11gR1/2 Installation and Upgrade Steps

This document outlines the entire process to upgrade an 11gR1 Oracle database and Clusterware to 11gR2.

It is assumed that appropriate storage devices have been allocated (like ext3 for backups and shared drives for Clusterware/OCFS2).

It is also assumed that appropriate RPMs have been installed for ASM and OCFS2 (ensuring that they match the correct kernel version of the OS).

The /etc/hosts file is also assumed to have appropriate entries for all nodes participating in the Cluster.

Storage configuration

Partition the provisioned disks using any appropriate utility: I am using fdisk

```
# fdisk /dev/sdd
```

Use n, p, 1, <First cylinder: OS-selected; Enter>, <Last cylinder: OS-selected; Enter>, w

Once all disks are partitioned, run fdisk on the other node(s) with just the “w” option and the partition tables are updated on those node(s).

```
# fdisk /dev/sdd
```

```
Enter command: w
```

Format Disk Partitions

a. For ext3 file systems, use the following:

```
# mkfs –t <file_system_type> <device>
```

```
# mkfs –t ext3 /dev/sdj1
```

Check and make permanent:

```
# fsck –f –y /dev/sdj1
```

Add entry to /etc/fstab

```
/dev/sdj1          /u03            ext3    defaults       0 0
```

b. For OCFS2 file systems, use one of the following methods:

OCFS2 Console

As root, set the DISPLAY parameter

```
# export DISPLAY=172.xxx.xxx.xxx:0.0
```

Run ocfs2console GUI

```
# ocfs2console &
```

Click Tasks -> Format

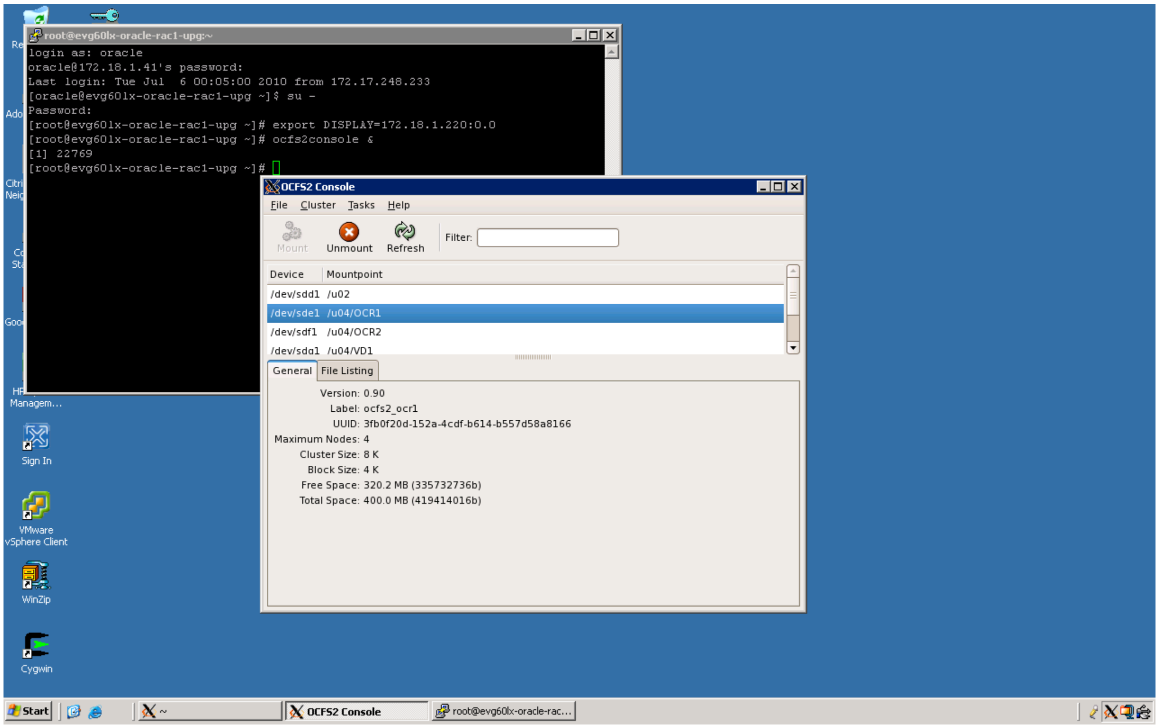
Select the appropriate device, for example, /dev/sdd1

Provide appropriate volume labels (the following is what I follow usually):
/u02 shared = ocfs2_shared
/u04/OCR1 = ocfs2_ocr1 (Oracle Cluster Registry file #1)
/u04/OCR2 = ocfs2_ocr2 (Oracle Cluster Registry file #2)
/u04/VD1 = ocfs2_vd1 (Voting Disk #1)
/u04/VD2 = ocfs2_vd2 (Voting Disk #2)
/u04/VD3 = ocfs2_vd3 (Voting Disk #3)

cluster size = 8k
block size = 4k
Max # of nodes = 4 (can be increased but cannot be decreased!)

