Technical report:

OracleRAC10*g* and IBM System Storage N series in a Linux Environment

Clustering using automatic storage-management functions

Document NS3349-0

October 29, 2007

© Copyright IBM Corporation, 2007. All Rights Reserved.
All trademarks or registered trademarks mentioned herein are the property of their respective holders

Table of contents

Abstract **3**

Introduction **3**

Assumptions **4**

Minimum requirements ad recommendations **5**

Storage infrastructure (recommended) 5

Linux server environment (recommended) 5

Oracle software requirements 6

Storage system configuration **6**

Node configuration **7**

Required Linux RPMs 7

Required Oracle installer interoperability patch for RHEL 3.0 7

Oracle user and NFS 7

Kernel settings 9

Network 9

iSCSI initiator software installation and setup **11**

iSCSI initiator support-kit installation **11**

Storage system iSCSI setup **12**

LUN creation and configuration **12**

Create iSCSI LUNs for the OCR device and voting disk device 12

Create iSCSI LUNs for the data file disk group 13

Create iSCSI LUNs for the logs' disk group 13

Install Oracle10*g* software (Oracle 10.1.0.2.0) **15**

Install Oracle CRS (Cluster Ready Service) 15

Install Oracle10g RDBMS 16

Install ASMLib and create ASM disks **17**

Obtaining the RPMs 17

Installing and configuring the ASMLib RPMs 18

Creating ASM disks 18

Use DBCA to create a database using ASM **19**

Backup and recovery **21**

Hot (online) backup procedure 21

Recovery procedure 22

Appendix A: Issues and examples **24**

Oracle issues 24

iSCSI issues 25

Backup-script example 25

Restore-script example 25

Trademarks and special notices **26**

Abstract

The Oracle10g automatic storage management feature provides the functionality of a file system and volume manager without the overhead; it provides the performance of raw disk-based databases without the complexities of managing raw disks. The feature provides a single point of storage for all nodes in a Real Application Cluster. IBM System Storage N series provides considerable value to enterprise data centers. Using the iSCSI protocol (either software or hardware), a system can provide flexible, reliable storage for database files, and the same system using NFS can provide cluster-enabled storage for the shared Oracle home.

Introduction

Oracle10*g*™ includes a feature known as "automatic storage management" (ASM), which provides the functionality of a file system and volume manager without the overhead normally associated with file systems and volume managers. In effect, it was designed by Oracle® to provide the performance of raw disk-based databases without the complexities of managing raw disks. Additional advantages of using Oracle ASM include the following:

- Automated database file naming
- Automatic load balancing across disks in a disk group, triggered by adding and removing disks
- Dynamic allocation and reallocation of disks by allowing the database administrator (DBA) to add or drop disks while the database is online
- Clustered file system for database file storage in real application cluster (RAC) databases.

The advantages of using Oracle ASM are very clear. Another feature of Oracle10*g* is the shared Oracle home for RAC databases. The shared Oracle home feature reduces the amount of administrative overhead associated with multinode RAC databases by providing a single point of storage for the Oracle RDBMS (Oracle Database) software for all nodes in an RAC. To make use of this feature, a cluster-enabled file system is necessary. While ASM provides cluster-enabled storage for database files, redo logs, and control files, it cannot be used for any other type of file system storage. For example, it cannot be used for storage of Oracle executables.

IBM® System Storage™ N series used with these features provides considerable value to enterprise data centers. Using the iSCSI protocol (either software or hardware), a system provides flexible, reliable storage for database files. The same system using native file system (NFS) protocol also provides cluster-enabled storage for the shared Oracle home. Once a system is integrated into this environment, the DBA has access to a range of features available with system, such as the IBM System Storage N series with Snapshot™ and SnapRestore® features for backup and recovery, and SnapMirror® for disaster recovery.

The purpose of this document is to describe in a step-by-step fashion a procedure for installing OracleRAC10*g* on Red Hat® Advanced Server 3.0 Linux® (the installed version at the time of this writing) using Oracle10*g* ASM and iSCSI raw devices on an IBM N series. The installation involves the following steps, which will be covered in this report:

- Linux host setup (two nodes)
- System setup
- Network setup
- Installation and setup of the iSCSI tools and SourceForge iSCSI software initiator
- Creation and preparation of iSCSI logical unit numbers (LUNs)
- Installation and configuration of the Oracle10*g* Cluster Ready Services (CRS)
- Installation and configuration of the Oracle10*g* RDBMS software
- Creation of Oracle ASM disk groups using Oracle Database Configuration Assistant (DBCA)
- Sample database creation using Oracle DBCA
- Backup and recovery of the database using Snapshot copies.

The installation procedure outlined in this document is consistent with both Oracle and IBM best practices and is therefore suitable for creating a production environment.

Assumptions

It is assumed that the target audience of this paper is already familiar with the following:

- Basic Linux system administration
- Installation and administration of OracleRAC10*g*
- Implementation and administration of Oracle10*g* ASM instances, disks, and disk groups
- Basic storage system administration and configuration
- Principles of iSCSI.

Minimum requirements ad recommendations

Storage infrastructure (recommended)

- One IBM N series which has IBM System Storage N series with Data ONTAP™ 7.1 or later
- One Gigabit Ethernet switch with at least five ports
- One Gigabit Ethernet NIC in the storage system for private access by RAC nodes
- One 10/100MB or GB NIC for storage system public access
- Two or more disk shelves, based on the disk space requirements of the data
- iSCSI license (see page 12 of “Setup Guide for iSCSI Linux Initiator Support Kit 1.0" for details).

Figure 1 is a diagram showing how the hosts and storage system are to be networked.

