White Paper

Reducing Backup Window and Recovery Time with Oracle Database 11g RMAN and EMC TimeFinder/Clone

Applied Technology

Abstract

The need to reduce backup and recovery time is increasingly critical to business. The integration between EMC® TimeFinder®/Clone and Oracle RMAN technology significantly cuts down recovery time and shortens the backup window. This white paper demonstrates the usage of TimeFinder/Clone with the latest RMAN technology to optimally address emerging needs in backup and recovery strategies for large databases.

October 2008

Part Number H4262.1

Table of Contents

Executive summary **4**
Introduction **4**
Audience 5
Technology overview **5**
Symmetrix DMX-4 overview 5
EMC TimeFinder family 6
TimeFinder/Clone 6
TimeFinder Consistent Split 7
RMAN’s Block Change Tracking 7
Oracle Database 11g and TimeFinder integration **8**
Offload the backup with storage replication 8
Reducing RTO with parallel database recovery 8
Faster incremental backup with Block Change Tracking 9
Lab configuration and sample best practices **10**
Production and backup servers software stack 10
Logical Oracle data storage layout 11
Offload RMAN BCT-enabled backup with TimeFinder/Clone 11
On the production server 11
On the backup server 12
Restoring from offload backup using parallel database recovery 13
TimeFinder/Clone restore 13
RMAN recovery 14
Conclusion **15**
References **15**

Executive summary

Many enterprises are experiencing tremendous growth in their data centers, driving both the capacity and performance requirements of their mission-critical databases higher and higher. These critical large databases may soon overwhelm their current host-based backup mechanisms as business requirements often demand shorter backup window and recovery time objectives (RTO). In addition, running an online backup on a production host consumes additional CPU cycles and I/O bandwidth, which can lead to a performance degradation of the database.

These impending challenges can effectively be addressed by the integration of EMC® TimeFinder®/Clone with Oracle technology as follows:

Challenges	Solutions
Shrinking backup window	Offload backup operations from production using TimeFinder. When doing so, production is in backup mode only for a few minutes at the most, allowing TimeFinder to perform a split.
Shorter RTO	Integrate TimeFinder with RMAN. The integration enables parallel database recovery. TimeFinder restores the last full backup in seconds, allowing RMAN to recover additional committed transactions from that time forward.
Performance impact from running backup on production	Offload backup operations from production using TimeFinder. The backup is performed on another server utilizing different devices. RMAN Block Change Tracking works with offload backup and can considerably speed up incremental backups.

Offloading backup with TimeFinder reduces backup window and slashes recovery time. The database recovery can take place either on the production server or on the backup server. The latter is commonly used for surgical restore of tables. In addition, the TimeFinder database image on the backup server can be opened in read-only mode after the backup has completed for other purposes such as reporting.

Prior to the RMAN Block Change Tracking (BCT) feature, which was introduced in Oracle Database 10g, even RMAN incremental backups needed to scan all database blocks in order to find the changed blocks for the backup. When BCT is used, Oracle keeps tracks of these changed blocks in a BCT file. Thus, an incremental backup performed from either production or TimeFinder target devices makes use of the BCT file and directly reads only the changed blocks.

The combination of offloading backup operations with TimeFinder and using RMAN incremental backups with BCT is a great way to perform backup and restore operations with minimum impact to mission-critical databases while maintaining a very short RTO if a disaster occurred.

Introduction

This white paper demonstrates seamless integration of TimeFinder/Clone with the latest Oracle RMAN technology to optimally address emerging needs in backup and recovery strategies for large databases.

Traditionally, an online database backup is executed directly on the production server. This can adversely impact the performance of the database because the backup itself also competes for production's resources such as CPU, memory, and disk utilization. TimeFinder, when used with Oracle hot backup mode, can create a recoverable copy of the database on a different server for backup. This process, known as offloading, works with all types of Oracle backups in ASM and non-ASM environments as well as with the latest RMAN Block Change Tracking (BCT) feature in Oracle Database 10g and Database 11g. When

TimeFinder/Clone is used, a backup operation from the clone will be using the backup server's CPU and memory resources, and the TimeFinder target devices instead of production devices, freeing production resources to focus on serving transactions.

