Backup and Recovery of Oracle8i Parallel Server: Guidelines for using Recovery Manager (RMAN)

An Oracle Technical White Paper

September 1999

AUTHORS: JUERGEN PAASCH, HEIKO ROBERT, HOGAN FLAKE

DATA STORAGE TECHNOLOGIES, PARALLEL SERVER GROUP

Table of contents

1. Overview

This document provides some guidelines on how to configure Recovery Manager (RMAN) and the production database in order to get a robust backup strategy for a Oracle8*i* Parallel Server installation. It discusses the pros and cons of several different solutions and also provides sample of the scripts that were used in the different scenarios.

The Oracle8i database - together with Oracle Parallel Server - yields dramatically higher scalability and availability for high volume, mission-critical online transaction processing (OLTP), data warehouse, and Internet applications. These benefits are achieved by exploiting the power and redundancy of clustered computer systems.

However, although clustered computer systems try to avoid any single point of failure, they don't eliminate the DBAs need for having a robust backup strategy. Backups are always needed to provide the ability to recover from an unforeseen media failure or human error. For example:

- The unlikely event of losing two mirrored disks
- A user who accidentally drops an important table

Both are examples in which restoring and recovering from a recent backup provides the solution.

Oracle8 introduced a new utility called Recovery Manager (RMAN) that can be used for all backup and recovery tasks. The Recover Manager is available with a GUI as well as a command line interface. It can be used with the production database's controlfile or with a recovery catalog. Oracle strongly recommends the use of a recovery catalog to get maximum benefits from the utility.

RMAN also provides an application programming interface for data movement between Oracle and other vendor's products. This provides the integration with a tape management systems necessary for backups directly to tape.

Using RMAN, the risks due to DBA errors can be considerably minimized, especially when using a recovery catalog database. It keeps all necessary information about the target database and the backups taken in this catalog. DBAs don't have to change backup scripts when new datafiles are added and they don't need to know which backups to apply in case of a media failure.

All of the Oracle7 backup, restore and recovery methods are still supported in Oracle8 and Oracle8iô. For example, Oracle8i still supports server manager and its recover command to perform recovery. Old scripts for Oracle7 can be used for Oracle8i databases, but the Oracle7 API called EBU was replaced by RMAN.

2. Introduction to Recovery Manager

Recovery manager (RMAN) is an Oracle tool introduced with Oracle8 that allows you to backup, copy, restore, and recover datafiles, control files and archived redo logs. You can invoke RMAN as a command line utility or use a GUI of the Oracle Enterprise Manager.

RMAN automates many of the backup and recovery tasks that were formerly performed manually. For example, instead of requiring one to locate appropriate backups for each datafile and copy them to the correct place using OS commands and then choosing which archived logs to apply, RMAN manages these tasks automatically.

When you use RMAN to connect to your target database, RMAN uses server sessions to perform the backup and recovery operations through a PL/SQL interface. RMAN physically stores its backups and copies on disks or, if you use media management software, on tape. RMAN stores metadata about its backup and recovery operations in the recovery catalog, which is a centralized repository of information, or exclusively in the target database's control file. We strongly recommend a separate database for the recovery catalog. We also recommend that the recovery catalog and the Oracle Enterprise Manager (OEM) repository be located in the same database.

3. Before you begin

For fast and reliable recovery from failures, you need to plan a robust backup and recovery strategy. This is true for both single instance and Parallel Server databases. You should be able to answer the following questions:

- How long can the database or some tablespaces be off-line to perform backups?
- What hardware and software resources are available?
- Do I need to recover from user or application errors?
- How far back in time should I be able to recover?

Several configuration parameters affect the availability of the database. These include, for example, the size of the Redo Logs, how many mirrors of the controlfile do I need or how often should I take backups. It is absolutely necessary to test some common backup and recovery scenarios before going into production. That includes the loss of a datafile, loss of the system tablespace, loss of a datafile containing rollback segments with open transactions and incomplete recovery.

There are also some Parallel Server specific issues that one needs to be aware of. Parallel Server is about having no single point of failure and the hardware and software configuration should take this into account.