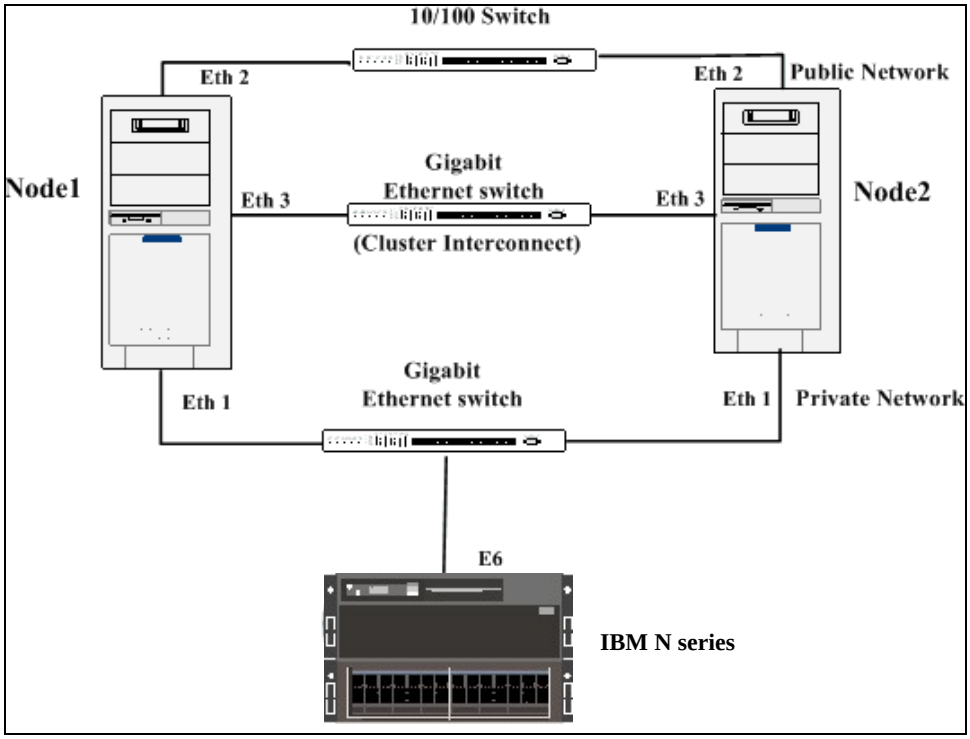

Figure 1) OracleRAC10*g* on Red Hat EL 3.0 server with an IBM N series.

Linux server environment (recommended)

Two 2-way 900 MHz Intel® processor-based enterprise-class servers or better equipped with the following:

- Red Hat Enterprise Linux Advanced Server 3.0 update 2 or later: kernel 2.4.21-15.EL or later. Note that Oracle also supports OracleRAC10*g* with ASM on Red Hat EL AS 2.1. This paper only addresses installation with RHEL AS 3.0 update 2 or later. Update 2 can be obtained from Red Hat.
- At least 4GB RAM.
- One GbE network interface for private interconnect between servers (Oracle’s best practices require a switch be used instead of crossover connection).
- One GbE network interface for private network connection to the system (through a switch).
- One network interface for public access (can be either GbE or 100Mb Ethernet).

Oracle software requirements

The procedures outlined in this document are based on Oracle10*g* Enterprise Edition 10.1.0.2.0; however, OracleRAC10*g* is also available with Oracle10*g* Standard Edition provided the Oracle CRS are installed and Oracle ASM is used. While the software release (10.1.0.2.0 or later) and patch requirements are the same for both versions, there may be some differences in the actual install and setup procedure. Below are the actual Oracle software and patch requirements for this installation:

- Oracle 10.1.0.2.0 with RAC license and ASM license
- Oracle10*g* patch 3639444 to correct Oracle bug 3557807, which results in "IPC using public Ethernet IP, instead of the private interconnect"
- Oracle10*g* patch 3591633 for Oracle ASMLIB bug 3591633.

Oracle patches can be downloaded from Oracle Metalink with a valid Oracle support agreement.

Storage system configuration

1. Follow normal instructions for installing the storage system.
2. Install Data ONTAP 7.1 or later. For the sake of simplicity it is assumed that the host name assigned to the storage system is "filer1."
3. Configure the storage system NICs. Throughout this document the following network configuration is assumed:
   - e6 Gigabit IP address: 10.60.140.35 netmask: 255.255.255.0
   - e0 Gigabit or 10/100 IP address: 10.60.136.35 netmask: 255.255.255.0

   Install license keys for NFS, SnapRestore, and iSCSI.
4. Create two volumes, one named "archive" for archive logs and one named "oraclehome" for Oracle executables by running the following commands on the storage system:
   ```
   vol create archive 5
   vol create oraclehome 2
   ```
   The archive log volume can be created on more than five disks if needed.
5. Create volumes for LUNs. You'll need one volume named "dbfile" for data file LUNs and one volume named "logs" for online redo logs, control files, voting disks, and OCR (Oracle Cluster Registry) devices. Create the two volumes by running the following commands on the system:
   ```
   vol create dbfile 14
   vol create logs 7
   ```
   This disk-per-volume configuration is recommended for good performance; however, it must be recognized that database load and performance requirements must be considered in determining the optimal configuration.
6. Add the following lines to the /etc/exports file on the storage system:
   ```
   /vol/orahome      -anon=0
   /vol/archive      -anon=0
   ```
7. Execute the following command on the storage system:
   ```
   Filer1> exportfs –a
   ```

Node configuration

Required Linux RPMs

Verify that the following required Red Hat Enterprise Linux 3.0 packages (or higher versions) have been installed on each node:

make-3.79.1
gcc-3.2.3-34
glibc-2.3.2-95.20
compat-db-4.0.14-5
compat-gcc-7.3-2.96.128
compat-gcc-c++-7.3-2.96.128
compat-libstdc++-7.3-2.96.128
compat-libstdc++-devel-7.3-2.96.128
openmotif21-2.1.30-8
setarch-1.3-1

The "`rpm -q` *`package_name`*" command can be used to check for packages and you can use "`rpm -Uvh` *`package_name`*" to install deficient packages.

Required Oracle installer interoperability patch for RHEL 3.0

p3006854_9204_LINUX.zip must be installed on each node before the Oracle installer can be started. This zip file can be downloaded from Oracle Metalink and installed as follows:

1. With a Web browser access Oracle's Metalink site.
2. Click the "Patches" link on the left.
3. Click "Simple Search."
4. Choose "Search By Patch Number(s)," enter "3006854" and click "Go."
5. Click the "Download Now" icon on the right of the screen to download R10.1.0.2 of the patch.
6. Download and copy the zip file to a working directory on each node (/tmp is suitable).
7. While logged in as the root user, change directories to the working directory and enter the following command: unzip p3006854_9204_LINUX.zip.
8. Change directories into the newly created "3006854" directory and run the install script to install the patch as follows: `. ./rhel3_pre_install.sh.`

Oracle user and NFS

1. Create the "oracle" user and "dba" group on both Linux hosts. Make sure the group ID and user ID are the same on all nodes and make the "oracle" user a member of the "dba" group.
2. This installation will utilize NFS-mounted volumes for the shared Oracle home and for archive logs. You will need to create the mountpoints on each Linux host and set the user and group to "oracle" and "dba," respectively:

```
mkdir /orahome
chown oracle:dba /orahome
mkdir /archive
chown oracle:dba /archive
```

It should be noted that archive logs can also be stored in an ASM disk group. In this example, NFS was chosen in order to retain SnapMirror functionality as a means of log transport for such applications as maintenance of a standby database for disaster recovery.

