Data Guard Standby Switchover SOP

The following outlines the steps followed in order to carry out a database switchover from the current primary to the current standby. While DGMGRL will carry out a switchover quite simply and easily, it is always best to check that the procedure should succeed and not fail due to some oversight.

For the purposes of this document, the primary database is AZDBA01 and the standby database is AZDBA91. Both are running on the same server, ORCDEVORC01, but this is not important and indeed, in normal circumstances, separate servers would be used.

# References

The following MOS (My Oracle Support) notes are valuable sources of information regarding this process:

* 1305019.1 - 11.2 Data Guard Physical Standby Switchover Best Practices using the Broker
* 1304939.1 - 11.2 Data Guard Physical Standby Switchover Best Practices using SQL\*Plus

Similar documents exists for Oracle 12c, should they be required.

* 1305019.1 - 11.2 Data Guard Physical Standby Switchover Best Practices using the Broker
* 1304939.1 - 11.2 Data Guard Physical Standby Switchover Best Practices using SQL\*Plus

# Pre-Switchover Checks

All of the following commands should return SUCCESS. If any do not, or if any Oracle errors are displayed, you cannot continue until such time as the problems have been resolved.

# Quick Version

The following instructions are extracted from Oracle docs, as listed, and are considered best practice.

## Verify Configuration

set oracle\_sid=azdba01

dgmgrl sys/password

show configuration verbose;

## Perform Switchover

switchover to <standby database name>;

## Post Switchover Tasks

See [these details](#_Post_Switchover_Checks) below.

# Explicit Version

The following instructions are extracted from Oracle docs, as listed, and are considered best practice.

## Verify Configuration

On the current primary database server, run the following commands. All of these should return SUCCESS. If any do not, or if any Oracle errors are displayed, you cannot continue until such time as the problems have been resolved.

set oracle\_sid=azdba01

dgmgrl sys/password

show configuration verbose;

show database azdba01;

show instance azdba01;

show database azdba91;

show instance azdba91;

You are expecting to see something similar to the following at the end of each of the above commands:

Configuration Status:

SUCCESS

## Test Connections & Flashback

The broker uses the StaticConnectIdentifier to reach the other database(s) in the configuration. You should check that they all work, from both servers.

show database azdba01 StaticConnectIdentifier

StaticConnectIdentifier = '(DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=ORCDEVORC

01)(PORT=1521))(CONNECT\_DATA=(SERVICE\_NAME=AZDBA01\_DGMGRL)(INSTANCE\_NAME=AZDBA01

)(SERVER=DEDICATED)))'

You should now check that the connection string, everything within - but excluding - the single quotes, can be contacted from SQL\*Plus on both servers:

sqlplus sys/password@"XXX" as sysdba

select flashback\_on, instance\_name, host\_name

from v$instance, v$database;

'XXX' is the full static connect identifier from the above query, wrapped in double quotes as opposed to single ones.

Ensure that the host\_name and instance\_name returned are correct for each test.

Ensure that the primary database has flashback on. If the primary shows up as having it turned off, enable it as follows:

alter database flashback on;

Ensure that the standby database has flashback on. If the standby shows up as having it turned off, enable it *after the switchover*:

## Check RMAN Archivelog Deletion Policy

Both databases have the same DBID, so if RMAN is in use for daily backups, then the archivelog deletion policy should be set to "APPLIED ON ALL STANDBY":

RMAN> show archivelog deletion policy;

RMAN configuration parameters for database with db\_unique\_name AZDBA01 are:

CONFIGURE ARCHIVELOG DELETION POLICY TO APPLIED ON ALL STANDBY;

and:

RMAN configuration parameters for database with db\_unique\_name AZDBA91 are:

CONFIGURE ARCHIVELOG DELETION POLICY TO APPLIED ON ALL STANDBY;

## Verify Tempfiles match

If any new Tempfiles have been added to the primary since the creation of the standby, or the most recent switchover, then they will not be present on the current standby. Run the following query on both databases.

First on the primary:

select tmp.name filename, bytes, ts.name tablespace

from v$tempfile tmp, v$tablespace ts

where tmp.ts# = ts.ts#;

FILENAME BYTES TABLESPACE

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

G:\MNT\ORADATA\AZDBA01\TEMP01.DBF 1368391680 TEMP

G:\MNT\ORADATA\AZDBA01\TEMP02.DBF 1368391680 TEMP

G:\MNT\ORADATA\AZDBA01\TEMP03.DBF 1368391680 TEMP

G:\MNT\ORADATA\AZDBA01\TEMP04.DBF 1369440256 TEMP

Then on the standby:

FILENAME BYTES TABLESPACE

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

G:\MNT\ORADATA\AZDBA91\TEMP01.DBF 1368391680 TEMP

G:\MNT\ORADATA\AZDBA91\TEMP02.DBF 1368391680 TEMP

G:\MNT\ORADATA\AZDBA91\TEMP03.DBF 1368391680 TEMP

G:\MNT\ORADATA\AZDBA91\TEMP04.DBF 1369440256 TEMP

There should be the same number of files, and they should match in size, on both databases. If any are missing or incorrectly sized, you can resolve this now or after opening the new primary.