The cluster needs to be setup by experts and a proper third party Media Management provider needs to be chosen. Oracle ships a special version of Legato's Networker with its Oracle8i Version 8.1.5 Server CD.

This version is not certified for Parallel Server by Legato. It has some shortcomings that make it unusable in a cluster. For example, if you want to distribute backups over two nodes with one tape drive each in a cluster, you could issue a RMAN command like follows (this assumes that RMAN has access to the archivelog files of both threads):

```
RMAN target internal/sys catalog RMAN USER/RMANPASS@RMANCAT
run {
        allocate channel ops1tape1 type `SBT_TAPE';
        allocate channel ops2tape1 type `SBT_TAPE';
        backup database;
        backup archivelog all:
}
```

With this command RMAN distributes the backup to both nodes in parallel. This means that the different datafiles and archivelog files are distributed to the nodes but RMAN does not know, on which tape it finds which datafile or archivelog file since LSM has only a local catalog on each node. If you want to recover you must either know, in which backupset to find a specific datafile or you must try for each file to find it in a specific backupset. Or you must be using version 8.1.6 which provides an AUTOLOCATE option, described later in this document, that causes RMAN to automatically search all channels for the backups it needs.

Therefore you should consider a full featured version of Networker or another Media management system like Omniback, otherwise Recovery using LSM gets very difficult.

Another issue that should be taken into account is to find a strategy for the archived redo logs. If RMAN wants to backup these files it needs to have access to all redo log files mentioned in the backup archivelog clause. With the above command, RMAN needs to have access to the files from both threads, otherwise the backup job will fail. In the section "Tests with different configurations" we will discuss some options with or without the use of NFS/HANFS. Another important point is the fact that if you don't mirror the archived log files and you need to recover while one node is down, you possibly will lose data, since you can only do incomplete recovery.

4. Installing and setting up a recovery catalog database

Oracle strongly recommends the use of a recovery catalog, since the following commands are not available when just using the control files of the target database:

- change...available, change unavailable, change backupset...crosscheck, change backuppiece...crosscheck (supported in 8.1.6 without a catalog)
- change...uncatalog
- create catalog, upgrade catalog, drop catalog
- create script, delete script, replace script, print script
- crosscheck backup (supported in 8.1.6 without a catalog)
- delete expired backup (supported in 8.1.6 without a catalog)
- list incarnation
- register database
- report schema at time
- reset database
- restore (when no controlfile is available)
- resync catalog
- set auxname

Although a recovery catalog database can be backed up while the main database is down, at times when no backup on the target database is done, we can't afford to lose any data here. Therefore this DB should operate in archivelog mode and the same recommendations, e.g. mirroring of controlfiles and online redo logs, should be followed as for a production system. We recommend installing the a recovery catalog on a different system from the Parallel Server cluster since you need access if one or more nodes are down.

We strongly recommend that the recovery catalog be used as a repository database for Oracle Enterprise Manager, as well. Therefore the PROCESSES setting in init.ora should be set to at least 200.

A Net8 Listener should be configured on each Parallel Server node and at the RMAN repository, since RMAN needs to connect through Net8. The TNSNAMES.ORA in the Oracle Home where you'll run RMAN should contain entries for all nodes of the Parallel Server database.

The sizes of the datafiles are proportional to the number of datafiles and archivelogs at the target database, and to how frequently those datafiles and archivelogs are backed up. The minimum allocations are:

SYSTEM tablespace	50 megabytes
TEMP tablespace	5 megabytes

Rollback tablespace	5 megabytes
recovery catalog tablespace (RECOVERYTS)	10 megabytes per year at least
controlfiles	2*5 megabytes
2 Online redo log groups, each with at least 2 members	4*1 megabytes
OEM_REPOSITORY tablespace (if needed)	50 megabytes

After installing the software and configuring the database, a schema for the recovery catalog should be created and it should be granted the role RECOVERY_CATALOG_OWNER. You can use the following command with SVRMGRL:

```
grant connect, resource, recovery_catalog_owner to RMANUSER identified by RMANPASS;
```

The catalog itself is now created with a RMAN command:

```
RMAN catalog RMANUSER/RMANPASS@RMANCAT

create catalog tablespace RECOVERYTS;
```

This procedure has changed since Oracle8 version 8.0.5. It is no longer required to run the script catrman.sql in RMANUSER's schema.

