

MAA / Data Guard 10g Setup Guide –
Creating a RAC Physical Standby
for a RAC Primary

*Oracle Maximum Availability Architecture White Paper
April 2007*

Maximum Availability Architecture

Oracle Best Practices For High Availability

MAA / Data Guard 10g Setup Guide – Creating a RAC Physical Standby for a RAC Primary

Overview.....	2
Task 1: Gather Files and Perform Back Up	3
Task 2: Configure Oracle Net SERVICES on the Standby.....	3
Task 3: Create the Standby Instances and Database.....	4
Task 4: Configure The Primary Database For Data Guard.....	9
Task 5: Verify Data Guard Configuration.....	10
References	11

MAA / Data Guard 10g Setup Guide – Creating a RAC Physical Standby for a RAC Primary

OVERVIEW

[Oracle Maximum Availability Architecture \(MAA\)](#) [1] is Oracle's best practices blueprint based on proven Oracle high-availability technologies and recommendations. The goal of MAA is to remove the complexity in designing the optimal high-availability architecture.

Published as part of the MAA series of white papers, this paper focuses on creating a RAC physical standby database for a RAC primary database. This document assumes that there is an existing RAC database and you want to implement Data Guard by adding a standby database to the configuration. The end configuration for this document is a RAC primary database with a RAC standby database. The steps outlined in this document use SQL*Plus, apply to both Oracle Database 10g Release 1 and Oracle Database 10g Release 2, and they assume using ASM/OMF, and that the software and ASM instance on the standby host have already been installed/created.

The example used in this document has the database unique name of the RAC database as CHICAGO. The instance names of the two RAC instances are CHICAGO1 (on node chicago_host1) and CHICAGO2 (on node chicago_host2). The database unique name of the RAC standby database is BOSTON, and the two standby instance names are BOSTON1 (on node boston_host1) and BOSTON2 (on node boston_host2).

This document includes the following tasks:

- Task 1: Gather Files and Perform Back Up
- Task 2: Configure Oracle Net on the Standby
- Task 3: Create the Standby Instances and Database
- Task 4: Configure the Primary Database for Data Guard
- Task 5: Verify Data Guard Configuration

This document assumes that the following conditions are met:

- The primary RAC database is in archivelog mode
- The primary RAC database is using ASM.

- The standby RAC cluster already has ASM instances created
- The primary and standby databases are using a flash recovery area.
- The standby RAC hosts have existing Oracle software installation.
- Oracle Managed Files (OMF) is used for all storage.

TASK 1: GATHER FILES AND PERFORM BACK UP

1. On the primary node, create a staging directory. For example:

```
[oracle@chicago_host1 oracle]$ mkdir -p /opt/oracle/stage
```
2. Create the same exact path on one of the standby hosts:

```
[oracle@boston_host1 oracle]$ mkdir -p /opt/oracle/stage
```
3. On the primary node, connect to the primary database and create a PFILE from the SPFILE in the staging directory. For example:

```
SQL> CREATE PFILE='/opt/oracle/stage/initCHICAGO.ora' FROM SPFILE;
```
4. On the primary node, perform an RMAN backup of the primary database that places the backup pieces into the staging directory. For example:

```
[oracle@chicago_host1 stage]$ rman target /  
  
RMAN> BACKUP DEVICE TYPE DISK FORMAT '/opt/oracle/stage/%U'  
DATABASE PLUS ARCHIVELOG;  
  
RMAN> BACKUP DEVICE TYPE DISK FORMAT '/opt/oracle/stage/%U'  
CURRENT CONTROLFILE FOR STANDBY;
```
5. Place a copy of the listener.ora, tnsnames.ora, and sqlnet.ora files into the staging directory. For example:

```
[oracle@chicago_host1 oracle]$ cp  
$ORACLE_HOME/network/admin/*.ora /opt/oracle/stage
```
6. Copy the contents of the staging directory on the RAC primary node to the standby node on which the staging directory was created on in step 2. For example:

```
[oracle@chicago_host1 oracle]$ scp /opt/oracle/stage/* \  
oracle@boston_host1:/opt/oracle/stage
```

TASK 2: CONFIGURE ORACLE NET SERVICES ON THE STANDBY

1. Copy the listener.ora, tnsnames.ora, and sqlnet.ora files from the staging directory on the standby host to the \$ORACLE_HOME/network/admin directory on all standby hosts.
2. Modify the listener.ora file each standby host to contain the VIP address of that host.

