs/catupgrd*.log)
TRACE FILE: (/home/oracle/logs/catupgrd_trace.log)

catcon: See [/home/oracle/logs/catupgrd*.log] files for output generated by scripts

catcon: See [/home/oracle/logs/catupgrd_*.lst] files for spool files, if any


!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

catcon version: /st_rdbms_19/10
       catconInit2: start logging catcon output at 2019-12-18 10:49:30

!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!



2019-12-18 10:56:13 INFO> /u01/app/oracle/product/19/rdbms/admin/catcon.pm:18092 catcon::catcon_HandleSigINT


Grand Total Time: 404s

 LOG FILES: (/home/oracle/logs/catupgrd*.log)
[FTEX] oracle@hol:~
```

恢复 FTEX 到升级前状态

关闭 FTEX 数据库

SQL> shut immediate

```
Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
Version 19.3.0.0.0

SQL> shut immediate
Database closed.
Database dismounted.
ORACLE instance shut down.
SQL> exit
Disconnected from Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
Version 19.3.0.0.0
```

还原 FTEX 数据库

. /home/oracle/scripts/restoreFTEX.sh

```
$ . /home/oracle/scripts/restoreFTEX.sh
[FTEX] oracle@hol:~
$ cat /home/oracle/scripts/restoreFTEX.sh
#!/bin/sh
cp /home/oracle/FTEX/bck/* /u02/oradata/FTEX

[FTEX] oracle@hol:~
```

打开 11204 FTEX 并验证

. ftex
sqlplus / as sysdba
startup
alter tablespace USERS read write;
select * from SYSTEM.TRACKING_TAB;
exit

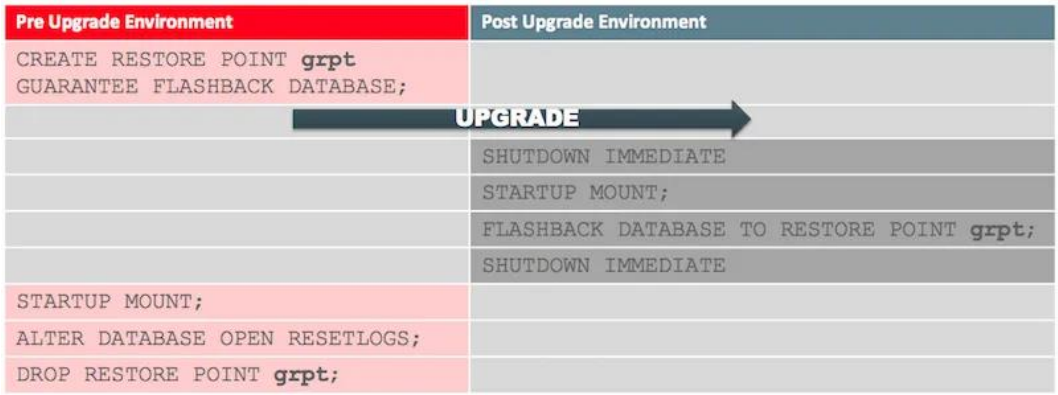
```
[FTEX] oracle@hol:~
$ . ftex
[FTEX] oracle@hol:~
$ sqlplus / as sysdba

SQL*Plus: Release 11.2.0.4.0 Production on Wed Dec 18 11:03:04 2019

Copyright (c) 1982, 2013, Oracle.  All rights reserved.

Connected to an idle instance.

SQL> startup
ORACLE instance started.

Total System Global Area 1152450560 bytes
Fixed Size                  2252584 bytes
Variable Size             335544536 bytes
Database Buffers          805306368 bytes
Redo Buffers                9347072 bytes
Database mounted.
Database opened.
SQL> alter tablespace USERS read write;

Tablespace altered.

SQL> select * from SYSTEM.TRACKING_TAB;

        NO ACTION
---------- ------------------------------
         0 Nothing has been done yet
         1 partial offline backup

SQL>
```

闪回数据库到保证恢复点(GRP)

到目前为止，保护 Oracle 数据库升级过程的最佳和最简单的技术是借助闪回数据库特性，通过保证 Guaranteed Restore Points 来实现。但也必须在同时满足以下要求时才能使用:

- 数据库必须运行在归档模式下
- 数据库必须是企业版，或者 XE 和 PE 版本
- 升级以后不能修改兼容性参数

下表是关于如何使用一个有保证的恢复点 GRP1 进行闪回数据库的概述，它允许我们多次闪回数据库



升级前环境	升级后环境
create restore point grpt guarantee flashback database;	
执行升级操作	
	shut immediate; startup mount; flashback database to restore point grpt; **shut immediate**;
startup mount; alter database open resetlogs; drop restore point grpt;	

在 FTEX 中执行下面的操作

打开归档模式

$. ftex
$ sqlplus / as sysdba
startup mount
alter database archivelog;
alter database open;
archive log list;
insert into SYSTEM.TRACKING_TAB values (2,'guaranteed restore point');
注意：这个插入一条数据是为了验证用
commit;

然后创建还原点 GRP1

create restore point GRP1 guarantee flashback database;
shut immediate
exit

```
$ sqlplus / as sysdba

SQL*Plus: Release 11.2.0.4.0 Production on Wed Dec 18 11:24:26 2019

Copyright (c) 1982, 2013, Oracle.  All rights reserved.

Connected to an idle instance.

SQL> startup mount
ORACLE instance started.

Total System Global Area 1152450560 bytes
Fixed Size                  2252584 bytes
Variable Size             335544536 bytes
Database Buffers          805306368 bytes
Redo Buffers                9347072 bytes
Database mounted.
SQL> alter database archivelog;

Database altered.

SQL> alter database open;

Database altered.

SQL> archive log list;
Database log mode              Archive Mode
Automatic archival             Enabled
Archive destination            USE_DB_RECOVERY_FILE_DEST
Oldest online log sequence     22
Next log sequence to archive   24
Current log sequence           24
SQL> insert into SYSTEM.TRACKING_TAB values (2,'guaranteed restore point');

1 row created.

SQL> commit;

Commit complete.

SQL> create restore point GRP1 guarantee flashback database;

Restore point created.

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

升级 FTEX 数据库到 19c

. ftex19
sqlplus / as sysdba
startup upgrade

```
$ . ftex19
[FTEX] oracle@hol:~
$ sqlplus / as sysdba

SQL*Plus: Release 19.0.0.0.0 - Production on Wed Dec 18 11:36:58 2019
Version 19.3.0.0.0

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

Connected to an idle instance.

SQL> startup upgrade
ORACLE instance started.

Total System Global Area 1157627168 bytes
Fixed Size                  8895776 bytes
Variable Size             369098752 bytes
Database Buffers          771751936 bytes
Redo Buffers                7880704 bytes
Database mounted.
Database opened.
SQL> exit
Disconnected from Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
Version 19.3.0.0.0
[FTEX] oracle@hol:~
```

dbupgrade -l /home/oracle/logs -n 2

```
SQL> startup upgrade
ORACLE instance started.

Total System Global Area 1157627168 bytes
Fixed Size                  8895776 bytes
Variable Size             369098752 bytes
Database Buffers          771751936 bytes
Redo Buffers                7880704 bytes
Database mounted.
Database opened.
SQL> exit
Disconnected from Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
Version 19.3.0.0.0
[FTEX] oracle@hol:~
$ dbupgrade -l /home/oracle/logs -n 2

Argument list for [/u01/app/oracle/product/19/rdbms/admin/catctl.pl]
For Oracle internal use only A = 0
Run in                       c = 0
Do not run in                C = 0
Input Directory              d = 0
Echo OFF                     e = 1
Simulate                     E = 0
Forced cleanup               F = 0
Log Id                       i = 0
Child Process                I = 0
Log Dir                      l = /home/oracle/logs
Priority List Name           L = 0
```

```
Serial   Phase #:106  [FTEX] Files:1    Time: 0s
Serial   Phase #:107  [FTEX] Files:1     Time: 27s

------------------------------------------------------
Phases [0-107]         End Time:[2019_12_18 12:05:53]
------------------------------------------------------

Grand Total Time: 1256s

 LOG FILES: (/home/oracle/logs/catupgrd*.log)

Upgrade Summary Report Located in:
/home/oracle/logs/upg_summary.log

Grand Total Upgrade Time:    [0d:0h:20m:56s]
[FTEX] oracle@hol:~
```

升级过程大约需要 15~30 分钟，这取决于你的硬件环境。
说明：如果你不想等这么久，可以按 CTRL-C,模拟升级中故障，直接闪回到 GRP1
在本练习中，我们等待升级完成，将数据库升级到 19c 在后面的降级回退中也使用这个数据库。

执行升级后的步骤

. ftex19
sqlplus / as sysdba
startup
@?/rdbms/admin/utlrp.sql
插入一条升级后的验证记录
insert into SYSTEM.TRACKING_TAB values (3,'upgrade completed');
commit;
select * from SYSTEM.TRACKING_TAB;
查看回退验证记录

```
ERRORS DURING RECOMPILATION
---------------------------
                          0


Function created.


PL/SQL procedure successfully completed.


Function dropped.


PL/SQL procedure successfully completed.

SQL> insert into SYSTEM.TRACKING_TAB values (3,'upgrade completed');

1 row created.

SQL> commit;

Commit complete.

SQL> select * from SYSTEM.TRACKING_TAB;

        NO ACTION
---------- ------------------------------
         0 Nothing has been done yet
         1 partial offline backup
         2 guaranteed restore point
         3 upgrade completed

SQL>
```

创建还原点 GRP2

create restore point GRP2 guarantee flashback database;

闪回到升级前

在升级后的环境中，创建一个新的保证恢复点 GRP2，然后闪回到升级前的 GRP1
shut immediate
startup mount
flashback database to restore point GRP1;
shut immediate
exit

```
SQL> select * from SYSTEM.TRACKING_TAB;

        NO ACTION
---------- ------------------------------
         0 Nothing has been done yet
         1 partial offline backup
         2 guaranteed restore point
         3 upgrade completed

SQL> create restore point GRP2 guarantee flashback database;

Restore point created.

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

Total System Global Area 1157627168 bytes
Fixed Size                  8895776 bytes
Variable Size             536870912 bytes
Database Buffers          603979776 bytes
Redo Buffers                7880704 bytes
Database mounted.
SQL> flashback database to restore point GRP1;

Flashback complete.

SQL> shut immediate
ORA-01109: database not open


Database dismounted.
ORACLE instance shut down.
SQL> exit
Disconnected from Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
Version 19.3.0.0.0
[FTEX] oracle@hol:~
```

用升级前的 11.2.0.4 HOME 打开数据库

. ftex

sqlplus / as sysdba

startup open read only

select * from SYSTEM.TRACKING_TAB;

查看回退验证记录

```
SQL> flashback database to restore point GRP1;

Flashback complete.

SQL> shut immediate
ORA-01109: database not open


Database dismounted.
ORACLE instance shut down.
SQL> exit
Disconnected from Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
Version 19.3.0.0.0
[FTEX] oracle@hol:~
$ . ftex
[FTEX] oracle@hol:~
$ sqlplus / as sysdba

SQL*Plus: Release 11.2.0.4.0 Production on Wed Dec 18 12:14:02 2019

Copyright (c) 1982, 2013, Oracle.  All rights reserved.

Connected to an idle instance.

SQL> startup open read only
ORACLE instance started.

Total System Global Area 1152450560 bytes
Fixed Size                  2252584 bytes
Variable Size             335544536 bytes
Database Buffers          805306368 bytes
Redo Buffers                9347072 bytes
Database mounted.
Database opened.
SQL> select * from SYSTEM.TRACKING_TAB;

        NO ACTION
---------- ------------------------------
         0 Nothing has been done yet
         1 partial offline backup
         2 guaranteed restore point

SQL>
```

然后你会看到马上闪回到升级前状态了，这个时候你可以继续 open resetlogs 然后重复执行升级操作了

当然了，flashback 数据库动作可以工作在多个方向，比如前面的向后，也可以向前（向前可能会需要较长一点时间）

闪回到升级后

. ftex
sqlplus / as sysdba
shut immediate
exit

. ftex19
sqlplus / as sysdba
startup mount

flashback database to restore point GRP2;
alter database open resetlogs;
select * from SYSTEM.TRACKING_TAB;
查看回退验证记录

```
SQL> exit
Disconnected from Oracle Database 11g Enterprise Edition Release 11.2.0.4.0 - 64bit Production
With the Partitioning, OLAP, Data Mining and Real Application Testing options
[FTEX] oracle@hol:~
$ . ftex19
[FTEX] oracle@hol:~
$ sqlplus / as sysdba

SQL*Plus: Release 19.0.0.0.0 - Production on Wed Dec 18 12:17:45 2019
Version 19.3.0.0.0

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

Connected to an idle instance.

SQL> startup mount
ORACLE instance started.

Total System Global Area 1157627168 bytes
Fixed Size                  8895776 bytes
Variable Size             536870912 bytes
Database Buffers          603979776 bytes
Redo Buffers                7880704 bytes
Database mounted.
SQL> flashback database to restore point GRP2;

Flashback complete.

SQL> alter database open;
alter database open
*
ERROR at line 1:
ORA-01589: must use RESETLOGS or NORESETLOGS option for database open


SQL> alter database open resetlogs;

Database altered.

SQL> select * from SYSTEM.TRACKING_TAB;

        NO ACTION
---------- ----------------------------
         0 Nothing has been done yet
         1 partial offline backup
         2 guaranteed restore point
         3 upgrade completed

SQL>
```

