Joint Engineering White Paper

EMC TimeFinder and SRDF Best Practices for Oracle Database 10g Automatic Storage Management

Symmetrix DMX

Authors: Nitin Vengurlekar, Bill Bridge - Oracle
Bob Goldsand - EMC
Co-author: Ara Shakian - Oracle

ORACLE

EMC²
where information lives

Abstract: The purpose of this document is to provide a comprehensive set of best practices and procedures when deploying Oracle Database 10g and the Automatic Storage Management (ASM) feature with EMC Symmetrix storage based replication technologies.

Table of Contents

Introduction

The purpose of this document is to provide a comprehensive set of best practices and procedures when deploying Oracle Database 10g and Automatic Storage Management (ASM) with EMC Symmetrix storage based replication and Consistency Technologies. This includes EMC TimeFinder/Mirror and Symmetrix Remote Data Facility (SRDF-Asynchronous and Synchronous) , which have been validated in accordance with the Oracle's Storage Compatibility Program (OSCP) already, and now being extended to include validation when using Oracle Database 10g Automatic Storage Management.

This paper will document the procedures and best practices for the following use cases:

- Oracle Database 10g Hot Backup with TimeFinder/Mirror for Database Backup
- TimeFinder/Mirror for Database Cloning
- SRDF and TimeFinder/Mirror for DR and Remote Database Cloning
- Oracle Database 10g Hot Backup with SRDF and TimeFinder/Mirror for DR and Remote Database Backup

This document assumes the reader has a basic understanding of Oracle Database 10g Automatic Storage Management and EMC TimeFinder/Mirror and SRDF technologies.

Related Documents

- *Using Oracle 10g's Automatic Storage Management with EMC Storage Technology*
- *Using SYMCLI to Perform Consistent Splits with the TimeFinder Product Family.*
- *Understanding EMC Consistent Split with Oracle Databases*
- *Oracle Database 10g Automatic Storage Management Best Practices*

Oracle Automatic Storage Management

Automatic Storage Management (ASM) is a storage manager that provides file system, volume management and clustering capabilities integrated into the Oracle Database 10g at no additional cost. ASM lowers your total cost of ownership, increases storage utilization without compromising performance or availability. With ASM, a fraction of the time is needed to manage your database files.
ASM eliminates over provisioning and maximizes storage resource utilization facilitating database consolidation. The ASM self-tuning feature evenly distributes the data files across all available storage. It delivers high performance similar to raw, sustained over time, with the ease of use of a file system. ASM's intelligent mirroring technology enables up to triple data protection, even on non-RAID storage arrays. ASM benefits are:

- Simplify and automate storage management
- Increase storage utilization and agility
- Predictably deliver on performance and availability service level agreements

ASM simplifies storage management tasks, such as creating/laying out databases and disk space management. Since ASM allows disk management to be done using familiar create/alter/drop SQL statements, DBAs do not need to learn a new skill set or make crucial decisions on provisioning. Additionally, ASM operations can be completely managed with 10g Enterprise Manager. ASM is a management tool specifically built to simplify the job of the DBA. It provides a simple storage management interface across all server and storage platforms. ASM provides the DBA flexibility to manage a dynamic database environment with increased efficiency. This feature is a key aspect of Grid Computing. For more information about ASM, please refer to the OTN ASM homepage:
http://www.oracle.com/technology/products/database/asm/index.html

Oracle Recovery Manager (RMAN)

Recovery Manager is Oracle's utility to manage the backup, and more importantly the recovery, of the database. It eliminates operational complexity while providing superior performance and availability of the database. Recovery Manager debuted with Oracle8 to provide DBAs an integrated backup and recovery solution. Recovery Manager determines the most efficient method of executing the requested backup, restore, or recovery operation, and then executes these operations in concert with the Oracle database server. Recovery Manager and the server automatically identify modifications to the structure of the database and dynamically adjust the required operation to adapt to the changes.

Oracle Flash Recovery Area

The Flash Recovery Area is a unified storage location for all recovery related files and activities in an Oracle database. By defining one init.ora parameter, all RMAN backups, archive logs, control file autobackups, and datafile copies are automatically written to a specified file system or ASM disk group. In addition, RMAN automatically manages the files in the Flash Recovery Area by deleting obsolete backups and archive logs that are no longer required for recovery. Allocating sufficient space to the Flash Recovery Area will ensure faster, simpler, and automatic recovery of the Oracle database.

EMC Consistency Technology

Beginning with EMC Solutions Enabler version 5.1, you can use the Enginuity Consistency Assist (ECA) feature to perform consistent splits on BCV pairs across multiple, heterogeneous hosts. Consistent split is an implementation of instant split that avoids inconsistencies and restart problems that can occur if you split a database-related BCV without first quiescing the database. The difference between a normal instant split and a consistent split is that during consistent split the database writes are held at the storage level for a very short time while the foreground split occurs, maintaining dependent-write order consistency on the target devices.

Consistency technology, whether applied to SRDF, TimeFinder BCVs, clones, or snaps, provides the capability to create an image of one or more databases that are DBMS restartable copies. It does this by momentarily holding all write IO to the specified Symmetrix volumes while performing a split operation.

The resultant databases on the target volumes are in a data state that is equivalent to the state they would be in after a power failure. In an Oracle context, a more appropriate analogy would be that they look the same as if all database instances performed shutdown abort simultaneously.

Since restarting an aborted instance does not in any way require the database to be in Oracle's hot backup mode, we are able to provide customers with a way to create restartable database clones without requiring the user to place the databases tablespaces in hot backup mode.