Click OK and Click Yes on the subsequent window. The disk formatting will happen and will be displayed in the 2nd panel below with all details.

Repeat for the other OCFS2 drives.



```
login as: oracle
oracle@172.18.1.41's password:
Last login: Tue Jul  6 00:05:00 2010 from 172.17.248.233
[oracle@evg60lx-oracle-rac1-upg ~]$ su -
Password:
[root@evg60lx-oracle-rac1-upg ~]# export DISPLAY=172.18.1.220:0.0
[root@evg60lx-oracle-rac1-upg ~]# ocfs2console &
[1] 22769
[root@evg60lx-oracle-rac1-upg ~]#
```

Device	Mountpoint
/dev/sdd1	/u02
/dev/sde1	/u04/OCR1
/dev/sdf1	/u04/OCR2
/dev/sdg1	/u04/VD1

```
         Version: 0.90
           Label: ocfs2_ocr1
            UUID: 3fb0f20d-152a-4cdf-b614-b557d58a8166
   Maximum Nodes: 4
    Cluster Size: 8 K
      Block Size: 4 K
      Free Space: 320.2 MB (335732736b)
     Total Space: 400.0 MB (419414016b)
```

Using command-line tool to format OCFS2 volumes:

```
# mkfs.ocfs2 –b <block_size> –C <cluster_size> –N <#_of_nodes> -L <volume_label>
<mount_point> <device>
```

```
# mkfs.ocfs2 –b 4k –C 8k –N 4 –L ocfs2_vd1 /u04/VD1 /dev/sdg1
```

!!! Unless specifically mentioned, all steps indicated in this document needs to be done on only one node!!!

Add entries in /etc/fstab for all devices:

```
/dev/sdd1     /u02        ocfs2   _netdev,datavolume,nointr   0 0
/dev/sde1     /u04/OCR1   ocfs2   _netdev,datavolume,nointr   0 0
/dev/sdf1     /u04/OCR2   ocfs2   _netdev,datavolume,nointr   0 0
/dev/sdg1     /u04/VD1    ocfs2   _netdev,datavolume,nointr   0 0
/dev/sdh1     /u04/VD2    ocfs2   _netdev,datavolume,nointr   0 0
/dev/sdi1     /u04/VD3    ocfs2   _netdev,datavolume,nointr   0 0
```

Reboot all nodes to ensure that disk partitions are properly mounted.

Create ASM disk

Assuming ASMLib drivers have been installed & configured and disk devices identified and partitioned, execute the following to mark a device as an ASM disk:

```
# /etc/init.d/oracleasm createdisk <volume_name> <device>

# /etc/init.d/oracleasm createdisk VOL1 /dev/sdc1
```

Scan the system

```
# /etc/init.d/oracleasm scandisks
```

Check if the ASM volume is created

```
# /etc/init.d/oracleasm listdisks
VOL1
```

Scan other nodes for the ASM disk(s):

```
#/etc/init.d/oracleasm scandisks
```

Check if the ASM volume is created

```
# /etc/init.d/oracleasm listdisks
VOL1
```

Ensure that permissions are set to oracle:dba on all nodes

```
# ls –l /dev/oracleasm/disks/*

brw-rw---- 1 oracle dba 8, 33 Jul  6 02:03 /dev/oracleasm/disks/VOL1
```

OCR and Voting Disk Configuration

Before installing clusterware, the OCR and VD mount points need a few configurations to be done.
As root, change the permissions of /u04 or whatever the mount point of the devices are:

```
# chown –R oracle.dba /u04
```

Create empty files for all files

```
$ cd /u04/OCR1
$ touch ocfs2_file_ocr1
$ cd /u04/OCR2
$ touch ocfs2_file_ocr2
.
.
$ cd /u04/VD31
$ touch ocfs2_file_vd3
```

The permissions for clusterware files will be set appropriately when root.sh is run at the end of clusterware installation.

Install 11gR1 Clusterware

Configuring password-less SSH

Before installing clustering software, password-less ssh needs to be setup on all nodes

Login to the first node as oracle user

```
$ cd /home/oracle/.ssh
$ /usr/bin/ssh-keygen –t rsa
$ /usr/bin/ssh-keygen –t dsa
```

Provide password(s) when prompted. Providing same passwords across all nodes is suggested for ease of administration

Repeat the above on all nodes

On the first node, execute the following

```
$ cat id_rsa.pub >> authorized_keys
$ cat id_dsa.pub >> authorized_keys
```

Repeat the above on all nodes

Finally copy over all nodes' authorized_keys file contents to the first node's authorized_keys

Now, the first node's authorized_keys contains keys from all nodes participating in the proposed cluster.

Copy over the first node's authorized keys to the respective folder of all other nodes.
For a two-node cluster, there should be 4 lines in each node's authorized_keys.

