9.2.2 Performing a Failover to a Physical Standby Database

This section describes how to perform a failover to a physical standby database.

**Step 1   Flush any unsent redo from the primary database to the target standby database.**

If the primary database can be mounted, it may be possible to flush any unsent archived and current redo from the primary database to the standby database. If this operation is successful, a zero data loss failover is possible even if the primary database is not in a zero data loss data protection mode.

Ensure that Redo Apply is active at the target standby database.

Mount, but do not open the primary database. If the primary database cannot be mounted, go to [Step 2](https://docs.oracle.com/database/121/SBYDB/role_management.htm#DAFGBIEE).

If not already done, then set up the remote LOG\_ARCHIVE\_DEST\_*n* configured at the primary to point to the target destination. (You may not have any remote LOG\_ARCHIVE\_DEST\_*n* configured if the target destination was serviced by a far sync instance, or was a terminal standby in a cascaded configuration.) Also, ensure that the primary will be able to connect to the target destination by verifying that the NET\_ALIAS\_TARGET\_DB\_NAME is valid and properly established.

SQL> ALTER SYSTEM SET LOG\_ARCHIVE\_DEST\_6='SERVICE=NET\_ALIAS\_TARGET\_DB\_NAME -

> ASYNC VALID\_FOR=(online\_logfile, primary\_role) -

> DB\_UNIQUE\_NAME="target\_db\_unique\_name"' SCOPE=memory;

SQL> ALTER SYSTEM SET LOG\_ARCHIVE\_DEST\_STATE\_6=ENABLE;

It is also assumed that the LOG\_ARCHIVE\_CONFIG specification includes the DB\_UNIQUE\_NAME of the target destination at the primary (and LOG\_ARCHIVE\_CONFIG at the target destination includes the DB\_UNIQUE\_NAME of the primary). If not, then add that information to the LOG\_ARCHIVE\_CONFIG at the primary and target destination as required.

Issue the following SQL statement at the primary database:

SQL> ALTER SYSTEM FLUSH REDO TO *target\_db\_name*;

For target\_db\_name, specify the DB\_UNIQUE\_NAME of the standby database that is to receive the redo flushed from the primary database.

This statement flushes any unsent redo from the primary database to the standby database, and waits for that redo to be applied to the standby database.

If this statement completes without any errors, go to [Step 5](https://docs.oracle.com/database/121/SBYDB/role_management.htm#DAFIBCGJ). If the statement completes with any error, or if it must be stopped because you cannot wait any longer for the statement to complete, continue with [Step 2](https://docs.oracle.com/database/121/SBYDB/role_management.htm#DAFGBIEE).

**Step 2   Verify that the standby database has the most recently archived redo log file for each primary database redo thread.**

Query the V$ARCHIVED\_LOG view on the target standby database to obtain the highest log sequence number for each redo thread.

For example:

SQL> SELECT UNIQUE THREAD# AS THREAD, MAX(SEQUENCE#) -

> OVER (PARTITION BY thread#) AS LAST from V$ARCHIVED\_LOG;

THREAD LAST

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

1 100

If possible, copy the most recently archived redo log file for each primary database redo thread to the standby database if it does not exist there, and register it. This must be done for each redo thread.

For example:

SQL> ALTER DATABASE REGISTER PHYSICAL LOGFILE 'filespec1';

**Step 3   Identify and resolve any archived redo log gaps.**

Query the V$ARCHIVE\_GAP view on the target standby database to determine if there are any redo gaps on the target standby database.

For example:

SQL> SELECT THREAD#, LOW\_SEQUENCE#, HIGH\_SEQUENCE# FROM V$ARCHIVE\_GAP;

THREAD# LOW\_SEQUENCE# HIGH\_SEQUENCE#

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

1 90 92

In this example, the gap comprises archived redo log files with sequence numbers 90, 91, and 92 for thread 1.

If possible, copy any missing archived redo log files to the target standby database from the primary database and register them at the target standby database. This must be done for each redo thread.

For example:

SQL> ALTER DATABASE REGISTER PHYSICAL LOGFILE 'filespec1';

**Step 4   Repeat** [**Step 3**](https://docs.oracle.com/database/121/SBYDB/role_management.htm#DAFFEBFJ) **until all gaps are resolved.**

The query executed in [Step 3](https://docs.oracle.com/database/121/SBYDB/role_management.htm#DAFFEBFJ) displays information for the highest gap only. After resolving a gap, you must repeat the query until no more rows are returned.