EMC TimeFinder Overview

TimeFinder software works by creating multiple, independently addressable business continuance volumes (BCVs) for independent storage. The BCV is a Symmetrix® device with special attributes created when the Symmetrix is configured. It can function either as an additional mirror to a Symmetrix logical volume or as an independent, host-addressable volume. Establishing BCV devices as mirror images of active production volumes allows you to run multiple simultaneous business continuance tasks in parallel. The principal device, known as the standard device, remains online for regular Symmetrix operation from the production server. Each BCV contains a unique host address, making it accessible to a separate backup/recovery server. When you establish a BCV as a mirror of a standard device, that relationship is known as a BCV pair. Any time you split one of the BCVs from the standard device, the BCV has the mirrored data from the standard device and so it is available for backup, testing, analysis, or snapping (making instant copies).

TimeFinder consistent split is used to create valid point-in-time restartable images of the Oracle database. These point-in-time restartable images are not valid Oracle backups; Oracle backups require additional procedures, such as putting the tablespace(s) into hot backup mode prior to splitting the BCVs.

TimeFinder Consistent Split

TimeFinder software provides a consistent-split implementation of instant split that allows you to split off a consistent, DBMS-restartable BCV copy of your database without having to shut down the database or put the database files into hot backup mode. It is able to do this by simultaneously holding all write I/O to database devices momentarily before splitting the BCVs. After mounting the BCVs to a host, a subsequent Oracle startup will perform instance crash recovery, ensuring the integrity of the database image.

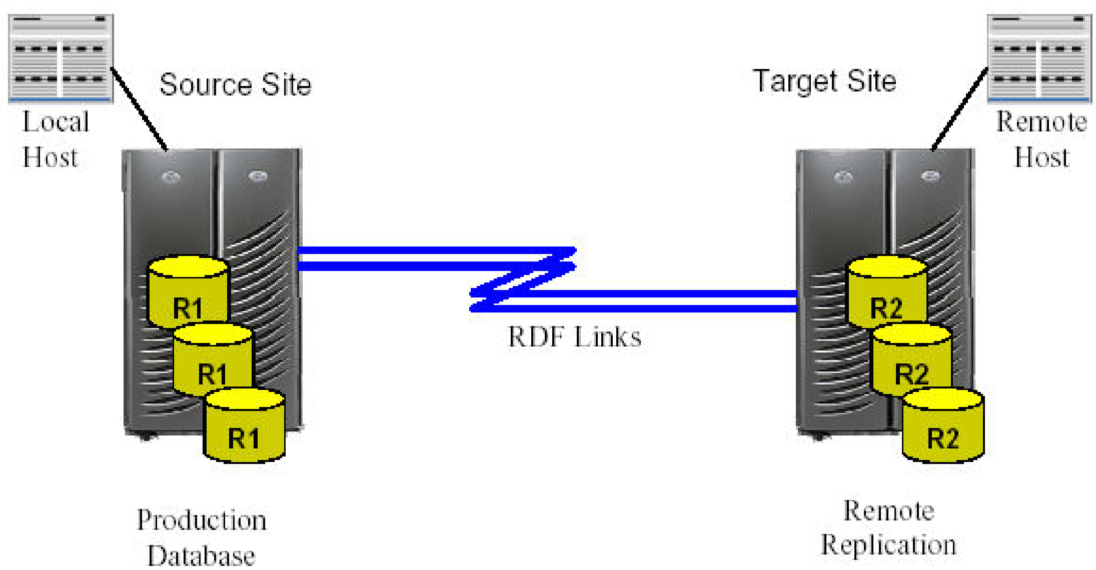
A point-in-time database image taken with a consistent split is not a valid Oracle backup without additional procedures such as putting the database in hot backup mode prior to the split. A consistent split is used for the purpose of creating a restartable image of the database at a specific point in time. For more information about TimeFinder consistent splits, refer to the white paper Using SYMCLI to Perform Consistent Splits with the TimeFinder Product Family (P/N 300-000-283).

EMC SRDF Overview

Symmetrix Remote Data Facility (SRDF) is a Symmetrix-based business continuance and disaster restart solution. In simplest terms, SRDF is a configuration of multiple Symmetrix units whose purpose is to maintain real-time copies of logical data volume in more than one location. The Symmetrix units can be in the same room, in different buildings within the same campus, or hundreds of miles apart. SRDF provides data mobility and disaster restart spanning multiple host platforms, operating systems, and applications.

The local SRDF device, known as the source (R1) device, is configured in a pairing relationship with a remote target (R2) device, forming an SRDF pair. While the R2 device is mirrored with the R1 device, the R2 device is write-disabled to the host. After the (R2) device becomes synchronized with its (R1) device, you can split the (R2) device from the (R1) device at any time, making the (R2) device fully accessible again to its host. After the split, the target (R2) device contains (R1) data and is available for performing business continuance tasks through its original device address or restoring (copying) data back to the source (R1) device. Figure 1 shows a typical SRDF configuration.

Figure 1 Typical SRDF Configuration



SRDF Protection Modes

SRDF currently supports the following modes of operations for database restart or database recovery solutions with Oracle Databases.

SRDF Synchronous Mode

In SRDF synchronous mode, every I/O from the production host is first written to the local Symmetrix cache, and is then sent over the SRDF links to the remote Symmetrix unit. Once the remote Symmetrix unit reports that the data has reached its cache successfully, the I/O is acknowledged to the production host. Synchronous mode guarantees that the remote image is a complete duplication of the source image.

