**Technical Design Document**

***RRDB Setup - Active Dataguard Configuration***

**Table of Contents**

[RRDB Setup Active Dataguard Configuration 3](#_Toc450723667)

[Brief Description 3](#_Toc450723668)

[Pre-Requisites 3](#_Toc450723669)

[Setting up Oracle Active Dataguard for RRDB database 3](#_Toc450723670)

[Primary Database setup for Active Dataguard 3](#_Toc450723671)

[Standby Database setup for Active Dataguard 5](#_Toc450723672)

[Wallet setup for the newly created standby database to enable recovery of the standby database 8](#_Toc450723673)

[Finishing the standby database setup 8](#_Toc450723674)

[Synchronization between primary and standby database 9](#_Toc450723675)

[Referenced By 10](#_Toc450723676)

[Notes/Comments 10](#_Toc450723677)

[Revision History: 10](#_Toc450723678)

[Appendix A 10](#_Toc450723679)

# RRDB Setup Active Dataguard Configuration

## Brief Description

This document details the following

* Setting up Oracle Active Dataguard for RRDB (Reporting) database
* Primary Database setup for Active Dataguard
* Standby Database setup for Active Dataguard
* Wallet setup for the newly created standby database to enable recovery of the standby database
* Finishing the standby database setup
* Synchronization between primary and standby database

## Pre-Requisites

* Already installed Oracle 12c database available with parameters for ORACLE\_HOME being set.
* Network communication available between the primary and standby site.
* Primary database is already in archivelog mode
* In case of RRDB database already existing in Oracle 11g home then it has to be de-installed using DBCA of 11g and the service should be removed using the below command:

srvctl remove database -d $ORACLE\_SID

# Setting up Oracle Active Dataguard for RRDB database

The reporting database is meant to be made available for querying purposes and also for real time synchronization. The real time synchronization is required for the standby database as the results are queried from the standby database and using ODI and ETL process is initiated to perform loads in the datawarehouse database. Considering the above requirement it was decided to go ahead with the Oracle Active Dataguard configuration which is easier to implement and serves the purpose for the client.

Also, the RRDB database is built as a standby for UAT databases for each of the 3 states and hence all references in the document w.r.t primary will be PRIMDB and standby will be STDBRDB.

# Primary Database setup for Active Dataguard

*Make sure to turn on the primary database to FORCE LOGGING mode by running the below queries:*

SQL> SELECT force\_logging FROM v$database;

FORCE\_LOGGING

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

NO

SQL> ALTER DATABASE FORCE LOGGING;

Database altered.

SQL> SELECT force\_logging FROM v$database;

FORCE\_LOGGING

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

YES

*Add standby redo log files as below:*

ALTER DATABASE ADD STANDBY LOGFILE

'+DG\_DATA/PRIMDB/srl01.log'

SIZE 52428800

/

ALTER DATABASE ADD STANDBY LOGFILE

'+DG\_DATA/PRIMDB/srl02.log'

SIZE 52428800

/

ALTER DATABASE ADD STANDBY LOGFILE

'+DG\_DATA/PRIMDB/srl03.log'

SIZE 52428800

/

ALTER DATABASE ADD STANDBY LOGFILE

'+DG\_DATA/PRIMDB/srl04.log'

SIZE 52428800

/

*Put TNS entry for primary database as well as standby database as below:*

PRIMDB =

(DESCRIPTION =

(ADDRESS = (PROTOCOL = TCP)(HOST = 10.72.65.XXX)(PORT = 1521))

(CONNECT\_DATA =

(SERVER = DEDICATED)

(SERVICE\_NAME = PRIMDB)

)

)

STDBRDB =

(DESCRIPTION =

(ADDRESS = (PROTOCOL = TCP)(HOST = 10.72.65.XX)(PORT = 1522))

(CONNECT\_DATA =

(SERVER = DEDICATED)

(SERVICE\_NAME = STDBRDB)

)

)

*Following initialization parameters are to be set in the primary database*

ALTER SYSTEM SET log\_archive\_config='dg\_config=(PRIMDB,STDBRDB)'

ALTER SYSTEM SET log\_archive\_dest\_2 =

'service=STDBRDB lgwr async valid\_for=(online\_logfile,primary\_role) db\_unique\_name=STDBRDB';

ALTER SYSTEM SET fal\_server=STDBRDB;