3. Add the following entries to /etc/fstab on each Linux host, where “filer1” is the name of the storage system:
   - `filer1:/vol/oraclehome /orahome nfs rw,bg,hard,nointr,rsize=32768,wsize=32768,tcp,vers=3,timeo=600`
   - `filer1:/vol/archive /archive nfs rw,bg,hard,nointr,rsize=32768,wsize=32768,tcp,vers=3,timeo=600`
4. Mount the volumes “oraclehome” and “archive” on each Linux host; create the Oracle10*g* directory; and set permissions, owner, and group as follows:

```
mount /orahome
mkdir /orahome/10g
chown oracle:dba /orahome
chown oracle:dba /orahome/10g
chmod 755 /orahome
mount /archive
chown oracle:dba /archive
chmod 755 /archive
```

   These commands must be executed on all Linux hosts.
5. Create the Oracle product directory on each host. Make “oracle” the owner and “dba” the group:

```
mkdir /home/oracle/product
chown oracle:dba /home/oracle/product
```

6. Create a symbolic link under the /home/oracle/product subdirectory for Oracle home:

```
ln –s /orahome/10g /home/oracle/product/10g
```

7. Set environmental variables.
   Several environmental variables need to be set. It’s very convenient to utilize a shell resource file for this purpose. Examples include the following: .bashrc for bash, .profile for the bourne shell, and .csh for the C shell. In Linux, bash is very popular, so for the sake of brevity and example we will assume the use of bash in this environment. Please note, however, that the intent behind doing this is not to imply that bash is the only shell that should be used for this type of configuration.
   a. Add the following entries to the Oracle user’s .bashrc file on node1:

```
ORACLE_BASE=/home/oracle/product; export ORACLE_BASE
ORACLE_HOME=$ORACLE_BASE/10g; export ORACLE_HOME
CRS_HOME=$ORACLE_BASE/CRS; export CRS_HOME
ORACLE_SID=rac1; export ORACLE_SID
LD_ASSUME_KERNEL=2.4.19; export LD_ASSUME_KERNEL
THREADS_FLAG=native; export THREADS_FLAG
PATH=$PATH:$ORACLE_BASE:$ORACLE_HOME:$ORACLE_HOME/bin
export PATH
```

   b. Add the following entries to the Oracle user’s .bashrc file on node2:

```
ORACLE_BASE=/home/oracle/product; export ORACLE_BASE
ORACLE_HOME=$ORACLE_BASE/10g; export ORACLE_HOME
CRS_HOME=$ORACLE_BASE/CRS; export CRS_HOME
ORACLE_SID=rac2; export ORACLE_SID
LD_ASSUME_KERNEL=2.4.19; export LD_ASSUME_KERNEL
THREADS_FLAG=native; export THREADS_FLAG
PATH=$PATH:$ORACLE_BASE:$ORACLE_HOME:$ORACLE_HOME/bin
export PATH
```

Kernel settings

Append the following lines to the /etc/sysctl.conf file on each node to set kernel parameters:

```
kernel.shmall = 2097152
kernel.shmmax = 2147483648
kernel.shmmni = 4096
kernel.sem = 250 32000 100 128
fs.file-max = 65536
net.ipv4.ip_local_port_range = 1024 65000
```

These settings can be activated by rebooting each node or by executing the following command on each node while logged in as the root user: `/sbin/sysctl –p`.

Network

1. The interconnect between the Linux nodes must be done through a switch. This is required by Oracle for the avoidance of direct connect autonegotiation issues that can occur during the recovery of a failed or evicted node. Oracle Metalink note 220970.1 strongly recommends the use of Gigabit Ethernet for the interconnect.
2. The dedicated network between the storage system and the Linux nodes should be Gigabit Ethernet. Gigabit Ethernet provides sufficient bandwidth and sufficiently low latency, which is required for best performance. A switch is required to allow both nodes to connect to the system.
3. The public networks for the storage system and the Linux nodes can be either gigabit or 100Mb.
4. Configure network interfaces. Below is an excerpt from a sample /etc/hosts file listing the IP addresses and associated host names described in this document:

```
10.60.136.121      node1 node1-eth2  (public network)
10.60.140.121      node1-eth1        (private storage system network)
10.60.139.121      node1-eth3        (private interconnect)
10.60.136.119      node2 node2-eth2  (public network)
10.60.138.119      node2-eth1        (private storage system network)
10.60.139.119      node2-eth3        (private interconnect)
```

   c. Use the subnet mask of 255.255.255.0 in configuring each interface. Verify connectivity by pinging each interface from each host.

   d. Add the above entries to the /etc/hosts file on each of the two nodes. Also add the following entry to the /etc/hosts files on both Linux hosts:

```
filer1 filer1-e6
```

   e. Verify connectivity to all nodes using the "`ping`" command from the storage command line.

5. One difference between OracleRAC10*g* and OracleRAC9*i*™ is that the OracleRAC10*g* listener must be configured to listen on a virtual IP (VIP) address instead of the actual public IP address of the node. This results in a faster failover in the event of a node failure. VIP configuration is part of the RAC installation process; however, an /etc/hosts entry must be made on each node before running the installer. Below are two entries for the two-node RAC configuration being discussed in this paper:

```
10.60.136.166      node1-vip
10.60.136.167      node2-vip
```

   The virtual IP addresses are for use by Oracle VIP only and should not be bound to any physical network interfaces. Adding them to the /etc/hosts files of each node is all that needs to be done. They must be on the same subnet as the public network interfaces.

6. Enable “rlogin” (and “rsh”) capabilities for the Oracle user between all nodes. This will require the creation of an /etc/hosts.equiv on each node. Below is a sample of a suitable /etc/hosts.equiv file on node1:

```
node1 oracle
node1-eth0 oracle
node1-eth2 oracle
node2 oracle
node2-eth0 oracle
node2-eth2 oracle
```

The rsh functionality between nodes and to themselves should be verified:

a. From node1 to node2:

b.
```
# su - oracle
$ rsh node2  uname -a
$ rsh node2-eth2 uname -a
```

c. From node2 to node1:

```
# su – oracle
$ rsh node1 uname -a
$ rsh node1-eth2 uname –a
```

d. From node1 to node1:

```
# su - oracle
$ rsh node1  uname -a
$ rsh node1-eth2 uname -a
```

e. From node2 to node2:

```
# su – oracle
$ rsh node1 uname -a
$ rsh node1-eth2 uname –a
```

iSCSI initiator software installation and setup

1. Download version 3.4.2 (the version used at the time of this writing) of the Linux iSCSI initiator software to a working directory on both Linux hosts.
2. Change directory to the one containing the linux-iscsi-3.4.2.tgz file and unbundle it as follows:
   ```
   tar xvzf linux-iscsi-3.4.2.tgz
   ```
3. Change to directory "linux-iscsi-3.4.2" and run the following commands:
   ```
   make all
   make install
   ```
4. Edit the /etc/initiatorname.iscsi file. You'll see a line similar to the following:
   ```
   InitiatorName=iqn.1987-05.com.cisco:<random_number>
   ```
5. Change the "<random_number>" part of the string to "rac1" on the first Linux node and "rac2" on the second Linux host. The string should appear on each host as follows:
   ```
   "InitiatorName=iqn.1987-05.com.cisco:rac1"          on the first host
   "InitiatorName=iqn.1987-05.com.cisco:rac2"          on the second host
   ```
6. Record the InitiatorName value for each node. These values will be used later in the installation.
7. Edit the /etc/iscsi.conf file on each Linux host and add the following lines to the top of the file:
   ```
   Continuous=no
   HeaderDigest=never
   DataDigest=never
   ImmediateData=yes
   MaxDiskCommandTimeout=180
   ConnFailTimeout=180
   DiscoveryAddress=<GBE_IP_ADDRESS_OF_FILER>
   ```
   where <*GBE_IP_ADDRESS_OF_FILER*> is the address of the Gigabit Ethernet interface in the storage system.