Before any cluster-related installation or upgrade or configuration, run the following so that password is not asked during ssh operations by the Oracle installer:

```
$exec /usr/bin/ssh-agent $SHELL
$ /usr/bin/ssh-add
```

The ssh password is asked only once and then subsequent ssh requests should not prompt for a password

```
$ ssh <node> date
[oracle@evg60lx-oracle-rac1-upg software]$ ssh 172.18.1.42 date
Tue Jul  6 02:22:09 CDT 2010
```

To be safe, use the fully qualified host name to ssh and ensure that they also work fine.

```
[oracle@evg60lx-oracle-rac1-upg software]$ ssh evg60lx-oracle-rac2-upg date
Tue Jul  6 02:23:01 CDT 2010
[oracle@evg60lx-oracle-rac1-upg software]$ ssh evg60lx-oracle-rac2-upg.internal.seatoncorp.com date
Tue Jul  6 02:23:26 CDT 2010
```

Install Clusterware

Create a Clusterware software directory on all nodes that is **not** under a database-software directory:

```
$ mkdir –p /u01/crs/oracle/product/11g/crs_1/
```

Navigate to the directory where Oracle installation files are present and unzip files, if not already done

```
$ unzip linux.x64_11gR1_clusterware.zip
$ cd clusterware
$ export DISPLAY=172.xxx.xxx.xxx:0.0
$. ./runInstaller
```

Provide appropriate inputs when prompted for.
For network interface, edit the 172.18.xx.xx and make it Public. The 10.xx.xx.xx will be left as Private.

At the end of installation, the installer prompts to run root.sh as root on each node.

```
$ su –
# /u01/crs/oracle/product/11g/crs_1/root.sh
```

On the first node, this formats the OCR and Voting devices, adds CRS daemons and starts the clusterware services/processes.

Login to the other node(s) and repeat (Do NOT run simultaneously!)

On the last node, root.sh configures and starts nodeapps (VIP, ONS and EVM). It also runs vipca in silent mode, the assistant that creates VIP resources.

Once the root.sh finishes, clusterware should be running on all nodes.
Come back to the installer and click OK and Exit from the main window.

Install ASM/Database software

Create appropriate directories on all nodes, separating ASM and Database software installations

```
$ mkdir –p /u01/app/oracle/product/11.1.0/asm_1
$ mkdir –p /u01/app/oracle/product/11.1.0/db_1
```

Unzip the Oracle software installation files

```
$ cd /home/oracle/software
$ unzip linux.x64_11gR1_database_1013.zip
```

Run the installer and select Enterprise edition and “Install software Only” options and appropriate software home directories.

```
$ cd database
$ ./runInstaller
```

Run the installer for each Oracle Home (ASM and Database) one after the other.

At the end of installation, run root.sh as root on each node, one after the other.

Patching Clusterware and Oracle Homes

Unpack the Patchset

```
$ cd /home/oracle/patches
$ unzip p6890831_111070_Linux-x86-64.zip
```

Run the installer

```
$ cd Disk1
$ ./runInstaller
```

Select appropriate Oracle homes and run through the installer steps and Click “Install”.

!!!Always patch the Clusterware first before patching ASM & Database!!!

Run root.sh after installation on each node one after the other when prompted.

Creating Listener and ASM Instances

Once installation and upgrade of 11gR1 is complete, create the listener and ASM instances on all nodes.

The important thing to note here is that the listener is recommended to be created and configured from ASM Home and not Database Home.

Listener

After ensuring SSH connectivity, run NETCA from ASM_HOME

```
$ cd /u01/app/oracle/product/11.1.0/asm_1/bin
$ ./netca &
```

Create the listener with name "LISTENER" after selecting all cluster nodes, going through the configuration assistant prompts. Select TCP/IP network Protocol and Listener Port #1521.

ASM

Before creating ASM instances, it is suggested to create a directory for placing the server parameter file under a shared drive.

```
$ mkdir –p /u02/oradata/asm_spfile
```

Run the Database Configuration Assistant from ASM_HOME

```
$ cd /u01/app/oracle/product/11.1.0/asm_1/bin
$ ./dbca &
```

Select Oracle Real Application Clusters database
Select Automatic Storage Management
Select all nodes
The Assistant will ask you if you want to create ASM instances. Choose Yes.
Provide the ASM administrator password, Select "Create SPFile" and enter the directory created under the shared drive.

It will take a while for the Assistant to bring up the ASM instances on all nodes.

The next step would be to configure ASM disk group(s).

Select "Create New"
Enter disk group name and choose Redundancy "External"

By default, the disk discovery path is "ORCL:*".
If, for some reason, a correctly configured disk is not visible and disk is configured through ASMLib, try to change the disk discovery path to "/dev/oracleasm/disks/*"

Select the disk click OK

The disk group will be created and mounted on all ASM instance-running nodes.

