**Standby database** is a backup database that mirrors the primary database. It is part of Oracle's **Data Guard** technology, which ensures high availability, disaster recovery, and **ZERO** data loss. Standby databases are crucial for maintaining business continuity and minimizing downtime during unexpected failures. Standby database is in either **MOUNT** or **READ ONLY**.

**There are two main types of standby databases:**

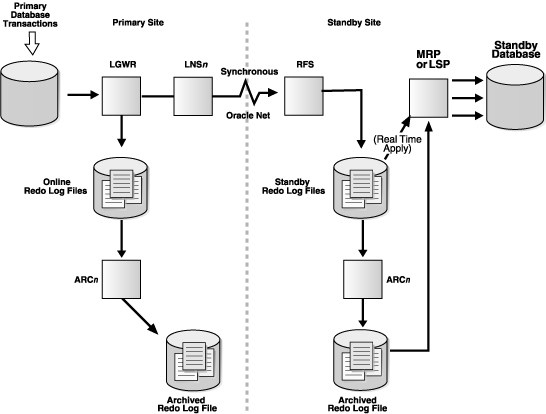
**Physical Standby**: This is an exact replica of the primary database, maintained through redo log application.

**Logical Standby**: This contains the same data as the primary database but allows for additional querying and reporting.

* From 11g 30 standby database can be created.
* For standby creation primary database should be enable with **ARCHIVELOG** mode.
* Password file of primary is required.
* **PRIMARY** and **STANDBY** version should be same.
* **FORCE LOGGING** must be enables at database level.

**SQL>** **alter database force logging;**

**SQL>** **ALTER DATABASE NO FORCE LOGGING;**



**LNS**: The **Log Network Service** process reads redo data from the redo log buffer or online redo logs on the primary database and is responsible for transmitting redo data from the primary database to the standby database

**RFS**: The **Remote File Server** process receives redo data sent from the primary database and writes it to the standby redo logs or archived redo logs on the standby database.

**MRP**: The **Managed Recovery Process** is responsible for applying redo data to the standby database to keep it synchronized with the primary database.

**Active duplicate standby create without backup:**

* Same db name in primary and standby.
* We can have different SID.

**Primary:**

1. Add Standby TNS\_names
2. Create password and copy to standby
3. Conn to rman

* rman target / auxiliary=sys/welcome@tns\_stby
* rman> duplicate target database for standby from active database nofilenamecheck;

**Standby**:

1. Create required directories.
2. Copy pfile from primary and edit accordingly
3. Copy password file from primary
4. Create listener
5. If **PRIMARY** and **STANDBY** directory structure is differnet add this parameters.

* db\_file\_name\_convert='/data/oracle /jms/','/SSD/STBY/ORADATA/' '/data/oracle/JMSUATJP/','/SSD/STBY/ORADATA/'
* log\_file\_name\_convert='/data/oracle/app/oracle/oradata/jms/','/SSD/STBY/ORADATA/'

1. Startup nomount the database.

**Enable Auto ARCHIVE shiping from primary to standby:**

1. **ARCHIVE** shipping
2. **LOG** shipping (standby redo logfiles)

**ARCHIVE Shipping:**

**Primary:**

**SQL> alter system set Log\_archive\_dest\_1=‘LOCATION=/prod/ram/admin/arch’;**

**SQL>Log\_archive\_dest\_2=‘SERVICE=JMSSTBY ARCH’;**

**Standby:**

**SQL> ALTER SYSTEM SET STANDBY\_FILE\_MANAGEMENT=AUTO;**

**MRP Enable / Disable:**

* alter database recover managed standby database cancel;
* alter database recover managed standby database disconnect;
* alter database recover managed standby database using current logfile disconnect;

**To check MRP status:**

* select process,status from v$managed\_standby;

**To check archives applied or not:**

* Select sequence# ,applied from v$archived\_log;
* Select max(sequence#) from v$archivde\_log where applied=’YES’;

**Manual standby creation using Rman hot backup:**

**Primary:**

1. Take Rman hot backup

* backup database;
* Backup archivelog all;
* Backup current controlfile for standby;

1. Transfer backups to standby.

**Standby:**

1. Create required directories.
2. Copy pfile and edit
3. Put db in nomount

**Rman> restore standby controlfile from ‘backup /location and/file name’;**

**Rman> alter database mount;**

1. register backups in standby controlfile

**RMAN>catalog start with ‘/backup/location/‘;**

**Restore database;**

**Recover database;**

**alter database open;**

1. **If directory structure is different prepare script**

* run

{

Set newname for datafile 1 to ‘/prod/hydtst/oradata/system01.dbf’;

Set newname for datafile 2 to ‘/prod/hydtst/oradata/sysaux01.dbf’;

Set newname for datafile 3 to ‘/prod/hydtst/oradata/undotbs01.dbf’;

Set newname for datafile 4 to ‘/prod/hydtst/oradata/users01.dbf’;

restore database;

}

**Rman>switch database to copy;**

**Rman> recover database;**

1. **Rename LOG files:**

* alter database rename file ‘/prod/oradata/redo01.log’ to ‘/prod/oradata/redo01.log’;
* alter database rename file ‘/prod/oradata/redo02.log’ to ‘/prod/oradata/redo02.log’;
* alter database rename file ‘/prod/oradata/redo03.log’ to ‘/prod/oradata/redo03.log’;

1. **alter database open;**
2. start MRP.

**Snapshot Standby:**

* This can be performed only on standby database.
* FRA should be enabled.
* Flashback should be enabled.
* Cancel MRP process.

**Convert Physical Standby to Snapshot standby**

* Alter database convert to snapshot standby;
* Bounce the database.

**Snapshot standby To Physical standby**

* Shutdown snapshot standby.
* Startup in mount.
* Alter database convert to physical standby;
* Start MRP process.

**Switch Over**

* Before doing switchover activity both the primary and standby database should be in sync.

**Primary**

* select name,open\_mode,database\_role,switchover\_status from v$database;
* Disconnect the active sessions of user.

**SQL>Alter database commit to switchover to standby;**

* Bounce the db and Check the db status.

**Standby**

* select name,open\_mode,database\_role,switchover\_status from v$database;

**SQL>Alter database commit to switchover to primary;**

* Bounce the db and Check the db status.
* enable the sync between primary and standby

**Failover:**

* In case of worst situation primary database will not work or not available for production than we can activated standby database as a primary production database.
* If once the MPR is finished in standby again it will not be converted as standby.

1. Check the database role and open\_mode.

* Select name,open\_mode,database\_role from v$database;

1. Cancel the MRP process.

**SQL>alter database recover managed standby database cancel;**

1. The below commands will help to bring up standby as primary.

**SQL>alter database recover managed standby database finish;**

**SQL> alter database activate standby database;**

1. Bounce your database and verify database name its open mode and its role.

SQL>shutdown immediate;

SQL>Startup

SQL>select name,open\_mode,database\_role from v$database;