



Oracle10g™ Real Application Clusters Release 1: Installation with a Filer in a SUSE Linux® Enterprise Server 8/9 (SLES 8, SLES 9) Environment

Andrew Firebaugh, Antonio Robinson & Tushar Patel | Network Appliance | July 2005 | TR-3413

TECHNICAL REPORT

Network Appliance, a pioneer and industry leader in data storage technology, helps organizations understand and meet complex technical challenges with advanced storage solutions and global data management strategies.

Table of Contents

1. Introduction	3
2. Assumption	3
3. The Server/Filer Environment.....	3
4. Requirements	4
5. Preinstallation Setup Tasks (for Network Appliance Filer)	9
6. Preinstallation Setup Tasks (for Cluster Nodes)	10
7. Installation Procedure	12
8. Known Issues	24
9. Best Practices and Recommendations	25
10. Appendix	26
11. Disclaimer	26

1. Introduction

This technical report will cover the installation of a NetApp filer in a NAS environment and Intel[®] architecture-based enterprise-class servers using SUSE Linux Enterprise Server 8/9 (SLES 8, SLES 9) with a Linux operating system running Oracle10g Real Application Clusters release 1 (hereafter referred to as Oracle10g RAC). This is a certified configuration, and, as such, the components presented in this paper have to be used in the same combination to gain support from all parties involved. The only exception to this is the application of certain patches (as defined and required by all the vendors in this configuration). This document will also cover the best practices and recommendations for running Oracle10g RAC on NetApp filers in a NAS environment.

2. Assumption

We assume that readers are familiar with Oracle10g release 1 and the operation of Network Appliance[™] filers. We also assume that readers are familiar with the operation of the Linux OS and installation of Oracle[®] patches and any relevant Linux patches. It is also important to be familiar with all networking terminology and implementations.

3. The Server/Filer Environment

The configuration presented in this document is based on the Oracle10g RAC certification environment specified by Oracle and Network Appliance.

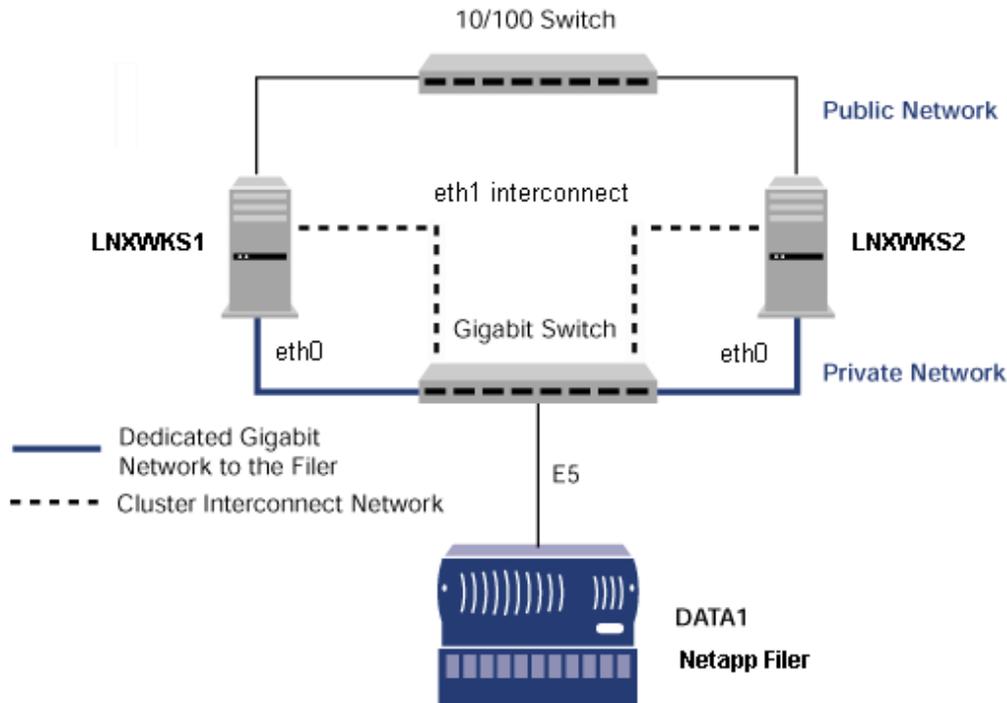


Figure 1) Oracle10g RAC on SUSE Linux Enterprise Server 8/9 (SLES 8, SLES 9) server with a Network Appliance filer.

