

GR740

Remote Equivalent Copy

Disaster Recovery Guideline
for Oracle Database

November 22, 2001

| | |
|--|----|
| INTRODUCTION | 3 |
| REMOTE MIRRORING FOR DISASTER RECOVERY | 4 |
| OVERVIEW OF GR740 REMOTE EQUIVALENT COPY..... | 5 |
| SYNCHRONOUS AND ASYNCHRONOUS MODE..... | 5 |
| PRESERVATION OF WRITE ORDERING | 5 |
| ATOMIC BREAK OF MIRRORS | 5 |
| VALIDITY OF STANDBY DATABASE..... | 6 |
| SUSPEND AND RESUME | 6 |
| COMMANDS LINE INTERFACE TO CONTROL REC | 7 |
| OSCP TEST CONFIGURATION | 9 |
| MIRRORING ONLINE LOGS FOR ORACLE STANDBY DATABASE..... | 11 |
| INITIAL CREATION OF STANDBY DATABASE..... | 11 |
| PRIMARY SITE TO STANDBY SITE FAIL-OVER | 11 |
| STANDBY SITE TO PRIMARY SITE FALL-BACK..... | 12 |
| MIRRORING THE WHOLE DATABASE | 17 |
| PRIMARY TO SECONDARY FAIL-OVER | 17 |
| SECONDARY TO PRIMARY FALL-BACK..... | 18 |
| SUMMARY..... | 19 |
| APPENDIX EXAMPLE OF COMMAND LINE INTERFACE | 20 |
| FOR REDO LOG MIRRORING..... | 20 |
| FOR WHOLE DATABASE MIRRORING | 21 |

Introduction

GR740 is a Fujitsu RAID storage system for Open Systems with *Remote Advanced Copy* features. These features consist of *Remote Equivalent Copy* (REC) for remote mirroring and *Remote One Point Copy* (ROPC) for remote snapshot. Remote Equivalent Copy (REC) is the capability to mirror a whole volume or a part of a volume across Fibre Channel links and transfer data from a primary box to a secondary box without any host CPU usage. The Fibre Channel link can be extended more than 10km by an extender box that can transfer data via WAN or MAN connection (for example ATM). Remote One Point Copy (ROPC) is the capability to make a replication of a whole volume or a part of a volume across Fibre Channel links.

Oracle offers the Oracle Storage Compatibility Program (OSCP) to assist third parties in confirming the compatibility with the Oracle database environment. There are several areas to be confirmed by OSCP. The remote mirroring area using Remote Equivalent Copy (REC) is addressed in this document. The other area of OSCP and Remote One Point Copy (ROPC) is not included in this document.

As part of the Oracle Storage Compatibility Program (OSCP), Oracle provides a test suit for third parties to perform self-test to confirm compatibility for remote mirroring. This test suit confirms the compatibility to recover database for disaster recovery in various scenarios. Based on these scenarios and the reference document issued from Oracle, this guideline describes how to recover an Oracle database using the GR740 Remote Equivalent Copy (REC) feature.

Within this document, the term "volume" is used to denote the storage space defined within the GR740, which is seen as a disk drive by the typical Operating System. It may be defined within the GR740 to include all or part of a RAID Group. A RAID Group may be composed of several disk drives organized in one of the standard protection schemes, such as RAID1, RAID0+1, or RAID5.

Terminology

The descriptions in following sections use the following terms:

- *Primary site* refers to the main site for production database. This does not change during fail-over.
- *Standby site* refers to the initial site for standby database. This also does not change during fail-over.
- *Primary database* refers to the database handling user requests in a production environment. This may also be called the *production database*. The primary database may be on either primary or standby site. For example, after fail over, the standby site may run the primary database.
- *Standby database* refers to the Oracle standby database. This may also change sites. For example, after a role reversal, the primary site may run the standby database.
- *Primary volume* refers to the source volume (or area) of mirroring. The write data to *Primary volume* is reflected to *Secondary volume*. It might be called the *Primary GR740*.
- *Secondary volume* refers to the target volume (or area) of mirroring. *Secondary volume* keeps the image of *Primary volume*. It might be called the *Secondary GR740*.

References

- Guidelines for Using Remote Mirroring Storage Systems for Oracle Database, J. Bill Lee, Nov. 1999

Remote Mirroring for Disaster Recovery

There are two ways to use remote mirroring for disaster recovery with an Oracle database;

- Remotely mirror online logs for an Oracle Standby database. If a disaster happens to the primary database, one can use the mirrored online redo logs to recover changes in the current logs.
- Remotely mirror the whole database including data files and log files. If a disaster happens to the primary database, one can quickly fail over to remote site.