3. Modify the tnsnames.ora file on each node, including the primary RAC nodes and standby RAC nodes, to contain all primary and standby net service names. You should also modify the Oracle Net aliases that are used for the local_listener and remote_listener parameters to point to the listener on each standby host. In this example, each tnsnames.ora file should contain all of the net service names in the following table:

Example Entries in the tnsnames.ora Files

Primary Net Service Names	Standby Net Service Name
<pre> CHICAGO = (DESCRIPTION = (ADDRESS = (PROTOCOL = TCP) (HOST = chicago_host1vip) (HOST = chicago_host2vip) (PORT = 1521)) (CONNECT_DATA = (SERVER = DEDICATED) (SERVICE_NAME = CHICAGO))) </pre>	<pre> BOSTON = (DESCRIPTION = (ADDRESS = (PROTOCOL = TCP) (HOST = boston_host1vip) (HOST = boston_host2vip) (PORT = 1521)) (CONNECT_DATA = (SERVER = DEDICATED) (SERVICE_NAME = BOSTON))) </pre>

4. Start the standby listeners on all standby hosts.

TASK 3: CREATE THE STANDBY INSTANCES AND DATABASE

1. To enable secure transmission of redo data, make sure the primary and standby databases use a password file, and make sure the password for the SYS user is identical on every system. For example:

```

$ cd $ORACLE_HOME/dbs
$ orapwd file=orapwBOSTON password=oracle

```

The naming and location of the password file varies on different platforms. See [“Creating and Maintaining a Password File”](#) in the *Oracle Database Administrator’s Guide* for more information.

2. Copy and rename the primary database PFILE from the staging area on all standby hosts to the \$ORACLE_HOME/dbs directory on all standby hosts. For example:

```

[oracle@boston_host1 stage]$ cp initCHICAGO1.ora
$ORACLE_HOME/dbs/initBOSTON1.ora

```

3. Modify the standby initialization parameter file copied from the primary node to include Data Guard parameters as illustrated in the following table:

Initialization Parameter Modifications

Parameter Category	Before	After
RAC Parameters	<pre> *.cluster_database=true *.db_unique_name=CHICAGO CHICAGO1.instance_name=CHICAGO1 CHICAGO2.instance_name=CHICAGO2 CHICAGO1.instance_number=1 CHICAGO2.instance_number=2 CHICAGO1.thread=1 CHICAGO2.thread=2 CHICAGO1.undo_tablespace=UNDOTBS1 CHICAGO2.undo_tablespace=UNDOTBS2 *.remote_listener=LISTENERS_CHICAGO CHICAGO1.LOCAL_LISTENER=LISTENER_CHICAGO_HOST1 CHICAGO2.LOCAL_LISTENER=LISTENER_CHICAGO_HOST2 </pre>	<pre> *.cluster_database=true *.db_unique_name=BOSTON BOSTON1.instance_name=BOSTON1 BOSTON2.instance_name=BOSTON2 BOSTON1.instance_number=1 BOSTON2.instance_number=2 BOSTON1.thread=1 BOSTON2.thread=2 BOSTON1.undo_tablespace=UNDOTBS1 BOSTON2.undo_tablespace=UNDOTBS2 *.remote_listener=LISTENERS_BOSTON BOSTON1.LOCAL_LISTENER=LISTENER_BOSTON_HOST1 BOSTON2.LOCAL_LISTENER=LISTENER_BOSTON_HOST2 </pre>
Data Guard Parameters		<pre> *.log_archive_config='dg_config= (BOSTON,CHICAGO)' *.log_archive_dest_2='service=CHICAGO valid_for=(online_logfiles,primary_role) db_unique_name=CHICAGO' *.db_file_name_convert='+DATA/CHICAGO/', '+DATA/BOSTON/','+RECOVERY/CHICAGO', '+RECOVERY/BOSTON' *.log_file_name_convert='+DATA/CHICAGO/', '+DATA/BOSTON/','+RECOVERY/CHICAGO', '+RECOVERY/BOSTON' *.standby_file_management=auto *.fal_server='CHICAGO' *.fal_client='BOSTON' *.service_names='BOSTON' </pre>
Other parameters	<pre> *.background_dump_dest= /opt/oracle/admin/CHICAGO/bdump *.core_dump_dest= /opt/oracle/admin/CHICAGO/cdump *.user_dump_dest= /opt/oracle/admin/CHICAGO/udump *.audit_file_dest= /opt/oracle/admin/CHICAGO/adump *.db_recovery_dest='+RECOVERY' *.log_archive_dest_1 = 'LOCATION='+DATA/CHICAGO/' *.dispatchers=CHICAGOXDB </pre>	<pre> *.background_dump_dest= /opt/oracle/admin/BOSTON/bdump *.core_dump_dest= /opt/oracle/admin/BOSTON/cdump *.user_dump_dest= /opt/oracle/admin/BOSTON/udump *.audit_file_dest= /opt/oracle/admin/BOSTON/adump *.db_recovery_dest='+RECOVERY' *.log_archive_dest_1= 'LOCATION=USE_DB_RECOVERY_FILE_DEST' *.dispatchers=BOSTONXDB </pre>

