

Oracle9i Real Application Clusters
Quick Installation Guide
for Linux x86

October 2003

Oracle9i Real Application Clusters
Quick Installation Guide
for Linux x86

Introduction	1
Release 2 Updates and Changes	1
JVM Requirement	1
Installation Procedure with Oracle Universal Installer	1
Global Services Daemon	1
Cluster Communication Protocol	1
System Requirements	1
Operating System Requirements	2
Oracle Software Requirements	2
Hardware Requirements	2
Storage Configuration Options	3
Network Attached Storage (NAS)	3
Preinstallation	3
Operating System Kernel and Utilities Setup	4
Operating System Kernel and User Limits Configuration	4
Users and Groups Configuration	6

Activate Remote Services	7
Network Configuration	7
Configuring the Public Network	7
Configuring the Private Network	8
Network Connectivity Test	10
Storage with Oracle Cluster File System	11
Installing OCFS	11
Configuring OCFS	13
Creating Partitions and Formatting OCFS Mount Points	14
Starting OCFS	15
Creating Quorum Disk and SERVCONFIG Files	16
Cluster Software Layer with Cluster Manager	16
Preinstallation Steps for Oracle Cluster Manager Software	16
Installing Oracle Cluster Manager Software	16
Installing 9204 Oracle Cluster Manager Patch Set	18
Patching Oracle Cluster Manager to Release 9.2.0.4	19
Modifying the Oracle Cluster Manager Configuration Files	20
Creating Oracle Cluster Manger and Hangchecktimer Startup Scripts	21
Installing Oracle9i Release 2 (9.2.0.1)	23
Installing Oracle Universal Installer Release 2.2.0.18	25
Applying the Oracle9i 9.2.0.4.0 Patch Set	25
Database Creation and Configuration	26

Creating the Server Configuration File	26
Initializing the Server Configuration File	26
Starting the Global Services Daemon	26
Running Database Configuration Assistant	26
Configuring SRVCTL	28
Verifying Seed Database	29
Appendix A: OCFS Startup Script	30
REFERENCES	34

Oracle9i Real Application Clusters Quick Installation Guide for Linux x86

Introduction

The purpose of this document is to provide a step-by-step overview of the Oracle9i Real Application Cluster installation process for Linux x86 platforms. It is intended to be used in concert with the standard Oracle documentation for Real Application Clusters installation, and to assist users with installation on x86-based processor Linux platforms.

Release 2 Updates and Changes

The following is a list of significant changes for Oracle9i Release 2 (9.2.0.1) with Oracle Real Application Clusters on the Linux x86 platform:

JVM Requirement

It is no longer required to load JVM prior to beginning the Oracle installer. JVM is now provided with the release.

Installation Procedure with Oracle Universal Installer

Installation with the Oracle Universal Installer now requires two steps:

1. Install and configure the Oracle Cluster Manager.
2. Install Oracle Real Application Clusters.

Global Services Daemon

The global services daemon is now started with the new `gsdctl` command.

Cluster Communication Protocol

Cluster Manager (CM) communication now uses UDP instead of TCP.

System Requirements

The following sections describe the system requirements:

- Operating System Requirements
- Oracle Software Requirements

- Hardware Requirements

Operating System Requirements

Ensure that your system meets the following minimum operating system requirements:

- Red Hat Advanced Server 2.1, with kernel version 2.4.9 and with the latest errata

Oracle Software Requirements

Ensure that you have the following Oracle software installed, or available for installation

- Oracle9i Release 2 (9.2.0.1)
- Oracle9i Release 2 (9.2.0.1) Patch Set 9.2.0.4 or later
- Latest Oracle Cluster File System (OCFS) release

Hardware Requirements

Ensure that your system meets the following requirements:

- Two or more Intel-based servers, with at least two CPUs of 700 Mhz or faster for each server
- 512 MB Memory (2 GB or more recommended)
- Swap space equal to 2 to 3 times physical memory (4 GB recommended)
- Shared SCSI disks, or fiber-channel (FC) disk array
- Daisy-chain connection of SCSI or FC host adapters and shared disks
- One-to-one connection of SCSI or FC adapters and shared disks
- Multi-port disk
- Dedicated interconnect network connection
- Public Networks
- Servers racked and stacked with all network and shared storage interconnects in place
- At least three LUNs should be configured on the shared storage for use by the Oracle9i Real Application Clusters seed database
- Recommended internal disk partition size for each server and Oracle Database instance (18 GB in size) in accordance with the following table:

Partition	Size
/	2 GB, ext3
/boot	64 MB, ext3
/home	2 GB, ext3
/opt/oracle	Mount point for Oracle executables: 3.5 GB, to a maximum of 7000 GB, ext3
/tmp	1 GB, ext3
/usr	3 GB (3 GB Advanced Server install, 7 GB for all package install, ext3
/var	2148 MB, ext3

Partition	Size
swap	At least the size of physical memory, 2 or 3

Storage Configuration Options

For Oracle9i Real Application Clusters shared storage is required for the database. Select from the following options:

- SAN
 - Raw Devices
 - Clustered File Systems

Currently, the only cluster file system option is Oracle Cluster File System (OCFS) which can be downloaded from the following Web site:

<http://otn.oracle.com>

OCFS is supported on Red Hat AS 2.1, and on United Linux 1.0.

Note: For information on how to install, configure, and optimize Oracle Cluster File System, refer to Oracle*MetaLink* Note 238635.1 available at the following Web site:
<http://metalink.oracle.com>

Network Attached Storage (NAS)

Refer to your network vendor's 'Best Practices' configuration instructions for network attached storage hardware requirements. The operating system and hardware must be set up and configured in accordance with the requirements for Oracle9i Real Application Clusters.

The storage setup and configuration (SAN, NAS, or SCSI) and LUNs must be done at the time of hardware setup.

Preinstallation

The following sections describe the preinstallation steps for each physical node (server) in the cluster:

- Operating System Kernel and Utilities Setup
- Operating System Kernel and User Limits Configuration
- Users and Groups Configuration
- Activate Remote Services
- Network Configuration
- Network Connectivity

Operating System Kernel and Utilities Setup

Complete the following system file and environment variable setup:

1. Upgrade the default kernel version (2.4.9-e.3) to Errata e.16 or later (2.4.9-e.16) immediately after initial installation.

Note that the command to install a new kernel [# rpm -ivh kernel-...] is different from the command to install a kernel upgrade [# rpm -Uvh kernel-...]. Installation creates a new kernel of the later version, but retains the original kernel as a fail back option. An upgrade replaces the original kernel.

If you upgrade to a later kernel and use lilo boot manager, then modify /etc/lilo.conf to reflect the upgraded /boot/kernel file names, remembering to lilo the file afterward. The grub utility automatically modifies its own configuration file, /etc/grub.conf.

See Also: For a complete list of Oracle supported kernel versions, review OracleMetaLink Note:225303.1: How To Check the Supportability of RedHat AS.

2. Restart the server to the new kernel. Configure any devices newly discovered by kudzu.
3. Enter the command `uname -a`. You should receive a response similar to the following example:

```
# uname -a
Linux arachnid 2.4.9-e.16 #1 Fri Mar 21 05:55:06 PST 2003 i686 \
unknown
```

4. Use the following command to determine system memory use:

```
$ grep MemTotal /proc/meminfo
```

5. Use the following command to determine swap space use:

```
$ grep SwapTotal /proc/meminfo
```

6. Ensure correct kernel and glibc versions.

- a. To determine the kernel version, enter the following command:

```
$ uname -a | awk '{print $3}'
```

- b. To determine the glibc version:

```
$ rpm -qa|grep glibc
```

The command should return the following:

```
glibc-2.2.4-32.3
```

Operating System Kernel and User Limits Configuration

Review Oracle documentation to ensure that your system meets the minimum Oracle requirements, and then configure the kernel and user limits according to available resources.