SRDF Asynchronous Mode

Many SRDF customers use synchronous mode to protect data on a primary storage system. Synchronous mode creates a consistent copy of data on the secondary storage system (R2), but carries a price, both in performance (response time on the R1 side host), and cost (high-capacity links). The main premise of SRDF asynchronous mode is to provide a consistent, point-in-time image on the (R2) side, which is not too far behind the (R1) side, and that results in minimal data loss in the event of a disaster at the primary site.

Disaster Restart and Disaster Recovery

Classical disaster recovery techniques have evolved over several decades. In most cases, a disaster recovery activity implies usage of data tapes that have been stored offsite in a secure location. Full backup tapes of disk data are usually taken periodically—often during a low transaction period, such as a Saturday or Sunday night. During the rest of the week, incremental (or tapes capturing changed disk data since the last full or previous incremental backup) are gathered and sent offsite. By being stored offsite, these tapes use geographic separation to guard against any local disaster, such as a fire or a flood. In a disaster situation, the user must gather all the tapes and apply them in sequence. Considering the times associated with running the full and incremental tape backups from disk, packaging the tapes for offsite transport, gathering them in the event of a disaster, and recovering them back to disk storage at the backup site, 48 hours may be considered a realistic expectation for the duration of a disaster recovery activity. These activities are also susceptible to human error, such as the tapes incorrectly applied in the disaster recovery sequence, lost tapes, damaged tapes, incompatible tapes, etc. Once the remote site is running, a similarly lengthy outage (usually involving tapes) occurs to go home once repairs have been made at the original data center.

Disaster restart, on the other hand, does not use computer tapes. Rather, data is transported by communications links to remote data storage. The remote replica of data serves as the restart point, and the user may restart the application using disk images at the remote site.

Oracle's recommended disaster restart and recovery solution is Oracle Data Guard, which is a built-in feature of the Oracle Database. However, for this document, the disaster restart and recovery discussions will focus on EMC SRDF.

A major issue is whether the data in the remote location is logically consistent. EMC SRDF synchronous mode and EMC SRDF asynchronous mode ensure the dependent write order consistency of the replication by synchronizing each and every dependent I/O (SRDF synchronous mode) or by synchronizing delta sets of data (SRDF asynchronous mode). In a true physical disaster at the source location, database restart operations can be completed at the remote site without the delays associated with finding and applying tapes in the correct sequence. Because the remote site has physical disk replicas, the go home activity is likewise very fast and easy.

In addition to disaster restart benefits, SRDF significantly enhances disaster recovery operations by using fast and reliable replication technology to offload the Oracle backup operations to a remote site and later return the restored data to the local site.

When a disaster recovery solution is required, in order to create valid Oracle backups with any split mirror or snapshot technology additional procedures are required, such as to ensure that the database is in hot backup mode during the split or by using Oracle Recovery Manager (RMAN). Refer to Oracle documentation for further details regarding RMAN.

Rebalancing and Consistency Technology

ASM provides a seamless and non-intrusive mechanism to expand and shrink the diskgroup storage. When disk storage is added or removed, ASM will perform a redistribution (rebalancing) of the striped data[1]. This entire rebalance operation is done while the database is online, thus providing higher availability to database. The main objective of the rebalance operation is to always provide an even distribution of file extents and space usage across all disks in the diskgroup.

It is considered a best practice to use ASM external redundancy for data protection when using EMC arrays. The Symmetrix will provide protection against loss of media, as well as transparent failover in the event of a specific disk or component failure.

The split operation of storage based replicas is sensitive to the rebalancing process which may cause ASM disk group inconsistencies if the disk group device members are split at slightly different times. These inconsistencies are a result of ASM metadata changes occurring while a split operation is in process. . In addition Oracle provides tools and procedural steps to avoid inconsistencies when splitting storage based replicas, however these procedure can be simplified and streamlined with the use of EMC Consistency Technology.

Since EMC consistent split technology suspends database I/O to preserve write order consistency, it also has the side effect of preventing any ASM metadata changes during the split. Performing a consistent split will prevent ASM metadata inconsistencies during the replication process eliminating the otherwise extra steps or possible unusable replica if ASM rebalance was active while performing a non-consistent split.

Test Cases and Best Practices

The following test cases and results show that Oracle 10g database and Automatic Storage Management can be deployed non-disruptively with EMC TimeFinder/Mirror and SRDF family of products. If a rebalance operation is triggered while a consistent split is being performed, any ASM metadata changes are held until the source and target are in a synchronous state.

General Tests Configuration

Host names:

- The term Production host refers to the primary host where the source devices are used, and Target or Backup host refers to the host where the BCV, R2 or Remote BCV (RBCV) devices are used.

Assumptions:

- The target host is configured with Operating System level, user and group id, Oracle binaries and directory structure similar to production and is also configured for ASM
- A copy of the production init.ora files for the ASM instance and the database instance were copied to the backup host (target host) and modified if required to fit the target host environment (specifically ASM_DISKSTRING contains the appropriate BCV, R2 or remote BCV devices).
- The copy of the production orapwd file is available on the target host.
- The appropriate BCV, R2 or Remote BCV (whichever is appropriate for the test) are accessible by the target host and have Oracle permissions.
- An RMAN recovery catalog is configured and operational.
- The backup server (target host) has Oracle Net to the recovery catalog database.
- Flash Recovery Area is used

[1] Note, a disk failure, will also trigger a rebalance activity if ASM redundancy is not external.