iSCSI initiator support-kit installation

1. Load the Linux tools from the CD provided to a working directory on each Linux host.
2. From the working directory, unbundle the files by running the following commands on each Linux host while logged in as root:
   ```
   gunzip filename_linux_tools_1_1.tar.gz
   tar xvf filename_linux_tools_1_1.tar
   ```
3. Change to the "*filename*_linux_tools_1_1" directory and run the command "`./install`" on both Linux hosts to install diagnostic scripts in the /opt/sanlun/bin directory.
4. Start the iSCSI service on each Linux host with the following command while logged in as root:
   ```
   # service iscsi start
   ```

Storage system iSCSI setup

Start iSCSI by entering the following on a storage system command line: "`iscsi start`."

LUN creation and configuration

Keep in mind that all ASM metadata is stored on the ASM disks and that it is extremely important that the metadata be consistent across ALL disks in a given disk group. For that reason, the ability to perform online (hot) database backups with storage system Snapshot copies is dependent on each ASM disk group being stored in a single system volume. In other words, the disks for a given disk group must not be spread across multiple volumes. If this condition is not met, it will be necessary to unmount each disk group before creating a Snapshot copy, which will require that the database be shut down as well. Below is the recommended model/procedure for creating disk group LUNs so as to enable hot database backups using storage system Snapshot copies:

Create iSCSI LUNs for the OCR device and voting disk device

From the storage system command line do the following for the lun setup OCR (Oracle Cluster Registry) device and the voting disk device:

1. Enter "y" at the prompt `Do you want to create a LUN?`
2. For the LUN path enter "`/vol/logs/OCRdisk.`"
3. Accept the default "y" response to enable space reservation for the LUN.
4. Enter "`100m`" for the LUN size.
5. Enter "`OCR Disk`" for the comment string.
6. Enter "`oralogs`" for the initiator group name and "`iSCSI`" for the igroup type.
7. At the "`Enter comma separated nodenames:`" prompt, enter the initiator names from the /etc/initiatorname.iscsi file on each Linux host and press the "Enter" key.

   For example: `iqn.1987-05.com.cisco:node1,iqn.1987-05.com.cisco:node2`
8. You will receive the prompt "`Enter comma separated nodenames`" once more. Just press the "Enter" key to continue.
9. Enter "`linux`" for the igroup OS type.
10. Enter "`1`" for the LUN ID.
11. Accept the configuration.
12. Choose the option to create the next LUN.
13. Repeat the above process to create another LUN with the following characteristics:
    Multiprotocol type of LUN = linux
    LUN path = /vol/logs/VOTINGdisk
    Space reservation for LUN enabled (y)
    LUN size = 20m
    Comment string = "Voting Disk"
    Initiator group = oralogs
    igroup type = iSCSI
    igroup OS type = linux
    LUN ID = 2

Create iSCSI LUNs for the data file disk group

Use the procedure just described (in “Create iSCSI LUNs for the OCR device and voting disk device”) to create LUNs with the following characteristics:

LUN Path	/vol/dbfile/lun3	/vol/dbfile/lun4	/vol/dbfile/lun5
Multiprotocol Type of LUN	Linux	Linux	Linux
Space Reservation for LUN Enabled	y	y	y
LUN Size	6g	6g	6g
Comment String	dbfile lun3	dbfile lun4	dbfile lun5
Initiator Group	oradb	oradb	oradb
iGroup Type	iSCSI	iSCSI	iSCSI
iGroup OS Type	Linux	Linux	Linux
LUN ID	3	4	5

Create iSCSI LUNs for the logs' disk group

Use the procedure described described in “Create iSCSI LUNs for the OCR device and voting disk device” to create LUNs with the following characteristics:

LUN path	/vol/logs/lun6	/vol/logs/lun7	/vol/logs/lun8
Multiprotocol type of LUN	Linux	Linux	Linux
Space reservation for LUN enabled	y	y	y
LUN size	4g	4g	4g
Comment String	logs lun6	logs lun7	logs lun8
Initiator group	oralogs	oralogs	oralogs
igroup type	iSCSI	iSCSI	iSCSI
igroup OS type	Linux	Linux	Linux
LUN ID	6	7	8

It should be noted that the number and sizes of LUNs were chosen for creation of a small test database and should not be considered as limits. However, keep in mind that if the desired number of SCSI devices (including iSCSI devices) should be kept at 16 or less.

1. After all the LUNs have been completed, you will need to restart the iSCSI service on each Linux host in order to scan the new LUNs. As the root user, run the following command on each node:
   /etc/init.d/iscsi restart.
2. The Oracle ASMLib kernel driver must access the LUNs as raw devices, so you’ll need to map a raw device to each iSCSI device listed in the /dev/iscsi directory. Below are the necessary steps:
   a. Append the following lines to the /etc/sysconfig/rawdevices file on each Linux host:

```
/dev/raw/raw1 /dev/iscsi/bus0/target0/lun1/disk
/dev/raw/raw2 /dev/iscsi/bus0/target0/lun2/disk
/dev/raw/raw3 /dev/iscsi/bus0/target0/lun3/disk
/dev/raw/raw4 /dev/iscsi/bus0/target0/lun4/disk
/dev/raw/raw5 /dev/iscsi/bus0/target0/lun5/disk
/dev/raw/raw6 /dev/iscsi/bus0/target0/lun6/disk
/dev/raw/raw7 /dev/iscsi/bus0/target0/lun7/disk
/dev/raw/raw8 /dev/iscsi/bus0/target0/lun8/disk
```

   The first field is the path to the target raw device; the second field is the full path to the name of the iSCSI device created above. Also, the bus# and target# may differ in your environment.

b. Verify the correct paths to the iSCSI devices before continuing.
c. After appending the above lines to the /etc/sysconfig/rawdevices file, you will need to create the bindings by running the following command on each Linux host, while logged in as the root user:

```
service rawdevices restart
```

A reboot will also create the bindings. One more thing: the rawdevices service must be restarted each time new bindings are added to the /etc/sysconfig/rawdevices in order for the OS to recognize them. You can get a listing of rawdevices by running the following command as the root user:

```
service rawdevices status
```

3. The next step is to set the correct owner, group, and permissions on the new raw devices. As the root user, run the following on each Linux host:

```
chown oracle:dba /dev/raw/raw1
chown oracle:dba /dev/raw/raw2
chown oracle:dba /dev/raw/raw3
chown oracle:dba /dev/raw/raw4
chown oracle:dba /dev/raw/raw5
chown oracle:dba /dev/raw/raw6
chown oracle:dba /dev/raw/raw7
chown oracle:dba /dev/raw/raw8
chmod 775 /dev/raw/raw1
chmod 755 /dev/raw/raw2
chmod 755 /dev/raw/raw3
chmod 755 /dev/raw/raw4
chmod 755 /dev/raw/raw5
chmod 755 /dev/raw/raw6
chmod 755 /dev/raw/raw7
chmod 755 /dev/raw/raw8
```

In order for the new settings to survive a reboot, you will also need to append these lines to the /etc/rc.local file on each Linux host.