Depending on the mode of remote mirroring, both approaches can guarantee either no loss of committed data (synchronous mode) or reduced loss of committed data (asynchronous mode). When selecting a disaster recovery strategy using the remote mirroring feature, one will have to pick either mirroring the whole database or mirroring just the online log for the standby database, The choice may depend on the remote mirroring storage system, mirroring distance, fail-over time limit, and Oracle versions;

- *Performance.* Mirroring the whole database typically has a much bigger performance impact to the primary database, especially as the mirroring distance increases. This characteristic alone may make mirroring online logs the only choice when one has to mirror over a long distance, such as hundreds or thousands of miles,
- *Expense.* Mirroring the whole database requires a larger communication bandwidth between local and remote storage systems. A larger pipeline means more cost for the overall system.
- *Availability.* Mirroring online logs provides better availability characteristics. If you mirror the whole database, any corruption to the production database will propagate to the secondary database, whether the corruption is caused by the database, OS, or storage systems. Standby database can protect you from a wide variety of these failures by checking consistency during redo operation. Standby database can also be run with a delay to protect the database from user errors.
- *Fail-over time.* Fail-over is faster when mirroring the whole database. When mirroring the whole database, the fail over only needs to wait for crash recovery to complete at the secondary site. When mirroring online log for the standby database, the fail over needs to wait for all logs not yet archived to be applied.
- *Fall-back time.* When mirroring online logs for standby database, fall-back is more flexible. It does not require copying the database across network. When mirroring the whole database, fall-back can be time consuming because it requires copying the database from secondary site to primary site. This is OK if the underlying remote mirroring system supports resynchronization, so that only data changed is copied. Otherwise the time it takes to copy the database may be too long to be practical for a large database.
- *Simplicity.* The procedure for mirroring the whole database is simpler. Note that maintaining a standby database is made simpler by the Oracle81 Managed Standby Database Recovery feature.

Overview of GR740 Remote Equivalent Copy

Remote Equivalent Copy (REC) is the remote mirroring feature to mirror a whole volume or part of a volume across Fibre Channel. To start the mirroring, one makes a pair of the area within the primary GR740 and the secondary GR740. One pair composes one *session* that is the unit of remote mirroring control. One session has an operating mode, synchronous or asynchronous, and an internal attribute and state. The mirrored area must be defined as one continuous area within one volume.

Synchronous and Asynchronous Mode

In synchronous mode, the primary GR740 acknowledges a write command to the host after storing the write data to both primary GR740 and secondary GR740 cache memories. In Asynchronous mode, the primary GR740 acknowledges a write command to the host after storing to the primary cache. The copy to the secondary volume would be performed later in background.

Asynchronous mode allows the database to continue the activity during a link failure or when the secondary volume is down, but any amount of data may be lost when a disaster occurs. In synchronous mode, the data loss never happens, because primary database activity would be prevented by problems like a link failure or the secondary volume down.

Synchronous mode performance is affected with the distance between primary site and secondary site. Asynchronous mode is not.

Remote Equivalent Copy (REC) feature supports both modes. One can select it for each session when starting the mirroring operation.

Preservation of Write Ordering

Write Ordering is important for Oracle databases. The recovery operation by Oracle is performed based on Write Ordering, which would keep the secondary volume to be the image of the primary volume at any time.

The synchronous mode preserves Write Ordering by nature, since copy operations to the secondary GR740 are synchronous with a write from a host.

In asynchronous mode, the primary GR740 sends the write data with a sequence number. The secondary GR740 applies the write data in turn, re-ordering the write data by its sequence number to keep Write Ordering because there is no insurance that the secondary GR740 would receive the data in order through WAN and it might be lost in the network. The sequence number is also used for Atomic Break of Mirrors.

The sequence number for asynchronous mode is assigned within one session, in which the write ordering is preserved. One session can have one mirrored area pair only. It means that the write ordering in asynchronous mode is preserved within one volume, not across volumes.

Atomic Break of Mirrors

Atomic break of mirrors means to stop the reflection of write data to the secondary volume at a point in time. After atomic break of mirrors, the secondary volume must be a valid image of primary volume. Breaking the mirrors may be caused by specific command to control REC or by a link failure between GR740s. REC satisfies atomic break of mirrors for both modes, synchronous and asynchronous, in any cases.

It is important that the standby database be valid at a break. See Validity of Standby Database.

Validity of Standby Database

The standby database is not *valid* during the synchronizing operation because the secondary volume is not the image of the primary volume at any time. It is in either of the following cases ;

- *Synchronizing just after start mirroring.* When mirroring is started, the REC system assumes that whole area to be mirrored is different between the primary volume and the secondary volume, and tries to make both volumes become *synchronized* as soon as possible. Until synchronized, the secondary volume has inconsistent data and the data transfer is performed by a method most suitable for performance, but not preserving Write Ordering.
- *Re-synchronizing after break of mirrors.* When resuming mirroring again after a break of mirrors, REC performs the re-synchronizing task without preserving Write Ordering. In ordinal cases, the re-synchronized portion contains only the blocks marked as dirty. Re-synchronizing will start after link recovery in asynchronous mode or RESUME command after SUSPEND command.

A Query command is provided to get the status of a mirroring session. One can confirm if synchronization is being performed by using the Query command, and can know the validity of the standby database. An invalid standby database can not be used for recovery. One must be careful that the link is in a stable state when confirming the validity with the Query command after a disaster, because an unexpected link recovery may start re-synchronization unexpectedly.

Figure 1 Validity of Standby Database shows when the standby database is valid. A break of mirrors is atomic, so the validity remains unchanged after a break. While re-synchronizing, the update to secondary volume is performed without preserving Write Ordering. The standby database is invalid while re-synchronizing.