数据库又回到了之前升级后的状态了。
尽管我们用 resetlogs 方式打开数据库，但依然可以根据需要经常重复闪回数据库操作。

查看一下现在数据库的组件和版本

set linesize window
select comp_id,version,status from dba_registry order by comp_id;

```
SQL> set linesize window
select comp_id,version,status from dba_registry order by comp_id;SQL>

COMP_ID                        VERSION                        STATUS
------------------------------ ------------------------------ -----------
CATALOG                        19.0.0.0.0                     VALID
CATPROC                        19.0.0.0.0                     VALID
RAC                            19.0.0.0.0                     OPTION OFF
XDB                            19.0.0.0.0                     VALID

SQL>
SQL>
```

清理并删除恢复点

因为 GRP 也是需要一定的日志空间的，另外再把数据库切换回 NOARCHIVELOG 模式
drop restore point GRP1;
drop restore point GRP2;
shut immediate
startup mount
alter database noarchivelog;
alter database open;
archive log list
exit

```
SQL> drop restore point GRP1;

Restore point dropped.

SQL> drop restore point GRP2;

Restore point dropped.

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

Total System Global Area 1157627168 bytes
Fixed Size                  8895776 bytes
Variable Size             536870912 bytes
Database Buffers          603979776 bytes
Redo Buffers                7880704 bytes
Database mounted.
SQL> alter database noarchivelog;

Database altered.

SQL> alter database open;

Database altered.

SQL> archive log list
Database log mode              No Archive Mode
Automatic archival             Disabled
Archive destination            USE_DB_RECOVERY_FILE_DEST
Oldest online log sequence     1
Current log sequence           1
SQL> exit
Disconnected from Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
Version 19.3.0.0.0
[FTEX] oracle@hol:~
```

到这里我们成功地完成了回退策略的第二部分练习。在这个实验的最后环节，我们将数据库保留在了 19c 状态，在后面的通过 downgrade 脚本回退练习中使用。

升级后遇到问题回退

在这个环节，我们再体验两种技术来保护数据库，但这次针对的是升级以后发生的问题，这个回退常常被业内人士称之为"降级"。

通过 DataPump 导出和导入进行降级

这个方案就是用大家最熟悉的 Data Pump 重新将升级之后的数据从 19c 环境导出然后导入到升级前环境，方案简单但停机时间相对较长。

导出数据

. ftex19
sqlplus / as sysdba
插入一条跟踪记录
insert into SYSTEM.TRACKING_TAB values (4,'full export downgrade');

commit;
select * from TRACKING_TAB;
create directory dpump as '/home/oracle/IMP';
grant read, write on directory dpump to public;
exit

```
$ . ftex19
[FTEX] oracle@hol:~
$ sqlplus / as sysdba

SQL*Plus: Release 19.0.0.0.0 - Production on Thu Dec 19 06:43:01 2019
Version 19.3.0.0.0

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


Connected to:
Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
Version 19.3.0.0.0

SQL> insert into SYSTEM.TRACKING_TAB values (4,'full export downgrade');

1 row created.

SQL> commit;

Commit complete.

SQL> select * from TRACKING_TAB;

        NO ACTION
---------- ------------------------------
         0 Nothing has been done yet
         1 partial offline backup
         2 guaranteed restore point
         3 upgrade completed
         4 full export downgrade

SQL> create directory dpump as '/home/oracle/IMP';

Directory created.

SQL> grant read, write on directory dpump to public;

Grant succeeded.

SQL> exit
```

expdp system/oracle directory=dpump dumpfile=downgrade.dmp logfile=downgrade.log version=12.2 full=y reuse_dumpfiles=y exclude=statistics logtime=all

```
$ expdp system/oracle directory=dpump  dumpfile=downgrade.dmp  logfile=downgrade.log  version=12.2 full=y reuse_dumpfiles=y exclude=statistics logtime=all

Export: Release 19.0.0.0.0 - Production on Thu Dec 19 06:44:09 2019
Version 19.3.0.0.0

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

Connected to: Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
19-DEC-19 06:44:15.838: Starting "SYSTEM"."SYS_EXPORT_FULL_01":  system/******** directory=dpump dumpfile=downgrade.dmp logfile=downgrade.log version=12.2 full=y reuse_dumpfiles=y exclude=s
tatistics logtime=all
19-DEC-19 06:44:21.420: Processing object type DATABASE_EXPORT/EARLY_OPTIONS/VIEWS_AS_TABLES/TABLE_DATA
19-DEC-19 06:44:31.921: Processing object type DATABASE_EXPORT/NORMAL_OPTIONS/TABLE_DATA
19-DEC-19 06:44:33.655: Processing object type DATABASE_EXPORT/NORMAL_OPTIONS/VIEWS_AS_TABLES/TABLE_DATA
19-DEC-19 06:44:35.307: Processing object type DATABASE_EXPORT/SCHEMA/TABLE/TABLE_DATA
19-DEC-19 06:44:35.620: Processing object type DATABASE_EXPORT/PRE_SYSTEM_IMPCALLOUT/MARKER
19-DEC-19 06:44:35.656: Processing object type DATABASE_EXPORT/PRE_INSTANCE_IMPCALLOUT/MARKER
19-DEC-19 06:44:35.791: Processing object type DATABASE_EXPORT/TABLESPACE
19-DEC-19 06:44:36.068: Processing object type DATABASE_EXPORT/PROFILE
19-DEC-19 06:44:36.275: Processing object type DATABASE_EXPORT/ROLE
19-DEC-19 06:44:36.333: Processing object type DATABASE_EXPORT/RADM_FPTM
```

```
19-DEC-19 06:46:21.341: Processing object type DATABASE_EXPORT/POST_SYSTEM_IMPCALLOUT/MARKER
19-DEC-19 06:46:22.017: . . exported "SYS"."KU$_USER_MAPPING_VIEW"                5.843 KB      22 rows
19-DEC-19 06:46:22.175: . . exported "SYS"."DAM_CONFIG_PARAM$"                   6.539 KB      14 rows
19-DEC-19 06:46:22.181: . . exported "SYS"."AUD$"                                    0 KB       0 rows
19-DEC-19 06:46:22.185: . . exported "SYS"."DAM_CLEANUP_EVENTS$"                     0 KB       0 rows
19-DEC-19 06:46:22.190: . . exported "SYS"."DAM_CLEANUP_JOBS$"                       0 KB       0 rows
19-DEC-19 06:46:22.196: . . exported "SYS"."TSDP_ASSOCIATION$"                       0 KB       0 rows
19-DEC-19 06:46:22.201: . . exported "SYS"."TSDP_CONDITION$"                         0 KB       0 rows
19-DEC-19 06:46:22.208: . . exported "SYS"."TSDP_FEATURE_POLICY$"                    0 KB       0 rows
19-DEC-19 06:46:22.244: . . exported "SYS"."TSDP_PARAMETER$"                     5.953 KB       1 rows
19-DEC-19 06:46:22.276: . . exported "SYS"."TSDP_POLICY$"                        5.921 KB       1 rows
19-DEC-19 06:46:22.283: . . exported "SYS"."TSDP_PROTECTION$"                        0 KB       0 rows
19-DEC-19 06:46:22.290: . . exported "SYS"."TSDP_SENSITIVE_DATA$"                    0 KB       0 rows
19-DEC-19 06:46:22.299: . . exported "SYS"."TSDP_SENSITIVE_TYPE$"                    0 KB       0 rows
19-DEC-19 06:46:22.307: . . exported "SYS"."TSDP_SOURCE$"                            0 KB       0 rows
19-DEC-19 06:46:22.350: . . exported "SYS"."TSDP_SUBPOL$"                        6.328 KB       1 rows
19-DEC-19 06:46:22.357: . . exported "SYSTEM"."REDO_DB"                              0 KB       0 rows
19-DEC-19 06:46:22.364: . . exported "SYSTEM"."REDO_LOG"                             0 KB       0 rows
19-DEC-19 06:46:22.576: . . exported "SYS"."AUDTAB$TBS$FOR_EXPORT"               5.960 KB       2 rows
19-DEC-19 06:46:23.211: . . exported "SYS"."DBA_SENSITIVE_DATA"                      0 KB       0 rows
19-DEC-19 06:46:23.238: . . exported "SYS"."DBA_TSDP_POLICY_PROTECTION"              0 KB       0 rows
19-DEC-19 06:46:23.246: . . exported "SYS"."FGA_LOG$FOR_EXPORT"                      0 KB       0 rows
19-DEC-19 06:46:23.361: . . exported "SYS"."NACL$_ACE_EXP"                           0 KB       0 rows
19-DEC-19 06:46:23.476: . . exported "SYS"."NACL$_HOST_EXP"                      6.914 KB       1 rows
19-DEC-19 06:46:23.485: . . exported "SYS"."NACL$_WALLET_EXP"                        0 KB       0 rows
19-DEC-19 06:46:23.492: . . exported "SYS"."SQL$TEXT_DATAPUMP"                       0 KB       0 rows
19-DEC-19 06:46:23.497: . . exported "SYS"."SQL$_DATAPUMP"                           0 KB       0 rows
19-DEC-19 06:46:23.505: . . exported "SYS"."SQLOBJ$AUXDATA_DATAPUMP"                 0 KB       0 rows
19-DEC-19 06:46:23.512: . . exported "SYS"."SQLOBJ$DATA_DATAPUMP"                    0 KB       0 rows
19-DEC-19 06:46:23.528: . . exported "SYS"."SQLOBJ$PLAN_DATAPUMP"                    0 KB       0 rows
19-DEC-19 06:46:23.535: . . exported "SYS"."SQLOBJ$_DATAPUMP"                        0 KB       0 rows
19-DEC-19 06:46:23.616: . . exported "SYSTEM"."SCHEDULER_JOB_ARGS"                   0 KB       0 rows
19-DEC-19 06:46:23.812: . . exported "SYSTEM"."SCHEDULER_PROGRAM_ARGS"           9.507 KB      12 rows
19-DEC-19 06:46:23.877: . . exported "SYSTEM"."TRACKING_TAB"                     5.625 KB       5 rows
19-DEC-19 06:46:26.016: Master table "SYSTEM"."SYS_EXPORT_FULL_01" successfully loaded/unloaded
19-DEC-19 06:46:26.047: ******************************************************************************
19-DEC-19 06:46:26.048: Dump file set for SYSTEM.SYS_EXPORT_FULL_01 is:
19-DEC-19 06:46:26.050:   /home/oracle/IMP/downgrade.dmp
19-DEC-19 06:46:26.068: Job "SYSTEM"."SYS_EXPORT_FULL_01" successfully completed at Thu Dec 19 06:46:26 2019 elapsed 0 00:02:13
```