Integrating TimeFinder into a backup strategy also reduces RTO. A conventional database recovery comprises of two sequential phases. The first phase restores the needed bits of the last full backup back to production volumes. After the first phase finished, which normally can take a long time for large databases, the second phase initiates Oracle media recovery. With TimeFinder incremental restore, only changes between the source and target devices are copied back to production (source devices). In addition, the copy happens in the background, so as soon as the TimeFinder restore is initiated, the production server sees the final image of the target devices (the last full backup), and therefore almost eliminates the time of the first phase – regardless of the size of the database. The background operation will depend on the amount of changes since the last full backup. This allows the DBA to start Oracle media recovery (phase two) while TimeFinder is still restoring the data from clone to production volumes in the background, resulting in an extensive reduction in RTO, even for large databases. This integration is also known as parallel database recovery.

Audience

This white paper is for database administrators, storage administrators, and EMC personnel who want to understand the integration of TimeFinder/Clone with Oracle Database. The audience is expected to have familiarity with EMC Symmetrix® technologies, Oracle Database 10g Automatic Storage Management, Oracle Recovery Manager, and EMC TimeFinder/Clone.

Technology overview

Symmetrix DMX-4 overview

The new Symmetrix DMX-4 system is the next generation in the DMX series and extends EMC's leadership in the high-end enterprise storage market. The DMX-4 delivers immediate support for the latest generation of disk drive technologies: 4 Gb/s FC drives for high performance SATA II for high capacity and Flash SSD for ultra-fast tier 0 storage needs. The DMX-4 is based on Enginuity™ 5772 (and later), which has been optimized for maximum performance and tiered storage functional flexibility.

- 96 to 2,400 disks
- 2 to 12 channel directors*
- 2 to 8 4 Gb disk directors*
- Fibre Channel, iSCSI, Gigabit Ethernet, FICON, ESCON connectivity
- Up to 512 (256 usable) GB global memory

**Combinations may be limited or restricted based on configuration*

Figure 1. Symmetrix DMX-4: World's largest high-end storage array

- 32 to 360 disks
- 2 or 4 front-end/back-end directors
- FICON, Fibre Channel, iSCSI, Gigabit Ethernet connectivity
- Up to 128 (64 usable) GB global memory

**Combinations may be limited or restricted based on configuration*

Figure 2. Symmetrix DMX-4 950: Lower entry point for DMX-4 technology

EMC TimeFinder family

TimeFinder is a family of products consisting of TimeFinder/Clone, TimeFinder/Snap, and TimeFinder/Mirror. All three products can create instantly accessible read/write replicas between Symmetrix devices, where TimeFinder/Mirror and TimeFinder/Clone create a full-size replica, and TimeFinder/Snap creates a virtual replica that doesn't take any storage space initially, and only takes as much space as the application change rate since it was taken. While the solution described in this paper can be accomplished with TimeFinder/Mirror, TimeFinder/Clone or TimeFinder/Snap, this paper will focus on TimeFinder/Clone.