Install Oracle10*g* software (Oracle 10.1.0.2.0)

Install Oracle CRS (Cluster Ready Service)

1. Log in as the Oracle user. The correct environmental parameters should have already been set up in the Oracle user's ./bashrc script as described in steps 4.1.7 and 4.1.8 above.
2. Switch directories to the /crs subdirectory on the Oracle Cluster Ready Services (CRS) Release 1 (10.1.0.2.0) CD-ROM and start the installer by running the runInstaller command (./runInstaller).
3. Click "next" on the first (Welcome) screen.
4. Accept the following defaults:
   "/home/oracle/product/oraInventory" for inventory directory
   "dba" for the operating system group name

   Please note that while a shared Oracle home can be used, each instance must have its own oraInventory directory.
5. When prompted, run the script /home/oracle/product/oraInventory/orainstRoot.sh as the root user on the node from which you are running the installer and click "Continue" when finished.
6. On the "Specify File Locations" screen, accept the defaults for "Source Path" and "Destination Name." Then change the "Destination Path" to "/home/oracle/product/CRS." When finished, click "Next." Keep in mind that the CRS_HOME must be under the ORACLE_BASE directory and in a different directory from the one used for ORACLE_HOME. In other words, CRS_HOME and ORACLE_HOME cannot share the same directory.
7. Choose the desired language and click "Next."
8. Accept the default "Cluster Name," specify the following information for public and private node names and click "Next" to continue:
   Public Node Names: node1, node2
   Private Node Names: node1-eth3, node2-eth3
9. On the "Private Interconnect Enforcement" screen, designate the public and private interface types and set the others to "Do Not Use." Click "Next" to continue.
10. The next screen should be entitled "Oracle Cluster Registry." Specify the path to the raw device to be used as the OCR disk; "/dev/raw/raw1" has already been created for this purpose, so specify that path and click "Next."
11. On the "Voting Disk" screen, enter "/dev/raw/raw2," which is the raw device already created for this purpose. Click "Next" to continue.
12. Use additional telnet windows to execute the script "/home/oracle/product/oraInventory/orainstRoot.sh" on both nodes as directed while logged in as the root user. Click "Continue" after this has been done.
13. On the "Summary" screen, verify that all the information displayed is correct and click "Install."

14. When prompted, run the following script as root on each node as directed, starting with the node on which the installer is being run:

```
/home/oracle/product/CRS/root.sh
```

    You may receive the following error message:

    Failed to upgrade Oracle Cluster Registry configuration

    If you receive the above error message, immediately run the script again.

    CLSFMT returned with error [4].
    Failed to initialize Oracle Cluster Registry for cluster0

    These messages appear to be the result of CRS being slow to start and the result of a bug in the script. If this happens, wait about five minutes and rerun it. If it still fails, wait another five minutes and run the script on each node again. CRS has been observed to require as long as eight minutes for its initial start. This should not be a problem, but if after a few attempts the problem persists, contact Oracle for assistance.

    Once the script runs successfully on all nodes, click “OK” on the “Setup Privileges” screen and “Exit” on the “End of Installation” screen. The CRS install is complete, and you are now ready to begin the Oracle RDBMS install.

15. The “root.sh” script that was run in the previous step changed the owner of the OCR disk from “oracle” to “root” and changed the permissions from “775” to “640." Before continuing to the next step, log in to each node as the root user, edit the /etc/rc.local file, and make the following changes in order to make the new permissions and ownership of the OCR disk persistent across reboots:

    a. Change line “`chown oracle:dba /dev/raw/raw1`” to “`chown root:dba /dev/raw/raw1`”
    b. Change line “`chmod 775 /dev/raw/raw1`” to “`chmod 640 /dev/raw/raw1`”

Install Oracle10g RDBMS

1. Before continuing with the installation, please verify that the preinstallation tasks have been performed, including those outlined in the section above, which include the following items:

    a. The oraclehome volume has been mounted under /orahome.
    b. The symbolic link for the shared Oracle home has been created in –s /orahome/10g /home/oracle/product/10g.
    c. The Oracle user’s .bashrc file has been set up properly.
    d. Kernel settings have been implemented.

2. Confirm the CRS configuration with the following command while logged in as the Oracle user:

```
/home/oracle/product/CRS/bin/olsnodes –n
```

    This should return each node name with the corresponding node number:

```
node1  1
node2  2
```

3. Insert the Oracle10*g* RDBMS CD and, while logged in as the Oracle user, start the installer (`./runinstaller`). Click “Next” on the first (Welcome) screen.

4. Accept the default name for Oracle home, verify that the destination path is defined as /home/oracle/product/10g, and click “Next."
5. Verify that “Cluster Installation” is checked and click “Select All” to select all nodes for installation. Click “Next” to continue.
6. Select “Enterprise Edition” and click “Next."
7. The installer will now run the “Product-specific Prerequisite Checks." Click “Next” upon successful completion.
8. On the “Select Database Configuration” screen, choose the option “Do Not Create a Starter Database." Click “Next” and then click “Install” on the “Summary” screen.
9. Logged in as the root user, execute the script "/home/oracle/product/10g/root.sh" on the current node as/when directed.
    a. The script will start the “VIP Configuration Assistant” (VIPCA) on the current node. Click “Next” on the first (Welcome) screen.
    b. When prompted to select network interfaces, choose the public interfaces only (node1 and node2) and click “Next."
    c. When prompted to enter IP alias names and IP addresses, enter the ones created earlier:
        ```
        node1-vip 10.60.136.166
        node2-vip 10.60.136.167
        ```
    d. Click “Next” when finished. Examine the “Summary” screen and verify the configuration. Click “Finish."
    e. On the “Configuration Results” screen, click “Exit."
    f. On the remaining node(s), run the same script as the root user. VIPCA will not be run on the remaining nodes. The following messages will indicate the script has finished successfully:
        Now product-specific root actions will be performed.
        CRS resources are already configured.
    g. When the script is finished on all nodes, click “OK” on the “Setup Privileges” screen.
10. The RDBMS installation is now complete. Click “Exit” on the “End of Installation” screen.