Figure 1 illustrates a typical configuration of Oracle10g RAC with a NetApp filer and Intel architecture–based enterprise-class servers running SUSE Linux Enterprise Server 8/9 (SLES 8, SLES 9). This is a scalable configuration and allows the user to scale horizontally and internally in terms of processor, memory, and storage.

As shown in the network diagram, we recommend that you dedicate a private network connection between the Oracle10g RAC servers and the filer. This is accomplished using a dedicated Gigabit network (with a Gigabit switch) to the filer. A dedicated network connection is beneficial for the following reasons:

- In an Oracle10g RAC environment, it is important to eliminate any contentions and latencies
- Providing a separate network ensures security

The cluster interconnect is used to monitor the heartbeat of the two servers in the cluster. This is a typical configuration that can be deployed in a customer's environment.

4. Requirements

4.1. Hardware Used for Tests

Cluster Nodes

- Two two-way 900 MHz, 2GB RAM Intel architecture–based enterprise-class servers
- Two Intel Pro1000 Gigabit Ethernet network interface cards (NICs; for cluster interconnect)
- Additional NIC on each machine for public network
- Two Intel Pro1000 Gigabit Ethernet NICs (for Network Appliance filer I/O)

Storage Infrastructure

- One Network Appliance F2xx/F7xx/F8xx/F9x filer with Data ONTAP™
- One Gigabit switch with at least four ports
- One Gigabit NIC in the filer
- One or more disk shelves, based on the disk space requirements

4.2. Software Requirements

For both nodes in the participating cluster unless specified otherwise:

- SUSE SLES 8 or SLES 9
- Oracle10g with Real Application Clusters

Software Requirements: SUSE SLES 8 (x86)

- Service Pack 3 (SP3) or higher (kernel version 2.4.21-138)
- gcc-3.2.2-38 or higher

The following packages (or higher versions) must also be installed:

- make-3.79
- binutils-2.12
- openmotif-2.2.2-124

Software Requirements: SUSE SLES 9

Kernel version 2.6.5-7.97 or later must be installed.

To check your kernel version use:

- # `uname -r`

The following packages (or later versions) must also be installed:

- gcc-3.3.3-43
- gcc-c++-3.3.3-43
- glibc-2.3.3-98
- libaio-0.3.98-18
- libaio-devel-0.3.98-18
- make-3.80
- openmotif-libs-2.2.2-519.1

To ensure that the system meets these requirements, follow these steps:

1. To determine which distribution and version of Linux are installed, enter the following command:

- # `cat /proc/version`

2. To determine whether the required packages are installed, enter commands similar to the following:

- \$ `rpm -q package_name`

If a required package is not installed, or if the version is lower than the required version, install the package from your operating system distribution media or download the required package version from your Linux vendor's Web site.

Cluster Node Kernel Parameters and Shared Memory

On all cluster nodes, verify that the kernel parameters shown in the following table are set to values equal to or greater than the recommended value shown. The procedures following the tables describe how to verify and set the values.

Note: The kernel parameter and shell limit values shown in the following sections are basic recommended values only. For production database systems, Oracle recommends that you tune these values to optimize the performance of the system. Refer to your operating system documentation for more information about tuning kernel parameters.

To view the current value of these kernel parameters and to change them if necessary, follow these steps:

1. Enter the commands shown in the following table to view the current values of the kernel parameters:
2. If the value of any kernel parameter is different than the recommended value, complete the following steps:

For SLES 8 and SLES 9:

Parameter	Value	File
-----	-----	-----
semmsl	250	/proc/sys/kernel/sem
semmns	32000	
semopm	100	
semmni	128	
shmall	2097152	/proc/sys/kernel/shmall
shmmax	Half the size of physical memory (bytes)	/proc/sys/kernel/shmmax
shmmni	4096	/proc/sys/kernel/shmmni
file-max	65536	/proc/sys/fs/file-max
ip_local_port_range	1024 65000	/proc/sys/net/ipv4 /ip_local_port_range

For SLES 9 only (in addition to those above):

Parameter	Value	File
-----	-----	-----
rmem_default	262144	/proc/sys/net/core/rmem_default
rmem_max	262144	/proc/sys/net/core/rmem_max
wmem_default	262144	/proc/sys/net/core/wmem_default
wmem_max	262144	/proc/sys/net/core/wmem_max

Note: If the current value for any parameter is higher than the value listed in this table, do not change the value of that parameter.