导入数据

通过在 cdb1（12.2.0.1）中创建 PDB3，然后创建目录，最后通过 impdp 导入数据

```
. cdb1
sqlplus / as sysdba
create pluggable database PDB3 admin user adm identified by adm
file_name_convert=('pdbseed','pdb3');
alter pluggable database PDB3 open;
alter session set container=pdb3;
create directory dpump as '/home/oracle/IMP';
grant read, write on directory dpump to public;
exit
```

```
$ . cdb1
[CDB1] oracle@hol:~
$ sqlplus / as sysdba

SQL*Plus: Release 12.2.0.1.0 Production on Thu Dec 19 06:51:39 2019

Copyright (c) 1982, 2016, Oracle.  All rights reserved.

Connected to an idle instance.

SQL> startup
ORACLE instance started.

Total System Global Area 1459617792 bytes
Fixed Size                  8792872 bytes
Variable Size             486540504 bytes
Database Buffers          956301312 bytes
Redo Buffers                7983104 bytes
Database mounted.
Database opened.
SQL> show pdbs

    CON_ID CON_NAME                       OPEN MODE  RESTRICTED
---------- ------------------------------ ---------- ----------
         2 PDB$SEED                       READ ONLY  NO
SQL> create pluggable database PDB3 admin user adm identified by adm file_name_convert=('pdbseed','pdb3');

Pluggable database created.

SQL> alter pluggable database PDB3 open;

Pluggable database altered.

SQL> alter session set container=pdb3;

Session altered.

SQL> create directory dpump as '/home/oracle/IMP';

Directory created.

SQL> grant read, write on directory dpump to public;

Grant succeeded.

SQL> exit
Disconnected from Oracle Database 12c Enterprise Edition Release 12.2.0.1.0 - 64bit Production
[CDB1] oracle@hol:~
```

vi $OH19/network/admin/tnsnames.ora 添加以下内容

```
CDB1PDB3 =
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL = TCP)(HOST = hol)(PORT = 1521))
    (CONNECT_DATA =
      (SERVER = DEDICATED)
      (SERVICE_NAME = pdb3.localdomain)
    )
  )
```

impdp system/oracle@CDB1PDB3 directory=dpump dumpfile=downgrade.dmp logfile=impdp-downgrade.log

```
[CDB1] oracle@hol:~
$ impdp system/oracle@CDB1PDB3 directory=dpump dumpfile=downgrade.dmp logfile=impdp-downgrade.log

Import: Release 12.2.0.1.0 - Production on Thu Dec 19 07:02:50 2019

Copyright (c) 1982, 2017, Oracle and/or its affiliates.  All rights reserved.

UDI-28002: operation generated ORACLE error 28002
ORA-28002: the password will expire within 7 days

Connected to: Oracle Database 12c Enterprise Edition Release 12.2.0.1.0 - 64bit Production
Master table "SYSTEM"."SYS_IMPORT_FULL_01" successfully loaded/unloaded
Starting "SYSTEM"."SYS_IMPORT_FULL_01":  system/********@CDB1PDB3 directory=dpump dumpfile=downgrade.dmp logfile=impdp-downgrade.log
```

```
Processing object type DATABASE_EXPORT/FINAL_POST_INSTANCE_IMPCALLOUT/MARKER
Processing object type DATABASE_EXPORT/AUDIT
Processing object type DATABASE_EXPORT/POST_SYSTEM_IMPCALLOUT/MARKER
Job "SYSTEM"."SYS_IMPORT_FULL_01" completed with 6 error(s) at Thu Dec 19 07:03:55 2019 elapsed 0 00:01:02

[CDB1] oracle@hol:~
```

校验数据

. cdb1

sqlplus system/oracle@cdb1pdb3

select * from SYSTEM.TRACKING_TAB;

```
$ . cdb1
[CDB1] oracle@hol:~
$ sqlplus system/oracle@cdb1pdb3

SQL*Plus: Release 12.2.0.1.0 Production on Thu Dec 19 07:09:31 2019

Copyright (c) 1982, 2016, Oracle.  All rights reserved.

ERROR:
ORA-28002: the password will expire within 7 days


Last Successful login time: Thu Dec 19 2019 07:02:50 +01:00

Connected to:
Oracle Database 12c Enterprise Edition Release 12.2.0.1.0 - 64bit Production

SQL> select * from SYSTEM.TRACKING_TAB;

        NO ACTION
---------- ------------------------------
         0 Nothing has been done yet
         1 partial offline backup
         2 guaranteed restore point
         3 upgrade completed
         4 full export downgrade

SQL>
```

到这里我们通过导出/导入的方式，就快速的回顾了这种经典的降级方式。对于生产环境，如果数据库中包含的数据和对象越多，那么执行时间就会越长，特别是 LOB 数据类型可能更慢，这时也可以考虑并行的方式来优化导出导入脚本。

通过 downgrade 脚本进行降级

这个方案就是运行 downgrade scripts，相对来说这时比较强大的技术，执行速度快也比较容易操作。数据库降级是目前为止最简单的升级后几天进行回退的方法。一般在实际生产环境下，极其罕见用户在升级后又进行降级处理的，但至少 Oracle 也提供了降级选项作为可选的后备方案，所以我们在这个实验中也练习一下。

源数据库	可能降级到
Non-CDB 19c	18c,12.2.0.1,12.1.0.2,12.1.0.1,11.2.0.4
PDB/CDB 19c	18c,12.2.0.1,12.1.0.2

必须加以说明的是，将数据库实例从当前版本降级到升级前的版本时，数据库不会返回到升级前的完全相同状态。根据所涉及的版本，升级过程会进行不可逆的更改。用户使用降

级过程可以打开和访问以前版本的数据库实例，这通常已足够。

使用此方案的注意事项：

- 此过程旨在降级已成功升级到 19c 的数据库，并非用于从失败的升级退回。您只能降级到升级前所用的版本和补丁程序级别。
- 从 12c 版本开始，XMLDB 组件是必需的，所以升级到 19c 会安装 XMLDB 组件（如果以前没装得话），那么降级到 11g 同样也会删除 XMLDB 组件。
- 如果是从 11g 版本升级到 19c，因为在升级过程中需要删除 DatabaseControl 资料档案库，那么降级后如果需要 DBControl 得重新配置。
- 如果数据库参数 COMPATIBLE 被设置为高于被升级版本，则不能降级到原版本
- XE 版本不能降级
- 对于 ASM 磁盘组，如果在升级数据库时将磁盘组兼容性更改为 12.1.0.0.0，那么在降级到较早版本时，将无法挂载 ASM 磁盘组。

更多关于 19c 降级的注意事项，请参考 Oracle 19c - How to Downgrade a 19c Multitenent Database (CDB) or Non CDB Database to Previous Release (Doc ID 2548962.1)

在这个实验中，我们使用前面使用闪回数据库到保证恢复点回退练习中升级后的 FTEX 数据库。

创建测试数据

```
. ftex19
sqlplus / as sysdba
insert into SYSTEM.TRACKING_TAB values (5,'database downgrade');
commit;
shut immediate
exit
```

```
$ . ftex19
[FTEX] oracle@hol:~
$ sqlplus / as sysdba

SQL*Plus: Release 19.0.0.0.0 - Production on Thu Dec 19 07:19:25 2019
Version 19.3.0.0.0

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


Connected to:
Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
Version 19.3.0.0.0

SQL> insert into SYSTEM.TRACKING_TAB values (5,'database downgrade');

1 row created.

SQL> commit;

Commit complete.

SQL> select * from SYSTEM.TRACKING_TAB;

        NO ACTION
---------- ------------------------------
         0 Nothing has been done yet
         1 partial offline backup
         2 guaranteed restore point
         3 upgrade completed
         4 full export downgrade
         5 database downgrade

6 rows selected.

SQL> shut immediate
Database closed.
Database dismounted.
ORACLE instance shut down.
SQL> exit
Disconnected from Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
Version 19.3.0.0.0
[FTEX] oracle@hol:~
```

执行降级脚本

startup downgrade
set echo on termout on serveroutput on timing on
spool /home/oracle/logs/downgrade.log
执行降级脚本，注意这里 ORACLE_HOME 应指向 19c 主目录
@?/rdbms/admin/catdwgrd.sql
shut immediate
exit

```
$ sqlplus / as sysdba

SQL*Plus: Release 19.0.0.0.0 - Production on Thu Dec 19 07:35:14 2019
Version 19.3.0.0.0

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

Connected to an idle instance.

SQL> startup downgrade
ORACLE instance started.

Total System Global Area 1157627168 bytes
Fixed Size                  8895776 bytes
Variable Size             570425344 bytes
Database Buffers          570425344 bytes
Redo Buffers                7880704 bytes
Database mounted.
Database opened.
SQL> set echo on termout on serveroutput on timing on
SQL> spool /home/oracle/logs/downgrade.log
SQL> @?/rdbms/admin/catdwgrd.sql
SQL> Rem
SQL> Rem $Header: rdbms/admin/catdwgrd.sql /main/103 2018/07/05 19:32:42 amunnoli Exp $
SQL> Rem
SQL> Rem catdwgrd.sql
SQL> Rem
SQL> Rem
SQL> Rem Copyright (c) 2000, 2018, Oracle and/or its affiliates.
SQL> Rem All rights reserved.
SQL> Rem
SQL> Rem    NAME
SQL> Rem         catdwgrd.sql -  DataBase DoWnGrade from the current release
SQL> Rem                      to the original release (if supported)
SQL> Rem
SQL> Rem    DESCRIPTION
SQL> Rem
SQL> Rem         This script is to be used for downgrading your database from the
SQL> Rem         current release you have installed to the release from which
SQL> Rem         you upgraded.
SQL> Rem
SQL> Rem    NOTES
```

```
SQL> @?/rdbms/admin/sqlsessend.sql
SQL> Rem
SQL> Rem $Header: rdbms/admin/sqlsessend.sql /main/3 2018/07/25 13:50:02 surman Exp $
SQL> Rem
SQL> Rem sqlsessend.sql
SQL> Rem
SQL> Rem Copyright (c) 2013, 2018, Oracle and/or its affiliates.
SQL> Rem All rights reserved.
SQL> Rem
SQL> Rem    NAME
SQL> Rem         sqlsessend.sql - SQL session end
SQL> Rem
SQL> Rem    DESCRIPTION
SQL> Rem         Any commands which should be run at the end of all oracle
SQL> Rem         supplied scripts.
SQL> Rem
SQL> Rem    NOTES
SQL> Rem         See sqlsessstart.sql for the corresponding start script.
SQL> Rem
SQL> Rem    BEGIN SQL_FILE_METADATA
SQL> Rem    SQL_SOURCE_FILE: rdbms/admin/sqlsessend.sql
SQL> Rem    SQL_SHIPPED_FILE: rdbms/admin/sqlsessend.sql
SQL> Rem    SQL_PHASE: MISC
SQL> Rem    SQL_STARTUP_MODE: NORMAL
SQL> Rem    SQL_IGNORABLE_ERRORS: NONE
SQL> Rem    END SQL_FILE_METADATA
SQL> Rem
SQL> Rem    MODIFIED    (MM/DD/YY)
SQL> Rem    surman       05/04/18 - 27464252: Update SQL_PHASE
SQL> Rem    surman       03/08/13 - 16462837: Common start and end scripts
SQL> Rem    surman       03/08/13 - Created
SQL> Rem
SQL>
SQL> alter session set "_ORACLE_SCRIPT" = false;

Session altered.

Elapsed: 00:00:00.00
SQL>
SQL>
SQL>
SQL> Rem ********************************************************************************
SQL> Rem END catdwgrd.sql
SQL> Rem ********************************************************************************
```

```
SQL> Rem ********************************************************************************
SQL> Rem END catdwgrd.sql
SQL> Rem ********************************************************************************
SQL>
SQL> shut immediate
Database closed.
Database dismounted.
ORACLE instance shut down.
SQL> exit
Disconnected from Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
Version 19.3.0.0.0
[FTEX] oracle@hol:~
```

启动升级前环境并加载脚本

. ftex
sqlplus / as sysdba
startup upgrade

set echo on termout on timing on
spool /home/oracle/logs/relod.log
@?/rdbms/admin/catrelod.sql
catrelod.sql 脚本在降级的数据库中重新加载各个数据库组件的合适版本

```
SQL> @?/rdbms/admin/catrelod.sql
SQL> Rem
SQL> Rem $Header: rdbms/admin/catrelod.sql /st_rdbms_11.2.0/4 2012/03/21 14:55:04 bmccarth Exp $
SQL> Rem
SQL> Rem catrelod.sql
SQL> Rem
SQL> Rem Copyright (c) 2001, 2012, Oracle and/or its affiliates.
SQL> Rem All rights reserved.
SQL> Rem
SQL> Rem    NAME
SQL> Rem      catrelod.sql - Script to apply CATalog RELOaD scripts to a database
SQL> Rem
SQL> Rem    DESCRIPTION
SQL> Rem      This script encapsulates the "post downgrade" steps necessary
SQL> Rem      to reload the PL/SQL and Java packages, types, and classes.
SQL> Rem      It runs the "old" versions of catalog.sql and catproc.sql
SQL> Rem      and calls the component reload scripts.
SQL> Rem
SQL> Rem    NOTES
SQL> Rem      Use SQLPLUS and connect AS SYSDBA to run this script.
SQL> Rem      The database must be open for MIGRATE
SQL> Rem
SQL> Rem    MODIFIED   (MM/DD/YY)
SQL> Rem    bmccarth    03/06/12 - 11.2.0.4
SQL> Rem    cdilling    02/24/11 - add support for 11.2.0.3
SQL> Rem    skabraha    07/29/10 - Backport skabraha_bug-9928461 from main
SQL> Rem    cmlim       07/27/10 - Backport cmlim_bug-9803834 from main
SQL> Rem    cmlim       04/26/10 - bug 9546509: suggest to force a checkpoint prior
SQL> Rem                           to shutdown abort
SQL> Rem    cdilling    05/21/09 - check for 8 digits for prv_version
SQL> Rem    jciminsk    10/22/07 - Upgrade support for 11.2
SQL> Rem    jciminsk    10/10/07 - fix typo
SQL> Rem    cdilling    10/09/07 - update version to 11.2.0.0.0
SQL> Rem    cdilling    12/07/06 - add DV support
SQL> Rem    rburns      04/15/06 - remove ODM
SQL> Rem    rburns      01/10/06 - release 11.1.0
SQL> Rem    rburns      10/28/05 - no utlip for patch downgrade
SQL> Rem    rburns      02/27/05 - record action for history
SQL> Rem    rburns      01/18/05 - comment out htmldb for 10.2
SQL> Rem    rburns      11/11/04 - move CONTEXT
```

```
SQL> column comp_name format a35
SQL> SELECT comp_name, status, substr(version,1,10) as version
  2  from dba_server_registry order by modified;

COMP_NAME                           STATUS
----------------------------------- -----------------------------------------
VERSION
----------------------------------------
Oracle Database Packages and Types  VALID
11.2.0.4.0

Oracle Database Catalog Views       VALID
11.2.0.4.0


2 rows selected.

Elapsed: 00:00:00.01
SQL>
SQL> DOC
DOC>#####################################################################
DOC>#####################################################################
DOC>
DOC>   The above query lists the SERVER components now loaded in the
DOC>   database, along with their current version and status.
DOC>
DOC>   Please review the status and version columns and look for
DOC>   any errors in the spool log file.  If there are errors in the spool
DOC>   file, or any components are not VALID or not the correct 10.1.0
DOC>   patch version, consult the downgrade chapter of the current release
DOC>   Database Upgrade book.
DOC>
DOC>   Next shutdown immediate, restart for normal operation, and then
DOC>   run utlrp.sql to recompile any invalid application objects.
DOC>
DOC>#####################################################################
DOC>#####################################################################
DOC>#
SQL>
SQL> Rem **********************************************************************
SQL> Rem END catrelod.sql
SQL> Rem **********************************************************************
SQL>
SQL>
```