ALTER SYSTEM SET fal\_client=PRIMDB;

ALTER SYSTEM SET standby\_file\_management='AUTO';

# Standby Database setup for Active Dataguard

*Create Service:*

srvctl add database -db sid -oraclehome /opt/app/oracle/product/12.1.0.2/db\_1

*Creating pfile and spfile:*

Create pfile with the name **initSTDBRDB.ora\_bak** as well as setting the below mentioned initialization parameters and keep the pfile in $ORACLE\_HOME/dbs

STDBRDB.\_\_data\_transfer\_cache\_size=0

STDBRDB.\_\_db\_cache\_size=1040187392

STDBRDB.\_\_java\_pool\_size=16777216

STDBRDB.\_\_large\_pool\_size=50331648

STDBRDB.\_\_oracle\_base='/opt/app/oracle'#ORACLE\_BASE set from environment

STDBRDB.\_\_pga\_aggregate\_target=117440512

STDBRDB.\_\_sga\_target=3221225472

STDBRDB.\_\_shared\_io\_pool\_size=150994944

STDBRDB.\_\_shared\_pool\_size=1912602624

STDBRDB.\_\_streams\_pool\_size=33554432

\*.audit\_file\_dest='/opt/app/oracle/admin/STDBRDB/adump'

\*.audit\_trail='db'

\*.compatible='12.1.0.2.0'

#\*.control\_files='+DG\_DATA/STDBRDB/CONTROLFILE/current.328.894113801'#Restore Controlfile

\*.control\_files='+DG\_DATA'

\*.db\_block\_size=8192

\*.db\_create\_file\_dest='+DG\_DATA'

\*.db\_domain=''

\*.db\_name='PRIMDB'#Reset to original value by RMAN

\*.db\_unique\_name='STDBRDB'

\*.db\_recovery\_file\_dest='+DG\_DATA'

\*.db\_recovery\_file\_dest\_size=4322230272

\*.diagnostic\_dest='/opt/app/oracle'

\*.dispatchers='(PROTOCOL=TCP) (SERVICE=STDBRDBXDB)'

\*.log\_archive\_dest\_1='LOCATION=/arch/STDBRDB'

\*.log\_archive\_format='%t\_%s\_%r.dbf'

\*.memory\_target=0

\*.open\_cursors=300

\*.pga\_aggregate\_target=104857600

\*.processes=300

\*.remote\_login\_passwordfile='EXCLUSIVE'

\*.sga\_max\_size=3221225472

\*.sga\_target=3221225472

\*.undo\_tablespace='UNDOTBS1'

\*.log\_archive\_max\_processes=5

\*.log\_file\_name\_convert='+DG\_DATA/PRIMDB/ONLINELOG','+DG\_DATA/STDBRDB/ONLINELOG'

\*.db\_file\_name\_convert='+DG\_DATA/PRIMDB/DATAFILE','+DG\_DATA/STDBRDB/DATAFILE','+DG\_DATA/PRIMDB/TEMPFILE','+DG\_DATA/STDBRDB/TEMPFILE'

event="10298 trace name context forever, level 32"

Start the database in nomount mode by using the above pfile:

export ORACLE\_SID=STDBRDB

export ORACLE\_HOME=/opt/app/oracle/product/12.1.0.2/db\_1

sqlplus / as sysdba

SQL> startup nomount pfile=initSTDBRDB.ora\_bak

SQL> create spfile='+DG\_DATA/STDBRDB/spfileSTDBRDB.ora' from pfile='/opt/app/oracle/product/12.1.0.2/db\_1/dbs/initSTDBRDB.ora\_bak';

SQL> shutdown immediate;

Create pfile with the name **initSTDBRDB.ora** and reference spfile location in it as shown below:

**initSTDBRDB.ora**

SPFILE='+DG\_DATA/STDBRDB/spfileSTDBRDB.ora'

Shutdown the standby database and start it up using spfile as below:

export ORACLE\_SID=STDBRDB

export ORACLE\_HOME=/opt/app/oracle/product/12.1.0.2/db\_1

sqlplus / as sysdba

SQL> startup nomount

*Create password file*

Create password file for standby database

cd $ORACLE\_HOME/dbs

orapwd file=orapwSTDBRDB password=\*\*\*\*\*\*\* entries=10

*Following initialization parameters are to be set in the standby database*

ALTER SYSTEM SET log\_archive\_config='dg\_config=(PRIMDB,STDBRDB)'