The following section shows the contents of core initialization, and of the configuration files used to tune the kernel and increase user limits. Consult with your system administrator (SA) or Red Hat before implementing any changes with which you are unfamiliar.

As the root user, adjust interprocess communication settings and shared memory settings in the

/etc/sysctl.conf file on both the nodes and restart for the new settings to take effect.

For node 1:

```
[root@stc-raclin01 root]# vi /etc/sysctl.conf
# IPC memory and filesystem settings
fs.aio-max-size =1048576
fs.file-max = 327680
fs.aio-max-nr = 1048576
vm.pagecache = 10 20 30
net.core.rmem_max=262144
net.core.wmem_max=262144
net.core.rmem_default=262144
net.core.wmem_default=262144

# Shared memory and Semaphore memory settings
kernel.shmmax = 47483648
kernel.shmmin = 1
kernel.shmmni = 1024
kernel.shmseg = 10
kernel.sem = 100 32000 32 128

~
"/etc/sysctl.conf" 28L, 622C written
```

For node 2:

```
[root@stc-raclin02 root]# vi /etc/sysctl.conf
# IPC memory and filesystem settings
fs.aio-max-size =1048576
fs.file-max = 327680
fs.aio-max-nr = 1048576
vm.pagecache = 10 20 30
net.core.rmem_max=262144
net.core.wmem_max=262144
net.core.rmem_default=262144
net.core.wmem_default=262144

# Shared memory and Semaphore memory settings
kernel.shmmax = 47483648
kernel.shmmin = 1
kernel.shmmni = 1024
kernel.shmseg = 10
kernel.sem = 100 32000 32 128

~
"/etc/sysctl.conf" 28L, 622C written
```

For [/etc/rc.local]

```
#!/bin/sh
touch /var/lock/subsys/local
/sbin/inssmod hangcheck-timer hangcheck_tick=30 hangcheck_margin=180
```

Note: In Real Application Clusters configurations with 2 or more nodes, an I/O fencing model is required to detect when one or more nodes die or become unresponsive. This helps to prevent data corruption such as a node in an unknown state continuing to write to the shared disk.

In release 9.2.0.1.0, Oracle originally recommended using the `softdog` module (also known as `watchdog`) as the I/O fencing model. However, due to performance and stability issues when using `watchdog` with the `/dev/watchdog` device, Oracle recommends using `/dev/null` as the `watchdog` device file.

In releases 9.2.0.2.0 and later, Oracle recommends using a new I/O fencing model, the `hangcheck-timer` module. Oracle Cluster Manager configuration changes are required if you have already implemented Real Application Clusters using release 9.2.0.1.0, and then upgrade to release 9.2.0.2.0 or later. The reason for the I/O fencing model change and `hangcheck-timer` configuration requirements are discussed in the Oracle Server release 9.2.0.2.0 and later patch set notes.

Users and Groups Configuration

For the Oracle installation to work successfully there needs to be an `oracle` user created at the operating system level. Once the `oracle` account is created, it must be assigned membership in specific operating system groups.

Use the following guide line for creating the 'oracle' user account

Log in the server as the `root` user and type the following commands to add the `oracle` user to the groups:

```
# groupadd -g 500 dba
# groupadd -g 501 oinstall
# useradd -u 500 -g dba -G oinstall -m -s /bin/bash oracle
Please note /bin/bash could be changed according to shell being used
Run "passwd oracle" to set oracle user's password.
a. At the "New password:" prompt, enter "oracle".
b. At the "Retype new password:" prompt, enter "oracle" again.
```

Note that you use the uppercase `G` to assign primary groups, and the lowercase `g` to assign secondary groups.

To verify that the `oracle` user has been added to the groups and that the `oinstall` group is set as primary, enter the following commands:

```
# su - oracle
$ id
$ uid=500(oracle) gid=501(oinstall) groups=501(oinstall),500(dba)
```

Note: You can use the Red Hat utility NEAT to set up users and groups.

Use the following table to set the Oracle user environment variables:

Environment Variable	Suggested value
CLASSPATH	<code>\$ORACLE_HOME/JRE:\$ORACLE_HOME/jlib \</code> <code>\$ORACLE_HOME/rdbms/jlib:</code> <code>\ \$ORACLE_HOME/network/jlib</code>
LD_LIBRARY_PATH	Should contain <code>\$ORACLE_HOME/lib:\$ORACLE_HOME/oracm/lib</code>
NLS_LANG	Your preferred language variable, such as <code>AMERICAN_AMERICA.UTF8</code>
ORA_NLS33	<code>\$ORACLE_HOME/ocommon/nls/admin/data</code>
ORACLE_BASE	<code>/opt/oracle</code> , or similar
ORACLE_HOME	<code>\$ORACLE_BASE/product/9.2.0</code>
PATH	Should contain <code>\$ORACLE_HOME/bin</code> and <code>\$ORACLE_HOME/oracm/bin</code>
THREADS_FLAG	<code>native</code>
TNS_ADMIN	<code>\$ORACLE_HOME/network/admin</code>

Activate Remote Services

As the `root` user, run the following commands:

```
Run "chkconfig rsh on".
Run "chkconfig rcp on".
Run "chkconfig rlogin on".
Run "chkconfig telnet on".
```

Network Configuration

After the groups and the environment variables are set for the oracle user, set up the network interface card (NIC), and to make appropriate changes to the `/etc/hosts` and `/etc/hosts.equiv` files.

The purpose of these files is to allow communication between each of the cluster nodes.

Network setup must be done on all nodes in the cluster. There are two types of networks, public and private. Each must be configured for all nodes on the cluster.

Configuring the Public Network

Use the following procedure to assign a public host name to a node:

1. Enter the following command to open the network configuration file in the vi editor:

```
vi /etc/sysconfig/network
```

2. Edit the line containing the parameter `HOSTNAME=yourhostname.yourdomain` to assign the node name, and to assign the domain.

For example, to assign `node1-pub` as the node name and `oracle.com` as the domain, the parameter `HOSTNAME` entry should look like the following:

```
HOSTNAME=node1-pub.oracle.com
```

3. Restart the system, or restart the network services to implement the change. To restart network services, enter the following command:

```
service network restart
```

You must restart the system, or restart the network services to implement the change.

4. Use one of the following methods to assign a public IP address for NIC:
 - Use the Red Hat utility `netconfig/neat` to assign either a static IP address on each node in the cluster
 - Edit the public network interface card (NIC) file `/etc/sysconfig/network-scripts/ifcfg-xxx`, where `xx` is the public NIC that is connected to a public network switch.

If you edit the public NIC file, then use the following example as a guide:

```
DEVICE=eth0
ONBOOT=yes
BOOTPROTO=static
IPADDR=node1-pub.oracle.com
netmask=255.255.255.0
```

Note: In the preceding example, a public NIC was addressed with the IP address `node1-pub.oracle.com`, and the netmask address `255.255.255.0`.

5. Restart the system, or restart the network services to implement the change. To restart network services, enter the following command:

```
service network restart
```

Configuring the Private Network

You must assign a hostname and a private hostname to each node in the cluster, if your hardware vendor or system administrator has not already done so.

To assign a private hostname to a node, use one of the following methods:

- Use the Red Hat utility `netconfig/neat` to assign IP address on each node in the cluster.
- Edit the private NIC file `/etc/sysconfig/network-scripts/ifcfg-xxx`, where `xxx` is the private NIC that is connected to a private network switch.