Install ASMLib and create ASM disks

ASMLib is a library add-on for use with the Oracle automatic storage management feature. It was designed to simplify the management of raw devices used in the creation of ASM disk groups. Oracle recommends its use. According to Oracle, it provides for more efficient and capable access to ASM disk groups. Instructions for obtaining and installing the necessary Linux RPMs as well as instructions for creating ASM disk groups are listed below.

Obtaining the RPMs

1. At the time of this writing, the ASMLib RPMs were not being provided on Oracle distribution media. To obtain them, you will need to download them.
2. Navigate to the section that describes your Linux environment.
3. You will need to download three packages to each Linux host:
    - The RPM containing the userspace library
    - The RPM containing the driver support files
    - The driver for the kernel you are running on your Linux hosts

4. This document is based on a multiprocessor system running 32-bit Red Hat Enterprise Linux 3 (x86), so the required RPMs are as follows:
   - `oracleasmlib-1.0.0-1.i386.rpm: userspace library`
   - `oracleasm-support-1.0.2-1.i386.rpm: driver support files`
   - `oracleasm-2.4.21-EL-smp-1.0.0-1.i686.rpm: driver for “smp” kernels`

Installing and configuring the ASMLib RPMs

1. Copy the RPMs to a working directory and install each of them as follows in the order listed while logged in as the root user:
   - `rpm -Uvh oracleasm-support-1.0.2-1.i386.rpm`
   - `rpm -Uvh oracleasm-2.4.21-EL-smp-1.0.0-1.i686.rpm`
   - `rpm -Uvh oracleasmlib-1.0.0-1.i386.rpm`
2. Configure the ASM library driver on each Linux host while logged in as root:
   a. From a command line run the following:
      ```
      /etc/init.d/oracleasm configure
      ```
   b. Respond to the configuration prompts as follows:
      ```
      Default user to own the driver interface []: oracle
      Default group to own the driver interface []: dba
      Start Oracle ASM library driver on boot (y/n) [n]: y
      Fix permissions of Oracle ASM disks on boot (y/n) [y]: y
      ```

Creating ASM disks

1. To create the ASM disks, run these commands on the first Linux host while logged in as root:
   ```
   /etc/init.d/oracleasm createdisk VOLDB1
   /dev/iscsi/bus0/target1/lun3/disk
   /etc/init.d/oracleasm createdisk VOLDB2
   /dev/iscsi/bus0/target1/lun4/disk
   /etc/init.d/oracleasm createdisk VOLDB3
   /dev/iscsi/bus0/target1/lun5/disk
   /etc/init.d/oracleasm createdisk VOLLOG1
   /dev/iscsi/bus0/target1/lun6/disk
   /etc/init.d/oracleasm createdisk VOLLOG2
   /dev/iscsi/bus0/target1/lun7/disk
   /etc/init.d/oracleasm createdisk VOLLOG3
   /dev/iscsi/bus0/target1/lun8/disk
   ```
2. lun1 and lun2 will not be used by ASM. They will be used as the OCR disk and voting disk, respectively. You can verify the success of this operation by running the following command:
   ```
   /etc/init.d/oracleasm listdisks
   ```

   Below is the desired output:

   ```
   VOLDB1
   VOLDB2
   VOLDB3
   VOLLOG1
   VOLLOG2
   VOLLOG3
   ```

Use DBCA to create a database using ASM

Oracle strongly recommends its DBCA utility for creating OracleRAC10*g* databases, especially OracleRAC10*g* databases that will use ASM. The following instructions will give step-by-step procedures for doing this. Figure 2 is a graphical representation of the layout for our database:

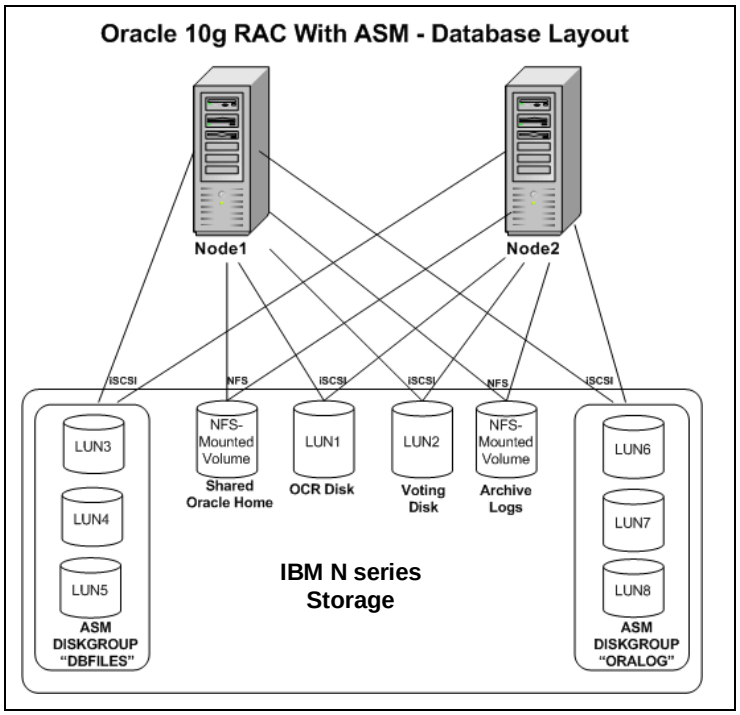

Figure 2) Database layout.

1. Log in to the first node as the Oracle user, change to the ORACLE_HOME/bin directory, and run the command "`./dbca`."
2. Choose the "Oracle Real Application Clusters Database" option and click "Next."
3. Choose the "Create a Database" option and click "Next."
4. Click "Select All" to select all nodes and then click "Next."
5. Choose the "Custom Database" option and click "Next."
6. For "Global Database Name," enter "rac." Note that the "SID Prefix" field will populate automatically. Click "Next" when you are ready to continue.
7. Check the "Configure the Database with Enterprise Manager" block, leave all the other blocks unchecked, and click "Next."
8. Configure passwords according to your organization's requirements and click "Next."
9. Choose "Automatic Storage Management (ASM)" for the storage mechanism and click "Next."
10. Enter the SYS password and confirm. Choose "Create initialization parameter file (IFILE)," accept the default "Initialization Parameter Filename," and click "Next" and then "OK."
11. On the "Available Disk Groups" screen, choose "Create New."
    1.1.1. Enter "dbfiles" for the "Disk Group Name" and click the "External" radio button. If you do not see a listing of member disks with a path beginning with "ORCL:," click "Change Disk Discovery Path" and enter "ORCL:*" and click "OK."