shut immediate

```
SQL> shut immediate
Database closed.
Database dismounted.
ORACLE instance shut down.
SQL> exit
Disconnected from Oracle Database 11g Enterprise Edition Release 11.2.0.4.0 - 64bit Production
With the Partitioning, OLAP, Data Mining and Real Application Testing options
[FTEX] oracle@hol:~
```

编译无效对象并验证降级

sqlplus / as sysdba
startup
@?/rdbms/admin/utlrp.sql

```
[FTEX] oracle@hol:~
$ sqlplus / as sysdba

SQL*Plus: Release 11.2.0.4.0 Production on Thu Dec 19 08:02:48 2019

Copyright (c) 1982, 2013, Oracle.  All rights reserved.

Connected to an idle instance.

SQL> startup
ORACLE instance started.

Total System Global Area 1152450560 bytes
Fixed Size                  2252584 bytes
Variable Size             335544536 bytes
Database Buffers          805306368 bytes
Redo Buffers                9347072 bytes
Database mounted.
Database opened.
SQL> @?/rdbms/admin/utlrp.sql

TIMESTAMP
--------------------------------------------------------------------------------
COMP_TIMESTAMP UTLRP_BGN  2019-12-19 08:03:41
```

select count(*) from dba_objects where status='INVALID';
select * from SYSTEM.TRACKING_TAB;

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

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

SQL> select * from SYSTEM.TRACKING_TAB;

        NO ACTION
---------- ------------------------------
         0 Nothing has been done yet
         1 partial offline backup
         2 guaranteed restore point
         3 upgrade completed
         4 full export downgrade
         5 database downgrade

6 rows selected.
```

##检查已降级数据库的状态，也可以参考
Note 556610.1 Script to Collect DB Upgrade/Migrate Diagnostic Information (dbupgdiag.sql)
set linesize 160
select comp_id,version,status from dba_registry order by comp_id;

```
SQL> set linesize 160
SQL> select comp_id,version,status from dba_registry order by comp_id;

COMP_ID                        VERSION                        STATUS
------------------------------ ------------------------------ ----------
CATALOG                        11.2.0.4.0                     VALID
CATPROC                        11.2.0.4.0                     VALID

SQL>
```

到这里我们就完成了通过降级脚本进行数据库降级的基本练习，更多关于19c 数据库降级的信息，请参考这里的官方文档说明或My Oracle Support 上的Doc ID 2548962.1。

通过命令行升级整个 12.2.0.1 CDB 到 19c CDB

在前面实验中，我们学习了 11.2.0.4 non-CDB 到 19c 的升级，以及如何将升级后的 non-CDB 转换为 PDB 的练习，本节练习内容属于可选部分，大家可根据自己的兴趣课后自行练习。练习的目的是学习 12c 或 18c CDB 数据库到 19c CDB 的升级过程，一是体验一下和 non-CDB 升级的不同，二是体验针对整个 CDB 进行全库升级。

为完成本练习，我们需要先准备一个名为 CDB3 的 12.2.0.1 版本下的数据库。根据大家的硬件环境和 PDBs 个数的不同，执行可能相对很长，所以该试验大家自行课后练习。可采用 AutoUpgrade、命令行方式或 DBUA 等方式，这里使用命令行的方式来做。

准备工作

用 root 在/usr/local/bin 下创建 cdb3、19cdb3 两个环境变量概要文件。然后切换到 cdb3 环境，通过 DBCA 创建一个 12.2.0.1 版本下的多租户数据库，包含 2 个 pdb，分别是 pdb1,pdb2
为了本试验进行的更加流畅，建议先关闭或删除其他数据库，我这里直接删除了 CDB1、CDB2 两个库，创建一个新的 CDB3。

```
$ cd /usr/local/bin
$ cp cdb1 cdb3
$ cp cdb2 19cdb3
$ vi cdb3
$ vi 19cdb3
$ grep ORACLE_SID cdb3
$ grep ORACLE_SID 19cdb3
$
```

```
[FTEX] root@hol:~
$ cd /usr/local/bin
[FTEX] root@hol:/usr/local/bin
$ cp cdb1 cdb3
[FTEX] root@hol:/usr/local/bin
$ cp cdb2 19cdb3
[FTEX] root@hol:/usr/local/bin
$ vi cdb3
[FTEX] root@hol:/usr/local/bin
$ vi 19cdb3
[FTEX] root@hol:/usr/local/bin
$ grep ORACLE_SID cdb3
  ORACLE_SID=CDB3
  ORACLE_SID=$1
export ORACLE_BASE ORACLE_HOME ORACLE_SID OH18 NLS_LANG CLASSPATH PATH LD_LIBRARY_PATH TNS_ADMIN
export PS1="[$ORACLE_SID] \\u@\\h:\\w\n$ "
[FTEX] root@hol:/usr/local/bin
$ grep ORACLE_SID 19cdb3
  ORACLE_SID=CDB3
  ORACLE_SID=$1
export ORACLE_BASE ORACLE_HOME ORACLE_SID OH18 NLS_LANG CLASSPATH PATH LD_LIBRARY_PATH TNS_ADMIN
export PS1="[$ORACLE_SID] \\u@\\h:\\w\n$ "
[FTEX] root@hol:/usr/local/bin
```

```
SQL*Plus: Release 12.2.0.1.0 Production on Thu Dec 19 11:09:06 2019

Copyright (c) 1982, 2016, Oracle.  All rights reserved.


Connected to:
Oracle Database 12c Enterprise Edition Release 12.2.0.1.0 - 64bit Production

SQL> show pdbs;

    CON_ID CON_NAME                       OPEN MODE  RESTRICTED
---------- ------------------------------ ---------- ----------
         2 PDB$SEED                       READ ONLY  NO
         3 PDB1                           READ WRITE NO
         4 PDB2                           READ WRITE NO
SQL>
```

预升级&问题修复

$OH19/jdk/bin/java -jar $OH19/rdbms/admin/preupgrade.jar TEXT TERMINAL

```
$ $OH19/jdk/bin/java -jar $OH19/rdbms/admin/preupgrade.jar TEXT TERMINAL
Report generated by Oracle Database Pre-Upgrade Information Tool Version
19.0.0.0.0 Build: 1 on 2019-12-19T11:10:44

Upgrade-To version: 19.0.0.0.0

======================================
Status of the database prior to upgrade
======================================
      Database Name:  CDB3
     Container Name:  CDB$ROOT
       Container ID:  1
            Version:  12.2.0.1.0
     DB Patch Level:  DATABASE APR 2019 RELEASE UPDATE 12.2.0.1.190416
         Compatible:  12.2.0
          Blocksize:  8192
           Platform:  Linux x86 64-bit
      Timezone File:  26
  Database log mode:  NOARCHIVELOG
           Readonly:  FALSE
            Edition:  EE

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

```
$ $OH19/jdk/bin/java -jar $OH19/rdbms/admin/preupgrade.jar TEXT TERMINAL
Report generated by Oracle Database Pre-Upgrade Information Tool Version
```

```
19.0.0.0.0 Build: 1 on 2019-12-19T11:10:44

Upgrade-To version: 19.0.0.0.0

=====================================
Status of the database prior to upgrade
=====================================
      Database Name:  CDB3
     Container Name:  CDB$ROOT
       Container ID:  1
            Version:  12.2.0.1.0
     DB Patch Level:  DATABASE APR 2019 RELEASE UPDATE 12.2.0.1.190416
         Compatible:  12.2.0
          Blocksize:  8192
           Platform:  Linux x86 64-bit
      Timezone File:  26
  Database log mode:  NOARCHIVELOG
           Readonly:  FALSE
            Edition:  EE

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

==============
BEFORE UPGRADE
==============

  REQUIRED ACTIONS
  ================
  None
```

RECOMMENDED ACTIONS
==========

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

   EXECUTE DBMS_STATS.GATHER_DICTIONARY_STATS;

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

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

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

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

   None of the fixed object tables have had stats collected.

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

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

INFORMATION ONLY
==========

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

Tablespace	Size	Min Size For Upgrade
SYSAUX	620 MB	647 MB
SYSTEM	810 MB	921 MB
TEMP	33 MB	150 MB
UNDOTBS1	120 MB	439 MB

   Minimum tablespace sizes for upgrade are estimates.

```
  4.  No action needed.

      Using default parallel upgrade options, this CDB with 3 PDBs will first
      upgrade the CDB$ROOT, and then upgrade at most 2 PDBs at a time using 2
      parallel processes per PDB.

      The number of PDBs upgraded in parallel and the number of parallel
      processes per PDB can be adjusted as described in Database Upgrade Guide.

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

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

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

  ORACLE GENERATED FIXUP SCRIPT
  ============================
  All of the issues in database CDB3 container CDB$ROOT
  which are identified above as BEFORE UPGRADE "(AUTOFIXUP)" can be resolved by
  executing the following from within the container

    SQL>@/u01/app/oracle/cfgtoollogs/CDB3/preupgrade/preupgrade_fixups.sql

=============
AFTER UPGRADE
=============

  REQUIRED ACTIONS
  ================
  None

  RECOMMENDED ACTIONS
  ===================
  6.  Upgrade the database time zone file using the DBMS_DST package.

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

      Oracle recommends upgrading to the desired (latest) version of the time
      zone file.  For more information, refer to "Upgrading the Time Zone File
```

and Timestamp with Time Zone Data" in the 19 Oracle Database Globalization Support Guide.

7. To identify directory objects with symbolic links in the path name, run $ORACLE_HOME/rdbms/admin/utldirsymlink.sql AS SYSDBA after upgrade. Recreate any directory objects listed, using path names that contain no symbolic links.

   Some directory object path names may currently contain symbolic links.

   Starting in Release 18c, symbolic links are not allowed in directory object path names used with BFILE data types, the UTL_FILE package, or external tables.

8. (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.

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

   ```
   EXECUTE DBMS_STATS.GATHER_FIXED_OBJECTS_STATS;
   ```

   This recommendation is given for all preupgrade runs.

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

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

ORACLE GENERATED FIXUP SCRIPT

==========

```
  All of the issues in database CDB3 container CDB$ROOT
  which are identified above as AFTER UPGRADE "(AUTOFIXUP)" can be resolved by
  executing the following from within the container

    SQL>@/u01/app/oracle/cfgtoollogs/CDB3/preupgrade/postupgrade_fixups.sql


Report generated by Oracle Database Pre-Upgrade Information Tool Version
19.0.0.0.0 Build: 1 on 2019-12-19T11:10:57

Upgrade-To version: 19.0.0.0.0

====================================
Status of the database prior to upgrade
====================================
      Database Name:  CDB3
     Container Name:  PDB$SEED
       Container ID:  2
            Version:  12.2.0.1.0
     DB Patch Level:  DATABASE APR 2019 RELEASE UPDATE 12.2.0.1.190416
         Compatible:  12.2.0
          Blocksize:  8192
           Platform:  Linux x86 64-bit
      Timezone File:  26
  Database log mode:  NOARCHIVELOG
           Readonly:  TRUE
            Edition:  EE

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

```
==============
BEFORE UPGRADE
==============

  REQUIRED ACTIONS
  ================
  None

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

        EXECUTE DBMS_STATS.GATHER_DICTIONARY_STATS;

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

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

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

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

      None of the fixed object tables have had stats collected.