If you edit the private NIC file, then use the following example as a guide:

Assign the private host names racnode1 and racnode2 to a two-node cluster. The private IP address is 192.168.0.1 for node 1, and 192.168.0.2 for node 2. The netmask address is 255.255.255.0. Keep a record of the name you assign as the private hostname. In this example, add the following text to the NIC file:

For node 1:

```
DEVICE=eth1
ONBOOT=yes
BOOTPROTO=static
IPADDR=192.168.0.1
NETMASK=255.255.255.0
HOSTNAME=racnode1
```

For node 2:

```
DEVICE=eth1
ONBOOT=yes
BOOTPROTO=static
IPADDR=192.168.0.2
NETMASK=255.255.255.0
HOSTNAME=racnode2
```

To implement the change, you must restart the system, or restart the network services to implement the change. To restart network services, enter the following command:

```
# service network restart
```

For /etc/hosts, both the public and private network interfaces must be defined. The following is an example of a network interface definition file:

```
# Do not remove the following line, or various programs
# that require network functionality will fail.
#127.0.0.1 localhost.localdomain localhost
192.168.0.1 node1
192.168.0.2 node2
192.168.0.3 node3
192.168.0.4 node4
10.180.94.1 node1-pub.us.oracle.com node1-pub
10.180.94.2 node2-pub.us.oracle.com node2-pub
10.180.94.3 node3-pub.us.oracle.com node3-pub
10.180.94.4 node4-pub.us.oracle.com node4-pub
```

The following is an /etc/hosts.equiv file example:

```
Node1 oracle
Node2 oracle
Node3 oracle
Node4 oracle
Node1-pub oracle
Node2-pub oracle
```

```
Node3-pub oracle
Node4-pub oracle
```

In the preceding example, the first field contains the private or public node names, such as node1, and the second field contains the user account name, oracle, that is permitted to run rsh commands on the specified nodes.

The `/etc/hosts` and `/etc/hosts.equiv` files should be identical on all nodes in the cluster.

Network Connectivity Test

It is important to check the network connectivity between the planned nodes for the Real Application Clusters cluster before proceeding with any of the installation procedures below. Log in to each Real Application Clusters server and test connectivity of both the public and private network connections using the ping command

To confirm public and private network functionality, complete the following tests:

1. Check node one using the following procedure:
 - a. Ping node 2 using the private hostname.
 - b. Ping node 2 using the public hostname.
 - c. Ping node 2 using the private IP address.
 - d. Ping node 2 using the public IP address.
 - e. If testing a four-node cluster, then repeat the preceding checks for node 3 and 4.
2. Check node two using the following procedure:
 - a. Ping node 1 using the private hostname.
 - b. Ping node 1 using the public hostname.
 - c. Ping node 1 using the private IP address.
 - d. Ping node 1 using the public IP address.
 - e. If testing a four-node cluster, then repeat the preceding checks for node 3 and 4.
3. Test remote shell hostname equivalency using the following procedure:
 - a. Log on as the oracle user on node one.
 - b. Check for equivalency from node 1 using the following command syntax:

```
$ rsh privatehostnamenode2 [pwd | hostname]
```

In the preceding command, the variable *privatehostname2* is the hostname of node 2, using either `pwd` or `hostname` options. Use this command with its options to obtain a response, from either the current directory or hostname to node 2. Perform this step for each node in the cluster.

For example:

- Using the command option `pwd`, if the private hostname on node2 is sales, then enter the following command:

```
rsh sales pwd
```

The response is sales, and the current directory of node 2.

- Using the command option `hostname`, if the private hostname on node2 is sales, then enter the following command:

```
rsh sales hostname
```

The response is sales, and the hostname of node 2.

4. Test remote copy equivalency using the following procedure:

- a. Use the following command to create an rcp test file on node 1:

```
touch /tmp/rcptest
```

- b. Use the following command syntax to copy the rcp test file to node 2:

```
rcp /tmp/rcptest racnode2:/tmp/rcptest
```

In the preceding command, *racnode2* represents the private hostname of node 2.

- c. Check node 2 to ensure that the rcp test file was copied to `/tmp/rcptest` on node 2.
- d. Repeat this process for each node of the cluster, replacing the private hostname variable with the hostname of the node to which you copy the file.

Storage with Oracle Cluster File System

Oracle Cluster File System (OCFS) presents a consistent file system image across the servers in a cluster. OCFS allows administrators to take advantage of a filesystem for Oracle database files, such as data files, control files, and archive logs, and configuration files. This eases administration of Oracle9i Real Application Clusters. When installing Real Application Clusters on a 2-node or greater cluster, OCFS provides an alternative to having to use raw devices.

Create an OCFS configuration file to set up cluster file system storage. You can create the OCFS configuration file either by using the OCFS configuration tool, or by completing the task manually. Completing the task manually helps to ensure that you configure the correct node names.

Installing OCFS

Use the following procedure to install OCFS:

OCFS is offered under GNU Public License (GPL). At time of this writing, the latest version is 2.4.9-e-1.0.9-9, which is suitable for kernel versions 2.4.9-e.12 and later.

1. Download the appropriate version of OCFS for your system. OCFS is readily available for download from the following URL:

<http://oss.oraclecorp.com/>

2. Ensure you have the OCFS packages installed in a directory such as /tmp on all nodes.

You must download the smp or enterprise OCFS kernel which corresponds to your system's kernel type (smp or enterprise), and support and tools packages. The minimum required files are:

```
ocfs-2.4.9-e-`type`-1.0.9-9.i686.rpm
```

where `type` is either `smp` or `enterprise`

```
ocfs-support-1.0.9-9.i686.rpm
```

```
ocfs-tools-1.0.9-9.i686.rpm
```

- OracleMetaLink Note:220178.1 OCFS Setup
- OracleMetaLink Notes:224685.1, and 237997.1, OCFS Best Practices
- Red Hat Advanced Server Best Practices White Paper. This paper is available at the following link:

```
http://oss.oracle.com/projects/ocfs/dist/documentation/RHAS\_best\_practices.txt
```

3. Install the OCFS software. OCFS must first be configured before you create any OCFS volumes. Guidelines, limitations, and instructions for configuring OCFS are available.

The following commands should be run as the root user on all the nodes.

```
cd /tmp
```

```
rpm -Uvh ocfs-support-1.0.9-9.i686.rpm
```

```
rpm -Uvh ocfs-2.4.9-e-enterprise1.0.9-9.i686.rpm
```

```
rpm -Uvh ocfs-tools-1.0.9-9.i686.rpm
```

They all can be run as one command such as the following:

```
rpm -Uvh ocfs-support-1.0.9-9.i686.rpm -Uvh ocfs-2.4.9-e-enterprise-1.0.9-9.i686.rpm -Uvh ocfs-tools-1.0.9-9.i686.rpm
```

To receive a complete list of files installed as part of OCFS, query the RPM database or packages, as shown in the following example:

```
[root@arachnid /rpms]# rpm -qa | grep -i ocfs
ocfs-support-1.0.9-9
ocfs-2.4.9-e-enterprise-1.0.9-9
ocfs-tools-1.0.9-9
[root@arachnid /rpms]# rpm -ql ocfs-support-1.0.9-9
/etc/init.d/ocfs
/sbin/load_ocfs
/sbin/mkfs.ocfs
/sbin/ocfs_uid_gen
[root@arachnid /rpms]# rpm -ql ocfs-2.4.9-e-1.0.9-9
/lib/modules/2.4.9-e-ABI/ocfs
/lib/modules/2.4.9-e-ABI/ocfs/ocfs.o
```

```
[root@arachnid /rpms]# rpm -ql ocfs-tools-1.0.9-9
/usr/bin
/usr/bin/cdslctl
/usr/bin/debugocfs
/usr/bin/ocfstool
/usr/share
/usr/share/man
/usr/share/man/man1
/usr/share/man/man1/cdslctl.1.gz
/usr/share/man/man1/ocfstool.1.gz
```

Note: the OCFS installation automatically creates the necessary rc (init) scripts to start OCFS on server(see sample ocfs startup script) reboot i.e.:

```
[root@arachnid /]# find . -name '*ocfs*' -print
...
./etc/rc.d/init.d/ocfs
./etc/rc.d/rc3.d/S24ocfs
./etc/rc.d/rc4.d/S24ocfs
./etc/rc.d/rc5.d/S24ocfs
```

Configuring OCFS

Use the ofstool tool to generate the configuration file. This creates `/etc/ocfs.conf` file be used by `ocfs_uid_gen`. Each node in the cluster has a unique global user ID (guid). The following is an example for node 2:

```
node_name = node2
node_number =
ip_address = 192.168.0.2
ip_port = 7000
guid = B3DA4038931A55419A7900065B19C6BE
```

1. As the root user, run the following commands:

```
Load_ocfs
ocfstool &
```

2. Select Generate Config from the Tasks menu.
3. Select eth1 as the interface from the menu.
4. Enter `raclocal` for the node name. This the private host/node name.
5. Click **OK**.
6. Select Exit from the File menu.
7. Repeat these steps for each node in the cluster.