For SLES 8 and SLES 9:

Parameters	Command
-----	-----
semmsl	# /sbin/sysctl -a grep sem
semmn	This command displays the value of the semaphore parameters in the order listed.
semopm	
semmni	
shmall	# /sbin/sysctl -a grep shm
shmmax	
shmmni	
file-max	# /sbin/sysctl -a grep file-max
ip_local_port_range	# /sbin/sysctl -a grep ip_local_port_range This command displays a range of port numbers.

For SLES 9 only (in addition to those above):

Parameters -----	Command -----
rmem_default rmem_max wmem_default wmem_max	# /sbin/sysctl -a grep net.core

2. If the value of any kernel parameter is different than the recommended value, complete the following steps:

a. Using any text editor, create or edit the `/etc/sysctl.conf` file and add or edit lines similar to the following:

Note: Include lines only for the kernel parameter values that you want to change. For the semaphore parameters (`kernel.sem`), you must specify all four values. However, if any of the current values are larger than the recommended value, specify the larger value.

For SLES 8 and SLES 9:

- `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`

For SLES 9 only (in addition to those above):

- `net.core.rmem_default = 262144`
- `net.core.rmem_max = 262144`
- `net.core.wmem_default = 262144`
- `net.core.wmem_max = 262144`

Specify the values in the `/etc/sysctl.conf` file; they persist when you reboot the system.

b. Enter the following command to change the current values of the kernel parameters:

- `# /sbin/sysctl -p`

Review the output from this command to verify that the values are correct. If the values are incorrect, edit the `/etc/sysctl.conf` file, then enter this command again.

c. Enter the following command to cause the system to read the `/etc/sysctl.conf` file when it reboots:

- # /sbin/chkconfig boot.sysctl on

3. Repeat this procedure on all other cluster nodes.

Set shell limits for the Oracle user.

To improve the performance of the software on Linux systems, you must increase the following shell limits for the Oracle user:

Shell Limit -----	Item in limits.conf -----	Hard Limit -----
Maximum number of open file descriptors	nofile	65536
Maximum number of processes available to a single user	nproc	16384

To increase the shell limits:

1. Add the following lines to the `/etc/security/limits.conf` file:

- `oracle soft nproc 2047`
- `oracle hard nproc 16384`
- `oracle soft nofile 1024`
- `oracle hard nofile 65536`

2. Add or edit the following line in the `/etc/pam.d/login` file, if it does not already exist:

```
session required /lib/security/pam_limits.so
```

3. Depending on the Oracle user's default shell, make the following changes to the default shell startup file:

For the Bourne, Bash, or Korn shell, add the following lines to the `/etc/profile` file (or the `/etc/profile.local` file on SUSE systems):

```
if [ $USER = "oracle" ]; then
  if [ $SHELL = "/bin/ksh" ]; then
    ulimit -p 16384
    ulimit -n 65536
  else
    ulimit -u 16384 -n 65536
  fi
fi
```

For the C shell (csh or tcsh), add the following lines to the `/etc/csh.login` file (or the `/etc/csh.login.local` file on SUSE systems):

```

if ( $USER == "oracle" ) then
  limit maxproc 16384
  limit descriptors 65536
endif

```

4. Repeat this procedure on all other cluster nodes.

5. Preinstallation Setup Tasks (for Network Appliance Filer)

For more information, refer to the Network Appliance installation and setup guides at <http://now.netapp.com>.

1. Install the filer and connect the disk shelves to it.
2. Make sure each shelf has a unique ID. This should be on the back of the shelf.
3. Connect the shelves to the Fibre Channel adapter in a daisy chain. The output of one shelf should be connected to the input of the next. The last shelf's output should be terminated. The input to the first shelf should come from the Fibre Channel adapter in the filer.
4. Install Data ONTAP and configure the filer. We used hostname Data1 for the filer.
5. Install the NFS and SnapRestore[®] license keys.
6. Configure the Gigabit NIC in the filer.
 - a. Ensure that the Gigabit NIC is installed in the filer. This document assumes the NIC is inserted in slot 5 in the filer Data1.
 - b. Configure the Gigabit interfaces: e.g., e5 on Data1.
 - c. Specify the following IP address for the interface (run the setup command on the filer console to configure the network interfaces):

```
Data1>e5 - ip: 10.32.90.103, netmask: 255.255.255.0
```

7. Our filer administration examples use commands entered at the command line; however, FilerView[®], a GUI-based tool can also be used. To access FilerView use a browser and set the URL to:

```
http://Data1/na_admin/
```

Where Data1 is the hostname (or IP address) of the filer.