In the above example the primary and standby datafiles are in a single ASM diskgroup. If the primary and standby datafiles are distributed across multiple ASM diskgroups then the unset the DB_CREATE_FILE_DEST parameter prior to starting the standby instance. For further information refer to MetaLink note 340848.1.

For more information about these initialization parameters, see Chapter 13, “Initialization Parameters” in [Oracle Data Guard Concepts and Administration](#) manual.

If you are using an SPFILE instead of an initialization parameter file, then see the [“Managing Initialization Parameters Using a Server Parameter File”](#) section in the [Oracle Database Administrator’s Guide](#) for instructions on managing an SPFILE.

4. Connect to the ASM instance on one standby host, and create a directory within the DATA disk group that has the same name as the DB_UNIQUE_NAME of the standby database. For example:

```
SQL> ALTER DISKGROUP data ADD DIRECTORY '+DATA/BOSTON';
```

5. Connect to the standby database on one standby host, with the standby in the IDLE state, and create an SPFILE in the standby DATA disk group:

```
SQL> CREATE SPFILE='+DATA/BOSTON/spfileBOSTON.ora' FROM PFILE='?/dfs/initBOSTON.ora';
```

6. In the \$ORACLE_HOME/dfs directory on each standby host, create a PFILE that is named `initoracle_sid.ora` that contains a pointer to the SPFILE. For example:

```
[oracle@boston_host1 oracle]$ cd $ORACLE_HOME/dfs
[oracle@boston_host1 dfs]$ echo
"SPFILE='+DATA/BOSTON/spfileBOSTON.ora'" > initBOSTON1.ora
```

7. Create the dump directories on all standby hosts as referenced in the standby initialization parameter file. For example:

```
[oracle@boston_host1 oracle]$ mkdir -p
$ORACLE_BASE/admin/BOSTON/bdump
[oracle@boston_host1 oracle]$ mkdir -p
$ORACLE_BASE/admin/BOSTON/cdump
[oracle@boston_host1 oracle]$ mkdir -p
$ORACLE_BASE/admin/BOSTON/udump
[oracle@boston_host1 oracle]$ mkdir -p
$ORACLE_BASE/admin/BOSTON/adump
```

8. After setting up the appropriate environment variables on each standby host, such as ORACLE_SID, ORACLE_HOME, and PATH, start the standby database instance on the standby host that has the staging directory, without mounting the control file.

```
SQL> STARTUP NOMOUNT
```

9. From the standby host where the standby instance was just started, duplicate the primary database as a standby into the ASM disk group. For example:

```
$ rman target sys/oracle@CHICAGO auxiliary /
RMAN> DUPLICATE TARGET DATABASE FOR STANDBY;
```

10. Connect to the standby database, and create the standby redo logs to support the standby role. The standby redo logs must be the same size as

the primary database online logs. The recommended number of standby redo logs is:

```
(maximum # of logfiles +1) * maximum # of threads
```

This example uses two online log files for each thread. Thus, the number of standby redo logs should be $(2 + 1) * 2 = 6$. That is, one more standby redo log file for each thread.

```
SQL> ALTER DATABASE ADD STANDBY LOGFILE THREAD 1
GROUP 5 SIZE 10M,
GROUP 6 SIZE 10M,
GROUP 7 SIZE 10M;
```

```
SQL> ALTER DATABASE ADD STANDBY LOGFILE THREAD 2
GROUP 8 SIZE 10M,
GROUP 9 SIZE 10M,
GROUP 10 SIZE 10M;
```

These statements create two standby log members for each group, and each member is 10MB in size. One member is created in the directory specified by the `DB_CREATE_FILE_DEST` initialization parameter, and the other member is created in the directory specified by `DB_RECOVERY_FILE_DEST` initialization parameter. Because this example assumes that there are two redo log groups in two threads, the next group is group five.