Creating Partitions and Formatting OCFS Mount Points

Complete the following procedure to create partitions and format OCFS mount points:

1. As the `root` user, run the following command:

```
fdisk /dev/sdd (one of the devices )
```

2. Enter the following when prompted by the system:

- a. Enter `n` at the first prompt.

- b. Enter `p` at the second prompt.

- c. Enter `1` for the partition type.

- d. Press `Enter` when prompted for the first cylinder

- e. Press `Enter` when prompted for the last cylinder

- f. Enter `w` at the next prompt

- g. Repeat above steps for all devices that will be used for OCFS, such as `sdd`, and `sd`.

3. Make mount points for each of the OCFS partitions by creating the directories for the locations where partitions will be mounted, and set the ownership on both nodes:

```
mkdir -p /u01 /u02 /u03  
chown -R oracle.oinstall /u01 /u02 /u03
```

4. Format the partitions and assign to OCFS by running the following commands or use `ocfstool` on the first node only:

```
mkfs.ocfs -F -b 128 -L /u01 -m /u01 -u 200 -g 300 -p 0775 /dev/sdb1
```

```
mkfs.ocfs -F -b 128 -L /u02 -m /u02 -u 200 -g 300 -p 0775 /dev/sdc1
```

```
mkfs.ocfs -F -b 128 -L /u03 -m /u03 -u 200 -g 300 -p 0775 /dev/sdd1
```

In the preceding syntax example, `u` represents the oracle user identifier (uid), and `g` represents the `oinstall` group identifier (gid).

5. Run the following command as the `root` user,;

```
ocfstool &
```

6. Select `Format` from the `Tasks` menu.

7. Select `/dev/sdd1` as the device from the menu.

8. Enter `ocfs` for the volume label from the menu.

9. Enter `/u03` for the mount point from the menu.

10. Select `oracle` as the user from the menu.

11. Select `dba` as the group from the menu.
12. Click **OK**.
13. Click **Yes** when prompted by the system to format the disk.
14. Repeat for all other partitions for OCFS that need to be formatted.
15. Select Exit from the File menu when done formatting.
16. Create a file named `/etc/cfstab` with the following values on both nodes:

```
/dev/sdb1 /u01 uid=200,gid=300
/dev/sdc1 /u02 uid=200,gid=300
/dev/sdd1 /u03 uid=200,gid=300
```

The OCFS filesystems can also be entered into the `/etc/fstab` to be used by the `ocfs` startup script created during OCFS installation.

24. Run the following command as the `root` user:

```
xedit /etc/fstab &
```

25. Add the following lines to the end of the file:

```
/dev/sdd1 /ocfs ocfs _netdev 0 0
/dev/sdc1 /ocfs ocfs _netdev 0 0
/dev/sdb1 /ocfs ocfs _netdev 0 0
```

Note: Be sure to press Enter after the last 0 at the end of the line.

Otherwise, when the mount command is run, it will show the following warning:

```
"[mntent]: warning: no final newline at
the end of /etc/fstab"
```

26. Click **Save** at the top of the window.
27. Click **Quit** at the top of the window.
28. Modify the startup script, if needed. During OCFS configuration, OCFS creates the OCFS startup script `/etc/init.d/ocfs`. It loads automatically on restarting the system. You can modify the startup script to change the location of mounted files.

If the startup script was not created, then use the sample 'cfs' script in appendix B to load OCFS modules and mount the filesystem:

Starting OCFS

The OCFS services will start automatically on all nodes at restart of the system, if the startup script created by OCFS was used. If you created a new startup script as shown in Appendix B, then enter the following commands to start OCFS manually.

```
chkconfig cfs on
service cfs start
```

Creating Quorum Disk and SERVCONFIG Files

To create the CMDISK and SERVCONFIG files, enter the following commands:

```
su - oracle
touch /u01/cmdiskfile
touch /u01/srvconfigfile
exit
```

Cluster Software Layer with Cluster Manager

Real Application Clusters on Linux requires a cluster software layer. Oracle supplies the Oracle Cluster Manager software with Oracle9i for Linux. Oracle Cluster Manager software consists of the following components:

- Watchdog timer (release 9.2.0.1) or Hangcheck Thread driver (release 9.2.0.2 and later)
- Oracle Cluster Manager (ORACM)

Preinstallation Steps for Oracle Cluster Manager Software

You must set some Linux environment parameters used with Cluster Manager before installing the Oracle software.

To test the Oracle environment parameters are correctly set, enter the following command:

```
$ env | grep ORA
```

This should return the parameters ORACLE_BASE and ORACLE_HOME. Do not proceed if those parameters are not set. If the parameters are not returned by the system, then add them in the file /home/oracle/.bash_profile. The following is an example of what should be added to the file:

```
export ORACLE_BASE=/opt/oracle
export ORACLE_HOME=$ORACLE_BASE/product/9.2.0
OH=$ORACLE_HOME
export PATH=$PATH:$OH/bin
export LD_LIBRARY_PATH=$OH/lib:$OH/oracm/lib
export CLASSPATH=$OH/JRE:$OH/jlib:$OH/rdbms/jlib:$OH/network/jlib
export THREADS_FLAG=native
# chown oracle:dba /home/oracle/.bash_profile
# chmod u+x /home/oracle/.bash_profile
```

Installing Oracle Cluster Manager Software

Make sure that the gcc 2.9* compiler is installed, is in the path, and has at least a symlink at: /usr/bin/gcc, otherwise linking Oracle will fail. As the oracle user, enter the following commands:

```
$ gcc -v
$ ls -l /usr/bin/gcc
```

The first command should point to gcc binary, and the second command should return 2.96, not 3.*.

Complete the following procedure to install Oracle Cluster Manager Software release 9.2.0.1:

1. If you set environment parameters as the `root` user, then log out, and log in as the `oracle` user. To log out on Red Hat, click **Start**, then click Log out. Log in again as the `oracle` user.
2. Open a terminal text windows by right clicking the blue screen area and selecting New Terminal.
3. Log in as the `root` user on the Linux node designated as your primary server node.
4. Insert Disk 1 for Oracle9i on Linux in the CD-ROM on the primary server node.
5. Use the following command to mount the CD-ROM as the `root` user:

```
# mount /mnt/CDROM
```

6. Log in as the `oracle` user account by opening another window, or by typing the following command:

```
# su - oracle
```

7. Ensure that all required environment variables are set, including the JDK path before beginning the Oracle software installation process. Use the following command to display the environment variables.

```
$ env | more
```

If you fail to do this, then the installation will fail.