- The target host has connectivity to a LAN based Tape Backup System (applicable to "backup to tape" scenarios below)

Test conditions were:
- OLTP load was running during the split
- Transaction integrity test (defined by OSCP test kit) was running during the split.
- ASM rebalance was active during the split

Test success was measured by:
- ASM and database instance were opened successfully on target host without any errors reported.
- Transaction integrity test passed
- Rebalance operation continued automatically and completed successfully on target host.
- Database Verification utility verified the integrity of all the data files and no errors were found.

Normally, when consistent split it used, TimeFinder and SRDF commands are issued from a control host connected to the Symmetrix. However in the following test cases for the sake of simplicity, unless specified otherwise, they were issued from the production host.

Hardware

	Model	OS	Oracle Version
Local "Production" Host	SUN	Solaris 2.8	10g Release 2 (10.2.0.1)
"Backup" or "Target" Host	SUN	Solaris 2.8	10g Release 2 (10.2.0.1)

	Name/Serial Number	Type	Enginuity Version
Local Symmetrix	000187900754	DMX 800-M2	5671
Remote Symmetrix	000187900671	DMX 800-M2	5671

ASM Disk Groups / Mount Points

In all cases, the databases were built using three distinct ASM diskgroups.

- The Datafile disk group that contained all data files
- The Flash Recovery Area (FRA) disk group that contained files such as multiplexed control files, backup sets, archive logs and flashback logs. Oracle recommends that archive logs be placed in the Flash Recovery Area.
- The Online Redo disk group that contained online redo logs for the database

Disk Group Purpose	Disk Group Name / Mount Point	Path (on production host)	TimeFinder Standard/R1 Device	TimeFinder BCV Device	SRDF R2 Device	TimeFinder Remote BCV
Flash Recovery Area	**RECOVERY_AREA**	/dev/rdsk/c3t2d0s6	003	07B	003	07B
		/dev/rdsk/c3t2d1s6	004	07C	004	07C
Online Redo Logs	**REDO_AREA**	/dev/rdsk/c3t2d2s6	005	07E	005	07E
		/dev/rdsk/c3t2d3s6	006	07F	006	07F
Data Files	**DATA_AREA**	/dev/rdsk/c3t2d4s6	007	080	007	080
		/dev/rdsk/c3t2d5s6	008	081	008	081
		/dev/rdsk/c3t2d6s6	009	082	009	082
		/dev/rdsk/c3t2d7s6	00A	083	00A	083
		/dev/rdsk/c3t2d8s6	00B	084	00B	084
		/dev/rdsk/c3t2d9s6	00C	085	00C	085
		/dev/rdsk/c3t2d10s6	00D	086	00D	086
		/dev/rdsk/c3t2d11s6	00E	087	00E	087
		/dev/rdsk/c3t2d12s6	00F	088	00F	088
		/dev/rdsk/c3t2d13s6	010	089	010	089
		/dev/rdsk/c3t2d14s6	011	08A	011	08A
		/dev/rdsk/c3t2d15s6	012	08B	012	08B
		/dev/rdsk/c3t2d16s6	013	08C	013	08C
		/dev/rdsk/c3t2d17s6	014	08D	014	08D
		/dev/rdsk/c3t2d18s6	015	08E	015	08E

Note – each device size was approximately 8.6 GB. The Standard devices were mirrored and the BCV were Raid5 protected.

Example: Query the contents of the DATA_AREA diskgroup.

```
SQL> select path, dk.name "disk name", dg.name "diskgroup name"
from v$asm_disk dk, v$asm_diskgroup dg where
dk.group_number=dg.group_number  and dg.name='DATA_AREA;'
```

```
PATH                DISKGROUP_NAME       DISK_NAME
-------------------- -------------- -------------
/dev/rdsk/c3t2d4s6   DATA_AREA      DATA_AREA_0000
/dev/rdsk/c3t2d5s6   DATA_AREA      DATA_AREA_0001
/dev/rdsk/c3t2d6s6   DATA_AREA      DATA_AREA_0002
/dev/rdsk/c3t2d7s6   DATA_AREA      DATA_AREA_0003
/dev/rdsk/c3t2d8s6   DATA_AREA      DATA_AREA_0004
/dev/rdsk/c3t2d9s6   DATA_AREA      DATA_AREA_0005
/dev/rdsk/c3t2d10s6  DATA_AREA      DATA_AREA_0006
/dev/rdsk/c3t2d11s6  DATA_AREA      DATA_AREA_0007
/dev/rdsk/c3t2d12s6  DATA_AREA      DATA_AREA_0008
/dev/rdsk/c3t2d13s6  DATA_AREA      DATA_AREA_0009
/dev/rdsk/c3t2d14s6  DATA_AREA      DATA_AREA_0010
/dev/rdsk/c3t2d15s6  DATA_AREA      DATA_AREA_0011
/dev/rdsk/c3t2d16s6  DATA_AREA      DATA_AREA_0012
/dev/rdsk/c3t2d17s6  DATA_AREA      DATA_AREA_0013
/dev/rdsk/c3t2d18s6  DATA_AREA      DATA_AREA_0014
```

ASM Instance Parameter File

```
INSTANCE_TYPE=ASM
ASM_DISKSTRING='/dev/rdsk/*s6'
ASM_DISKGROUPS='DATA_AREA','REDO_AREA','RECOVERY_AREA'
```

Database Instance Parameter File

(These parameters are specific to the test environment only)

```
db_name = hrd10g
control_files = +DATA_AREA/control_001
DB_RECOVERY_FILE_DEST = +RECOVERY_AREA
LOG_ARCHIVE_DEST_1 = 'LOCATION=USE_DB_RECOVERY_FILE_DEST'
```

Case 1: Oracle Database 10g Hot Backup with TimeFinder/Mirror

While the Oracle database is in Hot Backup mode on the production host, a TimeFinder/Mirror consistent split is performed to create an image of the active database which can be used to perform a backup to tape by offloading this process to a target or backup server..