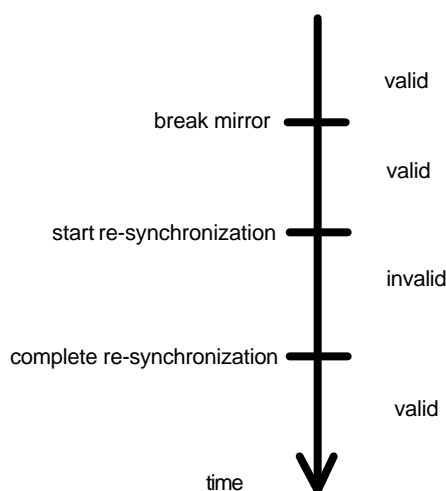


Figure 1 Validity of Standby Database

Suspend and Resume

REC supports Suspend and Resume functions. Suspend breaks the mirror, and Resume re-establishes the mirroring. During suspended state, the REC system keeps track updates to primary and secondary volumes, and marks the dirty blocks. The Resume operation performs the re-synchronizing operation for the marked blocks. The update to the secondary volume may be nullified or remain according to user options specified for RESUME command. This re-synchronization does not preserve Write Ordering. The standby database becomes invalid after Resume until re-synchronization is complete.

Commands Line Interface to Control REC

This section describes the command line interface for the commands to Start, Stop and Query REC on the host running Solaris operating system. This interface provides a very low level interface and is not be intended to be used by most customers. The customer should use another tool with GUI supplied from Fujitsu for GR740. The several examples of the command line interface are described in **Appendix Example of Command Line Interface** page20. These examples are referenced in the following sections to show the senario of fail-over and fall-back.

The basic operations for the REC feature consist of Start, Stop(Cancel), Query, and others. The interface routine transforms the request into one or more vendor specific commands, and sends it to GR740. Each command specifies the mirrored area by BoxID, LUN, start address and size. The commands which have the same value in those parameters are assumed to be specifying the same mirroring session. The Stop and Query commands specify the activated mirroring session by specifying the same parameters as the respective Start command. If the specified area is only a part of the mirroring area activated by a Start command, the Stop/Query command will fail.

```
STXCopyEC START/CANCEL/QUERY <Volume name>  
    <Source BoxID><Source LUN><Source LBA><Size>  
    <Taget BoxID><Target LUN><Target LBA><Copy Interval><Option flag>
```

The hex form number value is represented by the 'h' character following the number (Ex. 20h). Otherwise the number is assumed as decimal.

BoxID is the unique ID assigned to each GR740 device before shipment. It is never changed.

The command can be issued from either of the two sites and will cause the same effect if the links are active, and the parameters are the same except for the respective Volume name. The Volume name would be different at the primary and secondary sites.

Volume name

GR740 device name which should receive this command.

This volume name is prepared only for the Operating System to identify where to send the command. Any LUN and slice number can be used in the host system Volume name.

Source BoxID / Source LUN / Source LBA – source area

Source area to be mirrored. It is also called the Primary volume in this document. REC defines the mirrored area by BoxID, LUN, LBA and length. The area length is defined by the *Size* parameter.

Size

Length in blocks. This parameter is shared by both source and target areas.

Target BoxID / Taget LUN / Target LBA – target area

Target area to be mirrored. It is also called the Secondary volume in this document. The area length is specified by the *Size* parameter.

Copy Interval

Byte parameter for performance tuning. This parameter is not within the scope of this document.

Option flag

Byte parameter. The value 20h specifies Synchronous mode, valid for START command. A value of zero means Asynchronous mode.

The value 02h specifies the *Force* flag, valid for STOP command. The NoForce STOP command will complete successfully only when the link is active and both the primary and secondary volumes are *synchronized*. If mirroring is stopped successfully by the NoForce STOP command, then both volumes are insured to be identical at the break in both synchronous and asynchronous mode.

However one can not expect the NoForce STOP command to complete successfully in case of a disaster, because the link would very likely be down. When mirroring is terminated by Force STOP command with a link failure, the session is cleared only at GR740 that received the command and remains active at another. One must also issue the Force STOP command at the other site to clear it as well.

NOTE

The BoxID has to be specified in hex dump form, not ASCII form in the command line interface. The primary BoxID in our example must be specified as follows;

```
3030475237343023232323232323475237344230312323232323434135372323232323232323
```

This is not easily readable. The ASCII form will be used when showing examples in this document.

OSCP Test Configuration

Fujitsu GR740 storage system passed the OSCP test suite for remote mirroring. Here is the configuration used with the test suite. The recovery steps described later assume this same test configuration.