9. Run the Oracle installer using the following command:

```
$ /mnt/cdrom/runInstaller
```

Note: Do not change directory to `/mnt/CD-ROM` before running the `runInstaller` script, or you will be unable to unmount Disk1 and mount Disk 2 and Disk 3 when installing from CD-ROMs. Use the full path name when entering the command.

10. Select the option to install the Oracle Cluster Manager software only, and accept default values for watchdog timings.

The following procedure describes the steps that occur during installation:

- a. The Welcome window opens. Click **Next**.
- b. The Inventory Location window opens. Accept the default `/opt/oracle/oraInventory`, and click **OK**.
- c. The UNIX Group Name window opens. Accept the default `dba` or enter `oinstall`, and click **Next**.
- d. The Root Script window opens. When prompted, open another window, and log in as the `root` user.
- e. In the root terminal window, run the script `oraInstRoot.sh`.

```
# /tmp/orainstRoot.sh
```

- g. When the script completes, return to the Root Script window, and click **Continue**.
- h. In the File Locations window, check the defaults if not correct enter correct name and path. Click **Next**.
- i. The Available Products window opens. Select Oracle Cluster Manager 9.2.0.1.0, and click **Next**.
- j. The Public Node Information window opens. Enter the hostname in the first entry, and that of the peer node in the second. Click **Next**.
- k. The Private Node Information window opens. Enter the hostname in the first entry and that of the peer node in the second. Click **Next**.
- l. The Watchdog Parameter Information window opens. Accept the default value for the Watchdog Parameter Value, and click **Next**.
- m. The Quorum Disk Information window opens. Enter the full pathname.

Note: Both nodes should use the same filename. The filename syntax should be in the format `/u01/cmdiskfile`. The file was created in a preceding step.

- n. When you have entered the hostname, click **Next**. The Summary window opens.
- o. Click **Install**.
- p. When installation has completed, click **Exit**.
- q. Click **Yes** when prompted to exit.

Installing 9204 Oracle Cluster Manager Patch Set

Complete the following procedure to install the Oracle Cluster Manager release 9.2.0.4 or later patch set:

1. Download the latest patch set from *OracleMetalink*.
2. Enter 3095277 in the Patch Number field.
3. Click **Go**.
4. Click **Download**.
5. Place the file in a patch set directory on the node used to install Cluster Manager such as the following:

```
$ mkdir $ORACLE_HOME/9204  
$ cp p3095277_9204_LINUX.zip $ORACLE_HOME/9204
```

6. Unzip the file using the following commands:

```
$ cd $ORACLE_HOME/9204  
$ unzip p3095277_9204_LINUX.zip
```

```
Archive: p3095277_9204_LINUX.zip
inflating: 9204_lnx32_release.cpio
inflating: README.html
inflating: patchnote.css
```

7. Run CPIO against the file using the following commands:

```
$ cpio -idmv < 9204_lnx32_release.cpio
```

Patching Oracle Cluster Manager to Release 9.2.0.4

Complete the following procedure to patch Oracle Cluster Manager release 9.2.0.4 or later patch set. These steps only need to be performed on the node that you are installing from typically Node 1.

1. Log in as the `root` user on the Linux node designated as your primary server node. If the system is at release 9.2.0.1, then this step is not required, and you should proceed to Step 4.

2. Insert Disk 1 for Oracle9i on Linux in the CD-ROM on the primary server node.

3. Using the following command to mount the CD-ROM as the `root` user:

```
# mount /mnt/cdrom
```

4. Log in as the `oracle` user account by opening another window, or by typing the following command:

```
# su - oracle
```

5. Run the installer as the `oracle` user either from `/mnt/cdrom/runInstaller` or from `$ORACLE_HOME/bin/runInstaller` if release 9.2.0.1 is installed, and this is a patch set upgrade.

6. Respond to the installer prompts as follows:

- a. On the Welcome Screen, click **Next**.
- b. On the File Locations Screen, change the source location to the `products.jar` file in the 9.2.0.4 patch set. Navigate to the stage directory where you unpacked the patch set tar file usually `Disk1/stage`.
- c. Verify the destination listed is the Oracle home directory.
- d. On the Available Products Screen, select Oracle9i R2 Cluster Manager 9.2.0.4.0. Click **Next**.
- e. On the public node information screen, enter the public node names. Click **Next**.
- f. On the private node information screen, enter the interconnect node names. Click **Next**.
- g. Select Install at the summary screen.
- h. Click **Exit** and confirm by clicking **Yes**.

Note: If using kernel 2.4.9-e.16 or later, then the hangcheck-timer module exists as `/lib/modules/2.4.9-e.16/kernel/drivers/char/hangcheck-timer.o`. If using a kernel version of 2.4.9-e.3, e.8, e.9 or e.10, then download and install the hangcheck-timer from OracleMetaLink patch 2594820.

To check if the hangcheck-timer module is mounted, enter the following command:

```
lsmod
```

If hangcheck timer isn't mounted, then load the hangcheck-timer kernel module using the following command as root user:

```
# /sbin/insmod hangcheck-timer \  
hangcheck_tick=30 \  
hangcheck_margin=180
```

and append the following line to the `/etc/rc.local` file:

```
/sbin/insmod hangcheck-timer \  
hangcheck_tick=30 \  
hangcheck_margin=180
```

Repeat the above steps on all Oracle Real Applications Clusters nodes where the kernel module needs to be installed.

Modifying the Oracle Cluster Manager Configuration Files

You must manually modify the Oracle Cluster Manager files to update to release 9.2.0.4. To do this you must remove all references or calls to watchdog. These steps need to be performed on all nodes. Use the following procedure to modify the files:

1. Modify the `$ORACLE_HOME/oracm/bin/ocmstart.sh` and comment out the following lines:

```
# watchdog's default log file  
# WATCHDOGD_LOG_FILE=$ORACLE_HOME/oracm/log/wdd.log  
  
# watchdog's default backup file  
# WATCHDOGD_BAK_FILE=$ORACLE_HOME/oracm/log/wdd.log.bak  
  
# Get arguments  
# watchdog_args='grep '^watchdogd' $OCMARGS_FILE |\n# sed -e 's+^watchdogd *++'\n  
# Check watchdog's existence  
# if watchdogd status | grep 'Watchdog daemon active' >/dev/null  
# then  
# echo 'ocmstart.sh: Error: watchdogd is already running'  
# exit 1  
# fi  
  
# Backup the old watchdogd log  
# if test -r $WATCHDOGD_LOG_FILE  
# then  
# mv $WATCHDOGD_LOG_FILE $WATCHDOGD_BAK_FILE  
# fi  
# Startup watchdogd  
# echo watchdogd $watchdogd_args  
# watchdogd $watchdogd_args
```

2. Remove the following line from the `$ORACLE_HOME/oracm/admin/ocmargs.ora` file on both nodes:

```
watchdogd -d /dev/null -l 0 -m 0
```

3. Edit the file `$ORACLE_HOME/oracm/admin/cmcfg.ora` on node 1 so that it matches the following syntax:

```
HeartBeat=15000
ClusterName=Oracle Cluster Manager, version 9i
PollInterval=1000
MissCount=210
PrivateNodeNames=node1 node2
PublicNodeNames=node1-pub node2-pub
ServicePort=9998
HostName=node1(private node)
CmDiskFile=/u01/cmdiskfile
KernelModuleName=hangcheck-timer
```

4. Edit the file `$ORACLE_HOME/oracm/admin/cmcfg.ora` on node 2 so it matches the following syntax:

```
HeartBeat=15000
ClusterName=Oracle Cluster Manager, version 9i
PollInterval=1000
MissCount=210
PrivateNodeNames=node1 node2
PublicNodeNames=node1-pub node2-pub
ServicePort=9998
HostName=node2(private node)
CmDiskFile=/u01/cmdiskfile
KernelModuleName=hangcheck-timer
```

Creating Oracle Cluster Manager and Hangchecktimer Startup Scripts

By default the script `ocmstart` is installed on both nodes. If it is not installed, then you create the `ocmstart` script manually using the following procedure.