Create Symmetrix Device Groups and Associate BCV devices

Two device groups were created because hot (online) backup requires the archive logs to be split at a different time than the data files: one for the database files (DBFILES_DG) and the other for archive logs (RECOV_DG). In general it is best practice for online backups to only include the Oracle data files. However with ASM it is possible that control, redo, temp and data files may be all mixed together in small number of ASM disk groups. Note that the Symmetrix device group should always treat ASM disk groups as a unit and include all members.

```
# symdg create DBFILES_DG
# symld  -g DBFILES_DG addall -range 007:015
# symbcv -g DBFILES_DG associateall dev –range 080:08E

# symdg create RECOV_DG
# symld -g RECOV_DG addall -range 003:004
# symbcv -g RECOV_DG associateall dev –range 07B:07C
```

Establish (synchronize) The Device Groups

Use –full option only with the first establish. Consecutive establish are done incrementally.
Wait for the synchronization to complete before performing a TimeFinder Split.

```
# symmir –g DBFILES_DG establish [–full]
# symmir –g DBFILES_DG verify -i 30

# symmir –g RECOV_DG establish [–full]
# symmir –g RECOV_DG verify -i 30
```

ONLINE BACKUP ON PRODUCTION HOST:

Begin Backup Mode

```
# export ORACLE_SID=hrd10g
# sqlplus "/ as sysdba"
SQL> alter database begin backup;
```

Perform a Consistent Split Snapshot for Database Files

 # symmir DBFILES_DG split –consistent

End Backup Mode

```
SQL> alter database end backup;
```

Switch Logs and Create backup Controlfiles

Create two copies of the control file. One copy (control_start) will be used to start up the database in mount mode on the target server. The second copy (control_bakup) is a valid controlfile copy that will be part of the backup set used by RMAN.

```
SQL> alter system archive log current;
```

```
RMAN> run {
        Allocate channel ctl_file type disk;
        Copy current controlfile to
        '+RECOVERY_AREA/control_file/control_start';
        Copy current controlfile to
        '+RECOVERY_AREA/control_file/control_bak';
        Release Channel ctl_file;
        }
```

Resynchronize the RMAN Catalog

This adds the most recent archive log to the recovery catalog.

```
RMAN> resync catalog;
```

Perform a Consistent Split Snapshot of the Recovery Area to capture the Archive Log

```
# symmir RECOV_DG split –consistent
```

BACKUP PROCEDURES

On the Backup host the snapshot can be used as a disk backup or a source for a tape backup. Some backup applications require the database to be mounted to perform backups.

Once the BCVs are split, check that:

- The BCVs on backup host have correct Oracle permissions
- ASM init.ora file parameter ASM_DISKSTRING doesn't exclude the path to the BCVs.
- ASM init.ora file parameter ASM_DISKGROUPS contains the names of the disk groups.

Start ASM Instance

When the ASM instance is started, since the BCV physical names are included in the ASM_DISKSTRING parameter, it will identify them as the disk groups from production database. Also since the ASM_DISKGROUPS parameter contains the disk group names they will be mounted automatically.

```
# export ORACLE_SID=+ASM
# sqlplus "/ as sysdba"
SQL> startup
```

Mount Database Instance

A database backup that was taken with hot backup mode is valid for backup only as long as it wasn't open with resetlogs options or opened for read/write. For that reason it should be either mounted (pre-requisite for media recovery and many backup applications) or open read-only (after at least enough recovery was done to allow the database to open).

Before the database is mounted change the Backup database instance init.ora CONTROL_FILE parameter to point to the copied controlfile. For example:

```
control_files  = +RECOVERY_AREA/control_file/control_start
```

```
# export ORACLE_SID=hrd10g
# sqlplus "/ as sysdba"
SQL> startup mount
```

Backing Up The Database Instance

Perform a RMAN backup on the Backup host. The controlfile copy that was not used to mount the instance (control_bk) should be part of the backupset. The control_start controlfile should not be backed up, because once the database in mounted the SCN will be updated and is is inconsistent with production control file.

```
RMAN> run { ALLOCATE CHANNEL t1 TYPE SBT_TAPE
        BACKUP FORMAT 'ctl %d/%s/%p/%t'
        CONTROLFILECOPY '+RECOVERY_AREA/control_file/control_bak';
        BACKUP
                FULL
                FORMAT 'ctl %d/%s/%p/%t'
                (database);
        BACKUP
                FORMAT 'al %d/%s/%p/%t'
                (archive all);
        RELEASE CHANNEL t1
        }
```

Note: The format specifier %d is replaced with date, `%t` is replaced with a four byte time stamp, `%s` with the backup set number, and `%p` with the backup piece number.