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

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

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

                                                 Min Size
      Tablespace                      Size       For Upgrade
```

```
      ----------                      ----------  ----------
      SYSAUX                             470 MB       515 MB
      SYSTEM                             260 MB       370 MB
      TEMP                                64 MB       150 MB
      UNDOTBS1                           100 MB       439 MB

      Minimum tablespace sizes for upgrade are estimates.

  ORACLE GENERATED FIXUP SCRIPT
  =============================
  All of the issues in database CDB3 container PDB$SEED
  which are identified above as BEFORE UPGRADE "(AUTOFIXUP)" can be resolved by
  executing the following from within the container

    SQL>@/u01/app/oracle/cfgtoollogs/CDB3/preupgrade/preupgrade_fixups.sql

=============
AFTER UPGRADE
=============

  REQUIRED ACTIONS
  ================
  None

  RECOMMENDED ACTIONS
  ===================
  4.  Upgrade the database time zone file using the DBMS_DST package.

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

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

  5.  To identify directory objects with symbolic links in the path name, run
      $ORACLE_HOME/rdbms/admin/utldirsymlink.sql AS SYSDBA after upgrade.
      Recreate any directory objects listed, using path names that contain no
      symbolic links.

      Some directory object path names may currently contain symbolic links.

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

object path names used with BFILE data types, the UTL_FILE package, or external tables.

6. (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.

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

   EXECUTE DBMS_STATS.GATHER_FIXED_OBJECTS_STATS;

   This recommendation is given for all preupgrade runs.

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

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

ORACLE GENERATED FIXUP SCRIPT
==========
All of the issues in database CDB3 container PDB$SEED
which are identified above as AFTER UPGRADE "(AUTOFIXUP)" can be resolved by executing the following from within the container

  SQL>@/u01/app/oracle/cfgtoollogs/CDB3/preupgrade/postupgrade_fixups.sql

Report generated by Oracle Database Pre-Upgrade Information Tool Version 19.0.0.0.0 Build: 1 on 2019-12-19T11:10:58

Upgrade-To version: 19.0.0.0.0

```
======================================
Status of the database prior to upgrade
======================================
      Database Name:  CDB3
     Container Name:  PDB1
       Container ID:  3
            Version:  12.2.0.1.0
     DB Patch Level:  DATABASE APR 2019 RELEASE UPDATE 12.2.0.1.190416
         Compatible:  12.2.0
          Blocksize:  8192
           Platform:  Linux x86 64-bit
      Timezone File:  26
  Database log mode:  NOARCHIVELOG
           Readonly:  FALSE
            Edition:  EE

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

==============
BEFORE UPGRADE
==============

  REQUIRED ACTIONS
  ================
  None

  RECOMMENDED ACTIONS
  ===================
```

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

   EXECUTE DBMS_STATS.GATHER_DICTIONARY_STATS;

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

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

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

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

   None of the fixed object tables have had stats collected.

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

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

INFORMATION ONLY
================

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

Tablespace	Size	Min Size For Upgrade
SYSAUX	470 MB	516 MB
SYSTEM	260 MB	370 MB
TEMP	64 MB	150 MB
UNDOTBS1	100 MB	439 MB

   Minimum tablespace sizes for upgrade are estimates.

ORACLE GENERATED FIXUP SCRIPT
=============================

All of the issues in database CDB3 container PDB1

```
  which are identified above as BEFORE UPGRADE "(AUTOFIXUP)" can be resolved by
  executing the following from within the container

    SQL>@/u01/app/oracle/cfgtoollogs/CDB3/preupgrade/preupgrade_fixups.sql

==============
AFTER UPGRADE
==============

  REQUIRED ACTIONS
  ================
  None

  RECOMMENDED ACTIONS
  ===================
  4.  Upgrade the database time zone file using the DBMS_DST package.

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

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

  5.  To identify directory objects with symbolic links in the path name, run
      $ORACLE_HOME/rdbms/admin/utldirsymlink.sql AS SYSDBA after upgrade.
      Recreate any directory objects listed, using path names that contain no
      symbolic links.

      Some directory object path names may currently contain symbolic links.

      Starting in Release 18c, symbolic links are not allowed in directory
      object path names used with BFILE data types, the UTL_FILE package, or
      external tables.

  6.  (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.

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

        EXECUTE DBMS_STATS.GATHER_FIXED_OBJECTS_STATS;

      This recommendation is given for all preupgrade runs.

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

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

  ORACLE GENERATED FIXUP SCRIPT
  ============================
  All of the issues in database CDB3 container PDB1
  which are identified above as AFTER UPGRADE "(AUTOFIXUP)" can be resolved by
  executing the following from within the container

    SQL>@/u01/app/oracle/cfgtoollogs/CDB3/preupgrade/postupgrade_fixups.sql


Report generated by Oracle Database Pre-Upgrade Information Tool Version
19.0.0.0.0 Build: 1 on 2019-12-19T11:11:06

Upgrade-To version: 19.0.0.0.0

======================================
Status of the database prior to upgrade
======================================
      Database Name:  CDB3
     Container Name:  PDB2
       Container ID:  4
            Version:  12.2.0.1.0
     DB Patch Level:  DATABASE APR 2019 RELEASE UPDATE 12.2.0.1.190416
        Compatible:  12.2.0
         Blocksize:  8192
```

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

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

==============
BEFORE UPGRADE
==============

  REQUIRED ACTIONS
  ================
  None

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

        EXECUTE DBMS_STATS.GATHER_DICTIONARY_STATS;

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

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

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

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

      None of the fixed object tables have had stats collected.

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

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

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

                                                 Min Size
      Tablespace                      Size       For Upgrade
      ----------                   ----------  -----------
      SYSAUX                          470 MB       516 MB
      SYSTEM                          260 MB       370 MB
      TEMP                             64 MB       150 MB
      UNDOTBS1                        100 MB       439 MB

      Minimum tablespace sizes for upgrade are estimates.

  ORACLE GENERATED FIXUP SCRIPT
  =============================
  All of the issues in database CDB3 container PDB2
  which are identified above as BEFORE UPGRADE "(AUTOFIXUP)" can be resolved by
  executing the following from within the container

    SQL>@/u01/app/oracle/cfgtoollogs/CDB3/preupgrade/preupgrade_fixups.sql

=============
AFTER UPGRADE
=============

  REQUIRED ACTIONS
  ================
```

None

RECOMMENDED ACTIONS
==========

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

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

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

5. To identify directory objects with symbolic links in the path name, run $ORACLE_HOME/rdbms/admin/utldirsymlink.sql AS SYSDBA after upgrade. Recreate any directory objects listed, using path names that contain no symbolic links.

   Some directory object path names may currently contain symbolic links.

   Starting in Release 18c, symbolic links are not allowed in directory object path names used with BFILE data types, the UTL_FILE package, or external tables.

6. (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.

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

   ```
   EXECUTE DBMS_STATS.GATHER_FIXED_OBJECTS_STATS;
   ```

   This recommendation is given for all preupgrade runs.

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

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

  ORACLE GENERATED FIXUP SCRIPT
  =============================
  All of the issues in database CDB3 container PDB2
  which are identified above as AFTER UPGRADE "(AUTOFIXUP)" can be resolved by
  executing the following from within the container

    SQL>@/u01/app/oracle/cfgtoollogs/CDB3/preupgrade/postupgrade_fixups.sql


==================
PREUPGRADE SUMMARY
==================
  /u01/app/oracle/cfgtoollogs/CDB3/preupgrade/preupgrade.log
  /u01/app/oracle/cfgtoollogs/CDB3/preupgrade/preupgrade_fixups.sql
  /u01/app/oracle/cfgtoollogs/CDB3/preupgrade/postupgrade_fixups.sql

Execute fixup scripts across the entire CDB:

Before upgrade:

1. Execute preupgrade fixups with the below command
$ORACLE_HOME/perl/bin/perl -I$ORACLE_HOME/perl/lib -I$ORACLE_HOME/rdbms/admin
$ORACLE_HOME/rdbms/admin/catcon.pl -l /u01/app/oracle/cfgtoollogs/CDB3/preupgrade/ -b
preup_CDB3 /u01/app/oracle/cfgtoollogs/CDB3/preupgrade/preupgrade_fixups.sql

2. Review logs under /u01/app/oracle/cfgtoollogs/CDB3/preupgrade/

After the upgrade:

1. Execute postupgrade fixups with the below command
$ORACLE_HOME/perl/bin/perl -I$ORACLE_HOME/perl/lib -I$ORACLE_HOME/rdbms/admin
$ORACLE_HOME/rdbms/admin/catcon.pl -l /u01/app/oracle/cfgtoollogs/CDB3/preupgrade/ -b
postup_CDB3 /u01/app/oracle/cfgtoollogs/CDB3/preupgrade/postupgrade_fixups.sql

2. Review logs under /u01/app/oracle/cfgtoollogs/CDB3/preupgrade/
```

```
Preupgrade complete: 2019-12-19T11:11:11
[CDB3] oracle@hol:~
```

```
==================
PREUPGRADE SUMMARY
==================
  /u01/app/oracle/cfgtoollogs/CDB3/preupgrade/preupgrade.log
  /u01/app/oracle/cfgtoollogs/CDB3/preupgrade/preupgrade_fixups.sql
  /u01/app/oracle/cfgtoollogs/CDB3/preupgrade/postupgrade_fixups.sql

Execute fixup scripts across the entire CDB:

Before upgrade:

1. Execute preupgrade fixups with the below command
$ORACLE_HOME/perl/bin/perl -I$ORACLE_HOME/perl/lib -I$ORACLE_HOME/rdbms/admin $ORACLE_HOME/rdbms/admin/catcon.pl -l
e/cfgtoollogs/CDB3/preupgrade/preupgrade_fixups.sql

2. Review logs under /u01/app/oracle/cfgtoollogs/CDB3/preupgrade/

After the upgrade:

1. Execute postupgrade fixups with the below command
$ORACLE_HOME/perl/bin/perl -I$ORACLE_HOME/perl/lib -I$ORACLE_HOME/rdbms/admin $ORACLE_HOME/rdbms/admin/catcon.pl -l
le/cfgtoollogs/CDB3/preupgrade/postupgrade_fixups.sql

2. Review logs under /u01/app/oracle/cfgtoollogs/CDB3/preupgrade/
```

执行预修复脚本

1. Execute preupgrade fixups with the below command

$ORACLE_HOME/perl/bin/perl -I$ORACLE_HOME/perl/lib -I$ORACLE_HOME/rdbms/admin $ORACLE_HOME/rdbms/admin/catcon.pl -l /u01/app/oracle/cfgtoollogs/CDB3/preupgrade/ -b preup_CDB3 /u01/app/oracle/cfgtoollogs/CDB3/preupgrade/preupgrade_fixups.sql

```
[CDB3] oracle@hol:~
$ $ORACLE_HOME/perl/bin/perl -I$ORACLE_HOME/perl/lib -I$ORACLE_HOME/rdbms/admin $ORACLE_HOME/rdbms/admin/catcon.pl -l /u01/
app/oracle/cfgtoollogs/CDB3/preupgrade/ -b preup_CDB3 /u01/app/oracle/cfgtoollogs/CDB3/preupgrade/preupgrade_fixups.sql
catcon: ALL catcon-related output will be written to [/u01/app/oracle/cfgtoollogs/CDB3/preupgrade//preup_CDB3_catcon_26132.
lst]
catcon: See [/u01/app/oracle/cfgtoollogs/CDB3/preupgrade//preup_CDB3*.log] files for output generated by scripts
catcon: See [/u01/app/oracle/cfgtoollogs/CDB3/preupgrade//preup_CDB3_*.lst] files for spool files, if any
catcon.pl: completed successfully
[CDB3] oracle@hol:~
```

关闭数据库

```
SQL> shut immediate
Database closed.
Database dismounted.
ORACLE instance shut down.
SQL> exit
Disconnected from Oracle Database 12c Enterprise Edition Release 12.2.0.1.0 - 64bit Production
[CDB3] oracle@hol:~
```

准备 19c 版本下的参数和密码文件

cp $ORACLE_HOME/dbs/spfileCDB3.ora $OH19/dbs

cp $ORACLE_HOME/dbs/orapwCDB3 $OH19/dbs

```
$ cp $ORACLE_HOME/dbs/spfileCDB3.ora $OH19/dbs
[CDB3] oracle@hol:~
$ cp $ORACLE_HOME/dbs/orapwCDB3 $OH19/dbs
[CDB3] oracle@hol:~
```