TimeFinder/Clone

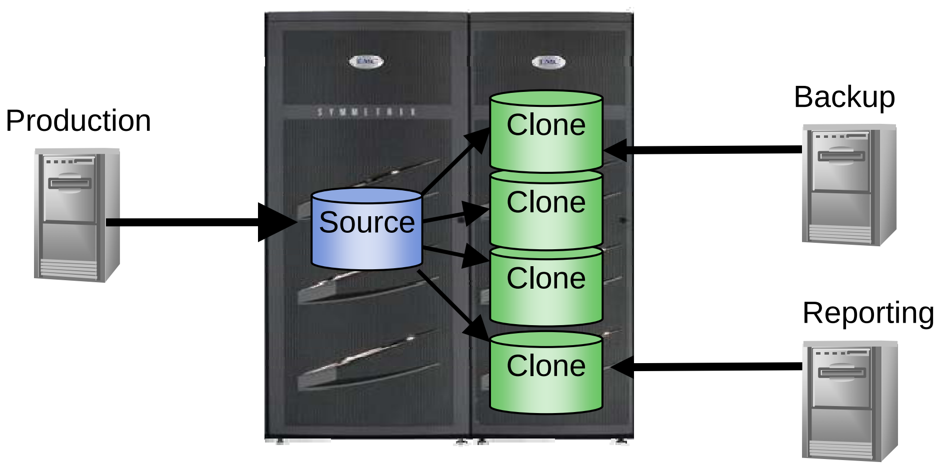


Figure 3. TimeFinder/Clone source and target device relationhip

Using TimeFinder/Clone, up to 16 simultaneous and immediately available full-size target copies (sessions) of source devices can be created. Both source and target devices can be any combination of standard or BCV Symmetrix devices and can have different RAID protection; however, the sizes of the source and target devices have to match. The relationship between a clone source and target is defined when a clone session is created. Once the session is later activated, the target device can be instantly accessed by a target's host. Clone target devices can be later refreshed from the source devices (re-activated) or can rewrite the source devices with their content (restored), or the clone session between source and target devices can be terminated.

A full copy will take place only during the first time a clone session is created. Any further operations on the clone, such as re-create or restore, will be done incrementally, passing just the final changes between the source and target devices, making TimeFinder/Clone a very fast and scalable database cloning solution, regardless of the database size. In addition, as soon as TimeFinder activates or restores a clone session, control is returned almost immediately to the user, although data copy may be still proceeding in the background. To a host accessing the devices, the data on them will seem as if the operation has already completed. These abilities make TimeFinder/Clone incredibly fast in creating or restoring database replicas of any size, and in fact, these replicas can include multiple databases, external data, and message queues, providing they are on Symmetrix devices.

TimeFinder/Clone is appropriate for repurposing in situations where multiple copies of production data are needed for testing, offloading backups, creating gold copies, generating reports, and more. When consistent split is used (as described in the next section), TimeFinder/Clone can create restartable and consistent images of multiple databases, external data, and message queues without the use of hot backup mode, which is ideal for cloning of service-oriented architectures (SOA) and databases with dependencies.

TimeFinder commands can be issued from any server connected to the Symmetrix with a few small devices (also called gate keepers). It is recommended that TimeFinder commands are issued from a management server rather than from production or backup servers. However, the commands in this paper are issued from the production server for demonstration purpose.

TimeFinder Consistent Split

TimeFinder software provides a consistent-split implementation of instant split that allows you to split off a consistent, DBMS-restartable image of your database without having to shut down the database or put the database into hot backup mode. It is able to do this by holding all dependent-write I/Os to database devices momentarily before splitting the target devices. After mounting the target devices to a host, a subsequent Oracle startup will perform instance crash recovery, ensuring the integrity of the database image. TimeFinder Consistent Split is tightly integrated with all products in the TimeFinder family.

Note that a point-in-time database image taken with a consistent split is not a valid Oracle backup unless the tablespaces were put into hot backup mode prior to the split. When Oracle ASM is used, consistent split provides an additional advantage by preserving ASM metadata consistency during a split operation. For that reason, when the database uses ASM, even when offloading backups with TimeFinder and the database is in hot backup mode, consistent split should always be used to preserve ASM metadata integrity during the split.

RMAN’s Block Change Tracking

RMAN's Block Change Tracking feature for incremental backups improves incremental backup performance by recording the changed blocks of each datafile in a change tracking file. When block change tracking is enabled, RMAN uses the change tracking file to identify changed blocks during incremental backup, thus avoiding the need to scan every block in the datafile. The end results are much faster incremental backups and reduced backup time. Note that the Block Change Tracking feature is disabled by default at this time for both Oracle Database 10g or Database 11g.

Oracle Database 11g and TimeFinder integration

Offload the backup with storage replication

Offloading backup with TimeFinder is a valid user-managed Oracle backup method. It allows the production server to utilize its resources for database transactions rather than spend CPU, memory, and I/O resources on running the backup operations. Offloading backup dramatically reduces the backup window requirement on production to just the amount of time it takes to put the database in hot backup mode and then quickly split off the mirrors. Therefore, offloading backup operations with TimeFinder increases scalability and maintains a high SLA of the production environment.