Creating/Cloning RAC Database(s)

Please follow the steps indicated in the “RAC-clone.doc” document to clone a database and then creating and configuring RAC database instances and services.

Oracle 11gR2 Clusterware Installation/Upgrade

When 11gR2 installer is run, the upgrade option is chosen automatically.

Before we can launch the installer a few checks/tasks need to be done.

1. Add asmadmin OS group

   ```
   # groupadd –g 12003 asmadmin
   # usermod –g oinstall –G asmadmin,dba oracle
   ```

   Execute the above on all nodes making sure that the same GID is assigned (12003 from the above example)

   ```
   #id oracle
   uid=11160(oracle) gid=12001(oinstall)
   groups=12002(dba),12003(asmadmin),12001(oinstall)
   ```

   Starting with 11g, the same user that owned the Oracle Clusterware software must perform the upgrade

2. Kernel Parameters

   The following kernel parameters need to added/modified:

   Edit /etc/sysctl.conf
   a. fs.file-max = 6815744
   b. net.ipv4.ip_local_port_range = 9000 65500
   c. fs.aio-max-nr = 1048576

3. RPM
   The package, libaio-devel needs to be installed (both i386 and x86_64 ARCH)

4. Change NTP configuration

   ```
   # vi /etc/sysconfig/ntpd
   ```
   Change the “OPTIONS” to include “-x”

Before:

```
OPTIONS=" -u ntp:ntp -p /var/run/ntpd.pid"
```

After:

```
OPTIONS="-x -u ntp:ntp -p /var/run/ntpd.pid"
```

Restart service ntpd

```
# /sbin/service ntpd restart
```

5. Ensure that at least 500MB is allocated to SWAP

6. Create an user with SYSASM privileges in the 11gR1 ASM instance and ensure that it is registered in the password file (orapw+<asm_instance>; orapw+ASM1 and orapw+ASM2)

```
$ export ORACLE_SID=”+ASM1”
$ export ORACLE_HOME=<ASM_HOME>
$ export PATH=$<ASM_HOME>/bin:$PATH
$ sqlplus / as sysdba

SQL> create user temp_sys identified by <pwd>;
SQL> grant sysasm, sysdba to temp_sys;
SQL> alter user temp_sys default role all;

SQL> connect temp_sys/<pwd>
SQL> grant sysasm to sys;
SQL> exit;
```

Repeat the above on all nodes.

7. Before running the installer, unset all ORA-related parameters except ORACLE_BASE

```
$ unset TNS_ADMIN ORACLE_HOME ORA_CRS_HOME ORACLE_SID
```

!!!This is very important as the relocation of the Listener during upgrade will fail, in turn failing to upgrade ASM!!!

8. Make sure that an entry exists in the /etc/oratab file for respective ASM instances pointing to the existing 11gR1 ASM Home and also ASM instances are up and running

```
$ cat /etc/oratab
# This file is used by ORACLE utilities.  It is created by root.sh
.
```

.

+ASM1:/u01/app/oracle/product/11.1.0/asm_1:N

9. Ensure that the SCAN (Single Client Access Name) is resolvable and exists in the DNS server

```
# nslookup <scan_name>

[root@evg60lx-oracle-rac1-upg ~]# nslookup ora-upg-scan
Server:         172.18.0.105
Address:        172.18.0.105#53

Name:   ora-upg-scan.internal.seatoncorp.com
Address: 172.18.1.45
Name:   ora-upg-scan.internal.seatoncorp.com
Address: 172.18.1.47
Name:   ora-upg-scan.internal.seatoncorp.com
Address: 172.18.1.46
```

Create a Clusterware software directory on all nodes that is **not** under a database-software directory:
$ mkdir –p /u01/grid_home/oracle/product/11gr2/grid_1

Navigate to the directory where Oracle installation files are present and unzip files, if not already done

```
$ unzip linux.x64_11gR2_grid.zip
$ cd grid
$ export DISPLAY=172.xxx.xxx.xxx:0.0
$. ./runInstaller
```

The installer automatically detects existing and running clusterware/ASM and defaults to “Upgrade Grid Infrastructure”.
Ignore the “Detected existing ASM instances” warning and continue.
Select language(s) (English is chosen by default)

On the “select nodes” page, click on SSH Connectivity and click “Setup”. This ensures that we don’t need to run the SSH password-less script upon each new session (
$ exec /usr/bin/ssh-agent $SHELL and $ /usr/bin/ssh-add need not be run after this step across reboots or sessions)

Specify OS privilege groups:
OSDBA = dba
OSOPER = oinstall
OSASM = asmadmin
Ignore warning and continue.

Specify ORACLE_BASE and GRID_HOME directories

Provide Cluster and SCAN names and click next.

Usually, in our systems, the Swap space is less and we will get a “Failed” status during the prerequisite checks. Select “Ignore All”

The summary screen should appear now.