1. On both nodes, create or modify the file `/etc/init.d/ocmstart` to match the following example:

```
#!/bin/sh
# Startup script for ocmstart
#
# chkconfig: 345 96 13
# description: auto start Oracle CM on each reboot.

# Source function library.
. /etc/rc.d/init.d/functions

# don't take any action to remove this line or this daemon will not
function.

prog="ocmstart"

start() {
    service cfs status >/dev/null 2>&1
    if [ "$?" != 0 ]; then
        echo "OCFS not running... aborting"
        echo "Run: service cfs start, and try again"
        exit
    fi
    if [ -e /lib/modules/`uname -
r`/kernel/drivers/addon/hangcheck/hangcheck-timer.o ]; then
        modprobe hangcheck-timer hangcheck_tick=30 hangcheck_margin=180
    fi
}
```

```

echo -n "Starting Oracle Cluster Manager: "
export ORACLE_HOME=/opt/oracle/product/9.2.0
rm -rf $ORACLE_HOME/oracm/log/*.ts
$ORACLE_HOME/oracm/bin/ocmstart.sh >/dev/null 2>&1
if [ "$?" != 0 ]; then
echo_failure;
else
echo_success;
fi
echo
}

stop() {
echo -n "Stopping Cluster Manager"
export ORACLE_HOME=/opt/oracle/product/9.2.0
killall oracm
rmmod hangcheck-timer
if [ "$?" != 0 ]; then
echo_failure;
else
echo_success;
fi
echo
}

status() {
cm_status="success"
echo -n "Checking the status of Cluster Manager: ... "
check=`pgrep oracm -l`
if [[ -z $check ]]
then
# I'd prefer to use echo_failure here...
echo "Cluster Manager is not running on this node."
else
# echo_success
echo "Cluster Manager is running on this node"
fi
#echo
}

case "$1" in
start)
start
;;

stop)
stop
;;

status)
status
;;

*)
echo $"Usage: $0 {start|stop|status}"
exit 1
esac

exit 0

```

Start Cluster Manager Services

2. Start and test the Cluster Manager service on all nodes. Start Oracle Cluster Manager manually, or use the

following commands:

```
chkconfig ocmstart on (to check to see if ocmstart is set to on)
service ocmstart start (to start service manually)
service ocmstart status (to check status of CM)
Checking the status of Cluster Manager
Cluster Manager is running on this node
```

Installing Oracle9i Release 2 (9.2.0.1)

These steps only need to be performed on the node that you are installing from, typically Node 1. Complete the following procedure to install Oracle9i Release 2 (9.2.0.1) software.

1. Ensure ORACM is running on all the nodes.
2. Define the Oracle environment parameters ORACLE_HOME, ORACLE_SID, and so on, to auto-populate field locations throughout the installation.
3. Start the Oracle Universal Installer as the `oracle` user using the following command:

```
$ /mnt/cdrom/runInstaller&
```

4. Install Oracle Enterprise edition software release 9.2.0.1 from CD-ROM. Run the installer, as the `oracle` user:

```
$ cd
```

```
$ /mnt/cdrom/runInstaller
```

Note: To prevent CD-ROM eject issues, start the installer from a directory other than the mount point such as `/mnt/cdrom`, or any part of the mounted volume.

5. Click **Next** when the Welcome screen appears.
6. Select all nodes using the shift key on the Cluster Node Selection screen.
7. Click **Next**. If node selection screen doesn't appear, then check that ORACM is up and running, and ORACLE_HOME is set before launching the installer. If they are not, then fix them and restart the installer.
8. Click **Next** on the File Locations screen.
9. Select Oracle9i Database 9.2.0.1 on the Available Products screen.
10. Click **Next**.
11. Select Enterprise Edition on the Installation Type screen.
12. Click **Next**.
13. Select: Software only on the Database Configuration screen.
14. Click **Next**. The Shared Configuration File Name window opens.

15. Enter the OCFS raw device name for the shared configuration file. This configuration file is used by the `srvctl` utility, the configuration and administration utility to manage Real Application Clusters instances and listeners.

16. Enter `/u01/srvconfigfile`.

17. Click **Install** on the Summary screen.

18. Insert the next CD-ROM when prompted by the system using the following command to unmount the first CD-ROM. You must be the `root` user.

```
# umount /mnt/cdrom
```

19. Click **OK** on the installer window.

20. Insert the next CD-ROM when prompted by the system.

11. Run `$ORACLE_HOME/root.sh` on each node as the `root` user.

21. Enter the full pathname of the local, and accept the default. You can ignore any errors about `/var/opt/oracle/srvConfig.loc`.

22. Enter the following command:

```
# mkdir -p /var/opt/oracle
# touch /var/opt/oracle/srvConfig.loc
# /u01/app/oracle/product/9.2.0/root.sh
```

Note: The preceding commands prevent issues running `root.sh` script when not specifying a shared configuration file or destination.

23. Click **OK** in installer window.

24. At the End of Installation screen, exit the installer by clicking **Exit** and confirming the exit.

25. When OEM discovery starts, select **Cancel** and exit.

26. Remove the CD-ROM, as `root` user:

```
# umount /mnt/cdrom
```

27. Eject the CD-ROM, if not done automatically.

Note: Create the following directories on the other nodes if they do not exist:

```
mkdir -p $ORACLE_HOME/rdbms/audit
mkdir -p $ORACLE_HOME/rdbms/log
mkdir -p $ORACLE_HOME/network/log
mkdir -p $ORACLE_HOME/Oracle/Oracle/logs
mkdir -p $ORACLE_HOME/Oracle/Jserv/logs
```

30. Create the following symbolic link:

```
$ cd $ORACLE_BASE/bin/linux
$ ln -s libclntsh.so.9.0 libclntsh.so
```

Installing Oracle Universal Installer Release 2.2.0.18

Complete the following procedure to install the Oracle Universal Installer release 2.2.0.18 using Oracle9i release 9.2.0.4 patch set:

1. Run the installer from the `$ORACLE_HOME/bin` directory using the following commands:

```
$ cd $ORACLE_HOME/bin
$ ./runInstaller
```

2. Respond to the installer prompts as follows:
 - a. On the Welcome Screen, click **Next**.
 - b. On the File Locations Screen, change the source location to the `products.jar` file in the release 9.2.0.4 patch set.
 - c. Navigate to the stage directory where you unpacked the patch set tar file usually under `Disk1/stage`.
 - d. Verify the destination listed is the Oracle home directory.
 - e. On the Available Products Screen, select Oracle Universal Installer 2.2.0.18. Click **Next**.
 - f. On the Component Locations Screen, review the information and click **Next**.
 - g. Select **Install** at the summary screen. You will get a progress window followed by the end of installation screen.
 - h. Click **Exit** and confirm by clicking **Yes**.

Applying the Oracle9i 9.2.0.4.0 Patch Set

Complete the following procedure to apply the Oracle9i release 9.2.0.4.0 patch set. These steps only need to be performed on the node that you are installing from.

1. From node 1, ensure that all Oracle Cluster Manager services are running.
2. Run the installer from the `$ORACLE_HOME/bin` directory using the following commands:

```
$ cd $ORACLE_HOME/bin
$ ./runInstaller
```
3. At the Welcome Screen, click **Next**.
4. At the Cluster Node Selection Screen, make sure that all Real Application Clusters nodes are selected.
5. At the File Locations Screen, make sure the source location is to the `products.jar` file in the release 9.2.0.4 patch set location under `Disk1/stage`. Also verify the destination listed is the Oracle home directory.
6. At the Available Products Screen, select Oracle9i R2 Patch Set 3 9.2.0.4.0. Click **Next**.
7. Click **Install** at the summary screen.