## Verify Datafiles

Prior to switching over, check that all data files on the current standby database, are online:

select file# from v$datafile where status='OFFLINE';

If any are offline,

alter database datafile <file#> online;

## Check For Running Jobs

There should be no jobs running on the primary database as these can interfere with the switchover. To check, run the following commands on the primary database:

select owner, job\_name, session\_id, running\_instance, elapsed\_time

from dba\_scheduler\_running\_jobs;

no rows selected

select job, sid, instance, this\_date

from dba\_jobs\_running;

no rows selected

The expected result for both is "no rows selected". Any running jobs should be allowed to finish, or be aborted as necessary before switching over.

## Check for Running Transactions with RollBack

Any transaction with any existing UNDO will be rolled back as part of the switchover. Large transactions may take a long time to rollback. Check for these as follows:

set lines 3000 trimspool on pages 200

col username format a15

col machine format a20

col tablespace\_name format a15

SELECT s.username, r.tablespace\_name, t.used\_ublk, t.start\_time "START\_TIME mm/dd/yyyy"

FROM sys.v\_$transaction t, dba\_rollback\_segs r, v$session s

WHERE (t.xidusn = r.segment\_id)

and S.TADDR = t.addr

ORDER BY t.start\_time;

The output will resemble the following (slightly contrived) example:

USERNAME TABLESPACE\_NAME USED\_UBLK START\_TIME mm/dd/yy

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

FRED UNDOTBS1 50 06/23/16 08:30:20

BARNEY UNDOTBS1 1 06/23/16 11:50:18

## Perform Switchover

The databases are now ready to switchover. Depending on the number of uncommitted transactions, and the size of these, there may well be quite a delay in the switchover process.

## Check Switchover Status

On both databases, make sure that the database will permit a switchover:

select switchover\_status from v$database;

* Not Allowed - There are no standby databases, or, this is the standby and the primary has not been switched yet.
* Session Active - There are active SQL sessions connected to the database. These need to be disconnected first, although they will be disconnected by the switchover.
* Switchover Pending - This is the standby database. The request to switchover has been received and is in progress, but not yet completed.
* Switchover Latent - The switchover *was* pending, but did not complete.
* To Primary - This is a standby database, with no active sessions, that is allowed to switch over to a primary database.
* To Standby - This is a primary database, with no active sessions, that is allowed to switch over to a standby database.
* Recovery Needed - This is a standby database that has not received the switchover request.

## Switch Over

In dgmgrl, on either server, run the following command:

connect sys/password

switchover to <standby database name>;

You *must* connect with the SYS username and password to actually carry out a switchover.

After the switchover completes, *and it may take some time*, check the configuration to ensure that the two databases have swapped roles.

If the standby doesn't come up correctly for any particular reason, simply login as SYSDBA and startup mount it in the normal manner. It will then come up and start processing redo in the normal manner.

Check the drc<database\_name>.log& the database alert.log file for the failure details.

# Post Switchover Checks

After a successful switchover, some additional checks are required to be carried out.

## Verify Configuration

In dgmgrl run the same commands as you did in the pre-switchover checks.

show configuration verbose

show database <primary database>

show database <standby database>

show instance <primary instance>

show instance <standby instance>

They should all show a similar result to the following:

Configuration Status:

SUCCESS

## Check Apply Gaps

Dgmgrl's show database <standby database name> command will quickly indicate if there's a gap or not. You should see 'NO GAP' reported.

Alternatively, run the following on the new primary database in SQL\*Plus:

set pages 300 lines 300 trimspool on

col destination format a30

col error format a30

col db\_unique\_name format a10

select destination, archived\_seq#, applied\_seq#, error, db\_unique\_name, gap\_status

from v$archive\_dest\_status

where status <> 'INACTIVE'

and dest\_name = 'LOG\_ARCHIVE\_DEST\_2';

## Confirm Flashback

Both databases should be running with flashback on. As per the preliminary checks above, the now current standby should be in this mode as the old primary was checked and enabled before the switch over. However, the old standby may not have been set and so the new primary now needs to be confirmed:

select flashback\_on from v$database;

If this returns "no", then enable it as follows:

alter database flashback on;

## Amend any Backup Scripts

If any scripts are configured to run backups against the old primary, these will now require amending to run against the new primary database instead.

# Important – RMAN Backup Tasks

When patching is in progress, or has completed, the database server, whichever one it happens to be, running as the primary server, must have the Windows Task Scheduler's RMAN Backup Tasks enabled. These will run at pre-determined times of the day to carry out RMAN backups of the various databases.

The servers running as standby or DR during and after patching, must have their task scheduler backup tasks disabled. We only run the backups on the primary servers in production and pre-production.