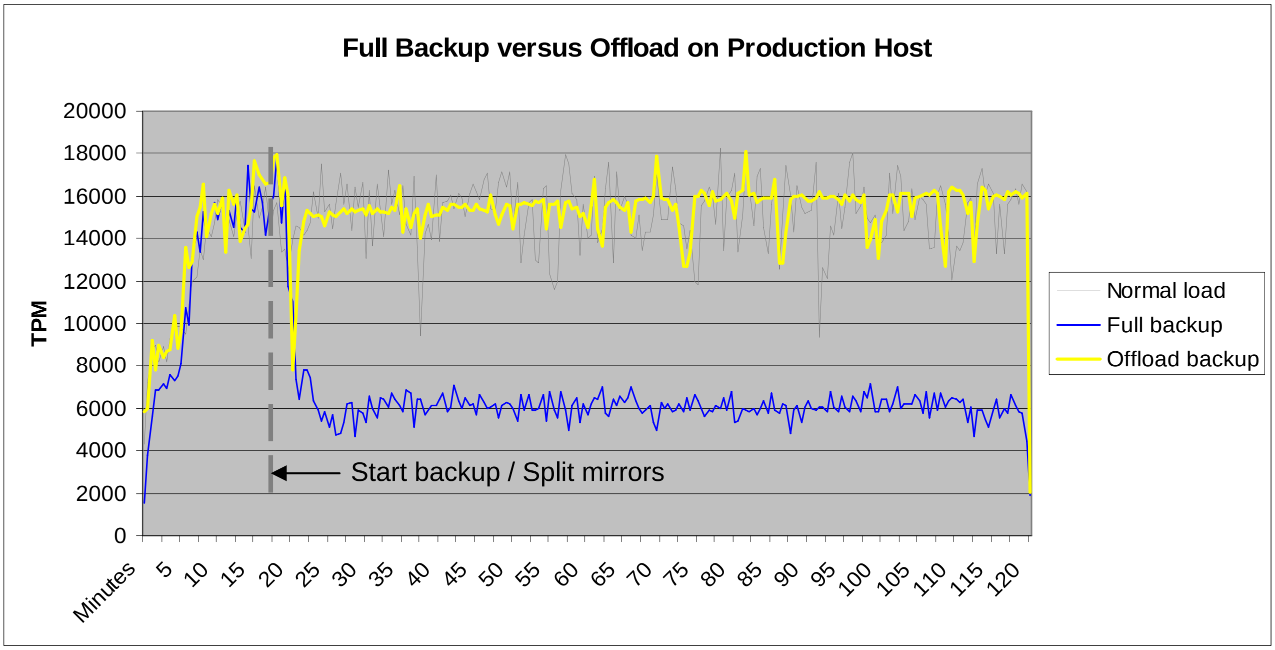


Figure 4. Full backup versus offload effect on a production workload

A full backup is a foundation of every backup strategy. Figure 4 shows an example of a traditional full backup operation performed directly on the primary database and how the operation adversely affects the transaction rate (in this example as much as 60 percent, down from 15,000 to 6,000). With offload backup, the transaction quickly rebounds after the mirrors were split. Both methods yield a full database backup at completion. However, a database with an integrated offload backup strategy is more poised to maintain its normal transaction rate and keeps the backup window very short, reaping the benefits of parallel database recovery, as described in the next section.

Reducing RTO with parallel database recovery

In a conventional database recovery, the DBA must wait for the initial database image restore from tape, disk, or the OS copy to complete, before they can apply additional archive logs and open the database. The time it takes to restore the initial image depends on the size of the database, and therefore it directly affects the RTO. TimeFinder technology eliminates that database size dependency as well as the need to wait for the initial restore, which could be a significant amount of time considering the current size and growth rate of databases. It achieves this by allowing the DBA to mount the database immediately after the TimeFinder restore operation has been initiated. The data from the TimeFinder target devices is immediately available on the source (production) devices. This allows the database to be mounted for applying additional archive logs while TimeFinder incremental restore continues simultaneously in the background.

When using TimeFinder to offload full and incremental backups from production, the TimeFinder replica always represents the last full backup image of production. If recovery is required, there is no need to first restore a full or incremental backup to it since it was in itself the last backup source. Therefore, applying archive or redo logs to it can start immediately, reducing RTO. Alternatively, by using the parallel database recovery concept, a TimeFinder restore operation needs only to start, and then the logs can be applied to the production volumes instead, still keeping the RTO very small. With this type of recovery strategy, even as database size increases, the RTO is only a factor of the amount of redo that needs to be applied and not of the database size.