Click “Install”

Once Installation files are copied over, linked, setup etc, the installer prompts you to run rootupgrade.sh from the new 11gR2 GRID_HOME.

Run on each node one after the other and return to the prompt window and click “OK” once rootupgrade.sh completes on the last node. This takes a while on slower systems.

The installer now runs the Configuration Assistants.

First, the Listener is migrated. Then ASM is migrated to the new GRID_HOME.

Other components are upgraded after this.

Oracle 11gR2 Database Installation/Upgrade

Installation

There is one prerequisite that must be resolved before starting 11gR2 upgrade.

In the databases that need to be upgraded, we need to set the initialization parameter, REMOTE_LISTENER to “SCAN:PORT”

```
$ export ORACLE_SID=webdev1
$ sqlplus / as sysdba
SQL> show parameter pfile

NAME                                 TYPE        VALUE
------------------------------------ ----------- ------------------------------
spfile                               string      +DATA_DG_1/webdev/spfilewebdev.ora

SQL> show parameter list

NAME                                 TYPE        VALUE
```

```
---------------------------------- ----------- -----------------------------
local_listener                   string    LISTENER_WEBDEV1
remote_listener                  string    LISTENER_WEBDEV2
```

```
SQL> create pfile='/tmp/temp_webdev1.ora' from spfile='
+DATA_DG_1/webdev/spfilewebdev.ora';
```

```
SQL> shutdown immediate;
```

Shutdown the Oracle database instances on the other nodes.

```
$ srvctl stop instance –d webdev –I webdev2
```

Edit /tmp/temp_webdev1.ora and include the following:

```
*.remote_listener = '<SCAN>:<Listener Port>'
```

Example:

```
*.remote_listener = 'ora-upg-scan:1521'
```

Remove individual references like:

```
Webdev1.remote_listener=' LISTENER_WEBDEV2'
Webdev2.remote_listener=' LISTENER_WEBDEV1'
```

```
SQL> startup mount pfile='/tmp/temp_webdev1.ora';
SQL> show parameter list

NAME                             TYPE       VALUE
---------------------------------- ----------- -----------------------------
local_listener                   string      LISTENER_WEBDEV1
remote_listener                    string     ora-upg-scan:1521
```

```
SQL> create spfile=' +DATA_DG_1/webdev/spfilewebdev.ora' from
pfile=/tmp/temp_webdev1.ora';
SQL> shutdown immediate;
```

```
SQL> startup;
```

Startup Oracle instances on all other nodes.

Create an installation folder for the new 11gR2 home

```
$ mkdir –p /u01/app/oracle/product/11.2.0/db_1
```

Navigate to the 11gR2 software-resident directory and extract the files

```
$ cd /home/oracle/software
$ unzip linux.x64_11gR2_database_1of2.zip
$ unzip linux.x64_11gR2_database_2of2.zip
```

Start the installer

```
$ export DISPLAY=172.xx.xx.xx:0.0
$ cd database
$ ./runInstaller
```

Enter Oracle support user-id/e-mail and optionally provide the password to receive updates from Oracle Support and Click Next.

Select “Install database software only”

Select “Real Application Clusters database installation” and ensure that all nodes are selected.

Select language(s): English is selected by default.

Select “Enterprise Edition”

Give appropriate directory names for installation

Select the following:
OSDBA = dba
OSOPER = oinstall

Review the prerequisite output and take necessary action.

Click “Finish” to start installation.

Run root.sh as prompted by the Installer at the end of installation.

Upgrade Database

!!!Take a full database backup before doing any upgrade!!!

Run the Pre-Upgrade Information Tool

```
$ cp /u01/app/oracle/product/11.2.0/db_1/rdbms/admin/utlu112i.sql /home/oracle/scripts
$ cd /home/oracle/scripts
$ sqlplus / as sysdba
SQL> spool upgrade_info_webdev.log
SQL> @utlu112i.sql
SQL> spool off
SQL> exit
```

Review the output file and make necessary changes, if needed.

Immediately before starting the upgrade, gather dictionary statistics and purge the recycle bin

```
SQL> exec dbms_stats.gather_dictionary_stats;
SQL> purge dba_recyclebin;
```

The Time Zone file can be updated using DBMS_DST package *after* the upgrade.

Complete Materialized Views Refreshes

```
SQL> select distinct(trunc(last_refresh)) from dba_snapshot_refresh_times;
```

Ensure no files need recovery

```
SQL> select * from v$recover_file;
```

Sync Standby/Reporting database, if any, with the Primary database by archiving current log and start recovery on the standby with the NODELAY option.