5. Creating Shared Volumes for Raw Devices

In a Parallel Server environment Oracle needs access to the tablespace datafiles from all nodes of the cluster running Oracle. Most of the operation systems don't support shared file systems, so you have to create raw devices to store the tablespace data.

In the following section we describe how to create raw devices if you use SUN Cluster Software from Veritas.

6. Creating Raw Devices on SUN Cluster

Before you can start creating shared volumes you have to create shared disk groups. This can be done with vxva, the GUI version for administrating Veritas volumes, or with the (often faster method) corresponding line commands.

You can create a small script using the following steps to create your shared volumes. The advantage of using a script instead of using the GUI is that your configuration is well documented and it is easier to repeat in case of a failure, or if you want to add more nodes.

6.1 Determine unused and not shared disks with the command

```
vxdisk list

...

c2t0d0s2     sliced    -          -          online

c2t0d1s2     sliced    -          -          online

...
```

6.2 Create a shared disk group

To create a shared (-s) disk group use the command:

```
vxdg -s init <groupname> <diskname> [disk2 disk3 ...]
```

If you get an error while creating the shared disk group like

Error in cluster processing

you might try to first create a private volume by omitting the -s parameter. You will then get more speaking errors like

Disk is in use by...

To share a disk group you can use the command

```
vxdg -s import ops
```

or

```
vxdg deport ops
```

to unshare a disk group.

6.3 Adding disks to a disk group

To add more disks to the disk group use the command:

```
vxdg -g <groupname> adddisk <diskname> [disk2 disk3 ...]
```

6.4 Create Volumes in the shared disk group

To create striped volumes (using 4 disks) use the command:

```
vxassist -g <groupname> -U fsgen make <Vol_name> <size>m columns=4 user=oracle group=dba
```

Please type vxassist help or read the Veritas Documentation for further information

In this example for creating the volumes using 4 stripes for all volumes, the complete script looks as follows:

```
#!/bin/sh
# creates Volumes for Oracle after creating
# the shared disk group

vxdg -s init ops c1t2d0s2
vxdg -g ops adddisk c1t3d0s2 c1t4d0s2 c1t5d0s2

vxassist -g ops -U fsgen make control1 11m columns=4 user=oracle group=dba
vxassist -g ops -U fsgen make control2 11m columns=4 user=oracle group=dba
vxassist -g ops -U fsgen make redo_1_1 11m columns=4 user=oracle group=dba
vxassist -g ops -U fsgen make redo_1_2 11m columns=4 user=oracle group=dba
vxassist -g ops -U fsgen make redo_2_1 11m columns=4 user=oracle group=dba
vxassist -g ops -U fsgen make redo_2_2 11m columns=4 user=oracle group=dba
vxassist -g ops -U fsgen make system 181m columns=4 user=oracle group=dba
vxassist -g ops -U fsgen make rbs1 31m columns=4 user=oracle group=dba
vxassist -g ops -U fsgen make temp 31m columns=4 user=oracle group=dba
vxassist -g ops -U fsgen make users 101m columns=4 user=oracle group=dba
vxassist -g ops -U fsgen make indx 51m columns=4 user=oracle group=dba
vxassist -g ops -U fsgen make oemrep 81m columns=4 user=oracle group=dba
```

Please pay attention to the host where you start the script. Disk names could be different on different nodes, because controllers could be installed in a different order on the hosts.

For example, if node 1 uses controller c1 to access the shared disks and node 2 uses the controller c3, the same disk t2d0 is listed on node 1 as c1t2d0 and on node 2 as c3t2d0!

6.5 Changing Ownership of Shared Volumes

Don't use the UNIX command chmod to change the ownership of a shared volume because the volume manager doesn't save these changes in its database and the changes will be lost after the next reboot. To change the properties of a shared volume use the vxedit command:

```
vxedit -g ops set mode=755 user=oracle group=dba control1
```

7. Setting up the Parallel Server Database for using with RMAN

After installing the Oracle software with the Parallel Server Option, it is necessary to configure the TNSNAMES.ORA and LISTENER.ORA files. Be sure to enter on both nodes an entry for the recovery catalog database, besides the entries for the Parallel Server instances.