Faster incremental backup with Block Change Tracking

Prior to Oracle Database 10g, RMAN used to scan all blocks in the datafiles, even when performing an incremental backup, resulting in increased backup window. With BCT enabled in Oracle Database 10g and later, Oracle tracks datafile blocks affected by each database update and stores tracking information in a Block Change Tracking (BCT) file. RMAN can then use this information to improve incremental backup performance by reading only those blocks known to have changed, instead of reading entire datafiles.

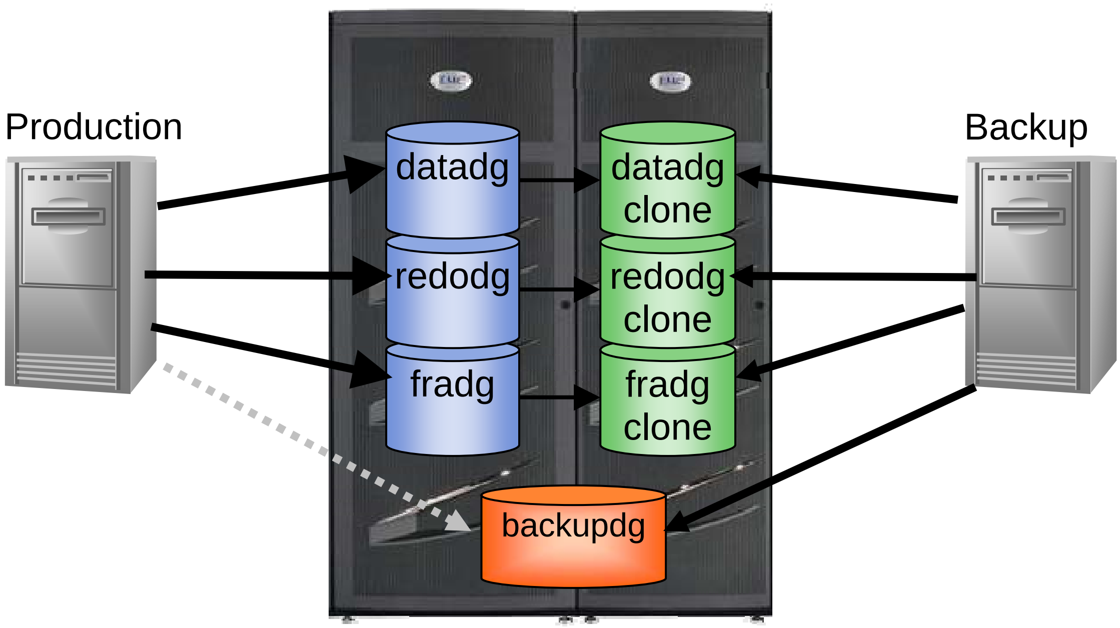
BCT does not change the amount of data that needs to be backed up and it does not have any effect on recovery operations. It only speeds up the incremental backup process. A BCT-enabled incremental backup strategy still relies on a periodic full backup that has to scan entire datafiles. RMAN will be able to leverage BCT for the following incremental backups.

Note that at the time this paper was written, the BCT file kept track of only seven incremental versions. If a full backup was not performed in between, the eighth incremental backup will defer to the old way of scanning all the blocks in the datafiles.

Offloading backup with TimeFinder/Clone is fully integrated with RMAN BCT. RMAN will able to utilize a copy of the BCT file to perform BCT-enabled incremental backups from the clone, just as if it was done from production.

For more information on RMAN Block Change Tracking, see the *Oracle Database Backup and Recovery User’s Guide 11g Release 1 (11.1)*.

Lab configuration and sample best practices



Symmetrix DMX

Figure 5. Hardware environment

Note that the examples shown are based on Oracle ASM, however, a similar strategy can be used when the database resides on file systems.

The production and backup servers use Symmetrix logical volumes that reside in the same DMX (although they can span multiple DMXs). Both servers have SAN connectivity to BACKUPDG ASM disks, but only one server had it mounted at any given time. Normally it was mounted by the backup server for keeping RMAN backups. However when an RMAN restore was performed directly to production, then the production server had the BACKUPDG ASM diskgroup mounted instead.