1.1.2. Choose "ORCL:VOLDB1," "ORCL:VOLDB2," and "ORCL:VOLDB3" and click "OK" to add the disks to disk group "dbfiles."

1.1.3. Once again, click "Create New." Enter "oralog" for the disk group name, click the "External" radio button, and choose "ORCL:VOLLOG1," "ORCL:VOLLOG2," and "ORCL"VOLLOG3." Click “OK” to add the disks to the disk group “oralog."

12. On the next screen, choose both disk groups (dbfiles and oralog) to be used as storage for the database and click “Next."
13. On the next screen, choose “Use Common Location for All Database Files." If “Database Files Location” has not defaulted to “+DBFILES,” click “Browse” and choose it from the list. Click “Next” to continue.
14. On the “Choose the recovery options for the database” screen, you may choose either “Specify Flash Recovery Area," “Enable Archiving,” or both. In this installation we will choose “Enable Archiving” only.
    a. Click “Edit Archive Mode Parameters." If you would like to store archive logs in an ASM disk group, set the “Archive Log Destination” to “+ORALOG/{DB_UNIQUE_NAME}/." (The string “{DB_UNIQUE_NAME}” is roughly equivalent to the database name and represents a parameter defined by the DBCA, which will be replaced with the actual value of the parameter by DBCA at the time of database creation. Unless you wish to hard code this portion of the destination path, it should be left unchanged. It should also be mentioned that this, along with the other database creation parameters, can be changed from within the DBCA GUI.) Please be advised that storing the archive logs in an ASM disk group will limit the IBM N series features you are able to use (such as IBM System Storage N series with SnapMirror and cloning) and make the management of archive log files dependent on Oracle’s Recovery Manager (RMAN) utility. RMAN is an excellent tool for managing backup and recovery of databases; however, for various reasons, many DBAs choose not to use it. In this installation we will use the NFS-mounted storage system volume “archive," mounted as “/archive” for storing archive logs. Therefore, set the “Archive Log Destination” to “/archive/” instead of “+ORALOG/{DB_UNIQUE_NAME}/," click “Automatic Archiving," accept the default “Archive Log File Format,” and click “OK." Click “Next” on the “Recovery Options” screen.
15. On the "Database Components" screen, choose the desired components, configure as desired, and click "Next" to continue.
16. On the “Database Services” screen, click “Next."
17. On the "Memory," "Sizing," etc. screen configure the database as desired and click "Next." For this installation “Typical" was chosen for memory allocation.
18. On the "Database Storage" screen, click "Controlfile." Change the disk group for each entry from +DBFILES to +ORALOG.
19. Click the “+” sign that precedes “Redo Log Groups." Click each group number and change the disk group from +DBFILES to +ORALOG in the same way. Please note that for storage system SnapRestore functionality you must not store redo logs or control files in the disk group containing Oracle data files. In the same way, verify that “+DBFILES” is set for the disk group to be used by tablespaces and data files.
20. Click “Next” to continue.
21. Choose the "Create Database" option and click “Finish."
22. Verify the configuration on the “Summary Screen” and click “OK” to begin database creation. Wait for the database creation screen to pop up. Database creation will start within a few seconds.
23. After the database creation is complete, enable asynchronous I/O as follows:

Asynchronous I/O is recommended for use with Oracle's ASMLib for Linux. Please note that asynchronous I/O should be used ONLY with the ASMLib driver. Do NOT use asynchronous I/O without the ASMLib driver. Use of asynch I/O without the ASMLib can result in iSCSI errors and database corruption. See section 13.2 below and Burt 139115 for more details. Below are the steps for enabling asynchronous I/O in Oracle:

a. The following libaio packages are required:

libaio-0.3.96-3.i386.rpm
libaio-devel-0.3.96-3.i386.rpm

They are included in the Red Hat EL AS 3.0 update 2 distribution (although versions are at 4.0 and later, 3.0 was the installed version at the time this paper was written), which is available from Red Hat. To determine whether they have been installed, run the following commands as the root user:

```
rpm –q libaio-0.3.96-3
rpm –q libaio-devel-0.3.96-3
```

If they are not installed, you must install them before continuing.

b. Shut down all Oracle instances cleanly (`shutdown immediate`).
c. Change directories into the RDBMS lib directory (`cd $ORACLE_HOME/rdbms/lib`).
d. Relink the Oracle binaries with the following command while logged in as the Oracle user:
```
make PL_ORALIBS=-laio -f ins_rdbms.mk async_on
```
e. Set the Oracle parameter "disk_asynch_io=true."
f. Set the Oracle parameter "`filesystemio_options =asynch`."
g. Restart all Oracle instances.

Backup and recovery

Hot (online) backup procedure

Backup of this database can be accomplished by placing the tablespaces in hot backup mode, creating a system Snapshot copy of the volume containing all the LUNs used in creating the database file ASM disk group, and then taking the tablespaces out of hot backup mode.

1. From one RAC node, place the database in hot backup mode. Example:
```
SQL> alter database begin backup;
```
2. Create a Snapshot copy on the system of the system volume that contains the LUNs used in creating the disk group containing all the Oracle database files. Example:
```
filer1> snap create dbfile backup1
```
Where "dbfile" is the name of the system volume and "backup1" is the name of the Snapshot copy.
3. Take the database out of hot backup mode. Example:
```
SQL> alter database end backup;
```
4. Sample listings of scripts for doing this are included in the appendix of this document.
5. To back up the archive logs, simply create a Snapshot copy of the /vol/archive volume. The database does not have to be in backup mode for this. It can be scheduled via cron job,

corresponding to the times the hot backups are performed. This requires no special interaction with the Oracle Database and therefore is not included in the sample script.

6. Backing up the spfile:
   a. If the spfile is stored in an ASM disk group, it will be backed up at the time the ASM disk group LUNs are backed up.
   b. If the spfile is not stored in an ASM disk group, it can be backed up by either creating a Snapshot copy of the volume it is stored in or simply executing a UNIX® "cp" command to make a copy of it.
   c. Regardless of where the spfile is located, it's a good idea to periodically to back it up as a pfile, using the sqlplus command "create pfile='pfile_path' from spfile='spfile_path';," where "pfile_path" is the target path for the pfile to be created and "spfile_path" is the path for the existing spfile. An editable pfile can be helpful in the event the spfile becomes corrupt or incompatible changes are made to the spfile, resulting in the database failing to open. Please note that the single quotes around the pfile path and spfile path, and the semicolon at the end of the command given in the example, are part of the required Oracle syntax for creating a pfile from an spfile.
   d. Backing up the spfile as described in items 12.1.6.2 and 12.1.6.3 above does not require the database to be in hot backup mode.
7. There are a few non-RMAN ways to back up a control file:
   a. `sqlplus> alter database backup control file to backup destination_path;`

      This command makes a binary copy of the control file that can be used during a recovery if the original one is lost or corrupt.

   b. `sqlplus> alter database backup control file to trace;`

      This command creates a SQL script for recreating the control file. It can be useful if you need to modify Oracle parameters. Simply modify the file and use it to create a new control file.