执行升级步骤

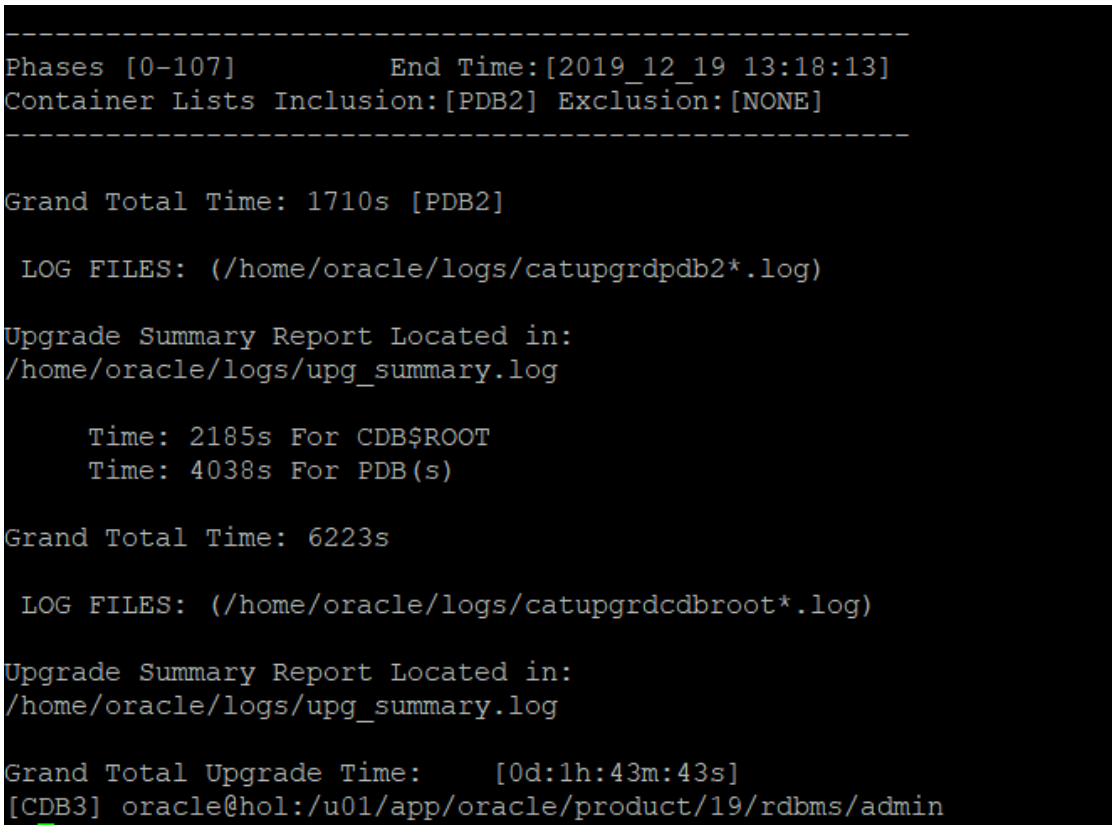
. 19cdb3
sqlplus / as sysdba
startup upgrade;
alter pluggable database all open upgrade force;
这一点和升级 non-CDB 不同
show pdbs
exit

```
$ . 19cdb3
[CDB3] oracle@hol:~
$ sqlplus / as sysdba

SQL*Plus: Release 19.0.0.0.0 - Production on Thu Dec 19 11:31:32 2019
Version 19.3.0.0.0

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

Connected to an idle instance.

SQL> startup upgrade;
ORACLE instance started.

Total System Global Area 2432695872 bytes
Fixed Size                  9137728 bytes
Variable Size             620756992 bytes
Database Buffers         1795162112 bytes
Redo Buffers                7639040 bytes
Database mounted.
Database opened.
SQL> alter pluggable database all open upgrade force;

Pluggable database altered.

SQL> show pdbs

    CON_ID CON_NAME                       OPEN MODE  RESTRICTED
---------- ------------------------------ ---------- ----------
         2 PDB$SEED                       MIGRATE    YES
         3 PDB1                           MIGRATE    YES
         4 PDB2                           MIGRATE    YES
SQL>
```

cd $ORACLE_HOME/rdbms/admin
$ORACLE_HOME/perl/bin/perl catctl.pl -n 4 -l /home/oracle/logs catupgrd.sql
其中：-n 表示并行度的意思，-l 表示输出日志目录

```
[CDB3] oracle@hol:/u01/app/oracle/product/19/rdbms/admin
$ $ORACLE_HOME/perl/bin/perl catctl.pl -n 4 -l /home/oracle/logs catupgrd.sql

Argument list for [catctl.pl]
For Oracle internal use only A = 0
Run in                       c = 0
Do not run in                C = 0
Input Directory              d = 0
Echo OFF                     e = 1
Simulate                     E = 0
Forced cleanup               F = 0
Log Id                       i = 0
Child Process                I = 0
Log Dir                      l = /home/oracle/logs
Priority List Name           L = 0
Upgrade Mode active          M = 0
SQL Process Count            n = 4
SQL PDB Process Count        N = 0
Open Mode Normal             o = 0
Start Phase                  p = 0
End Phase                    P = 0
Reverse Order                r = 0
AutoUpgrade Resume           R = 0
Script                       s = 0
Serial Run                   S = 0
RO User Tablespaces          T = 0
Display Phases               y = 0
Debug catcon.pm              z = 0
Debug catctl.pl              Z = 0

catctl.pl VERSION: [19.0.0.0.0]
          STATUS: [Production]
           BUILD: [RDBMS_19.3.0.0.0DBRU_LINUX.X64_190417]
```

```
Number of Cpus        = 4
Database Name         = CDB3
DataBase Version      = 12.2.0.1.0
catcon::set_log_file_base_path: ALL catcon-related output will be written to [/home/oracle/l

catcon::set_log_file_base_path: catcon: See [/home/oracle/logs/catupgrdcdbroot*.log] files f

catcon::set_log_file_base_path: catcon: See [/home/oracle/logs/catupgrdcdbroot_*.lst] files


Log file directory = [/home/oracle/logs]

Parallel SQL Process Count (PDB)      = 2
Parallel SQL Process Count (CDB$ROOT) = 4
Concurrent PDB Upgrades               = 2
Generated PDB Inclusion:[PDB$SEED PDB1 PDB2]
Components in [CDB$ROOT]
    Installed [APS CATALOG CATJAVA CATPROC CONTEXT DV JAVAVM OLS ORDIM OWM SDO XDB XML XOQ]
Not Installed [APEX EM MGW ODM RAC WK]
```


Phases [0-107] End Time:[2019_12_19 13:18:13]
Container Lists Inclusion:[PDB2] Exclusion:[NONE]
Grand Total Time: 1710s [PDB2]
LOG FILES: (/home/oracle/logs/catupgrdpdb2*.log)
Upgrade Summary Report Located in:
/home/oracle/logs/upg_summary.log
Time: 2185s For CDB$ROOT
Time: 4038s For PDB(s)
Grand Total Time: 6223s
LOG FILES: (/home/oracle/logs/catupgrdcdbroot*.log)
Upgrade Summary Report Located in:
/home/oracle/logs/upg_summary.log
Grand Total Upgrade Time: [0d:1h:43m:43s]
[CDB3] oracle@hol:/u01/app/oracle/product/19/rdbms/admin

```
Oracle Database Release 19 Post-Upgrade Status Tool    12-19-2019 13:17:5
Container Database: CDB3
[CON_ID: 4 => PDB2]

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

Oracle Server                          UPGRADED      19.3.0.0.0  00:14:04
JServer JAVA Virtual Machine           UPGRADED      19.3.0.0.0  00:00:50
Oracle XDK                             UPGRADED      19.3.0.0.0  00:00:47
Oracle Database Java Packages          UPGRADED      19.3.0.0.0  00:00:06
OLAP Analytic Workspace                UPGRADED      19.3.0.0.0  00:00:10
Oracle Label Security                  UPGRADED      19.3.0.0.0  00:00:04
Oracle Database Vault                  UPGRADED      19.3.0.0.0  00:00:38
Oracle Text                            UPGRADED      19.3.0.0.0  00:00:12
Oracle Workspace Manager               UPGRADED      19.3.0.0.0  00:00:21
Oracle Real Application Clusters       UPGRADED      19.3.0.0.0  00:00:00
Oracle XML Database                    UPGRADED      19.3.0.0.0  00:02:15
Oracle Multimedia                      UPGRADED      19.3.0.0.0  00:00:33
Spatial                                UPGRADED      19.3.0.0.0  00:03:44
Oracle OLAP API                        UPGRADED      19.3.0.0.0  00:00:08
Datapatch                                                        00:01:55
Final Actions                                                    00:02:11
Post Upgrade                                                     00:00:12

Total Upgrade Time: 00:27:27 [CON_ID: 4 => PDB2]

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


Upgrade Times Sorted In Descending Order

Total Upgrade Time: 00:42:59 [CON_ID: 2 => PDB$SEED * ]
Total Upgrade Time: 00:36:11 [CON_ID: 3 => PDB1]
Total Upgrade Time: 00:33:20 [CON_ID: 1 => CDB$ROOT]
Total Upgrade Time: 00:27:27 [CON_ID: 4 => PDB2]
Grand Total Upgrade Time:    [0d:1h:43m:43s]
[CDB3] oracle@hol:/u01/app/oracle/product/19/rdbms/admin
```

升级后续步骤

编译无效对象

shut immediate
. 19cdb3
sqlplus / as sysdba
startup
alter pluggable database all open;

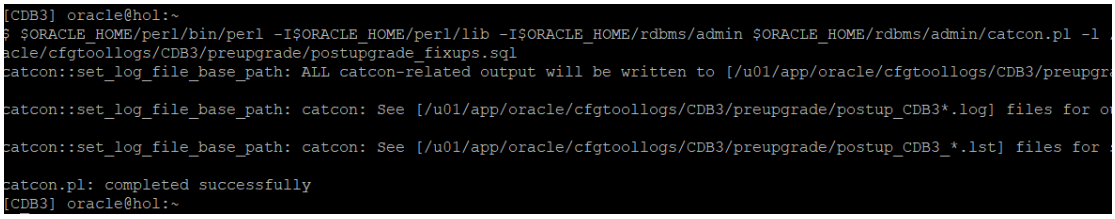
```
SQL> shut immediate
Database closed.
Database dismounted.
ORACLE instance shut down.
SQL> exit
Disconnected from Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
Version 19.3.0.0.0
[CDB3] oracle@hol:/u01/app/oracle/product/19/rdbms/admin
$ . 19cdb3
[CDB3] oracle@hol:/u01/app/oracle/product/19/rdbms/admin
$ sqlplus / as sysdba

SQL*Plus: Release 19.0.0.0.0 - Production on Thu Dec 19 13:21:20 2019
Version 19.3.0.0.0

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

Connected to an idle instance.

SQL> startup
ORACLE instance started.

Total System Global Area 2432695872 bytes
Fixed Size                  9137728 bytes
Variable Size             671088640 bytes
Database Buffers         1744830464 bytes
Redo Buffers                7639040 bytes
Database mounted.
Database opened.
SQL> alter pluggable database all open;

Pluggable database altered.

SQL> show pdbs

    CON_ID CON_NAME                       OPEN MODE  RESTRICTED
---------- ------------------------------ ---------- ----------
         2 PDB$SEED                       READ ONLY  NO
         3 PDB1                           READ WRITE NO
         4 PDB2                           READ WRITE NO
SQL>
```

$ORACLE_HOME/perl/bin/perl -I$ORACLE_HOME/perl/lib -I$ORACLE_HOME/rdbms/admin $ORACLE_HOME/rdbms/admin/catcon.pl -l /home/oracle/logs -b comp_invalid_objs -- --x"@$ORACLE_HOME/rdbms/admin/utlrp.sql"

```
$ $ORACLE_HOME/perl/bin/perl -I$ORACLE_HOME/perl/lib -I$ORACLE_HOME/rdbms/admin $ORACLE_HOME/rdbms/admin/catcon.pl -l /home/oracle/logs
/utlrp.sql"
catcon::set_log_file_base_path: ALL catcon-related output will be written to [/home/oracle/logs/comp_invalid_objs_catcon_26685.lst]

catcon::set_log_file_base_path: catcon: See [/home/oracle/logs/comp_invalid_objs*.log] files for output generated by scripts

catcon::set_log_file_base_path: catcon: See [/home/oracle/logs/comp_invalid_objs_*.lst] files for spool files, if any

catcon.pl: completed successfully
[CDB3] oracle@hol:~
```

执行 postupgrade 脚本

$ORACLE_HOME/perl/bin/perl -I$ORACLE_HOME/perl/lib -I$ORACLE_HOME/rdbms/admin $ORACLE_HOME/rdbms/admin/catcon.pl -l /u01/app/oracle/cfgtoollogs/CDB3/preupgrade/ -b postup_CDB3 /u01/app/oracle/cfgtoollogs/CDB3/preupgrade/postupgrade_fixups.sql