During normal backup operation, the production server mounted DATADG, REDODG, and FRADG ASM diskgroups. The backup host mounted the cloned volumes of these ASM diskgroups and also the BACKUPDG diskgroup as explained earlier.

The ASM_DISKGROUPS parameter in the init.ora file for both ASM instances on production and backup servers did not contain BACKUPDG to allow it to be mounted to the correct server as required. For example, the ASM_DISKGROUPS parameter used was as follows:

```
*.ASM_DISKGROUPS='datadg','redodg','fradg'
```

Production and backup servers software stack

- Sun AMD64 server: 4 x 2 GHz CPUs, 4 GB RAM
- Operating system: Solaris 10 64-bit
- HBAs: 2 x LP11002-E

- EMC PowerPath® 5.0.0
- Oracle Database 10g Enterprise Edition Release 10.2.0.3.0
- Symmetrix DMX-3 with Enginuity 5772 microcode
- EMC Solutions Enabler 6.4.1.0

Logical Oracle data storage layout

ASM diskgroup	Logical volumes (LUNs)	RAID	Oracle attributes
DATADG	20 X 32 GB	RAID 1	Datafiles
REDODG	4 X 32 GB	RAID 1	Online redo log files
FRADG	5 X 32 GB	RAID 5	Flash recovery area
BACKUPDG	43 X 32 GB	RAID 5	Backup to disk

Offload RMAN BCT-enabled backup with TimeFinder/Clone

This procedure creates a copy of a BCT-enabled database on the backup server for a RMAN backup to the BACKUPDG diskgroup. The process takes place in two parts. The first part creates a TimeFinder/Clone valid full backup image while production is in Oracle hot backup mode. The second part takes place on the backup server where RMAN backs up the database replica to the BACKUPDG ASM diskgroup (backup to disk). Alternatively other backup methods could have been used from the replica, such as RMAN backup to tape and EMC Disk Library (EDL).

On the production server

1. Create a clone session.

   ```
   # symclone -sid 571 -file alldev.txt create -precopy -diff  -nop
   ```

2. Archive the current log file, which automatically triggers a log switch and a database checkpoint.

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

3. Put the database in hot backup mode.

   ```
   SQL>alter database begin backup;
   ```

4. Activate the clone session for the datafiles in the DATADG diskgroup, with the `–consistent` option.

   ```
   # symclone -sid 571 -file datadev.txt activate -consistent –nop
   ```

5. End hot backup mode.

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

6. Archive the current log file.

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

7. Manually switch the bitmap in the change tracking file.

```
SQL> execute dbms_backup_restore.bctswitch();
```

8. Create two copies of the backup controlfile. The first one, control_start, will be used to start up the database for RMAN backup. The other one, control_backup, will be stored along with the backup set.

```
SQL>alter database backup controlfile to '+FRADG/control_start';

SQL>alter database backup controlfile to '+FRADG/control_backup';
```

9. Connect to the recovery catalog and resynchronize the catalog. A catalog resynchronization propagates the data from the controlfile to the recovery catalog. This assumes the catalog had been created and the database had been registered.

```
# rman target / catalog rman/rman@catdb
RMAN> resync catalog;

RMAN> exit;
```

10. Activate the clone session for the archive logs (in this example the archive logs were stored in Oracle FRA). The FRADG diskgroup also contained the two copies of the backup controlfile and the BCT file. After the clone operation they will all be available to the backup server.

```
#symclone -sid 571 -file archivedev.txt activate -consistent –nop
```

At this point, the TimeFinder/Clone volumes contain a full database backup. If there is an emergency need for database restore, TimeFinder can incrementally restore the entire database.

On the backup server

1. Start up the ASM instance.

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

2. Change the control_file parameter in the database instance init.ora file to use the backup control file.

```
control_files=+FRADG/control_start
```

3. Start up and mount the database. Do not open it.

```
$ export ORACLE_SID=ORABCT
$ sqlplus / as sysdba
SQL> startup mount
```

4. Connect to RMAN.

```
$ rman target / catalog rman/rman@catdb
Recovery Manager: Release 10.2.0.3.0 - Production on Fri Oct 26
15:45:30 2007
Copyright (c) 1982, 2005, Oracle.  All rights reserved.

connected to target database: ORABCT (DBID=3136004487, not open)
connected to recovery catalog database

RMAN>
```