You can check the number and group numbers of the redo logs by querying the `V$LOG` view:

```
SQL> SELECT * FROM V$LOG;
```

You can check the results of the previous statements by querying the `V$STANDBY_LOG` view:

```
SQL> SELECT * FROM V$STANDBY_LOG;
```

You can also see the members created by querying the `V$LOGFILE` view:

```
SQL> SELECT * FROM V$LOGFILE;
```

See the [“Configure a Standby Redo Log”](#) section in [Oracle Data Guard Concepts and Administration](#) manual for more information.

11. On only one standby host (and this is your designated Redo Apply instance), start managed recovery and real-time apply on the standby database:

```
SQL> ALTER DATABASE RECOVER MANAGED STANDBY DATABASE USING
CURRENT LOGFILE DISCONNECT;
```

12. On either node of the standby cluster, register the standby database and the database instances with the Oracle Cluster Registry (OCR) using the Server Control (SRVCTL) utility. For example:

```
$ srvctl add database -d BOSTON -o
/opt/oracle/product/10g_db_rac
$ srvctl add instance -d BOSTON -i BOSTON1 -n boston_host1
$ srvctl add instance -d BOSTON -i BOSTON2 -n boston_host2
```


The following are descriptions of the options in these commands:

The `-d` option specifies the database unique name (DB_UNIQUE_NAME) of the database.

The `-i` option specifies the database instance name.

The `-n` option specifies the node on which the instance is running.

The `-o` option specifies the Oracle home of the database.

Register the ASM instance with the OCR:

```
$ srvctl add asm -n boston_host1 -i +ASM1 -o
/opt/oracle/product/10g_db_rac -p
/opt/oracle/product/10g_db_rac/dbs/spfile+ASM1.ora

$ srvctl add asm -n boston_host2 -i +ASM2 -o
/opt/oracle/product/10g_db_rac -p
/opt/oracle/product/10g_db_rac/dbs/spfile+ASM2.ora
```

The following are descriptions of the options in these commands:

The `-i` option specifies the ASM instance name. If your ASM instance is named +ASM1, then specify it with the '+' included. In `crs_stat` output, the resource name will not have the '+' in the resource name. However, the '+' must be specified when an ASM instance name is specified in SRVCTL commands.

The `-n` option specifies the node name on which the ASM instance is running.

The `-o` option specifies the Oracle home for the ASM instance.

The `-p` option specifies the fully-qualified filename of the SPFILE, if the ASM instance is using an SPFILE. This option is not needed if you are using a PFILE located in \$ORACLE_HOME/dbs directory.

The following commands establish the dependency between the database instance and the ASM instance. Again, the ASM instance name must be specified with a '+' if that is the ASM instance name.

```
$ srvctl modify instance -d BOSTON -i BOSTON1 -s +ASM1
$ srvctl modify instance -d BOSTON -i BOSTON2 -s +ASM2
$ srvctl enable asm -n boston_host1 -i +ASM1
$ srvctl enable asm -n boston_host2 -i +ASM2
```

The following are descriptions of the options in these commands:

The `-d` option specifies the database unique name (DB_UNIQUE_NAME) of the database.

The `-i` option specifies the database instance name.

The `-s` option specifies the ASM instance name.

The following commands start, from the OCR standpoint, all ASM instances defined for the node specified by the `-n` option. If the ASM instance is already running, it will change the status in `crs_stat` output from OFFLINE to ONLINE.

```
$ srvctl start asm -n boston_host1
$ srvctl start asm -n boston_host2
```

If you have multiple ASM instances running on the same node, and only want to start a specific ASM instance, then specify the ASM instance name using the `-i` option, as in the following example:

```
$ srvctl start asm -n boston_host1 -i +ASM1
```

TASK 4: CONFIGURE THE PRIMARY DATABASE FOR DATA GUARD

1. Configure the primary database initialization parameters to support both the primary and standby roles.

```
*.log_archive_config='dg_config=(BOSTON,CHICAGO) '
*.log_archive_dest_2='service=BOSTON
    valid_for=(online_logfiles,primary_role)
    db_unique_name=BOSTON'
*.db_file_name_convert='+DATA/BOSTON/ ','+DATA/CHICAGO/ ',
'+RECOVERY/BOSTON', '+RECOVERY/CHICAGO'
*.log_file_name_convert='+DATA/BOSTON/ ','+DATA/CHICAGO/ ',
'+RECOVERY/BOSTON', '+RECOVERY/CHICAGO'
*.standby_file_management=auto
*.fal_server='BOSTON'
*.fal_client='CHICAGO'
*.service_names=CHICAGO
```

For more information about these initialization parameters, see Chapter 13, “Initialization Parameters” in the [Oracle Data Guard Concepts and Administration](#) manual.