$ORACLE_HOME/perl/bin/perl -I$ORACLE_HOME/perl/lib -I$ORACLE_HOME/rdbms/admin $ORACLE_HOME/rdbms/admin/catcon.pl -l /home/oracle/logs -b gather_dict_stats -- -- x"exec dbms_stats.gather_dictionary_stats" -c 'PDB2'

调整/etc/oratab

将
CDB3:/u01/app/oracle/product/12.2.0.1:Y
修改为
CDB3:/u01/app/oracle/product/19:Y

```
CDB3:/u01/app/oracle/product/19:N
```

调整兼容性参数

show parameter compatible
show spparameter compatible
alter system set compatible='19.0.0' scope=spfile;
show spparameter compatible
shut immediate
exit

```
$ sqlplus / as sysdba

SQL*Plus: Release 19.0.0.0.0 - Production on Thu Dec 19 13:45:43 2019
Version 19.3.0.0.0

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


Connected to:
Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
Version 19.3.0.0.0

SQL> show parameter compatible

NAME                                 TYPE        VALUE
------------------------------------ ----------- ------------------------------
compatible                           string      12.2.0
noncdb_compatible                    boolean     FALSE
SQL> show spparameter compatible

SID      NAME                          TYPE        VALUE
-------- ----------------------------- ----------- ----------------------------
*        compatible                    string      12.2.0
*        noncdb_compatible             boolean
SQL> alter system set compatible='19.0.0' scope=spfile;

System altered.

SQL> show spparameter compatible

SID      NAME                          TYPE        VALUE
-------- ----------------------------- ----------- ----------------------------
*        compatible                    string      19.0.0
*        noncdb_compatible             boolean
SQL> shut immediate
Database closed.
Database dismounted.
ORACLE instance shut down.
SQL> exit
Disconnected from Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
Version 19.3.0.0.0
[CDB3] oracle@hol:~
```

到这里我们就把12.2.0.1 版本多租户数据库CDB12 整个升级到19c 了

附录

catcon.pl

```
$ $ORACLE_HOME/perl/bin/perl $ORACLE_HOME/rdbms/admin/catcon.pl

  Usage: catcon    [-h, --help]
                   [-u, --usr username
                      [{/password | -w, --usr_pwd_env_var env-var-name}]]
                   [-U, --int_usr username
                      [{/password | -W, --int_usr_pwd_env_var env-var-name]]
                   [-d, --script_dir directory]
                   [-l, --log_dir directory]
                   [{-c, --incl_con | -C, --excl_con} container]
                   [-p, --catcon_instances degree-of-parallelism]
                   [-z, --ez_conn EZConnect-strings]
```

```
                    [-e, --echo]
                    [-s, --spool]
                    [-E, --error_logging
                      { ON | errorlogging-table-other-than-SPERRORLOG } ]
                    [-F, --app_con Application-Root]
                    [-V, --ignore_errors errors-to-ignore ]
                    [-l, --no_set_errlog_ident]
                    [-g, --diag]
                    [-v, --verbose]
                    [-f, --ignore_unavailable_pdbs]
                    [--fail_on_unopenable_pdbs]
                    [-r, --reverse]
                    [-R, --recover]
                    [-m, --pdb_seed_mode pdb-mode]
                    [--force_pdb_mode pdb-mode]
                    [--all_instances]
                    [--upgrade]
                    [--ezconn_to_pdb pdb-name]
                    [--sqlplus_dir directory]
                    [--dflt_app_module app-module]
                    -b, --log_file_base log-file-name-base
                    --
                    { sqlplus-script [arguments] | --x<SQL-statement> } ...

Optional:
   -h, --help
       print usage info and exit
   -u, --usr
       username (optional /password; otherwise prompts for password)
       used to connect to the database to run user-supplied scripts or
       SQL statements
       defaults to "/ as sysdba"
   -w, --usr_pwd_env_var
       name of environment variable which contains a password for a user
       whose name was specified with --usr;
       NOTE: should NOT be used if --usr specified a password
   -U, --int_usr
       username (optional /password; otherwise prompts for password)
       used to connect to the database to perform internal tasks
       defaults to "/ as sysdba"
   -W, --int_usr_pwd_env_var
       name of environment variable which contains a password for a user
       whose name was specified with --int_usr;
       NOTE: should NOT be used if --int_usr specified a password
```

```
       -d, --script_dir
           directory containing the file to be run
       -l, --log_dir
           directory to use for spool log files
       -c, --incl_con
           container(s) in which to run sqlplus scripts, i.e. skip all
           Containers not named here; for example,
              --incl_con 'PDB1 PDB2',
       -C, --excl_con
            container(s) in which NOT to run sqlplus scripts, i.e. skip all
           Containers named here; for example,
              --excl_con 'CDB PDB3'

         NOTE: --incl_con and --excl_con are mutually exclusive

       -p, --catcon_instances
           expected number of concurrent invocations of this script on a given
           host

         NOTE: this parameter rarely needs to be specified

       -z, --ez_conn
           blank-separated EZConnect strings corresponding to RAC instances
           which can be used to run scripts
       -e, --echo
           sets echo on while running sqlplus scripts
       -s, --spool
           output of running every script will be spooled into a file whose name
           will be
              <log-file-name-
base>_<script_name_without_extension>_[<container_name_if_any>].<default_extension
>
       -E, --error_logging
           sets errorlogging on; if ON is specified, default error logging table
           will be used, otherwise, specified error logging table (which must
           have been created in every Container) will be used
       -F, --app_con
           causes scripts to run in a Application Root and all Application PDBs
           belonging to it;
           ***CANNOT*** be specified concurrently with -{cC} flags
       -V, --ignore_errors
           causes catcon to ignore errors encountered during specified operations.
           The following options are supported:
              script_path == ignore errors while validating script path
```

```
-S, --user_scripts
    running user scripts, meaning that _oracle_script will not be set and
    all entities created by scripts will not be marked as Oracle-maintained
-l, --no_set_errlog_ident
    do not issue set Errorlogging Identifier (ostensibly because the
    caller already did it and does not want us to override it)
-g, --diag
    turns on production of diagnostic info while running this script
-v, --verbose
    turns on verbose output which is less verbose than debugging output
-f, --ignore_unavailable_pdbs
    instructs catcon to ignore PDBs which are closed or, if --incl_con or
    --excl_con was used, do not exist and process existing PDBs which
    were specified (explicitly or implicitly) and are open

    NOTE: if this flag is not specified and some specified PDBs do not
          exist or are not open, an error will be returned and none of
          the Containers will be processed.

--fail_on_unopenable_pdbs
    by default, if the caller instructs catcon to open PDBs against which
    scripts will be run in a certain mode (using --pdb_seed_mode or
    --force_pdb_mode), and some of them could not be opened in that mode,
    catcon will issue a warning and proceed to execute scripts, skipping
    such PDBs.

    This option should be specified if the caller prefers that
    in such cases catcon report an error and not execute supplied scripts
    against any PDBs.

-r, --reverse
    causes scripts to be run in all PDBs and then in the Root (reverse
    of the default order); required for running catdwgrd.sql in a CDB
-m, --pdb_seed_mode
    mode in which PDB should be opened; one of the following values
    may be specified:
    - UNCHANGED - leave PDB in whatever mode it is already open
    - READ WRITE (default)
    - READ ONLY
    - UPGRADE
    - DOWNGRADE

    NOTE: if the desired mode is different from the mode in which
          PDB is open, it is will be closed and reopened in the
```

desired mode before running any scripts; after all scripts were
run, it will be restored to the original mode

--pdb_seed_mode should not be specified if --force_pdb_mode
is specified because mode supplied with the latter will apply
to PDB

--force_pdb_mode
mode in which ALL PDBs against which scripts will be run must be
opened; one of the following values may be specified:
- UNCHANGED - leave PDBs in whatever mode they are already
open (default)
- READ WRITE
- READ ONLY
- UPGRADE
- DOWNGRADE

NOTE: if the desired mode is different from the mode in which
some of the PDBs specified by the caller are open, they will be
closed and reopened in the desired mode before running any
scripts; after all scripts were run, they will be restored to
the original mode

--force_pdb_mode should not be specified if --pdb_seed_mode
is specified because mode supplied with the latter will apply
to PDB

-R, --recover
causes catcon to recover from unexpected death of a SQL*Plus process
that it spawned; if not specified, such event will cause catcon to die

-D, --disable_lockdown
causes catcon to disable lockdown profile before running script(s) in
a PDB and reenable them before existing

--all_instances
if used to run scripts against a CDB and if --force_pdb_mode was
specified, catcon will attempt to run scripts on PDBs using all
instances on which a CDB is open

--upgrade
catcon is being invoked in the course of upgrading a database

--ezconn_to_pdb

```
        caller is expected to provide catcon with one or more EZConnect strings
        leading to the specified PDB; all specified scripts will be run ONLY
        against that PDB; neither --incl_con nor --excl_con may be specified
        concurrentrly with this flag

    --sqlplus_dir
        directory where sqlplus binary which catcon should use can be found
        (e.g. if    does not include it or if the caller wants
        catcon to use a particular version of sqlplus binary)

    --dflt_app_module
        if specified, value to which catcon should set APPLICATION MODULE

Mandatory:
    -b, --log_file_base
         base name (e.g. catcon_test) for log and spool file names

    sqlplus-script - sqlplus script to run OR
    SQL-statement    - a statement to execute

NOTES:
    - if --x<SQL-statement> is the first non-option string, it needs to be
        preceeded with -- to avoid confusing module parsing options into
        assuming that '-' is an option which that module is not expecting and
        about which it will complain
    - command line parameters to SQL scripts can be introduced using --p
    - interactive (or secret) parameters to SQL scripts can be introduced
        using --P
    - occupying middle ground between --p and --P, parameters whose values
        are stored in environment variables can be specified using --e
        (as in --e"env_var_holding_password")

    For example,
        perl catcon.pl ... x.sql --p"John" --P"Enter Password for John:" ...
    or store John's password in environment variable JOHNS_PASSWORD and
    then issue
        perl catcon.pl ... x.sql --p"John" --e"JOHNS_PASSWORD" ..
```

环境变量

upgr

```
#!/bin/sh
ORACLE_BASE=/u01/app/oracle
ORACLE_HOME=$ORACLE_BASE/product/11.2.0.4
OH19=$ORACLE_BASE/product/19

if [ "$1" = "" ] ; then
   ORACLE_SID=UPGR
else
   ORACLE_SID=$1
fi

ORACLE_HOSTNAME=hol.localdomain
TNS_ADMIN=/u01/app/oracle/product/19/network/admin
NLS_LANG=AMERICAN_AMERICA.AL32UTF8
PATH=$ORACLE_HOME/bin:$PATH
LD_LIBRARY_PATH=$ORACLE_HOME/lib:$LD_LIBRARY_PATH
CLASSPATH=.:$ORACLE_HOME/jdbc/lib/classes12.jar:$ORACLE_HOME/jdbc/lib/nls_charset1
2.jar:$ORACLE_HOME/rdbms/jlib/xdb.jar:$ORACLE_HOME/lib/xmlparserv2.jar:$ORACLE_HO
ME/sqlj/lib/utl_dbws.jar
export ORACLE_BASE ORACLE_HOME ORACLE_SID OH18 NLS_LANG CLASSPATH PATH
LD_LIBRARY_PATH TNS_ADMIN

export PS1="[$ORACLE_SID] \\u@\\h:\\w\n$ "

alias sql="sqlplus /nolog"
alias s="sqlplus / as sysdba"
alias oh="cd $ORACLE_HOME"
alias l="ls -la"

umask 022
```

upgr19

```
#!/bin/sh
ORACLE_BASE=/u01/app/oracle
ORACLE_HOME=$ORACLE_BASE/product/19
OH19=$ORACLE_BASE/product/19
```

```
if [ "$1" = "" ] ; then
   ORACLE_SID=UPGR
else
   ORACLE_SID=$1
fi

ORACLE_HOSTNAME=hol.localdomain
TNS_ADMIN=/u01/app/oracle/product/19/network/admin
NLS_LANG=AMERICAN_AMERICA.AL32UTF8
PATH=$ORACLE_HOME/bin:$PATH
LD_LIBRARY_PATH=$ORACLE_HOME/lib:$LD_LIBRARY_PATH
CLASSPATH=.:$ORACLE_HOME/jdbc/lib/classes12.jar:$ORACLE_HOME/jdbc/lib/nls_charset1
2.jar:$ORACLE_HOME/rdbms/jlib/xdb.jar:$ORACLE_HOME/lib/xmlparserv2.jar:$ORACLE_HO
ME/sqlj/lib/utl_dbws.jar
export ORACLE_BASE ORACLE_HOME ORACLE_SID OH18 NLS_LANG CLASSPATH PATH
LD_LIBRARY_PATH

export PS1="[$ORACLE_SID] \\u@\\h:\\w\n$ "

alias sql="/u01/app/oracle/product/12.2.0.1/sqldeveloper/sqlcl/bin/sql / as sysdba"
alias s="sqlplus / as sysdba"
alias oh="cd $ORACLE_HOME"
alias l="ls -la"

umask 022

unset TNS_ADMIN
```