   Note: Neither of these commands involves Snapshot copies, and both are purely Oracle procedures performed from within a sqlplus session, so they are not included in the sample script either.

Recovery procedure

1. Shut down all instances of the database:
   ```
   SQL> shutdown immediate
   ```
   This will have to be done on each node.
2. Shut down all ASM instances:
   ```
   $ srvctl stop asm –n node1
   $ srvctl stop asm –n node2
   ```
3. Take all the LUNs used in the dbfile group offline:
   ```
   filer1> lun offline /vol/dbfile/lun3
   filer1> lun offline /vol/dbfile/lun4
   filer1> lun offline /vol/dbfile/lun5
   ```
4. Restore the data file volume:
   ```
   filer1> snap restore -f -s backup1 -t vol /vol/dbfile
   ```

5. Restart the ASM instances:

```
$ srvctl start asm –n node1
$ srvctl start asm –n node2
```

6. Restart and recover one database instance as follows:

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

7. Restart the remaining instances with the sqlplus "`startup`" command as usual.
8. As with the backup procedure, sample script listings are included in Appendix A.

Appendix A: Issues and examples

Oracle issues

Oracle Bug 3557807

Description: Oracle Cluster Manager (OCM) inadvertently uses public network instead of private network for interconnect, causing degraded performance.

Fix: Install Oracle patch 3557807 and set the Oracle parameter “cluster_interconnects” on each node to the IP address of the private network interface. Examples of commands to run from a sqlplus prompt on node1 and node2, respectively:

```
SQL> alter system set cluster_interconnects = '10.60.139.121' scope=spfile sid='tpcc1';

SQL> alter system set cluster_interconnects = '10.60.139.119' scope=spfile sid='tpcc2';
```

Each instance will have to be shut down and restarted for the changes to take effect.

Oracle Bug 3591633

Description: RDBMS 10.1.0.3 ASM PRODID-5 PORTID-46. Process crash if it is attached by debugger during I/O through ASMLib.

Symptoms observed: Oracle Database crashes during heavy I/O with the following error messages:

Corrupt block relative dba: 0x01400641 (file #, block 1601)
Bad header found during buffer read
Data in bad block:
type: 32 format: 2 rdba: 0x01400641
last change scn: 0x0000, 099ff061 seq: 0x1 flg: 0x04
spare1: 0x0 spare2: 0x0 spare3: 0x0
consistency value in tail: 0xf0612001
check value in block header: 0x8ea8
Bad header found during reread.

Fix: Install Oracle patch 3591633. Obtain the patch by accessing the “Patches” section of Metalink. Choose “Simple Search” and “Search By Patch Numbers." Enter the patch number, click “Go," save the patch, unbundled the files, and install it with opatch per the Readme file.

Oracle patches can be downloaded from Oracle as follows:

- Log into Metalink and access the “Patches” page.
- Choose the "Simple Search” and “Search By Patch Numbers” options.
- Enter the patch number.
- Save the patch to disk.
- Unbundle the files and use opatch to install it per the included Readme file. If you do not have opatch, it can be downloaded from the Metalink Patch page as well. To do so, perform a simple search on the patch download page, searching for patch 2617419. Download, unbundle, install, and use as instructed in the README.txt file.

iSCSI issues

Oracle recommends the use of its ASMLib driver with ASM. It is possible; however, to use ASM without ASMLib by setting the Oracle parameter “asm_diskstring” in the ASM instance to the raw device path (example: /dev/raw/*) instead of “ORCL:*." Please be advised that there is an iSCSI bug that can result in iSCSI errors and database corruption in the non-ASMLib configuration ***with asynchronous I/O enabled***. Burt 139115 has been filed to address this issue. Currently, the workaround is to disable asynchronous I/O if you plan to use ASM without ASMLib. Below is an example of the errors you can expect to see in the filer /etc/messages file if you encounter this problem:

> *Mon Aug 16 13:23:31 GMT [iscsi.protocol.violation:warning]: ISCSI: iswta, iSCSI protocol violation, 'immediate delivery overflow, opcode 0'*

Backup-script example

Contents of script hotbak.sh:

```
sqlplus system/manager @hotbak.sql
rsh -l root filer1 snap create dbfile backup1
sqlplus system/manager  @endbak.sql
```

Contents of SQL script hotbak.sql:

```
alter database begin backup;
create pfile=’pfile_path’ from spfile=’spfile_path’;
exit
```

Contents of SQL script endbak.sql:

```
alter database end backup;
exit
```

The above sample script listings are given purely for illustration.

Restore-script example

do_snaprestore.sh:

```
srvctl stop instance -d rac -i rac1
srvctl stop instance -d rac -i rac2
srvctl stop asm -n node1
srvctl stop asm -n node2
rsh -l root filer1 lun offline /vol/dbfile/lun3
rsh -l root filer1 lun offline /vol/dbfile/lun4
rsh -l root filer1 lun offline /vol/dbfile/lun5
rsh -l root filer1 snap restore -f -s hot4 -t vol /vol/dbfile
rsh -l root filer1 lun online /vol/dbfile/lun3
rsh -l root filer1 lun online /vol/dbfile/lun4
rsh -l root filer1 lun online /vol/dbfile/lun5
```

The above sample script listing is given purely for illustration.

Trademarks and special notices

© International Business Machines 1994-2007. IBM, the IBM logo, System Storage, and other referenced IBM products and services are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. All rights reserved

References in this document to IBM products or services do not imply that IBM intends to make them available in every country.

The Network Appliance logo, Data ONTAP, SnapMirror and SnapRestore are registered trademarks and Network Appliance, DataFabric, and Snapshot are trademarks of Network Appliance, Inc., in the U.S. and other countries.

Oracle is a registered trademark of Oracle Corporation.

Intel, Intel Inside (logos), MMX, and Pentium are trademarks of Intel Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Linux is a trademark of Linus Torvalds in the United States, other countries, or both.

Other company, product, or service names may be trademarks or service marks of others.

Information is provided "AS IS" without warranty of any kind.

All customer examples described are presented as illustrations of how those customers have used IBM products and the results they may have achieved. Actual environmental costs and performance characteristics may vary by customer.

Information concerning non-IBM products was obtained from a supplier of these products, published announcement material, or other publicly available sources and does not constitute an endorsement of such products by IBM. Sources for non-IBM list prices and performance numbers are taken from publicly available information, including vendor announcements and vendor worldwide homepages. IBM has not tested these products and cannot confirm the accuracy of performance, capability, or any other claims related to non-IBM products. Questions on the capability of non-IBM products should be addressed to the supplier of those products.

Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. The actual throughput or performance that any user will experience will vary depending upon considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve throughput or performance improvements equivalent to the ratios stated here.

Photographs shown are of engineering prototypes. Changes may be incorporated in production models.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.