The LOG_ARCHIVE_FORMAT in init.ora must include an entry for the thread number, for example:

```
LOG_ARCHIVE_FORMAT=%t_%S.dbf
```

When available, Oracle recommends mirroring the archived redo log files to a NFS/HANFS mounted directory. Then, RMAN always has access to the logs of all threads. You will avoid loss of data in the following situation: If one node which has already written some archived redo log files since the last backup goes down and after that the database needs recovery, RMAN has no way of accessing the archived redo logs of the crashed node. Only incomplete recovery could be applied and the database would need to be opened with the resetlogs option and losing some data. The next chapter shows how to cross mount directories with soft mounted NFS and how to backup and recover in those setups.

Oracle8i introduces a new parameter that supports up to 5 destinations for mirroring of archived redo log files with the parameter

```
LOG_ARCHIVE_DEST_n="location=<directory>"  1<=n<=5
```

One of these locations can be an NFS mounted directory. You should contact your hardware vendor to see whether NFS is supported between the nodes of a cluster and to find the proper settings for the NFS mount. In our test environment we had good results with soft mounts performed with the following command:

```
mount -F NFS -o soft,rw,retry=10,timeo=30 ops1:/u01/app/oracle/product/815/admin/ops/arch

                /u01/app/oracle/product/815/admin/ops/arch
```

Hard mounted NFS directories caused problems if one node goes down and the Oracle Archiver tries to write to these mount points.

If the network is set up correctly and the recovery catalog and the Parallel Server database are up and running, the production system must be registered in the recovery catalog:

```
RMAN catalog RMANUSER/RMANPASS@RMANCAT target internal/sys@RMANCAT

register database;
```

RMAN supports backups directly to tape if a Media Management software is installed on the system. For a list of supported configurations, see http://backup.us.oracle.com (Oracle internal web sight) or http://www.oracle.com/database/recovery/backupsp.html.

At this point the media management software needs to be installed and configured for use with RMAN. The installation is dependent on the MML vendor.

Oracle also supports creating a standby database for an Parallel Server system. The standby database does not have to be a cluster and can operate in single instance mode. Both threads of the Parallel Server system are automatically applied in the correct order. Oracle8i introduces a new LOG_ARCHIVE_DEST setting which allows to enter a valid Net8 service name. This service name must belong to a standby database and the logs are then transferred via Net8 through the network. In version 8.1.5, this functionality can not be used to transfer the archived redo logs between the nodes of a cluster.

8. Configuring Directories for RMAN

To enable RMAN to backup and recover an Oracle Parallel Server database in one step and to use RMAN by way of Oracle Enterprise Manager, all nodes must have access to all archive logs. Furthermore, each archived log must be visible as the exact same filename on all nodes.

In the following sections we describe the recommended solutions for NT and UNIX for sharing archive log files.

- Configuring Shared Archive Log Destinations on Windows NT Using Netbios
- Shared Archive Log Destinations on UNIX Using HA-NFS
- Configuring Multiple Archive Log Destinations on UNIX

9. Configuring Shared Archive Log Destinations on Windows NT Using Netbios

To configure shared archive logs on Windows NT, assign an unused drive letter to each node in the cluster. For example, if you have a cluster comprising two nodes named "Node 1" and "Node 2", and if drive letters J and K are unused, assign these letters to the nodes as shown:

Node 1	J:
Node 2	K:

While logged in as Domain Administrator, on each node use the Windows NT Disk Administrator facility to create new NTFS partitions. Each partition will be a local archive log destination for the instance running on that node. To configure this, assign the drive letter owned by that node to the new partition. Continuing with the previous example, on Node 1, create a new partition named "J:", and on Node 2, create a new partition named "K:". When you create each new partition, also create a directory hierarchy called "archivelogs" as shown:

Node 1	mkdir J:\archivelogs
Node 2	mkdir K:\archivelogs

On each node, share the new NTFS partition with the other nodes using the following command syntax:

net share <database_name>_logs=<drive_letter>:\

using the variables database_name and drive_letter as in the example shown where the database name is "OPS":

Node 1	net share OPS_logs=J:\
Node 2	net share OPS_logs=K:\

Use Windows NT Explorer to set permissions on these shared drives. Then on each node, map the shared drives from the remote nodes in the cluster using the same drive letters with the command:

net use<*drive_letter*>: \\<node_name>\<database_name>_logs

For this example, use the variables node_name, database_name and drive letter as in the following entries:

On Node 1, that has local drive J:, enter

```
net use K:  \\node2\OPS_logs
```

On Node 2, that has local drive K:, enter:

```
net use J:  \\node1\OP_logs
```

9.1 Entries in the INIT<SID>.ORA file on Windows NT

For the example on Windows NT create the following initialization parameters on both nodes in the INIT<SID>.ORA file:

```
LOG_ARCHIVE_DEST_1 = "location=J:\archivelogs"

LOG_ARCHIVE_DEST_2 = "location=K:\archivelogs"
```

Because each destination defaults to be an Optional (not Mandatory) destination, the database will continue to run as long as redo logs can be written to at least one destination. However, if some logs are not written to both destinations, the database is not completely recoverable from both nodes. In 8.1.6, you should use this format:

```
LOG_ARCHIVE_DEST_1 = "location=J:\archivelogs,mandatory"

LOG_ARCHIVE_DEST_2 = "location=K:\archivelogs,mandatory"
```

9.2 Access Rights for OracleService<SID> on Windows NT

To access remote archive log directories from your database, configure the Windows NT Service "OracleService<SID>" to start with a Windows account that has permission to write to this directory. Otherwise, attempts to do so will produce the message:

"ORA-9291: sksachk: invalid device specified for archive destination."

From the NT Control Panel's Services Icon, select the Startup panel for the service OracleService<SID> on each Parallel Server node, and then select Log On As This Account and enter a valid domain user and password.

10. Shared Archive Log Destinations on UNIX Using HA-NFS

Using HA-NFS is the smartest way of sharing archive log destinations because backup and recovery is transparent to backing up and restoring a single Instance.

The architecture of HA-NFS (Highly Available Network Fileserver) consists of two server machines, each acting as a nondedicated backup for the other, and a set of shared disks. One machine is the designated

primary server for the file system on the dual ported disks. During failure free operation, each machine monitors the liveness of the other. If the primary server fails, the other machine will become the server of the filesystem. Additional to the hostname and IP-address of each machine, the HA-NFS Service has his own IP-Address to which all clients connect, so the surviving node can assume the network service under the same address. Thus it appears to the rest of the network as if the fileserver is still alive.

In the Parallel Server environment, HA-NFS can be either installed on the cluster members already running Oracle or on a separate cluster.

Please contact your system vendor how to install and configure HA-NFS.

Once installed HA-NFS, mount the directory of the virtual file server into the local archive destination directory using normal NFS. The mount point must be named identically on all Parallel Server nodes. The Optimal Flexible Architecture (OFA) procedure of naming Oracle directories can be used to organize these mount points in a systematic way. On Sun Solaris, for example, use the following command:

```
mount -F nfs node1:/u01/app/oracle/product/815/admin/ops/arch1 /u01/app/oracle/product/815/admin/ops/arch1
```

Don't configure your nodes to mount this directory at startup. This could prevent the cluster software to start up if it is on the same machine, because cluster software and HA-NFS services are started manually after reboot.

11. Configuring Multiple Archive Log Destinations on UNIX

If you don't use HA-NFS you should write your archive logs to multiple destinations to make sure to have access to all archived logs when a node fails.

To configure shared archive log destinations on UNIX, create the same directory structure for the archive logs on every instance. For the previous example with 2 nodes create the directories:

```
$ORACLE_HOME/admin/arch1

$ORACLE_HOME/admin/arch2
```

On of the directories is the local archive log directory and the other is the destination for the archive logs of the remote host.