Case 2: Database Cloning with TimeFinder/Mirror

While Oracle is open for read/write on the production host, a TimeFinder/Mirror consistent split is performed on an established TimeFinder/Mirror BCVs. This operation will create a restartable image of the active database that can serve as a repurposed database. The Symmetrix devices included in the ALLDB_DG device group match the ASM disk groups that contain redo logs, data files and a controlfiles. Archive logs are not used with cloning; however it may be beneficial to include the recovery area as well (especially if flashback logs are active). If recovery area is to be made available to the target host as well then include the recovery area devices in the ALLDB Symmetrix device group so they will be part of the consistent split, operation together with the database files.

Create Symmetrix Device Group and Associate BCV devices

Create a single device group for the Oracle data files, control files and Online Redo log files because consistent split requires all the database files to be split together.

```
# symdg create ALLDB_DG
# symld -g ALLDB_DG addall -range 003:015
# symbcv –g ALLDB_DG associateall dev –range 07B:08E
```

Establish (synchronize) The Device Group

Use –full option only with the first establish. Consecutive establish are done incrementally.
Wait for the synchronization to complete before performing a TimeFinder Split.

```
# symmir –g ALLDB_DG establish [–full]
# symmir –g ALLDB_DG verify -i 30
```

Perform a Consistent Split Snapshot for Database Files

```
# symmir ALLDB_DG split –consistent
```

ON TARGET HOST:

On the target host, once the BCVs are split, check that:

- The BCVs have Oracle permissions
- ASM init.ora file parameter ASM_DISKSTRING doesn't exclude the path to the BCVs.
- ASM init.ora file parameter ASM_DISKGROUPS contains the names of the disk groups.

Start ASM Instance

When the ASM instance is started, since the BCVs are included in the ASM_DISKSTRING parameter, it will identify them as the disk groups from production database. Also since the ASM_DISKGROUPS parameter contains the disk group names they will be mounted automatically.

```
# export ORACLE_SID=+ASM
# sqlplus "/ as sysdba"
SQL> startup
```

Start Database Instance

Startup and recover the clone database. Once the clone database is recovered , it should be assigned a new DBID and re-started with resetlogs . The following steps illustrate these steps.

```
# export ORACLE_SID=hrd10g
```

connect to RMAN

```
RMAN> startup mount
RMAN> recover database;
```

```
RMAN> exit

nid target=sys/manager1     optionally the db_name can be changed as well, see the
                            Oracle Recovery Guide for details on the nid utility.

SQL> startup mount
SQL> alter database open resetlogs
```

At the end of this step the database is opened and available for user connections.

Case 3: Remote Database Cloning with TimeFinder/Mirror and SRDF

When using SRDF/S or SRDF/A for database protection there is an advantage for using remote BCVs. The remote BCVs allow SRDF to remain synchronized and maintaining database protection, while at the same time the remote BCVs can be split as a database clone for test, development or reporting. It is possible to use them for backup in combination with Oracle hot backup (as described in the next section). Also they can serve as a gold copy for enhanced protection in situations when SRDF is about to start failback operation and the remote site contains a valid image of the database. It is a best practice to split the remote BCVs before synchronizing the SRDF for the possibility that before SRDF is fully synchronized and the database regain protection, a second failure occurs (also referred as “rolling disaster”).

Note: The following solution addresses the use of remote BCVs. However, to restart the database directly from the R2 devices, use the same steps as described below. The only difference is that instead of the remote BCVs it is the R2 devices that are used to start the ASM and database instances. If this wasn’t a planned failover and the R1 site is not accessible, issue a *symrdf failover* command from the R2 site to make the R2 devices read-writable.

Create Symmetrix Device Group and Associate Remote BCV devices

Because consistent split requires all the database files to be split together create a single device group for the Oracle data files, control files and Online Redo log files. To the same device group add the remote BCVs (remote is indicated by using the –rdf flag. That means that the BCVs are those attached to the R2 devices on the remote Symmetrix).

```
# symdg create ALLDB_DG –type R1
# symld -g ALLDB_DG addall -range 003:015
# symbcv –g ALLDB_DG –rdf associateall dev –range 07B:08E
```

Establish (synchronize) SRDF and Remote BCV devices

Use –full option only with the first establish. Consecutive establish are done incrementally.
Wait for the synchronization to complete before performing a TimeFinder Split. The synchronization of SRDF and the remote BCVs can happen simultaneously.
For SRDF/A, once the SRDF is in a consistent state use the *Enable* SRDF command to guarantee device level consistency.

For SRDF/S protection:

```
Note: SRDF/S is the default SRDF mode. If it was changed use:
# symrdf –g ALLDB_DG set mode sync

# symrdf –g ALLDB_DG establish [–full]
# symrdf –g ALLDB_DG verify -i 30
```

```
# symmir -g ALLDB_DG -rdf establish [-full]
# symmir -g ALLDB_DG -rdf verify -i 30
```

For SRDF/A protection:

```
# symrdf -g ALLDB_DG set mode async
# symrdf -g ALLDB_DG establish [-full]
# symrdf -g ALLDB_DG verify -i 30
# symrdf -g ALLDB_DG enable

# symmir -g ALLDB_DG -rdf establish [-full]
# symmir -g ALLDB_DG -rdf verify -i 30
```

Perform a Consistent Split Snapshot for Database Files

```
# symmir ALLDB_DG -rdf split -consistent
```

ON TARGET HOST:

On the target host, once the BCVs are split, check that:

- The BCVs have Oracle permissions. Note that if the same host has both remote BCVs as well as R2 devices mapped to it then the R2 devices should not have Oracle permissions. The reason is that ASM writes disk group information to the ASM members. As both R2 and Remote BCV contains the exact same Oracle information ASM cannot differentiate between them if both have Oracle permissions.
- ASM init.ora file parameter ASM_DISKSTRING includes the path to the BCVs.
- ASM init.ora file parameter ASM_DISKGROUPS contains the names of the disk groups.