8. Run the `root.sh` script when prompted.
9. A progress window will appear followed by the end of installation screen. Click **Exit** and confirm by clicking **Yes**.

Database Creation and Configuration

To create the database, you must first create the server configuration file. In Real Application Clusters with 2 or more node configurations, the Server Configuration file `srvConfig.loc` should reside on an Oracle Cluster File System, or on raw partitions. To ensure the server configuration file is correctly located, check it is located on one of the following paths:

```
/var/opt/oracle/srvConfig.loc  
/etc/srvConfig.loc  
$ORACLE_HOME/srvm/config/srvConfig.loc
```

Creating the Server Configuration File

If `srvConfig.loc` has not been created, then complete the following procedure:

1. Log in as the `root` user, and use the following commands to create the file:

```
# mkdir -p /var/opt/oracle  
# touch /var/opt/oracle/srvConfig.loc  
# chown oracle:dba /var/opt/oracle/srvConfig.loc  
# chmod 755 /var/opt/oracle/srvConfig.loc
```

2. Add the `srvconfig_loc` parameter to the `srvConfig.loc` on both nodes:

```
su - oracle  
echo "srvconfig_loc=/u01/srvconfigfile" >  
/var/opt/oracle/srvConfig.loc  
exit
```

Initializing the Server Configuration File

To initialize the `srvconfig.loc` file, run the following commands on the first node only:

```
su - oracle  
srvconfig -init  
exit
```

Starting the Global Services Daemon

To start the Global Services Daemon (GSD) as the `oracle` user, use the `gsdctl` utility on all nodes as follows:

```
su - oracle  
gsdctl start  
exit
```

Running Database Configuration Assistant

On the first node of the cluster, use the following procedure to run Oracle Database Configuration Assistant (DBCA) utility to create a seed or production database:

1. Open an X Terminal console, and start the Database Configuration Assistant (DBCA) as the `oracle` user using the following commands:

```
startx
xhost +
su - oracle
dbca -datafileDestination /u02
```

The first command will launch the Xterm console. In the last command, the command specifies the location of the database files. In addition, More mount points can be assigned during the file setup.

Note: The parameter DB_FILES must be set explicitly for the number that is required for your site other the value is set to the 32 default value.

2. The Welcome window opens. Select Oracle cluster database, and click **Next**.
3. The window Database Configuration Assistant: Step 1 of 10: Operations window opens. Select Create a Database, and click **Next**.
4. The window Database Configuration Assistant: Step 2 of 10: Node Selection opens. Select node1-pub and node2-pub and so on, and click **Next**.
5. The window Database Configuration Assistant: Step 3 of 10: Database Templates opens. Select New Database, and click **Next**.
6. The window Database Configuration Assistant: Step 4 of 10: Database Identification opens. Enter database name such as TEST in both fields, and click **Next**.
7. The window Network Configuration for the database: Step 5 of 10: Database Identification opens. Leave register this database with all listeners, and click **Next**.
8. The window Database Configuration Assistant: Step 6 of 10: Database Features opens. Clear all check boxes, and confirm deletion of tablespaces.
 - a. Select Standard database features.
 - b. Uncheck all options, confirm deletion of tablespaces.
 - c. Close Standard database features window.
 - d. Click **Next**.
9. The window Database Configuration Assistant: Step 7 of 10: Database Connection Options opens. Select Dedicated Server Mode, and click **Next**.
10. The window Database Configuration Assistant: Step 8 of 10: Initialization Parameters opens.
 - a. On the Memory tab, change Shared Pool to 32 MB, and Java Pool to 0 MB.
 - b. On the DB Sizing tab, change Block Size to 2 KB.
 - c. Click **Next**.

11. The window Database Configuration Assistant: Step 9 of 10 : Database Storage opens.
 - a. Select Controlfile on left side.
 - b. In General tab, delete control03.ctl.
 - c. Highlight, and click backspace.
 - d. On the Options tab, change the maximum number of instances to 4, and the maximum number of log history to 100.
 - e. Expand Tablespaces on left side, and select SYSTEM tablespace.
 - f. In General tab, change size to 110 MB.
 - g. In Storage tab, change to Managed in the dictionary.
 - h. Set Initial to 32 KB.
 - i. Set Next to 128 KB.
 - j. SetIncrement Size by 0.
 - k. Select TEMP on the left and change the size to 10 MB.
 - l. Select UNDOTBS1 on the left, and change the size to 50 MB.
 - m. Expand Redo Log Groups on the left side.
 - n. Select 1.
 - o. Change size to 10240 KB.
 - p. Select 2.
 - q. Change size to 10240 KB.
 - r. Select 3.
 - s. Delete group by selecting the Remove button, and confirm the deletion.
 - t. Click **Next**.
12. The window Database Configuration Assistant: Step 10 of 10: Creation Options opens. Click **Finish**.
13. The Database Configuration Assistant: Summary window opens, Review summary. Click **OK**.
14. When prompted to perform another operation, click **No**.

Configuring SRVCTL

SRVCTL will be configured automatically if Database Configuration Assistant was used to create the database. If necessary, use the following procedure for manual configuration:

1. Initialize the SRVCONFIG file using the following command:

```
srvconfig -f -init.
```

2. Start GSD using the following command:

```
gsdctl start.
```

3. Configure SRVCTL using the following command:

```
srvctl add database -d db_name
```

See Also: Refer to *OracleMetalink* Note:159371.1 for the complete procedure to configure SRVCTL.

Verifying Seed Database

Complete the following procedure on all nodes to verify seed database creation:

1. Start SQL*Plus with the following command:

```
$ sqlplus "/ as sysdba"
```

The SQL prompt appears.

2. Start the database instance on the node:

```
SQL> startup
```

3. Enter the following query at the SQL prompt:

```
SELECT inst_id, instance_number inst_no, instance_name inst_name,  
parallel, status, database_status db_status, active_state state,  
host_name host FROM gv$instance;
```

Screen output appears that resembles the following:

```
INST_ID INST_NO INST NAM PAR STATUS DB_STATUS STATE HOST  
-----  
1 oradb1 YES OPEN ACTIVE NORMAL racnode1-pub  
2 oradb2 YES OPEN ACTIVE NORMAL racnode2-pub
```

DB_STATUS indicates the database state, STATUS indicates the startup condition of the database, and PAR (parallel) indicates whether the database is operating in cluster mode.

4. At the SQL command prompt, type the following command:

```
srvctl status -p oradb
```

The following screen output appears:

```
Instance is running on node: racnode2  
Instance is running on node: racnode1  
All of the listeners are running on node: racnode2  
All of the listeners are running on node: racnode1
```

This output indicates that database instances have been started on all nodes, and all nodes are now ready to receive queries from clients.