If, after performing [Step 2](https://docs.oracle.com/database/121/SBYDB/role_management.htm#DAFGBIEE) through [Step 4](https://docs.oracle.com/database/121/SBYDB/role_management.htm#DAFDAIDC), you are not able to resolve all gaps in the archived redo log files (for example, because you do not have access to the system that hosted the failed primary database), some data loss will occur during the failover.

**Step 5   Stop Redo Apply.**

Issue the following SQL statement on the target standby database:

SQL> ALTER DATABASE RECOVER MANAGED STANDBY DATABASE CANCEL;

**Step 6   Switch the physical standby database to the primary role.**

Issue the following SQL statement on the target standby database:

SQL> ALTER DATABASE FAILOVER TO *target\_db\_name*;

For example, suppose the target standby database is named CHICAGO:

SQL> ALTER DATABASE FAILOVER TO CHICAGO;

If this statement completes without any errors, proceed to [Step 10](https://docs.oracle.com/database/121/SBYDB/role_management.htm#DAFBIAAF).

If there are errors, go to [Step 7](https://docs.oracle.com/database/121/SBYDB/role_management.htm#DAFDCFBJ).

**Step 7   If an error occurs, try to resolve the cause of the error and then reissue the statement.**

* If successful, go to [Step 10](https://docs.oracle.com/database/121/SBYDB/role_management.htm#DAFBIAAF).
* If the error still occurs and it involves a far sync instance, go to [Step 8](https://docs.oracle.com/database/121/SBYDB/role_management.htm#DAFBDFJB).
* If the error still occurs and there is no far sync instance involved, go to [Step 9](https://docs.oracle.com/database/121/SBYDB/role_management.htm#DAFBAFFA).

**Step 8   For far sync instance error cases only, use the FORCE option.**

If the error involves a far sync instance (for example, it is unavailable) and you have tried resolving the issue and reissuing the statement without success, then you can use the FORCE option. For example:

SQL> ALTER DATABASE FAILVOVER TO CHICAGO FORCE;

The FORCE option instructs the failover to ignore any failures encountered when interacting with the far sync instance and proceed with the failover, if at all possible. (The FORCE option has meaning only when the failover target is serviced by a far sync instance.)

If the FORCE option is successful, go to [Step 10](https://docs.oracle.com/database/121/SBYDB/role_management.htm#DAFBIAAF).

If the FORCE option is unsuccessful, go to [Step 9](https://docs.oracle.com/database/121/SBYDB/role_management.htm#DAFBAFFA).

**Step 9   Perform a data-loss failover.**

If an error condition cannot be resolved, a failover can still be performed (with some data loss) by issuing the following SQL statement on the target standby database:

SQL> ALTER DATABASE ACTIVATE PHYSICAL STANDBY DATABASE;

In the following example, the failover operation fails with an ORA-16472 error. That error means the database is configured in MaxAvailability or MaxProtection mode but data loss is detected during failover.

SQL> ALTER DATABASE FAILOVER TO CHICAGO;

ERROR at line 1:

ORA-16472: failover failed due to data loss

You can complete the data loss failover by issuing the following SQL statement:

SQL> ALTER DATABASE ACTIVATE PHYSICAL STANDBY DATABASE;

Database altered.

**Step 10   Open the new primary database.**

SQL> ALTER DATABASE OPEN;

**Step 11   Back up the new primary database.**

Oracle recommends that you perform a full backup of the new primary database.

**Step 12   Restart Redo Apply, if necessary.**

If Redo Apply has stopped at any of the other physical standby databases in your Data Guard configuration, then restart it. For example:

SQL> ALTER DATABASE RECOVER MANAGED STANDBY DATABASE DISCONNECT;

**Step 13    Optionally, restore the failed primary database.**

After a failover, the original primary database can be converted into a physical standby database of the new primary database using the method described in [Section 15.2](https://docs.oracle.com/database/121/SBYDB/scenarios.htm#i1049997) or [Section 15.7](https://docs.oracle.com/database/121/SBYDB/scenarios.htm#BACJCDGH), or it can be re-created as a physical standby database from a backup of the new primary database using the method described in [Section 3.2](https://docs.oracle.com/database/121/SBYDB/create_ps.htm#i67520).

Once the original primary database is running in the standby role, a switchover can be performed to restore it to the primary role.