There are two methods for configuring archive log destinations on UNIX as described under the following headings:

- Configuring Shared Archive Log Destinations using NFS
- Configuring Non-shared Log Destinations

11.1 Configuring Shared Archive Log Destinations using NFS

Mount on each node the remote archive log destinations in their corresponding local directories using soft mounting NFS (cross mounting). In the previous example, the directory arch1 on node1 is the local archive log destination, and the other directory is the NFS mounting point for the remote archive logs of node2.

11.2 Special NFS Considerations

Exercise caution when using NFS in Parallel Server environments. If you use "hard NFS" (default), you can block the entire cluster if the remote directories become inaccessible. This might occur as a result of a hardware failure. For this reason, Oracle strongly recommends that you use "soft mounted" NFS directories or HA-NFS. Soft mounting means that a process attempting to access the mounted directory is not blocked until the directory becomes available after a failure.

Please contact your hardware vendor if your cluster supports soft mounted NFS directories between the nodes in a cluster. Please consult your vendor documentation because the commands to configure this are operating system dependent.

On Sun Solaris, for example, create a soft mounted directory on node 2 using the following command:

```
mount -F nfs -o soft,rw,retry=10,timeo=30 node1:/u01/app/oracle/product/815/admin/ops/arch1
/u01/app/oracle/product/815/admin/ops/arch1
```

11.2.1 Entries in the INIT<SID>.ORA file on UNIX

For the example on UNIX with shared directories using NFS create the following initialization parameters on both nodes in the INIT<SID>.ORA file:

```
LOG_ARCHIVE_DEST_1 = "location=/u01/app/oracle/product/815/admin/ops/arch1"

LOG_ARCHIVE_DEST_2 = "location=/u01/app/oracle/product/815/admin/ops/arch2"
```

Because each destination defaults to be an Optional (not Mandatory) destination, the database will continue to run as long as redo logs can be written to at least one destination.

11.3 Configuring Non-Shared Log Destinations

If your cluster hardware does not support shared directories with NFS (UNIX), backup all local files with RMAN (see Backup local archive logs).

11.3.1 Entries in the INIT<SID>.ORA file on UNIX

For the example on UNIX without shared directories create the following initialization parameters in the INIT<SID>.ORA file:

On node 1 enter:

```
LOG_ARCHIVE_DEST_1 = "location=/u01/app/oracle/product/815/admin/ops/arch1"
```

On node 2 enter:

```
LOG_ARCHIVE_DEST_1 = "location=/u01/app/oracle/product/815/admin/ops/arch2"
```

12. Backup and Recovery Sample Scenarios using RMAN with Parallel Server

12.1 Parallel Server Specifics

12.1.1 Access to Archived Logs

For both backup and recovery, RMAN needs access to all archived redo log threads in order to perform its operations, otherwise the job will simply fail. If you call RMAN on one node (you can't start RMAN with multiple targets) and you don't use NFS then the following script will fail, because RMAN has no way to access the archived redo logs from thread2:

```
RMAN target internal/sys@ops1 catalog RMAN USER/RMANPASS@RMANCAT
run {
allocate channel ops1tape1 type `SBT_TAPE';
backup database;
backup archivelog all:
}
```

The previous chapter describes how to make all logs accessible to RMAN.

12.1.2 Backup to Multiple Nodes using several Channels

Although you can theoretically perform a concurrent backup of more than one node of a cluster by using the connect clause in the allocate channel command, you will not be able to recover from such a backup if you haven’t installed Oracle 8.1.6 or later.

RMAN stores in its recovery catalog the local path of the backup file. If you allocate multiple channels on recovery, RMAN assumes to find the backup file in any of these channels. If RMAN can’t find the required file in the channel it tries, recovery fails.

Oracle 8.1.6 introduces a new feature to handle this. With the command

```
set autolocate on
```

you force RMAN to try out all allocated channels before it gives up.