ALTER SYSTEM SET fal\_client=STDBRDB;

ALTER SYSTEM SET fal\_server=PRIMDB;

*Put TNS entry for primary database as well as standby database as below:*

PRIMDB =

(DESCRIPTION =

(ADDRESS = (PROTOCOL = TCP)(HOST = 10.72.65.XXX)(PORT = 1521))

(CONNECT\_DATA =

(SERVER = DEDICATED)

(SERVICE\_NAME = PRIMDB)

)

)

STDBRDB =

(DESCRIPTION =

(ADDRESS = (PROTOCOL = TCP)(HOST = 10.72.65.XX)(PORT = 1522))

(CONNECT\_DATA =

(SERVER = DEDICATED)

(SERVICE\_NAME = STDBRDB)

)

)

*Listener setup*

We need to create a separate listener for the standby database as below. The entries are to be inserted in **listener.ora**.

stdbrdb=

(DESCRIPTION\_LIST =

(DESCRIPTION =

(ADDRESS\_LIST =

(ADDRESS = (PROTOCOL = TCP)(HOST = 10.72.65.xx)(PORT = 1522))

)

)

)

SID\_LIST\_ stdbrdb=

(SID\_LIST =

(SID\_DESC =

(GLOBAL\_DBNAME = STDBRDB)

(ORACLE\_HOME =/opt/app/oracle/product/12.1.0.2/db\_1)

(SID\_NAME = STDBRDB)

)

)

# Wallet setup for the newly created standby database to enable recovery of the standby database

As primary database is encryption enabled to store SCUBI tables and indexes, we have to copy the wallet from the primary database and place it in standby database as per the paths mentioned in the sqlnet.ora file as below:

WALLET\_LOCATION=

(SOURCE=

(METHOD=FILE)

(METHOD\_DATA=

(DIRECTORY=/opt/app/oracle/admin/${ORACLE\_SID}/wallet)))

SSL\_CLIENT\_AUTHENTICATION=FALSE

SSL\_CIPHER\_SUITES= (SSL\_RSA\_WITH\_AES\_128\_CBC\_SHA, SSL\_DH\_anon\_WITH\_3DES\_EDE\_CBC\_SHA)

We need to zip the wallet file, transfer to standby database server, make similar folders in WALLET directory path in standby database under /opt/app/oracle/admin/${ORACLE\_SID}/wallet and unzip the files in wallet directory.

# Finishing the standby database setup

We need to now complete the standby database setup by restoring the database files from primary database to standby database using rman. The steps are as below:

*Connect to RMAN from primary database*

rman target sys/\*\*\*\*\*\*\*xx@PRIMDB auxiliary sys/\*\*\*\*\*\*\*@STDBRDB

*Run the below RMAN command from the rman prompt*

run {

allocate channel prmy1 type disk;

allocate channel prmy2 type disk;

allocate channel prmy3 type disk;

allocate channel prmy4 type disk;

allocate auxiliary channel stby type disk;

duplicate target database for standby from active database;

}

Once the rman command is completed, we need to go to the standby database and run the following commands in standby database to enable Active Dataguard.

$sqlplus / as sysdba

SQL> alter database recover managed standby database disconnect from session;

SQL> alter database open read only;

# Synchronization between primary and standby database

The following checks we are required to ensure the primary database is in sync with standby database.

*Primary*

* Connect as sys user and run the **archive log list** command, the current archive log sequence is known
* Run the below SQL to check the status of the dataguard

SQL> select \* from v$dataguard\_status;

* Check alert log to see if there are no errors shown for log\_archive\_dest\_2 if it is shown then check the trace file to see what the errors are.

*Standby*

* Run the below SQL to check the status of the dataguard.

SQL> select \* from v$dataguard\_status;

* Check alert log to see if there are no errors shown for fal processes.
* Query on v$archived\_log as below and check if all the transferred logs are applied.

SQL> select sequence#,applied from v$archived\_log;

## Referenced By

None

## Notes/Comments

None

## Revision History:

|  |  |  |
| --- | --- | --- |
| Revision Date | Author | Summary of Change |
| 2/17/16 | Kumar Viswanathan | 1. Initial version |
|  |  |  |
|  |  |  |