An entry for the database to be upgraded should exist in /etc/oratab file with the 11gR1 Oracle Home path

webdev:/u01/app/oracle/product/**11.1.0**/db_1:N

Making sure that *ORA* environment variables are unset except ORACLE_BASE and user equivalence (password-less SSH) is setup, run the Database Upgrade Tool from the **new** 11gR2 ORACLE_HOME.

```
$ cd /u01/app/oracle/product/11.2.0/db_1/bin
$ export DISPLAY=172.xx.xx.xx:0.0
$ ./dbua &
```

Select the database to be upgraded and provide details as requested. This includes setting degree of parallelism for invalid object recompilation, Flash Recovery Area and EM configuration. Make sure that archiving is *not* stopped during the upgrade process. Provide passwords for DBSNMP, SYSMAN and ASM user with SYSASM privileges.

Review the Database Upgrade Summary and start the upgrade process.

The DBUA will do all necessary tasks and the database will be upgraded to 11gR2.

Post-Installation Steps

Update OS Environment variables to point to the new Oracle 11gR2 Home.

Backup the database

Upgrade the Time Zone File Version

The steps that need to be done for upgrading the time zone file is highlighted in Oracle Support (formerly Metalink) Doc Id: 977512.1

However, the following steps have been executed and tested (picked from the same document) and can be used to upgrade the time zone file version.

The new DST version number that we should upgrade to is 11.

Check current RDBMS DST version and "DST UPGRADE STATUS"

```
conn / as sysdba
SELECT PROPERTY_NAME, SUBSTR(property_value, 1, 30) value
FROM DATABASE_PROPERTIES
WHERE PROPERTY_NAME LIKE 'DST_%'
ORDER BY PROPERTY_NAME;

-- check that the output gives

-- PROPERTY_NAME VALUE
-- ------------------------------ ------------------------------
-- DST_PRIMARY_TT_VERSION <the old DST version number>
-- DST_SECONDARY_TT_VERSION 0
-- DST_UPGRADE_STATE NONE

-- DST_PRIMARY_TT_VERSION should match the value found when selecting

SELECT version FROM v$timezone_file;
```

3b) Check UPFRONT if there is affected data that cannot be resolved automatically.

Note that this can be done on a working, live database. Of course it might that there is data added between this session and the actual upgrade of the RDBMS DST version that is affected. This is especially plausible if the update is done close to a DST change in your timezone and this timezone is affected by this RDBMS DST update.

```
conn / as sysdba

-- start prepare window
-- these steps will NOT update any data yet.

exec DBMS_DST.BEGIN_PREPARE(<the new DST version number>)
```

Sample error if the 11.2 DST patch for the requested DST version is not installed:

```
SQL> exec DBMS_DST.BEGIN_PREPARE(13)
BEGIN DBMS_DST.BEGIN_PREPARE(13); END;

*
ERROR at line 1:
ORA-30094: failed to find the time zone data file for version 13 in
$ORACLE_HOME/oracore/zoneinfo
ORA-06512: at "SYS.DBMS_DST", line 57
ORA-06512: at "SYS.DBMS_DST", line 1258
ORA-06512: at line 1
```

FIX: install the 11.2 patch for the DST version you want to use. See note 412160.1

Sample error if the requested new DST version is the current or a lower than the current timezone version:

```
SQL> exec DBMS_DST.BEGIN_PREPARE(4);
BEGIN DBMS_DST.BEGIN_PREPARE(4); END;

*
ERROR at line 1:
ORA-56921: invalid time zone version
ORA-06512: at "SYS.DBMS_SYS_ERROR", line 79
ORA-06512: at "SYS.DBMS_DST", line 1252
ORA-06512: at line 1
```

FIX: you cannot "downgrade" DST, there no need to do this. The new DST version needs to be higher than the current DST_PRIMARY_TT_VERSION

```
-- check for prepare status

SELECT PROPERTY_NAME, SUBSTR(property_value, 1, 30) value
FROM DATABASE_PROPERTIES
WHERE PROPERTY_NAME LIKE 'DST_%'
ORDER BY PROPERTY_NAME;

-- output should be
-- PROPERTY_NAME VALUE
-- ------------------------------ ------------------------------
-- DST_PRIMARY_TT_VERSION <the old DST version number>
-- DST_SECONDARY_TT_VERSION <the new DST version number>
-- DST_UPGRADE_STATE PREPARE

-- truncate logging tables if they exist.
```

```
TRUNCATE TABLE SYS.DST$TRIGGER_TABLE;
TRUNCATE TABLE sys.dst$affected_tables;
TRUNCATE TABLE sys.dst$error_table;

-- log affected data

BEGIN
DBMS_DST.FIND_AFFECTED_TABLES
(affected_tables => 'sys.dst$affected_tables',
log_errors => TRUE,
log_errors_table => 'sys.dst$error_table');
END;
/

-- check what tables have affected data that cannot be resolved automatically.
-- if this gives no rows then there is no problem at all

SELECT * FROM sys.dst$affected_tables;

-- IF previous select gives rows then you can see
-- what kind of problem there are in those rows

SELECT * FROM sys.dst$error_table;

-- error_on_overlap_time is error number ORA-1883
-- error_on_nonexisting_time is error number ORA-1878

-- for a explanation of the reported data please see
-- "Error Handling when Upgrading Time Zone File and Timestamp with Time Zone
Data"
-- For the "error_on_overlap_time" and "error_on_nonexisting_time" you do not HAVE
to
-- take action on this data to upgrade the DST version, but it is advised
-- to at least to check the results AFTER the update.

-- all "error_on_overlap_time" rows

SELECT * FROM sys.dst$error_table where ERROR_NUMBER= '1883';

-- all "error_on_nonexisting_time" rows

SELECT * FROM sys.dst$error_table where ERROR_NUMBER= '1878';

-- check for all other possible problems
```

```
SELECT * FROM sys.dst$error_table where ERROR_NUMBER not in ('1878','1883');
```