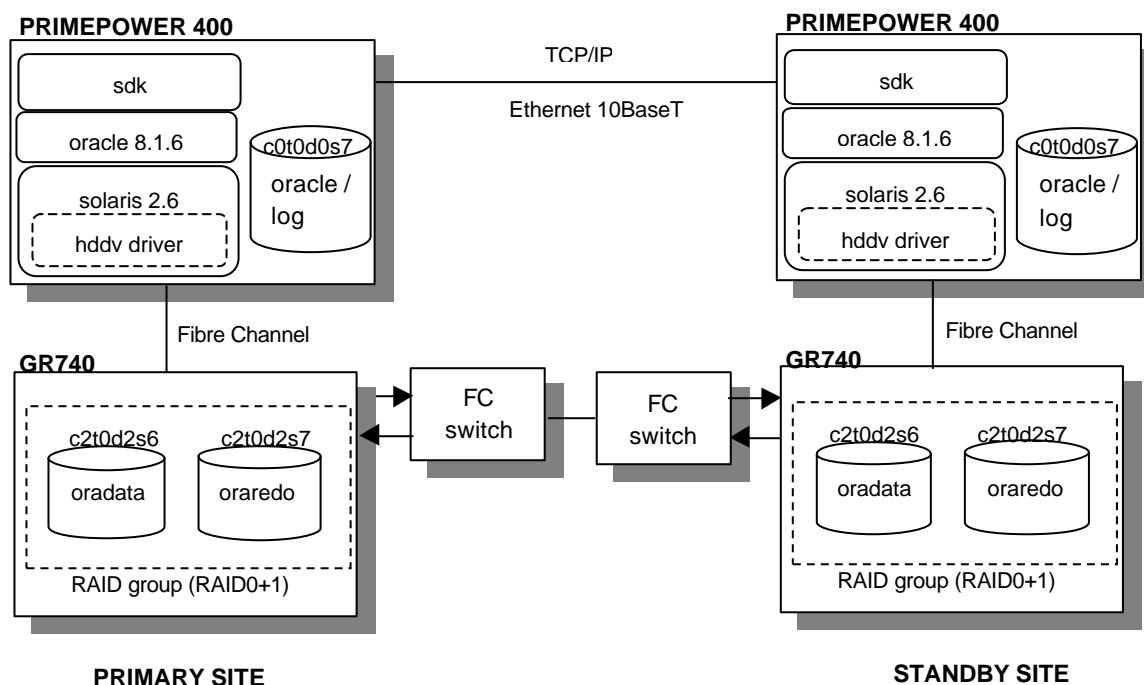


Figure 2 Test Configuration

The primary site and standby site are identical to each other in all elements, Hardware, Software, and volume configuration.

Hardware

The host machine is a Fujitsu PRIMEPOWER running solaris 2.6. The model used here has four SPARC64 CPUs. The hosts are connected by Ethernet.

Two GR740s with the Remote Advanced Copy optional feature are connected by two Fibre Channel links. There are two FC switches on this path.

Software

Solaris 2.6 and Oracle 8.1.6 are used. The *hddv* driver is used as disk driver to access GR740. This driver is specialized to access GR740 and supplied to GR740 customers from Fujitsu.

The *sdk* is the test suit from Oracle.

Volume configuration in GR740

Two slices are defined within one Logical Unit. One slice is for Oracle datafiles, the other is for the redo log files. Each slice makes up one volume on ufs. The other files except Oracle data files and redo log files are stored on the internal disks in the host machines.

| Slice for | | Value | |
|-----------|-----------------------------|---------|--------|
| | | (dec) | (hex) |
| datafile | Slice number | 6 | 6 |
| | LBA (Logical Block Address) | 524288 | 80000 |
| | Size (in blocks) | 2097152 | 200000 |
| redo log | Slice number | 7 | 7 |
| | LBA (Logical Block Address) | 2621440 | 280000 |
| | Size (in blocks) | 262144 | 40000 |

The datafile volume and redo log volume are prepared in one logical unit to enable mirroring the whole database while preserving Write Ordering. If the redo log only is mirrored, then two logical units may be used, one for each volume in another practical configuration.

File System

| Site | Host name | Volume | Mount point | Note | File system |
|--------------|-----------|-------------------|----------------|------------------------|-------------|
| Primary site | ora1 | /dev/dsk/c0t0d0s7 | /ora01 | oracle program and sdk | ufs |
| | | /dev/dsk/c2t0d2s6 | /ora04/oradata | datafile | |
| | | /dev/dsk/c2t0d2s7 | /ora04/oraredo | online redo log | |
| Standby site | ora2 | /dev/dsk/c0t0d0s7 | /ora01 | oracle program and sdk | ufs |
| | | /dev/dsk/c2t0d2s6 | /ora04/oradata | datafile | |
| | | /dev/dsk/c2t0d2s7 | /ora04/oraredo | online redo log | |

Configuration Data

The following is the mirroring configuration data to mirror only the online redo log volume.

| Site | Type | Value |
|--------------|----------------------------|-----------------------------------|
| Primary site | BoxID | 00GR740#####GR74B01#####CA57##### |
| | Volume name | /dev/dsk/c2t0d2s7 |
| | LUN(Logical Unit Number) | 2 |
| | LBA(Logical Block Address) | 2621440 (280000h) |
| | Size (in blocks) | 262144 (40000h) |
| Standby site | BoxID | 00GR740#####GR74B01#####BP56##### |
| | Volume name | /dev/dsk/c2t0d2s7 |
| | LUN(Logical Unit Number) | 2 |
| | LBA(Logical Block Address) | 2621440 (280000h) |
| | Size (in blocks) | 262144 (40000h) |

The following is the mirroring configuration data to mirror the whole database

| Site | Type | Value |
|--------------|----------------------------|-----------------------------------|
| Primary site | BoxID | 00GR740#####GR74B01#####CA57##### |
| | Volume name | /dev/dsk/c2t0d2s6 |
| | LUN(Logical Unit Number) | 2 |
| | LBA(Logical Block Address) | 524288 (80000h) |
| | Size (in blocks) | 2359296 (240000h) |
| Standby site | BoxID | 00GR740#####GR74B01#####BP56##### |
| | Volume name | /dev/dsk/c2t0d2s6 |
| | LUN(Logical Unit Number) | 2 |
| | LBA(Logical Block Address) | 524288 (80000h) |
| | Size (in blocks) | 2359296 (240000h) |

Mirroring Online Logs for Oracle Standby Database

The following sections describe how to fail-over and fall-back the Oracle database when mirroring only the online logs. Those descriptions can be applied to both of synchronous mode and asynchronous mode, unless a specific issue is indicated.