Start ASM Instance

When the ASM instance is started, since the BCVs are included in the ASM_DISKSTRING parameter, it will identify them as the disk groups from production database. Also since the ASM_DISKGROUPS parameter contains the disk group names they will be mounted automatically.

```
# export ORACLE_SID=+ASM
# sqlplus "/ as sysdba"
SQL> startup
```

Start Database Instance

Startup and recover the clone database. Once the clone database is recovered , it should be assigned a new DBID and re-started with resetlogs . The following steps illustrate these steps.

```
# export ORACLE_SID=hrd10g
```

<u>connect to RMAN</u>

```
RMAN> startup mount
RMAN> recover database;
RMAN> exit

nid target=sys/manager1     optionally the db_name can be changed as well, see the
                            Oracle Recovery Guide for details on the nid utility.

SQL> startup mount
SQL> alter database open resetlogs
```

At the end of this step the database is opened and available for user connections.

Case 4: Oracle Database 10g Hot Backup with TimeFinder/Mirror and SRDF

When using SRDF/S or SRDF/A for database protection there is an advantage for using remote BCVs. The remote BCVs allow SRDF to remain synchronized and maintaining database protection, while at the same time the remote BCVs can be split as a database clone for test, dev or reporting. It is possible to use them for backup in combination with Oracle hot backup (as described in this section). Also they can serve as a gold copy for enhanced protection in situations when SRDF is about to start failback operation and the remote site contains a valid image of the database. It is best practice to split the remote BCVs before synchronizing the SRDF for the possibility that before SRDF is fully synchronized and the database regain protection, a second failure occurs (also referred as “rolling disaster”).

Create Symmetrix Device Groups and Associate Remote BCV devices

Two device groups were created for remote TimeFinder operations because hot (online) backup requires the archive logs to be split at a different time than the data files: one device group was created for the database files (DBFILES_DG) and the other for archive logs (RECOV_DG). In general it is best practice for online backups to only include the Oracle data files. However with ASM it is possible that control, redo, temp and data files may be all mixed together in small number of ASM disk groups. Note that the Symmetrix device group should always treat ASM disk groups as a unit and include all members.

Note that remote BCVs are established with the R2 devices (remote is indicated by using the –rdf flag. That means that the BCVs are those attached to the R2 devices on the remote Symmetrix).

```
# symdg create DBFILES_DG –type R1
# symld -g DBFILES_DG addall -range 007:015
# symbcv –g DBFILES_DG –rdf associateall dev –range 080:08E

# symdg create RECOV_DG –type R1
# symld –g RECOV_DG add –range 003:004
# symbcv –g RECOV_DG associateall dev –range 07B:07C
```

Establish (synchronize) SRDF and Remote BCV devices

Use –full option only with the first establish. Consecutive establish are done incrementally.
Wait for the synchronization to complete before performing a TimeFinder Split. The synchronization of SRDF and the remote BCVs can happen simultaneously.
For SRDF/A, once the SRDF is in a consistent state use the *Enable* SRDF command to guarantee device level consistency.

Note: In general, SRDF is required to include all data, control and redo log files together to create a write order consistent and restartable image of the database. In addition, control operations when in SRDF/A mode always have to include ALL the SRDF/A devices in an RDF group together. If (like in this example) TimeFinder operations require two device groups: DBFILES_DG containing data files and RECOV_DG containing archive logs, and SRDF requires a different (larger) set of devices to operate on, a *device file* is used for the SRDF control operations. The device file contains the list of R1 devices including data, redo and control file devices. It may also contain archive log devices if the SRDF is used to replicate archive logs. Otherwise it is possible to ship archive logs over the network to the remote host.
When using a device file in SRDF control commands the Symmetrix ID and SRDF group are specified in the command line.

For SRDF/S protection:

```
Note: SRDF/S is the default SRDF mode. If it was changed use:
# symrdf –g DBFILES_DG set mode sync

# symrdf –sid 754 –rdfg 3 –file ./dev_srdf establish [–full]
# symrdf –sid 754 –rdfg 3 –file ./dev_srdf verify -i 30
```

For SRDF/A protection:

```
# symrdf –sid 754 –rdfg 3 –file ./dev_srdf set mode async

# symrdf –sid 754 –rdfg 3 –file ./dev_srdf establish [–full]
# symrdf –sid 754 –rdfg 3 –file ./dev_srdf verify -i 30
# symrdf –sid 754 –rdfg 3 –file ./dev_srdf enable
```

For TimeFinder:

```
# symmir –g DBFILES_DG –rdf establish [–full]
# symmir –g DBFILES_DG –rdf verify -i 30
```

ONLINE BACKUP ON PRODUCTION HOST:

Begin Backup Mode

```
# export ORACLE_SID=hrd10g
# sqlplus "/ as sysdba"
SQL> alter database begin backup;
```

Perform a Consistent Split Snapshot for Database Files

Note: When using SRFD/A, since the R2 is always 2 cycles behind the R1, in order for the begin hot backup mark in the data files to be included in the remote BCVs image (for backup and recovery sake) we use the SRDF *checkpoint* command that make sure the information on the R1 has reached the R2 before we issue the remote BCVs split.

```
# symrdf –sid 754 –rdfg 3 –file ./dev_srdf checkpoint

# symmir –g DBFILES_DG –rdf split –consistent
```

End Backup Mode

```
SQL> alter database end backup;
```

Switch Logs and Create Controlfiles

Create two copies of the control file. One copy (control_start) will be used to start up the database in mount mode on the backup server. The second copy (control_bakup) will be used as a component of the backup set used by RMAN.

```
SQL> alter system archive log current;
```

```
RMAN> run {
        Allocate channel ctl_file type disk;
        Copy current controlfile to
        '+RECOVERY_AREA/control_file/control_start';
        Copy current controlfile to
        '+RECOVERY_AREA/control_file/control_bakup';
        }
```

Resynchronize the RMAN Catalog

This adds the most recent archive log to the recovery catalog.