When updating from DSTv1 or DSTv2 (mainly after 9.2 upgrades) to a higher DST version it is possible to have also '1882' errors.
The cause is explained in Note 414590.1. These can be ignored, they will be corrected during the actual update of the dst version.

```
-- end prepare window, the rows above will stay in those tables.

EXEC DBMS_DST.END_PREPARE;

-- check if this is ended

SELECT PROPERTY_NAME, SUBSTR(property_value, 1, 30) value
FROM DATABASE_PROPERTIES
WHERE PROPERTY_NAME LIKE 'DST_%'
ORDER BY PROPERTY_NAME;

-- output should be
-- PROPERTY_NAME VALUE
-- --------------------------- -----------------------------
-- DST_PRIMARY_TT_VERSION <the old DST version number>
-- DST_SECONDARY_TT_VERSION 0
-- DST_UPGRADE_STATE NONE
```

4) Do the actual RDBMS DST version update of the database

Assuming all non-existing time and overlap times in previous step are solved or logged, so using for
DBMS_DST.UPGRADE_DATABASE error_on_overlap_time => FALSE and error_on_nonexisting_time => FALSE);

!!!
For RAC the database should be in single instance mode , as required by the "startup UPGRADE".

Before the database can be started in “UPGRADE” mode, all but one instance of the database should be shutdown.

On the lone RAC node, change parameter CLUSTER_DATABASE=false

```
SQL> alter system set cluster_database=false scope=spfile;
SQL> shutdown immediate;
```
!!!

```
startup upgrade;
set serveroutput on

-- check if previous prepare window is ended

SELECT PROPERTY_NAME, SUBSTR(property_value, 1, 30) value
FROM DATABASE_PROPERTIES
WHERE PROPERTY_NAME LIKE 'DST_%'
ORDER BY PROPERTY_NAME;

-- output should be
-- PROPERTY_NAME VALUE
-- --------------------------- ------------------------------
-- DST_PRIMARY_TT_VERSION <the old DST version number>
-- DST_SECONDARY_TT_VERSION 0
-- DST_UPGRADE_STATE NONE

-- If DST_UPGRADE_STATE is "PREPARE" then you did not ended
-- the prepare window in step 3)

-- If there are objects containing TSTZ data in recycle bin, please purge the bin now.
-- Otherwise dbms_dst.begin_upgrade will report "ORA-38301: Can not perform
DDL/DML over objects in Recycle Bin".

purge dba_recyclebin;

-- clean used tables

TRUNCATE TABLE SYS.DST$TRIGGER_TABLE;
TRUNCATE TABLE sys.dst$affected_tables;
TRUNCATE TABLE sys.dst$error_table;

-- start upgrade window

EXEC DBMS_DST.BEGIN_UPGRADE(<the new DST version number>);

-- the message
-- "An upgrade window has been successfully started."
-- will be seen
```

Sample error if a previous (prepare) window was not ended:

```
SQL> EXEC DBMS_DST.BEGIN_UPGRADE(11);
BEGIN DBMS_DST.BEGIN_UPGRADE(11); END;

*
```

```
ERROR at line 1:
ORA-56920: a prepare or upgrade window or an on-demand or datapump-job loading
of a secondary time zone data file is in an active state
ORA-06512: at "SYS.DBMS_SYS_ERROR", line 79
ORA-06512: at "SYS.DBMS_DST", line 1054
ORA-06512: at line 1
```

FIX: You NEED to end the "PREPARE" window in the previous step BEFORE doing the UPGRADE.
Or in other words, you did not do the "EXEC DBMS_DST.END_PREPARE;" step in point 3)

Sample error if the requested DST version / patch is not installed:

```
SQL> EXEC DBMS_DST.BEGIN_UPGRADE(13);
BEGIN DBMS_DST.BEGIN_UPGRADE(13); END;

*
ERROR at line 1:
ORA-30094: failed to find the time zone data file for version 13 in
$ORACLE_HOME/oracore/zoneinfo
ORA-06512: at "SYS.DBMS_DST", line 57
ORA-06512: at "SYS.DBMS_DST", line 1076
ORA-06512: at line 1
```