Disaster recovery by the mirroring the online logs requires coordination of the Oracle Standby database feature and the GR740 Remote Equivalent Copy feature. When disaster occurs at the primary site, one can apply the mirrored online logs to the standby database to recover the last bit of changes to the database. If online logs are not mirrored, the recovery can be performed only with the archive logs received by the standby database, in which case the last change just before the disaster would be lost. Even if the online logs are mirrored, a some changes may still be lost in asynchronous mode. The no-lost data recovery can be achieved with the synchronous mode, but one has to accept the performance impact of operating in synchronous mode. It may be not acceptable, especially in the cases where the distance between sites is large.

Initial Creation of Standby Database

To create the standby database, you can use standard procedures. After creating the standby database and unmounting the redo log volume at standby site, REC for mirroring for redo log volume must be started. The redo log volume has to be unmounted prior to starting mirroring because the ordinal filesystem, like ufs, performs various caching for mounted volumes. If volume is changed by mirroring while still mounted, the volume will be corrupted.

For the start mirroring command, See **Start PtoS command** at Page20.

Primary Site to Standby Site Fail-Over

When a disaster happens at the primary site, follow the procedures below to open the standby database.

| Steps to fail over | Action on primary site | Action on standby site |
|--|---|---|
| Construct a "create control file script" when changing database structures | Issue "alter database backup control file to trace" whenever you change online redo log structures or datafile structures. Copy the resulting "create controlfile script" to the standby database site. | |
| Disaster on primary site | Crash or Major Fault | |
| Query REC | | Query PtoS command at page 20 Confirm secondary volume is valid. If invalid, don't go further. Before Query, Confirm the link is disabled. |
| Stop REC | | Stop PtoS force command at page 20 |
| Mount redo log volume | | Mount redo log volume after checking consistency by fsck command. |
| Prepare to activate standby database | | Prepare a new init.ora file for activating standby database as primary. This init.ora should use a different location for the control file. Modify the "create control file script" so that all datafiles point to the datafiles of the standby database, and all log files point to mirrored online logs. Make sure that the "create control file command" uses the noresetlogs option. |
| Apply the current log | | Startup nomount and create control file Issue "recover database". If recovery asks for more logs, supply the current logs for recovery. |
| Activate Standby | | Issue "alter database open" |

Standby Site to Primary Site Fall-Back

After fail-over, there are two methods to switch back to the primary site. One method is *reverse role* – Continue to run the production database at the secondary site, and start to maintain a standby database at the primary site. Another method is *fall back directly* – Fall-back to run the production database at the primary site.

Reverse Role via DB Copy

One procedure to implement *reverse role* is to do a complete creation of the standby database. This involves copying all the datafiles from the standby site to the primary site. This means of reverse role can be used in any situation.

Reverse Role via Restoring Backup

Another procedure avoids copying the datafiles across the network. Instead, the user must restore a local full backup and recover the database at the primary site.

| Steps to Reverse Role via Restoring Backup | Action on primary site | Action on standby site |
|--|---|---|
| Periodic full backup | Make a full cold backup or hot backup of the database | |
| Disaster | | |
| Activate standby | | Fail over to standby site (as described above) |
| Stop REC at primary | Stop PtoS force command at page 20 The copy session would still remain active at the primary. You have to clear it explicitly. | |
| Restore backup and prepare for recovery | Restore the full backup of the database. Make sure that the backup is taken before the disaster. Pay more attention if either logs are mirrored asynchronously or the mirror is broken at the time of disaster. In these cases, the backup must have been taken before the primary database generated the last archive log shipped to the standby site. When in doubt, always use an older full backup. Copy all the archive logs from standby site since the backup was taken. These should be the only archive logs used for recovery. | |
| Prepare to create standby database | Create an init.ora file for the new standby database to be created | Issue "alter database create standby controlfile" |
| Setup sqlnet | Setup sqlnet connection | Setup sqlnet connection |
| Unmount redo log volume | Unmount redo log volume to be target of mirroring | |
| Reverse mirroring | Start StoP command at page 20 Start REC mirroring from the Standby site to the Primary site. It can be issued from either site. | |
| Create standby | Copy the standby control file from the standby site Use name convert init.ora parameters or rename command to let database point to the correct datafiles and log files Mount and recover the standby database, using only the archive logs copied from standby site. (See footnote ¹) | |

¹ For archive logs generated before the disaster, you may use the copy that is already on the primary site. However, you must make sure that these logs have been shipped to the standby site and applied to the standby database before the standby database is activated. All archive logs after the disaster must come from the standby site.