RMAN> resync catalog;

Perform a Consistent Split Snapshot of the Recovery Area to capture Archive Logs

```
# symrdf –sid 754 –rdfg 3 –file ./dev_srdf checkpoint
# symmir RECOV_DG –rdf split –consistent
```

BACKUP PROCEDURES:

On the backup host, once the BCVs are split, check that:

- The BCVs have Oracle permissions. Note that if the same host has both remote BCVs as well as R2 devices mapped to it then the R2 devices should not have Oracle permissions. The reason is that ASM writes disk group information to the ASM members. As both R2 and Remote BCV contains the exact same Oracle information ASM may confuse between them if they both have Oracle permissions.
- ASM init.ora file parameter ASM_DISKSTRING doesn't exclude the path to the BCVs.
- ASM init.ora file parameter ASM_DISKGROUPS contains the names of the disk groups.

Start ASM Instance

When the ASM instance is started, since the BCVs are included in the ASM_DISKSTRING parameter, it will identify them as the disk groups from production database. Also since the ASM_DISKGROUPS parameter contains the disk group names they will be mounted automatically.

```
# export ORACLE_SID=+ASM
# sqlplus "/ as sysdba"
SQL> startup
```

Mount Database Instance

Before the database is mounted change the target database instance init.ora CONTROL_FILE parameter to point to the copied controlfile. For example:

```
control_files  = +RECOVERY_AREA/control_file/control_start
#control_files = +DATA_AREA/control_001
```

```
# export ORACLE_SID=hrd10g
# sqlplus "/ as sysdba"
SQL> startup mount
```

Backup target Database Instance

Perform a rman backup on the backup host. The previously backed up control file must be part of the backupset because once the database in mounted the SCN will be updated and is no longer reflects the initial state of the control file.

```
RMAN> run { ALLOCATE CHANNEL t1 TYPE SBT_TAPE
        BACKUP FORMAT 'ctl %d/%s/%p/%t'
        CONTROLFILECOPY '+RECOVERY_AREA/control_file/control_bak';
        BACKUP
                FULL
                FORMAT 'ctl %d/%s/%p/%t'
                (database);
        BACKUP
                FORMAT 'al %d/%s/%p/%t'
```

```
        (archive all);
RELEASE CHANNEL t1
}
```

Note: The format specifier %d is replaced with date, `%t` is replaced with a four byte time stamp, `%s` with the backup set number, and `%p` with the backup piece number.

Case 5: Restoring a Database on the Production Host

If recovery time is critical, the recovery should be done on the production host. This method will provide minimal downtime, while protecting the *gold* copy (BCV) of the database. This case will restore the database from Case 1. This technique is very fast since only the changed tracks on the LUNs that make up the data files are restored. Only the additional archive logs since last backup time need to be applied for recovery. It is recommended that the restore process is done using –protect option. This ensures that any writes to the standard devices will not taint the BCVs. If corruption is reintroduced, or a mistake is made during the recovery procedure, the BCV can once again be used to perform a quick restore of the production host's database..

Shutdown the database instance on the production host

```
# export ORACLE_SID=hrd10g
# sqlplus "/ as sysdba"
SQL> shutdown [-immediate]
```

Dismount the ASM Datafiles diskgroup

SQL> alter diskgroup DATA_AREA dismount;

Note: In this case we only restore data files image to production host. We do not want to overwrite the online redo logs (if they still available on production) as they contain the last committed transactions. Also we do not want to overwrite the Flash Recovery Area, which contains recent archive logs. If Flash Recovery Areas was damaged as well then dismount and restore its ASM disk groups as well (if online redo logs are damaged on production then recreate their diskgroup. The logs will be recreated when the database is opened with resetlogs).

Perform a TimeFinder Restore on the datafiles diskgroup

The Recovery Area must not be restored to the production server

```
# symmir –g DBFILES_DG restore
```

Mount the ASM Diskgroups

```
# export ORACLE_SID=+ASM
# sqlplus "/ as sysdba"

SQL> alter diskgroup DATA_AREA mount;
```

Startup the Local Database in MOUNT

```
# export ORACLE_SID=hrd10g
# sqlplus "/ as sysdba"
SQL> startup mount
```

Perform complete or point-in-time recovery with RMAN.

If you are performing incomplete recovery, then set the "`until time`" or "`until SCN`" markers.

```
RMAN> run {
        SET UNTIL TIME '06-dec-05 13:00';
        RECOVER DATABASE;
        }
```

Opening the Database

After you are sure that all files are correctly restored and recovered, you can open the database using the `resetlogs` option. After you are sure that all files are correctly restored and recovered, you can open the database using the `resetlogs` option. The `open` with option `resetlogs` will create a new incarnation of the database, which must be also registered in the RMAN.

```
RMAN> alter database open resetlogs;
```