db12

```
#!/bin/sh
ORACLE_BASE=/u01/app/oracle
ORACLE_HOME=$ORACLE_BASE/product/12.2.0.1
OH19=$ORACLE_BASE/product/19

if [ "$1" = "" ] ; then
   ORACLE_SID=DB12
else
   ORACLE_SID=$1
fi

ORACLE_HOSTNAME=hol.localdomain
```

```
TNS_ADMIN=/u01/app/oracle/product/19/network/admin
NLS_LANG=AMERICAN_AMERICA.AL32UTF8
PATH=$ORACLE_HOME/bin:$PATH
LD_LIBRARY_PATH=$ORACLE_HOME/lib:$LD_LIBRARY_PATH
CLASSPATH=.:$ORACLE_HOME/jdbc/lib/classes12.jar:$ORACLE_HOME/jdbc/lib/nls_charset1
2.jar:$ORACLE_HOME/rdbms/jlib/xdb.jar:$ORACLE_HOME/lib/xmlparserv2.jar:$ORACLE_HO
ME/sqlj/lib/utl_dbws.jar
export ORACLE_BASE ORACLE_HOME ORACLE_SID OH18 NLS_LANG CLASSPATH PATH
LD_LIBRARY_PATH TNS_ADMIN

export PS1="[$ORACLE_SID] \\u@\\h:\\w\n$ "

alias sql="sqlplus /nolog"
alias s="sqlplus / as sysdba"
alias oh="cd $ORACLE_HOME"
alias l="ls -la"

umask 022
```

db19

```
#!/bin/sh
ORACLE_BASE=/u01/app/oracle
ORACLE_HOME=$ORACLE_BASE/product/19
OH19=$ORACLE_BASE/product/19

if [ "$1" = "" ] ; then
   ORACLE_SID=DB12
else
   ORACLE_SID=$1
fi

ORACLE_HOSTNAME=hol.localdomain
TNS_ADMIN=/u01/app/oracle/product/19/network/admin
NLS_LANG=AMERICAN_AMERICA.AL32UTF8
PATH=$ORACLE_HOME/bin:$PATH
LD_LIBRARY_PATH=$ORACLE_HOME/lib:$LD_LIBRARY_PATH
CLASSPATH=.:$ORACLE_HOME/jdbc/lib/classes12.jar:$ORACLE_HOME/jdbc/lib/nls_charset1
2.jar:$ORACLE_HOME/rdbms/jlib/xdb.jar:$ORACLE_HOME/lib/xmlparserv2.jar:$ORACLE_HO
ME/sqlj/lib/utl_dbws.jar
export ORACLE_BASE ORACLE_HOME ORACLE_SID OH18 NLS_LANG CLASSPATH PATH
LD_LIBRARY_PATH TNS_ADMIN
```

```
export PS1="[$ORACLE_SID] \\u@\\h:\\w\n$ "

alias sql="/u01/app/oracle/product/12.2.0.1/sqldeveloper/sqlcl/bin/sql / as sysdba"
alias s="sqlplus / as sysdba"
alias oh="cd $ORACLE_HOME"
alias l="ls -la"

umask 022
```

cdb1

```
#!/bin/sh
ORACLE_BASE=/u01/app/oracle
ORACLE_HOME=$ORACLE_BASE/product/12.2.0.1
OH19=$ORACLE_BASE/product/19

if [ "$1" = "" ] ; then
   ORACLE_SID=CDB1
else
   ORACLE_SID=$1
fi

ORACLE_HOSTNAME=hol.localdomain
TNS_ADMIN=/u01/app/oracle/product/19/network/admin
NLS_LANG=AMERICAN_AMERICA.AL32UTF8
PATH=$ORACLE_HOME/bin:$PATH
LD_LIBRARY_PATH=$ORACLE_HOME/lib:$LD_LIBRARY_PATH
CLASSPATH=.:$ORACLE_HOME/jdbc/lib/classes12.jar:$ORACLE_HOME/jdbc/lib/nls_charset1
2.jar:$ORACLE_HOME/rdbms/jlib/xdb.jar:$ORACLE_HOME/lib/xmlparserv2.jar:$ORACLE_HO
ME/sqlj/lib/utl_dbws.jar
export ORACLE_BASE ORACLE_HOME ORACLE_SID OH18 NLS_LANG CLASSPATH PATH
LD_LIBRARY_PATH TNS_ADMIN

export PS1="[$ORACLE_SID] \\u@\\h:\\w\n$ "

alias sql="sqlplus /nolog"
alias s="sqlplus / as sysdba"
alias oh="cd $ORACLE_HOME"
alias l="ls -la"

umask 022
```

cdb2

```
#!/bin/sh
ORACLE_BASE=/u01/app/oracle
ORACLE_HOME=$ORACLE_BASE/product/19
OH19=$ORACLE_BASE/product/19

if [ "$1" = "" ] ; then
   ORACLE_SID=CDB2
else
   ORACLE_SID=$1
fi

ORACLE_HOSTNAME=hol.localdomain
NLS_LANG=AMERICAN_AMERICA.AL32UTF8
TNS_ADMIN=/u01/app/oracle/product/19/network/admin
PATH=$ORACLE_HOME/bin:$PATH
LD_LIBRARY_PATH=$ORACLE_HOME/lib:$LD_LIBRARY_PATH
CLASSPATH=.:$ORACLE_HOME/jdbc/lib/classes12.jar:$ORACLE_HOME/jdbc/lib/nls_charset1
2.jar:$ORACLE_HOME/rdbms/jlib/xdb.jar:$ORACLE_HOME/lib/xmlparserv2.jar:$ORACLE_HO
ME/sqlj/lib/utl_dbws.jar
export ORACLE_BASE ORACLE_HOME ORACLE_SID OH18 NLS_LANG CLASSPATH PATH
LD_LIBRARY_PATH TNS_ADMIN

export PS1="[$ORACLE_SID] \\u@\\h:\\w\n$ "

alias sql="/u01/app/oracle/product/12.2.0.1/sqldeveloper/sqlcl/bin/sql / as sysdba"
alias s="sqlplus / as sysdba"
alias oh="cd $ORACLE_HOME"
alias l="ls -la"

umask 022
```

ftex

```
#!/bin/sh
ORACLE_BASE=/u01/app/oracle
ORACLE_HOME=$ORACLE_BASE/product/11.2.0.4
OH19=$ORACLE_BASE/product/19

if [ "$1" = "" ] ; then
   ORACLE_SID=FTEX
```

```
else
   ORACLE_SID=$1
fi

TNS_ADMIN=/u01/app/oracle/product/19/network/admin
NLS_LANG=AMERICAN_AMERICA.AL32UTF8
PATH=$ORACLE_HOME/bin:$PATH
LD_LIBRARY_PATH=$ORACLE_HOME/lib:$LD_LIBRARY_PATH
CLASSPATH=.:$ORACLE_HOME/jdbc/lib/classes12.jar:$ORACLE_HOME/jdbc/lib/nls_charset1
2.jar:$ORACLE_HOME/rdbms/jlib/xdb.jar:$ORACLE_HOME/lib/xmlparserv2.jar:$ORACLE_HO
ME/sqlj/lib/utl_dbws.jar
export ORACLE_BASE ORACLE_HOME ORACLE_SID OH18 NLS_LANG CLASSPATH PATH
LD_LIBRARY_PATH TNS_ADMIN

export PS1="[$ORACLE_SID] \\u@\\h:\\w\n$ "

alias sql="sqlplus /nolog"
alias s="sqlplus / as sysdba"
alias oh="cd $ORACLE_HOME"
alias l="ls -la"

umask 022
```

ftex19

```
#!/bin/sh
ORACLE_BASE=/u01/app/oracle
ORACLE_HOME=$ORACLE_BASE/product/19
OH19=$ORACLE_BASE/product/19

if [ "$1" = "" ] ; then
   ORACLE_SID=FTEX
else
   ORACLE_SID=$1
fi

ORACLE_HOSTNAME=hol.localdomain
TNS_ADMIN=/u01/app/oracle/product/19/network/admin
NLS_LANG=AMERICAN_AMERICA.AL32UTF8
PATH=$ORACLE_HOME/bin:$PATH
LD_LIBRARY_PATH=$ORACLE_HOME/lib:$LD_LIBRARY_PATH
CLASSPATH=.:$ORACLE_HOME/jdbc/lib/classes12.jar:$ORACLE_HOME/jdbc/lib/nls_charset1
2.jar:$ORACLE_HOME/rdbms/jlib/xdb.jar:$ORACLE_HOME/lib/xmlparserv2.jar:$ORACLE_HO
```

```
ME/sqlj/lib/utl_dbws.jar
export ORACLE_BASE ORACLE_HOME ORACLE_SID OH18 NLS_LANG CLASSPATH PATH
LD_LIBRARY_PATH

export PS1="[$ORACLE_SID] \\u@\\h:\\w\n$ "

alias sql="/u01/app/oracle/product/12.2.0.1/sqldeveloper/sqlcl/bin/sql / as sysdba"
alias s="sqlplus / as sysdba"
alias oh="cd $ORACLE_HOME"
alias l="ls -la"

umask 022

unset TNS_ADMIN
```

cdb3

```
#!/bin/sh
ORACLE_BASE=/u01/app/oracle
ORACLE_HOME=$ORACLE_BASE/product/12.2.0.1
OH19=$ORACLE_BASE/product/19

if [ "$1" = "" ] ; then
   ORACLE_SID=CDB3
else
   ORACLE_SID=$1
fi

ORACLE_HOSTNAME=hol.localdomain
TNS_ADMIN=/u01/app/oracle/product/19/network/admin
NLS_LANG=AMERICAN_AMERICA.AL32UTF8
PATH=$ORACLE_HOME/bin:$PATH
LD_LIBRARY_PATH=$ORACLE_HOME/lib:$LD_LIBRARY_PATH
CLASSPATH=.:$ORACLE_HOME/jdbc/lib/classes12.jar:$ORACLE_HOME/jdbc/lib/nls_charset1
2.jar:$ORACLE_HOME/rdbms/jlib/xdb.jar:$OR
ACLE_HOME/lib/xmlparserv2.jar:$ORACLE_HOME/sqlj/lib/utl_dbws.jar
export ORACLE_BASE ORACLE_HOME ORACLE_SID OH18 NLS_LANG CLASSPATH PATH
LD_LIBRARY_PATH TNS_ADMIN

export PS1="[$ORACLE_SID] \\u@\\h:\\w\n$ "

alias sql="sqlplus /nolog"
alias s="sqlplus / as sysdba"
```

```
alias oh="cd $ORACLE_HOME"
alias l="ls -la"

umask 022
```

19cdb3

```
#!/bin/sh
ORACLE_BASE=/u01/app/oracle
ORACLE_HOME=$ORACLE_BASE/product/19
OH19=$ORACLE_BASE/product/19

if [ "$1" = "" ] ; then
   ORACLE_SID=CDB3
else
   ORACLE_SID=$1
fi

ORACLE_HOSTNAME=hol.localdomain
NLS_LANG=AMERICAN_AMERICA.AL32UTF8
TNS_ADMIN=/u01/app/oracle/product/19/network/admin
PATH=$ORACLE_HOME/bin:$PATH
LD_LIBRARY_PATH=$ORACLE_HOME/lib:$LD_LIBRARY_PATH
CLASSPATH=.:$ORACLE_HOME/jdbc/lib/classes12.jar:$ORACLE_HOME/jdbc/lib/nls_charset1
2.jar:$ORACLE_HOME/rdbms/jlib/xdb.jar:$OR
ACLE_HOME/lib/xmlparserv2.jar:$ORACLE_HOME/sqlj/lib/utl_dbws.jar
export ORACLE_BASE ORACLE_HOME ORACLE_SID OH18 NLS_LANG CLASSPATH PATH
LD_LIBRARY_PATH TNS_ADMIN

export PS1="[$ORACLE_SID] \\u@\\h:\\w\n$ "

alias sql="/u01/app/oracle/product/12.2.0.1/sqldeveloper/sqlcl/bin/sql / as sysdba"
alias s="sqlplus / as sysdba"
alias oh="cd $ORACLE_HOME"
alias l="ls -la"

umask 022
```

Reference

https://mikedietrichde.com/database-upgrade-hands-on-lab-oracle-18c-and-19c/
https://docs.oracle.com/en/database/oracle/oracle-database/19/upgrd/database-preparation-tasks-to-complete-before-upgrades.html#GUID-58C1A4B9-32F4-44E5-

B82B-5D0EC78A6AE2
https://docs.oracle.com/en/database/oracle/oracle-database/19/upgrd/post-upgrade-tasks-oracle-database.html#GUID-637ADBD0-866B-4B64-9513-4C7CDE84895C