Reverse Role via Recovery

Yet another procedure of reverse role requires neither copying the database across the network nor restoring backups. One can only use it, however, under the following 3 conditions:

- the online logs are mirrored synchronously, and
- the mirror is established when the disaster happened, and
- no one has started the database at the primary site after the disaster.

| Steps to Reverse Role via Recovery | Action on primary site | Action on standby site |
|--|--|---|
| Disaster | | |
| Activate standby | | Fail over to standby site (as described above) |
| Stop REC at primary | Stop PtoS force command at page 20 The mirroring session would still remain active at the primary site. You have to clear it explicitly. | |
| Create Standby control file | | Issue "alter database create standby controlfile" |
| Prepare to create standby database at the primary site | Make sure the database is shutdown Make sure that, after fail-over, no one has either (1) opened the database or (2) mounted the database and performed any recovery. You may check the alert file. Copy the standby database control file from standby site Copy from standby site all archive log files generated after activating the standby database. All archived logs used for recovery must come from standby site Create a new init.ora file for the standby database to be created | |
| Setup sqlnet | Setup sqlnet connection | Setup sqlnet connection |
| Unmount redo log volume | Unmount the redo log volume to be target of mirroring | |
| Reverse mirroring | Start StoP command at page 20 Start REC mirroring from the Standby site to the Primary site. It can be issued from either site. | |
| Create standby database at primary site | Use name convert inti.ora parameter or "alter database rename file" command to let database point to the correct datafiles Mount and recover the standby database, using only archived logs copied from standby site | |

Direct Fall Back via DB Copy

You can also fall back directly to the primary site, skipping the steps to set up new init.ora files and configure archive logs to be sent in the reverse direction. However, this method typically causes more down time. There are, again, three different procedures that may be used to fall back directly. The first procedure is copying the database from the standby site to the primary site:

| Steps for fall back directly via DB copy | Action on primary site | Action on standby site |
|--|--|--|
| Make a full backup at standby | | Make a cold or hot full backup of the database |
| Stop REC at primary | Stop PtoS force command at page 20 The mirroring session would still remain active at the primary site. You have to clear it explicitly. | |

| | | |
|-----------------------|---|-------------------|
| Copying datafiles | Copy the full backup from standby site to primary site. Overwrite the old production datafiles on primary site. | |
| | | shutdown database |
| Prepare to fall back | Copy the online logs from standby site to primary site. (Overwrite the old production online logs on primary site.) Copy the current control file from the standby site to the primary site. (Overwrite the old production control files on primary site.) Copy from the standby site all logs archived during and after the full backup. | |
| Recover and fall back | Startup mount, use the original production init.ora file Query v\$datafile and v\$logfile. Make sure they point to the correct datafiles and log files. Rename those files to the correct location if necessary Issue "recover database", supplying only archive logs copied from standby site Issue "alter database open" | |

Direct Fallback via Restoring Backup

Another procedure to fall back directly is to restore a backup at the primary site and recover it all the way to current state. This is useful when the database is large, and it takes too long to copy the database across a network. The steps are as follows:

| Steps to fall back via restoring backup | Action on primary site | Action on standby site |
|---|---|--|
| Periodic full backup | Make a full cold backup or hot backup of the database. Also backup the control file via "alter database backup controlfile to location". | |
| Disaster | | |
| Activate standby | | Fail over to standby site (as described above) |
| Stop REC at primary | Stop PtoS force command at page 20 The mirroring session would still remain active at the primary site. You have to clear it explicitly. | |
| Restore backup and prepare for recovery | Restore the full backup of the database. Make sure that the backup was taken before the disaster. Pay more attention if either logs are mirrored asynchronously or the mirror is broken at the time of disaster. In these cases, the backup must have been taken before the primary database generated the last archive log shipped to the standby site. When in doubt, always use an older full backup. Copy all the archive logs from standby site since the backup. These should be the only archive logs used for recovery | Issue "alter database backup control file to trace" to generate a create control file script |
| Recover primary site ² | Restore the backup control file Startup mount, using the old production init.ora. Issue "recover database using backup controlfile | |

² You could skip this step, and recover the primary site in the next step, after shutting down the standby site. However, this step allows you to recover the primary site while the standby site is still running the production database. Thus if you include this step, the database will be more available.

| | | |
|-----------------------------------|---|-------------------|
| | until cancel". Recover the database until it has applied all the archive logs copied from the standby site. | |
| | shutdown database | |
| Shutdown database at standby site | | shtudown database |
| Fall back to primary site | <p>Copy the online logs from standby site to the primary site, by either establishing remote mirroring or using an OS copying facility. Also copy all new archive logs from standby site.</p> <p>Copy the create control file script from the standby site. Edit it if necessary so that it points to the correct datafiles and log files.</p> <p>startup nomount, using the old production init.ora file</p> <p>create control file</p> <p>Issue "recover database", supply archive logs if necessary</p> <p>alter database open</p> | |

Direct Fallback via Recovery

Yet another procedure to fallback directly requires neither copying the database across a network nor restoring backups. It only works, however, under the following 3 conditions:

- the online logs are mirrored synchronously, and
- the mirror is established when the disaster happened, and
- no one has started the database at the primary site after the disaster.

| Steps to fall back directly via recovery | Action on primary site | Action on standby site |
|--|--|---|
| Disaster | | |
| Activate standby | | Fail over to standby site (as described above) |
| Stop REC at primary | Stop PtoS force command at page 20 The mirroring session would still remain active at the primary site. You have to clear it explicitly. | |
| Generate create controlfile | | issue "alter database backup control file to trace" |
| shutdwon database at standby | | shutdown database |
| Prepare to recovery primary site | <p>Make sure the database is shutdown</p> <p>Make sure that, after fail-over, no one has either (1) opened the database or (2) mounted the database and performed any recovery. You may check the alert file.</p> <p>Copy from standby site all archive log files generated after activating the standby database. All archived logs used for recover must come from standby site.</p> <p>Copy online logs from stadnby site to primary site. You can also use remote mirroring to copy the files. Overwrite the old online logs on primary site.</p> <p>Copy the create controlfile script from the standby site. Modify it so that it points to the correct datafiles and log files. Make sure the script uses noresetlogs option.</p> | |
| Recover primary site | Startup the database nomount using the production init.ora file | |

| | | |
|--|--|--|
| | <p>Issue the create controlfile command</p> <p>Issue "recover database", recovering the database using only archive logs copied from the standby site</p> <p>Issue "alter database open"</p> | |
|--|--|--|