Appendix A: OCFS Startup Script

The following is the OCFS startup script created by OCFS during OCFS installation:

```
#!/bin/sh
# init fragment for ocfs
#
# chkconfig: 2345 24 20
# description: Run cvsman update at system boot

#
# Note that the start priority is 24. This is precisely one less than
# the start priority for netfs (25). This is because we expect OCFS
# volumes to be mounted in the netfs start.
#
# All OCFS volumes that are to be mounted at boot need to have the
# '_netdev' option added to their fstab entry. This prevents them
# from trying to mount before this script has run.
#

LOAD_OCFS=/sbin/load_ocfs
test -x "$LOAD_OCFS" || exit 0

test -r /etc/ocfs.conf || exit 0

KVER=`uname -r`
case "$KVER" in
2.4.9-e.*enterprise)
    MODPATH="/lib/modules/${KVER}/kernel/drivers/addon/ocfs"
    OBJPATH="/lib/modules/2.4.9-e-enterprise-ABI/ocfs"
    ;;
2.4.9-e.*smp)
    MODPATH="/lib/modules/${KVER}/kernel/drivers/addon/ocfs"
    OBJPATH="/lib/modules/2.4.9-e-smp-ABI/ocfs"
    ;;
2.4.9-e.*summit)
    MODPATH="/lib/modules/${KVER}/kernel/drivers/addon/ocfs"
    OBJPATH="/lib/modules/2.4.9-e-summit-ABI/ocfs"
    ;;
2.4.9-e.*)
    MODPATH="/lib/modules/${KVER}/kernel/drivers/addon/ocfs"
    OBJPATH="/lib/modules/2.4.9-e-ABI/ocfs"
    ;;
*)
    # Not a supported kernel
    exit 0
    ;;
esac

test -r "$OBJPATH/ocfs.o" || exit 0

# Source init.d functions
. /etc/init.d/functions

# Source networking configuration
. /etc/sysconfig/network

fstab_check()
{
    # Are there any volumes to mount?
    ANY=`awk `/^[ ]*#/{next}$4 ~ /^noauto$|^noauto,|,noauto$/{next}$3 ==
"ocfs"{print $2}' /etc/fstab 2>/dev/null`
```

```

test -z "$ANY" && exit 0

# Check if anything is missing netdev
ANY=`awk '/^[ ]*#/{next}$4 ~ /noauto$|^noauto,|,noauto$/{next}$4 ~
/^_netdev$|^_netdev,|,_netdev$/{next}$3 == "ocfs" {print $2}' /etc/fstab
2>/dev/null`
test -n "$ANY" && echo "Warning: OCFS filesystems missing \"_netdev\"
option: $ANY"
}

module_link()
{
# Assume any symlink is OK - caveat emptor for non-RPM users
test -L "${MODPATH}/ocfs.o" && return

echo -n "Linking OCFS module into the module path"

# Same deal for modules that are physically there
if test -f "${MODPATH}/ocfs.o" ; then
EXISTVER=`modinfo "${MODPATH}/ocfs.o" | awk
'/description.*version/{sub(/^.*\(\(version /, " ")\); sub(/\) "$/, " ")\);
print;}```
if test -n "${EXISTVER}" ; then
ABIVER=`modinfo "${OBJPATH}/ocfs.o" | awk
'/description.*version/{sub(/^.*\(\(version /, " ")\); sub(/\) "$/, " ")\);
print;}```

echo
echo "Leaving existing ocfs module in place (ABI module version
$ABIVER, existing module version $EXISTVER)"
success
echo
return
else
# The existing module is too old - pre-ABI packaging
echo
echo "Warning: Moving ancient ocfs.o out of the way"
mv -f "${MODPATH}/ocfs.o" "${MODPATH}/ocfs.o.preABI" >/dev/null 2>&1
if test "$?" != 0 ; then
failure $"Unable to move ancient ocfs module out of the way"
echo
exit 1
fi
fi
fi

if test ! -d "$MODPATH" ; then
if test -e "${MODPATH}"; then
failure $"Path \"${MODPATH}\" already exists, but is not a directory"
echo
exit 1
else
mkdir -p "$MODPATH" >/dev/null 2>&1
if test $? != 0; then
failure $"Unable to create directory \"${MODPATH}\""
echo
exit 1
fi
fi
fi

ln -s "${OBJPATH}/ocfs.o" "${MODPATH}/ocfs.o" >/dev/null 2>&1
if test $? != 0; then
failure $"Unable to create symbolic link \"${MODPATH}/ocfs.o\""

```

```

echo
exit 1
fi

success
echo
}

case "$1" in
start)
module_link
fstab_check

echo -n $"Loading OCFS:"

if test "$NETWORKING" = "no"; then
failure $"Network is unavailable"
echo
exit 1
fi

if grep `^ocfs` /proc/modules >/dev/null 2>&1; then
success $"OCFS is already loaded"
echo
exit 0
fi

# action - DO NOT REMOVE, this is to trick RH /etc/rc
initlog -q -c "$LOAD_OCFS" && success || failure
RETVAL=$?
echo

;;

stop|force-reload|restart)
RETVAL=0
;;

*)
echo "Usage: $0 {start|stop|restart|force-reload}"
RETVAL=1
esac

exit $RETVAL

```

Appendix B: CFS Startup Script

```

#!/bin/sh
# Startup Script for ocfs
#
# chkconfig: 345 98 04
# description: auto load ocfs on each reboot.
#
# source function library.
. /etc/rc.d/init.d/functions

prog="load_ocfs"

```

```

start() {
service cfs status >/dev/null 2>&1
if [ "$?" = 0 ]; then
echo "Oracle Cluster Filesystem already running..."
exit
fi
echo -n "Loading Oracle Cluster Filesystem"
export ORACLE_HOME=/opt/oracle/product/server32/9.2.0
/sbin/load_ocfs >/dev/null 2>&1
if [ -e /etc/cfstab]; then
while read source target options; do
mount $source $target -t ocfs -o $options
done < /etc/cfstab
fi
echo_success
echo
}

stop() {
echo -n "Unloading Oracle Cluster Filesystem..."
export ORACLE_HOME=/opt/oracle/product/server32/9.2.0
umount -a -t ocfs >/dev/null 2>&1
[ "$?" != "0" ] && echo " failed!" || echo_success; echo; rmmod ocfs
>/dev/null 2>&1;
}

status() {
cm_status="success"
echo -n "Checking the status of Cluster Filesystem..."
check=`pgrep -l ocfs`
if [[ -z $check ]]; then
echo " not running"
exit 2
else
echo " [$check] running"
fi
}

case "$1" in
start)
start
;;

stop)
stop
;;

status)

```

status

REFERENCES

<http://Certify.us.oracle.com>

<http://metalink.oracle.com>

<http://otn.oracle.com/>

<http://otn.oracle.com/linux>

<http://otn.oracle.com/deploy/availability/content.html>

<http://otn.oracle.com/products/oracle9i/content.html>

http://otn.oracle.com/tech/linux/installtips_final.pdf

<http://rac.us.oracle.com>

http://www.oracle.com/ip/index.html?rac_home.html

<http://www.oracle.com/oramag/index.html>

<http://www.oracletuning.com/>

<http://www.linuxjournal.com>

<http://www.orafaq.com/faq.htm>

Oracle*MetaLink* Note Titles

Note 188135.1 Documentation Index for Oracle9i Real Application Clusters

Note 200266.1 Linux on Real Application Clusters Best Practices

Note:225303.1 How To Check the Supportability of RedHat AS.

Note:237997.1 OCFS Best Practices

Note:224586.1 OCFS Best Practices

Note:226570.1 Oracle9i Real Application Clusters: OS Port-Specific Best Practices (INTERNAL ONLY)

Note:240575.1 Real Application Clusters on Linux Best Practices

Note:224586.1 Oracle Cluster File System (OCFS) on RedHat AS - FAQ

Part B10499-01 Oracle Cluster File System Release 1.0 for Red Hat Advanced Server 2.1 Installation Notes
November 2002



October 2003

Authors: Caroline Johnston, Yousef Kanani, Doug Williams

Oracle Corporation

World Headquarters

500 Oracle Parkway

Redwood Shores, CA 94065

U.S.A.

Worldwide Inquiries:

Phone: +1.650.506.7000

Fax: +1.650.506.7200

www.oracle.com

Oracle is a registered trademark of Oracle Corporation. Various product and service names referenced herein may be trademarks of Oracle Corporation. All other product and service names mentioned may be trademarks of their respective owners.

Copyright © 2003 Oracle Corporation

All rights reserved.