FIX: Install the 11.2 patch for the DST version you want to use. See note 412160.1

Sample error if the database is not in upgrade mode:

```
SQL> EXEC DBMS_DST.BEGIN_UPGRADE(11);
BEGIN DBMS_DST.BEGIN_UPGRADE(11); END;

*
ERROR at line 1:
ORA-56926: database must be in UPGRADE mode in order to start an upgrade windo
ORA-06512: at "SYS.DBMS_SYS_ERROR", line 79
ORA-06512: at "SYS.DBMS_DST", line 1091
ORA-06512: at line 1
```

FIX: start the database in UPGRADE mode

```
-- check if this select

SELECT PROPERTY_NAME, SUBSTR(property_value, 1, 30) value
FROM DATABASE_PROPERTIES
WHERE PROPERTY_NAME LIKE 'DST_%'
```

```
ORDER BY PROPERTY_NAME;

-- gives this output:
-- PROPERTY_NAME VALUE
-- --------------------------- ------------------------------
-- DST_PRIMARY_TT_VERSION <the new DST version number>
-- DST_SECONDARY_TT_VERSION <the old DST version number>
-- DST_UPGRADE_STATE UPGRADE

-- you can check what tables need to updated using

SELECT OWNER, TABLE_NAME, UPGRADE_IN_PROGRESS FROM
ALL_TSTZ_TABLES where UPGRADE_IN_PROGRESS='YES';
```

!!!

```
Change CLUSTER_DATABASE parameter

SQL> alter system set cluster_database=true scope=spfile;
```

!!!

```
-- restart the database

shutdown immediate

startup

-- now upgrade the tables who need action

set serveroutput on
VAR numfail number
BEGIN
DBMS_DST.UPGRADE_DATABASE(:numfail,
parallel => TRUE,
log_errors => TRUE,
log_errors_table => 'SYS.DST$ERROR_TABLE',
log_triggers_table => 'SYS.DST$TRIGGER_TABLE',
error_on_overlap_time => FALSE,
error_on_nonexisting_time => FALSE);
DBMS_OUTPUT.PUT_LINE('Failures:'|| :numfail);
END;
/

-- ouput of this will be a list of tables lie:
```

```
-- Table list: SYSMAN.AQ$_MGMT_NOTIFY_QTABLE_S
-- Number of failures: 0
-- ....
-- Table list: SYSMAN.MGMT_PROV_ASSIGNMENT
-- Number of failures: 0
-- Table list: SYSMAN.MGMT_CONFIG_ACTIVITIES
-- Number of failures: 0
-- Failures:0

-- if there where no failures then end the upgrade.

VAR fail number
BEGIN
DBMS_DST.END_UPGRADE(:fail);
DBMS_OUTPUT.PUT_LINE('Failures:'|| :fail);
END;
/

-- output that will be seen:
-- An upgrade window has been successfully ended.
-- Failures:0

-- last checks

SELECT PROPERTY_NAME, SUBSTR(property_value, 1, 30) value
FROM DATABASE_PROPERTIES
WHERE PROPERTY_NAME LIKE 'DST_%'
ORDER BY PROPERTY_NAME;

-- needed output:
-- PROPERTY_NAME VALUE
-- ---------------------------- ------------------------------
-- DST_PRIMARY_TT_VERSION <the new DST version number>
-- DST_SECONDARY_TT_VERSION 0
-- DST_UPGRADE_STATE NONE


SELECT * FROM v$timezone_file;

-- needed output:
-- FILENAME VERSION
-- ------------------ ----------
-- timezlrg_<new version>.dat <new version>
```

If needed, start over for the next database in the same ORACLE_HOME.

Upgrading Oracle Database with a Physical Standby Database in Place

1. Review and perform the steps listed in the "Preparing to Upgrade" chapter of the *Oracle Database Upgrade Guide*

2. Shut down the primary database.

3. Shut down the physical standby database(s).

4. Stop all listeners, agents and other processes running in the Oracle homes that are to be upgraded. Perform this step on all nodes in an Oracle Real Application Clusters (RAC) environment.

5. If Oracle Automatic Storage Management (Oracle ASM) is in use, shut down all databases that use Oracle ASM, and then shut down all Oracle ASM instance(s).

6. Install the new release of the Oracle software into a new Oracle home on the physical standby database and primary database systems, as described in the *Oracle Database Upgrade Guide*.

7. Restart all listeners, agents, and other processes stopped in step 4.

8. Mount the physical standby database(s).

9. Start Redo Apply on the physical standby database(s).

10. Upgrade the primary database as described in the *Oracle Database Upgrade Guide*. Note that the physical standby database(s) will be upgraded when the redo generated by the primary database as it is upgraded is applied.

11. Open the upgraded primary database.