Mirroring the Whole database

One can also remotely mirror a complete Oracle database, including datafiles, online log files, and control files. One can mirror a whole database either synchronously or asynchronously. The tradeoffs between mirroring a whole database and mirroring online logs for standby database are discussed on page 1. This section describes how to mirror a whole Oracle database with the procedures for mirroring a database synchronously and asynchronously.

Regardless of mirroring synchronously or asynchronously, one must mirror all datafiles, log files, and control files. In other words, put all datafiles, log files, and control files on the remotely mirrored storage system. One must include all log members, if logs are multiplexed. One can include just one control file, if the control file is also multiplexed. The Init.ora files are not mirrored.

If asynchronous mirroring is used, one must put the datafiles, online log files, and control files in the same volume in order to preserve Write Ordering, because the GR740 can preserve write order only within one volume. In synchronous mode, the online log files can be put into a volume separate from datafiles and control files, as ordering is maintained through the handling of the host I/O requests.

If the database is mirrored synchronously, you have two options with archive logs:

- Do not mirror archive logs. With this option, old backups may become invalid after fail-over.
- Mirror archive logs synchronously. In this case, you can continue to use old backups.

If the database is mirrored asynchronously, you also have two options:

- Do not mirror archive logs. With this option, old backups may become invalid after fail-over.
- Mirror archive logs in the same mirroring group used to mirror datafiles, log files, and control files. In other words, all files mirrored must be in the same volume in order to preserve write ordering.

Primary to Secondary Fail-Over

Suppose that a disaster happens at the primary site. One can fail over to the secondary site with the following steps:

1. Disable the link or Confirm that it has failed
2. Confirm the standby database is valid by using the QUERY command. If not valid, Don't go further.
3. Issue STOP Force command to stop mirroring.
4. Mount all volumes for database after fsck.
5. If all files have identical path on the primary and secondary sites, one can simply start Oracle on the secondary site. Otherwise, one can mount the database first, use ALTER DATABASE RENAME FILE command to rename datafiles and log files to the correct location, and then issue "alter database open". Note that there is no need to resetlogs.

If the database is mirrored synchronously, no committed data is lost after fail-over. If the database is mirrored asynchronously, it is possible that some committed data may be lost after fail-over.

If the archive logs are not mirrored, one should take a full backup after fail-over, and throw away old archive logs. If archive logs are mirrored, one must be very careful when using past archive logs. There are two cases:

- Both the database and the archive logs are mirrored synchronously. You must always maintain one current production database, and always use archive logs generated by the production database.
- Both the database and the archive logs are mirrored asynchronously, within the same volume. You must again always maintain one current production database, and always use archive logs generated by the production database. In addition, you should avoid copying archive logs between the primary and the secondary site, and only use archived logs from the mirroring group or volume.

Secondary to Primary Fall-Back

After the primary site has been fixed, one can fall back to the primary site. There are two ways:

- *Direct fallback*. Copy the database from the secondary site to the primary site. Use the primary site as production database.
- *Reverse role*. Establish the primary site as the secondary database. Continue to run the production database at the secondary site.

Direct Fallback via DB Copy

| Steps to fall back directly via recovery | Action on primary site | Action on standby site |
|--|--|---|
| Disaster | | |
| Activate standby | | Fail over to standby site (as described above) |
| Stop REC at primary | Stop PtoS force command at page 21 The mirroring session would still remain active at the primary site. You have to clear it explicitly. | |
| Shutdown database | | shutdown database |
| Copy all datafiles, log files, and control files to the primary site | 1. Unmount volumes 2. Start StoP command at page 21 3. Query StoP command at page 21, loop here until confirm synchronized 4. Stop StoP command at page 21 5. Mount volumes after fsck Mirroring is used to copy files here. Other options, like ftp, can be used. | |
| Unmount volumes | | Unmount datafiles and redo log volume to be target of mirroring |
| Start mirroring | Start PtoS command at page 21 | |
| Open database | Open the database as production database (noresetlogs). | |

Reverse Role via DB Copy

| Reverse Role via DB Copy | Action on primary site | Action on standby site |
|--------------------------|--|--|
| Disaster | | |
| Activate standby | | Fail over to standby site (as described above) |
| Stop REC at primary | Stop PtoS force command at page 21 The mirroring session would still remain active at the primary site. You have to clear it explicitly. | |
| Unmount volumes | Unmount redo log volume and datafile volume | |
| Start mirroring | Start StoP command at page 21 | |

Summary

GR740 Remote Equivalent Copy (REC) is the capability to mirror volumes between storage boxes through Fibre Channel without any host CPU usage. It can be used for disaster recovery with an Oracle database.

There is no absolute method for disaster recovery. One has to select an option which is most suitable to the operating environment among several available options. The major items to be chosen are (a) the transfer mode and (b) the range of mirroring. The transfer mode means synchronous mode or asynchronous mode. The range of mirroring is the choice between online logs mirroring or whole database mirroring.