5. Perform a full backup for the first time.

```
RMAN> run
{ allocate channel dev1 type disk;
  allocate channel dev2 type disk;
  backup format '+BACKUPDG/ctl%d%s%p%t' controlfilecopy
  '+FRADG/control_backup';
  backup incremental level=0 database format '+BACKUPDG' tag
  'level0_backup';
  release channel dev1;
  release channel dev2;
}
```

For subsequent incremental backups, change the backup level to 1 (incremental since the last full backup).

```
   RMAN> run
   { allocate channel dev1 type disk;
     allocate channel dev2 type disk;
     backup format '+BACKUPDG/ctl%d%s%p%t' controlfilecopy
     '+FRADG/control_backup';
     backup incremental level=1 database format '+BACKUPDG' tag
     'level1_backup';
     release channel dev1;
     release channel dev2;
}
```

6. Verify that change tracking took place.

```
SQL> select count(*) from v$backup_datafile where
used_change_tracking='YES';

COUNT(*)
----------
 103
```

Restoring from offload backup using parallel database recovery

TimeFinder/Clone restore

This method leverages TimeFinder incremental and protected restore to perform the operation. The database and ASM instance on both servers should be shut down prior to initiating the TimeFinder restore command.

Remember that the DBA can immediately bring the production environment online as soon as the TimeFinder restore command has successfully been issued. There is no need to wait for the background restore to complete.

New write I/Os to the production volume will not be propagated to the clone volume while a TimeFinder/Clone restore is in progress. This feature, protected restore, is built in to all TimeFinder family products to preserve the good copy on the clone should the restore be interrupted for any reason.

On the backup server

1. Shut down the database instance.

```
$ export ORACLE_SID=ORABCT
$ sqlplus / as sysdba
```

```
$ shutdown immediate
```

2. Shut down the ASM instance.

```
$ export ORACLE_SID=+ASM
$ sqlplus / as sysdba
SQL> shutdown immediate
```

On the production server

1. Shut down the database instance.

```
$ export ORACLE_SID=ORABCT
$ sqlplus / as sysdba
$ shutdown immediate
```

2. Shut down the ASM instance.

```
$ export ORACLE_SID=+ASM
$ sqlplus / as sysdba
SQL> shutdown immediate
```

3. Start a TimeFinder/Clone incremental restore for the datadg diskgroup.

```
# symclone -sid 571 -file datadev.txt restore
```

4. After the background TimeFinder restore finished, issue the split command on the session. This is done so that subsequent operations such as establish, re-create, or another restore will be performed in an incremental manner.

```
# symclone -sid 571 -file datadev.txt split
```

RMAN recovery

In order for RMAN to perform a restore directly to production, the DBA must dismount BACKUPDG from the backup server and mount it on the production server.

On the backup server

1. Dismount the BACKUPDG diskgroup (if still mounted).

```
$ export ORACLE_SID=+ASM
$ sqlplus / as sysdba
SQL> alter diskgroup backupdg dismount;
```

On the production server

1. Bring up ASM.

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

2. Mount the BACKUPDG diskgroup.

```
$ export ORACLE_SID=+ASM
$ sqlplus / as sysdba
SQL> alter diskgroup backupdg mount;
```

3. Mount the database.

```
$ export ORACLE_SID=ORABCT
$ sqlplus / as sysdba
SQL> startup mount
```

4. Start a RMAN restore procedure as needed.

```
$ export ORACLE_SID=ORABCT
$ rman target / catalog rman/rman@catdb
RMAN>recover database;
```

Conclusion

The need to reduce recovery time and improve backup speed will undoubtedly play a very important role in any IT organization as databases continue to grow and become strategically involved in critical decision-making and business operations.

The integration between EMC TimeFinder/Clone and Oracle RMAN technology significantly cuts down recovery time and shortens the backup window. At the same time, such an integration also maintains the performance service level agreement (SLA) of the production environment and enhances the level of protection for a traditional database backup and recovery strategy.

References

- *EMC TimeFinder and SRDF for Oracle Database 10g Automatic Storage Management - Best Practices Planning* white paper

- *Validating EMC TimeFinder/Mirror with Oracle8i, Oracle9i, and Oracle 10g* white paper

- *Oracle 10g and 11g Documentation Library*