If you are using an SPFILE instead of an initialization parameter file, then see the [“Managing Initialization Parameters Using a Server Parameter File”](#) section in the [Oracle Database Administrator’s Guide](#) for instructions on managing an SPFILE.

Note that all the parameters listed above can be dynamically modified with the exception of the standby role parameters `log_file_name_convert` and `db_file_name_convert`. It is recommended to set the parameters with `“scope=spfile”` so that they can be put into effect upon the next role change.

2. Create standby redo logs on the primary database to support the standby role. The standby redo logs are the same size as the primary database online logs. The recommended number of standby redo logs is one more than the number of online redo logs for each thread. Because this example has two online redo logs for each thread, three standby redo logs are required for each thread.

```
SQL> ALTER DATABASE ADD STANDBY LOGFILE THREAD 1  
GROUP 5 SIZE 10M,  
GROUP 6 SIZE 10M,  
GROUP 7 SIZE 10M;
```

```
SQL> ALTER DATABASE ADD STANDBY LOGFILE THREAD 2  
GROUP 8 SIZE 10M,  
GROUP 9 SIZE 10M,  
GROUP 10 SIZE 10M;
```

These statements create two standby log members for each group, and each member is 10MB in size. One member is created in the directory specified by the DB_CREATE_FILE_DEST initialization parameter, and the other member is created in the directory specified by DB_RECOVERY_FILE_DEST initialization parameter. Because this example assumes that there are two redo log groups in two threads, the next group is group five.

You can check the number and group numbers of the redo logs by querying the V\$LOG view:

```
SQL> SELECT * FROM V$LOG;
```

You can check the results of the previous statements by querying the V\$STANDBY_LOG view:

```
SQL> SELECT * FROM V$STANDBY_LOG;
```

You can also see the members created by querying V\$LOGFILE view:

```
SQL> SELECT * FROM V$LOGFILE;
```

See the [“Configure a Standby Redo Log”](#) section in *Oracle Data Guard Concepts and Administration* manual for more information.

TASK 5: VERIFY DATA GUARD CONFIGURATION

1. On the standby database, query the V\$ARCHIVED_LOG view to identify existing files in the archived redo log. For example:

```
SQL> SELECT SEQUENCE#, FIRST_TIME, NEXT_TIME  
FROM V$ARCHIVED_LOG ORDER BY SEQUENCE#;
```

2. On the primary database, issue the following SQL statement to force a log switch and archive the current online redo log file group:

```
SQL> ALTER SYSTEM ARCHIVE LOG CURRENT;
```

3. On the standby database, query the V\$ARCHIVED_LOG view to verify that the redo data was received and archived on the standby database:

```
SQL> SELECT SEQUENCE#, FIRST_TIME, NEXT_TIME  
FROM V$ARCHIVED_LOG ORDER BY SEQUENCE#;
```

REFERENCES

1. Oracle Maximum Availability Architecture website on OTN
<http://www.oracle.com/technology/deploy/availability/htdocs/maa.htm>



MAA / Data Guard 10g Setup Guide – Creating a RAC Physical Standby For a RAC Primary
April, 2007
Author: Mike Smith

Oracle Corporation
World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065
U.S.A.

Worldwide Inquiries:
Phone: +1.650.506.7000
Fax: +1.650.506.7200
oracle.com

Copyright © 2007, Oracle. All rights reserved.

This document is provided for information purposes only and the contents hereof are subject to change without notice.

This document is not warranted to be error-free, nor subject to any other warranties or conditions, whether expressed orally or implied in law, including implied warranties and conditions of merchantability or fitness for a particular purpose. We specifically disclaim any liability with respect to this document and no contractual obligations are formed either directly or indirectly by this document. This document may not be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without our prior written permission.

Oracle, JD Edwards, and PeopleSoft are registered trademarks of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.