12.1.3 Thread statement for backing up archivelogs

Also you should not use the thread statement of the backup archived log clause when you run RMAN on one node first and after that on the second one. This would look like:

```
RMAN target internal/sys@ops1 catalog RMAN USER/RMANPASS@RMANCAT
run {
...
backup archivelog thread 1;
...
}
RMAN target internal/sys@ops2 catalog RMAN USER/RMANPASS@RMANCAT
{
...
backup archivelog thread 2;
...
}
```

This procedure is not safe because if one node, lets say node 1, goes down for a longer period of time, the surviving node, node 2, performs log switches for the inactive instance in order to keep the SCNs nearly equal. The archiver of node 2 also archives the redo logs from ops1 into its log_archive_dest. This procedure is called redo log kick and this is why a script like above will simply fail.

12.2 About Sample Scenarios

The following Scenarios describe how to perform backup or recovery on one (any) node of a cluster if you have configured the archive log destinations as described in the previous chapters. Because you need the autolocate feature of 8.1.6 to use multiple hosts for backup and recovery, it is not discussed in this chapter.

There are in principle two procedures in Oracle8i backup/recovery in Parallel Server installations using RMAN:

- Shared Archive Log Destinations

- No Shared Archive Log Destinations

12.3 Shared Archive Log Destinations

12.3.1 Backup

If you share all archive logs with all nodes of a cluster using NetBios, NFS or HA-NFS, backup is very easy and can be executed from any node because every node can see and read all logs. In the example below, node 1 backs up all redo logs of all nodes. Make sure that the directories are configured for sharing as described in the previous chapter.

```
RMAN target internal/sys@node1 catalog rman user/rmanpass@rmancat
 run {
      allocate channel t1 type 'sbt_tape' format 'al_t%t_s%s_p%p';
      backup database;
      sql 'alter system archive log current';
     backup archivelog all delete input;
      release channel t1;
}
```

With the ALTER SYSTEM ARCHIVE LOG CURRENT statement, you force all nodes to back up their current log files.

12.3.2 Recovery

If you share archive log directories, you can change the destination of the automatic restoration of archive logs with the SET clause to restore the files to a local directory of the node from where you start recovery. This step does not apply if you use HA-NFS because you have only one shared directory which is mounted from all nodes.

To restore the USERS tablespace from node 1, use an RMAN command similar to the following:

```
RMAN target internal/sys@node1 catalog rman user/rmanpass@rmancat
 run {
      allocate channel t1 type 'sbt_tape';
      # set archivelog destination to '/u01/app/oracle/product/815/admin/ops/arch1';
      recover tablespace users;
```

```
sql 'alter tablespace users online';

release channel t1;

}
```

12.4 No Shared Archive Log Destinations

12.4.1 Backup

12.4.1.1 Backup local files from each node

If you do not share all archive logs, you can back up the logs locally on every node. In case of recovery, however, you need to have access from the node on which you begin recovery to all the archive logs on all nodes. For this reason Oracle recommends using a media management system which supports archiving over the network or shared directory services to simplify restoring log files. You should adapt the expression in the like clause to your own configuration (e.g. 'J:%' if you use NT).

The following RMAN script starts the local backup of all nodes using the CONNECT and LIKE clauses:

```
RMAN target internal/sys@node1 catalog rman/user/rmanpass@rmancat
 run {
      allocate channel t1 type 'sbt_tape' format 'db_n1_t%t_s%s_p%p'
      connect internal/sys@node1;
      backup database;
      sql 'alter system archive log current';
      backup archivelog like '%/arch1/%' delete input;
      release channel t1;
      }


run {
      allocate channel t1 type 'sbt_tape' format 'al_n2_t%t_s%s_p%p'
      connect internal/sys@node2;
      backup archivelog like '%/arch2/%' delete input;
      release channel t1;
```

```
}
```

12.4.1.2 Backup all Files from one node

Back up all the archive logs from one node into one backup archive instead of archiving them from each node separately. This makes it easier to find all backups on recovery. If you do not use shared directories to back up and restore archive logs, copy or move them using operating system tools, while retaining their exact names. You can easily create scripts to do this job before backing up or restoring the logs. If you are using Enterprise Manager, you can schedule the execution of this script before backup.