The asynchronous mode and online logs mirroring would be recommended for greatest performance and volume configuration freedom. However the practical requirement for each user affects the determination, such as the following;

- If lost data is never acceptable and no other mechanism except mirroring online logs can be defined to recover the last bit changed, then synchronous mode is the only option.
- If the fail-over time is important, then whole database mirroring may be preferred. Mirroring online logs requires the time to apply the archive log at fail over.

One must design the disaster recovery system with REC and Oracle database through a careful study of each of the options and the operational site requirements.

Appendix Example of Command Line Interface

Here are the command line examples to control GR740 remote mirroring feature using the values illustrated in **OSCP Test Configuration** at page9.

For redo log mirroring

Start PtoS command

Start the mirroring from the primary site to the standby site.

For synchronous mirroring;

```
STXCopyEC START /dev/dsk/c2t0d2s7 00GR740#####GR74B01#####CA57##### 02h 280000h 40000h  
00GR740#####GR74B01#####BP56##### 02h 280000h 0 20h
```

For asynchronous mirroring;

```
STXCopyEC START /dev/dsk/c2t0d2s7 00GR740#####GR74B01#####CA57##### 02h 280000h 40000h  
00GR740#####GR74B01#####BP56##### 02h 280000h 0 0h
```

Start StoP command

Start the mirroring from the standby site to the primary site.

For synchronous mirroring;

```
STXCopyEC START /dev/dsk/c2t0d2s7 00GR740#####GR74B01#####BP56##### 02h 280000h 40000h  
00GR740#####GR74B01#####CA57##### 02h 280000h 0 20h
```

For asynchronous mirroring;

```
STXCopyEC START /dev/dsk/c2t0d2s7 00GR740#####GR74B01#####BP56##### 02h 280000h 40000h  
00GR740#####GR74B01#####CA57##### 02h 280000h 0 0h
```

Query PtoS command

Query the status of mirroring from the primary site to the standby site. The mode, synchronous or asynchronous, is not sensitive in the Query command.

```
STXCopyEC QUERY /dev/dsk/c2t0d2s7 00GR740#####GR74B01#####CA57##### 02h 280000h 40000h  
00GR740#####GR74B01#####BP56##### 02h 280000h 0 0h
```

Stop PtoS force command

Stop mirroring from the primary site to the standby site anyway. If not sure about the link status, one may have to issue this command at another site to clear everything.

```
STXCopyEC CANCEL /dev/dsk/c2t0d2s7 00GR740#####GR74B01#####CA57##### 02h 280000h 40000h  
00GR740#####GR74B01#####BP56##### 02h 280000h 0 02h
```

For whole database mirroring

Start PtoS command

Start mirroring from the primary site to the standby site.

For synchronous mirroring;

```
STXCopyEC START /dev/dsk/c2t0d2s6 00GR740#####GR74B01#####CA57##### 02h 80000h 240000h  
00GR740#####GR74B01#####BP56##### 02h 80000h 0 20h
```

For asynchronous mirroring;

```
STXCopyEC START /dev/dsk/c2t0d2s6 00GR740#####GR74B01#####CA57##### 02h 80000h 240000h  
00GR740#####GR74B01#####BP56##### 02h 80000h 0 0h
```

Start StoP command

Start mirroring from the standby site to the primary site.

For synchronous mirroring;

```
STXCopyEC START /dev/dsk/c2t0d2s6 00GR740#####GR74B01#####BP56##### 02h 80000h 240000h  
00GR740#####GR74B01#####CA57##### 02h 80000h 0 20h
```

For asynchronous mirroring;

```
STXCopyEC START /dev/dsk/c2t0d2s6 00GR740#####GR74B01#####BP56##### 02h 80000h 240000h  
00GR740#####GR74B01#####CA57##### 02h 80000h 0 0h
```

Query PtoS command

Query the status of mirroring from the primary site to the standby site. The mode, synchronous or asynchronous, is not sensitive in the Query command.

```
STXCopyEC QUERY /dev/dsk/c2t0d2s6 00GR740#####GR74B01#####CA57##### 02h 80000h 240000h  
00GR740#####GR74B01#####BP56##### 02h 80000h 0 0h
```

Query StoP command

Query the status of mirroring from the standby site to the primary site. The mode, synchronous or asynchronous, is not sensitive in the Query command.

```
STXCopyEC QUERY /dev/dsk/c2t0d2s6 00GR740#####GR74B01#####BP56##### 02h 80000h 240000h  
00GR740#####GR74B01#####CA57##### 02h 80000h 0 0h
```

Stop PtoS force command

Stop mirroring from the primary site to the standby site anyway. If not sure about the link status, one may have to issue this command at another site to clear everything.

```
STXCopyEC CANCEL /dev/dsk/c2t0d2s6 00GR740#####GR74B01#####CA57##### 02h 80000h 240000h  
00GR740#####GR74B01#####BP56##### 02h 80000h 0 02h
```

Stop StoP command

Stop mirroring from the standby site to the primary site.

```
STXCopyEC CANCEL /dev/dsk/c2t0d2s6 00GR740#####GR74B01#####BP56##### 02h 80000h 240000h  
00GR740#####GR74B01#####CA57##### 02h 80000h 0 0h
```