To copy all archive logs to the local directories on node 1, you can use a script similar to the following:

```
#!/bin/sh
sqlplus system/manager@node1 @switchlog.sql
rcp node2:/u01/app/oracle/product/815/admin/ops/arch2/* /u01/app/oracle/product/815/admin/ops/arch2
```

The switchlog.sql script is used to make sure to get all necessary log files for recovery. It looks like this:

```
alter system archive log current;
exit
```

To back up the archived logs from node 1 using RMAN, the command is similar to the previous example except that the ALTER SYSTEM ARCHIVE LOG CURRENT statement is executed from the shell script:

```
RMAN target internal/sys@node1 catalog rman/rmanpass@rmancat run {
    allocate channel t1 type 'sbt_tape' format 'al_t%t_s%s_p%p';
    backup database;
    host 'copy_log_script';
    backup archivelog all delete input;
    release channel t1;
    }
```

12.4.2 Recovery

In this section is described how to restore datafiles without using shared directories for the following cases:

- Backup of the archive logs has been performed from one node
- Backup of the archive logs has been performed from every node local using Oracle 8.1.6

- ☐ Backup of the archive logs has been performed from every node local using Oracle 8.1.5 or earlier

12.4.2.1 Backup of the archive logs has been performed from one node

If you moved all archive logs to one node to back them up, recovery is as easy as recovery using shared directories. To make sure you have all log files, copy all remote log files with your shell script described in Backup:

```
./rcp_all_logs.sh
RMAN target internal/sys@node1 catalog rman/user/rmanpass@rmancat run {
        allocate channel t1 type 'sbt_tape';
        set autolocate on;

recover tablespace users;
       sql 'alter tablespace users online';
        release channel t1;
        }
```

12.4.2.2 Backup of the archive logs has been performed from every node local using Oracle 8.1.6

If you backed up from each node it's own logs using a central media management system you can use a new feature implemented in RMAN of Oracle 8.1.6:

If you use several channels for recovery, RMAN asks every channel for the required file if he doesn't find it in the first one. This feature is called autolocate and makes it possible to recover a database using the local tape drive on the remote nodes:

```
RMAN target internal/sys catalog rman/user/rmanpass@rmancat
  run {
        allocate channel t1 type 'sbt_tape' parms 'ENV=(NSR_CLIENT=node1)';
        allocate channel t2 type 'sbt_tape' parms 'ENV=(NSR_CLIENT=node2)';
        set autolocate on;
        recover tablespace users;
```

```
sql 'alter tablespace users online';

release channel t1;

}
```

12.4.2.3 Backup of the archive logs has been performed from every node local using Oracle 8.1.5 or earlier

If you backed up the logs from each node, restore all necessary log files from remote nodes before performing recovery. Recovery has to be performed in three steps:

1. Restore datafiles.
2. Restore archive logs from remote nodes.
3. Recover.

Either restore the files locally and move them to the recovering node or, if you have a central media management system, use the CLIENT parameter to tell your Media Management Software where look for the backup.

Because you have to restore the archive logs before RMAN can begin recovery, you must specify which log files to restore:

```
RMAN target internal/sys@node1 catalog rmanuser/rmanpass@rmancat

run {

allocate channel t1 type 'sbt_tape';

restore tablespace users;

release channel t1;

}

run {

allocate channel t1 type 'sbt_tape' parms 'ENV=(NSR_CLIENT=node2)';

# this next  line is optional, they will be restored to the first log_archive_dest at the restoring node

set archivelog destination to '/u01/app/oracle/product/815/admin/ops/arch2';

restore archivelog

# this next  line is optional, if you don't want to restore ALL archive logs:
```

```
from time "to_date('05.09.1999 00:00:00','DD.MM.YYYY HH24:Mi:SS')"
# the like operand should be used as each archive dest may contain copies from each thread
# (this is due to redo log kicks)
like '%/2_%';
release channel t1;
}
run {
allocate channel d1 type disk;
recover tablespace users;
sql 'alter tablespace users online';
}
```

Backup and Recovery of Oracle8i Parallel Server

Guidelines For Using Recovery Manager (RMAN)

An Oracle Technical White Paper

September 1999

Technical Authors: Juergen Paasch, Heiko Robert, Hogan Flake

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

Worldwide Inquiries:
+1.650.506.7000
Fax +1.650.506.7200
http://www.oracle.com/