8. Create an aggregate and volumes and export the volumes for storing Oracle Database files on the filers. Our example uses volumes created on one aggregate. Aggregates are a new feature in Data ONTAP 7G.

- a. Create an aggregate on the Data1 filer as shown below:

To create the aggregate, use the following command at the filer console:

```
Data1> aggr create aggr1 -r 10 10
```

This creates an aggregate named aggr1 with a 10-disk RAID array and 10 disks. The default RAID type is RAID-DP™ (double parity).

- b. Create three volumes on the aggr1 aggregate as shown below and mount them on all the cluster nodes:

volhome	Shared Oracle and CRS Home
voldb	Shared Oracle data files and control files
vollog	db logs, a copy of control files and archive log files

To create three volumes, 100GB each, use the following commands at the filer console:

```
Data1> vol create volhome aggr1 100g
Data1> vol create voldb aggr1 100g
Data1> vol create vollog aggr1 100g
```

If you prefer, you can create your aggregates and volumes with different parameters based on your workload needs.

Edit the /etc/exports file on Data1 and add the following entries to that file:

```
/vol/volhome -anon=0
/vol/voldb -anon=0
/vol/vollog -anon=0
```

- c. Execute the following command at the filer console:

```
Data1> exportfs -a
```

6. Preinstallation Setup Tasks (for Cluster Nodes)

This information is intended for both nodes in the participating cluster unless otherwise specified.

1. Install the Gigabit NICs in the cluster nodes (two per node).
 - a. On each node, connect one Gigabit NIC to the Gigabit switch for cluster interconnects.
 - b. On each node, connect one Gigabit NIC to the Gigabit switch, which will connect to the Gigabit NIC on the filer.
2. Install SUSE SLES 8 or 9 on the cluster nodes. (The cluster nodes in this document will be referred to as Inxwks1 and Inxwks2.) Refer to the SUSE Enterprise Linux installation manuals for more details.

3. Configure the two Gigabit network interfaces on each node using SUSE Enterprise Linux network management tools.

- a. Configure the two network interfaces and specify the following IP addresses:

```
#Inxwks1
eth0 - ip: 10.32.90.101, netmask: 255.255.255.0
eth1 - ip: 172.168.30.101, netmask: 255.255.255.0
#Inxwks2
eth0 - ip: 10.32.90.102, netmask: 255.255.255.0
eth1 - ip: 172.168.30.102, netmask: 255.255.255.0
```

Where:

Interface eth0 on both cluster nodes is connected to the Gigabit switch for filer I/O.
Interface eth1 on both cluster nodes is connected to the Gigabit switch for cluster interconnects.

- b. Update the /etc/hosts file on the cluster nodes and add the following entries:

```
#Inxwks1
Inxwks1 10.32.90.101
Inxwks1-i 172.168.30.101
data1 10.32.90.103
Inxwks2 10.32.90.102
Inxwks2-i 172.168.30.102
```

```
#Inxwks2
Inxwks1 10.32.90.101
Inxwks1-i 172.168.30.101
data1 10.32.90.103
Inxwks2 10.32.90.102
Inxwks2-i 172.168.30.102
```

4. Make sure the eth0 interface is connected to the switch and can communicate with the filer Data1. Use the /usr/sbin/ping command to verify the connectivity to the filer over the Gigabit network from both the cluster nodes, i.e., Inxwks1 and Inxwks2.

```
$/usr/sbin/ping data1
```

5. The eth1 interface on each node should also be connected to the Gigabit switch. Verify that eth1 on Inxwks1 is able to communicate with its counterpart, eth1 on Inxwks2. Use the /usr/sbin/ping command to verify the cluster interconnect from both cluster nodes, i.e., Inxwks1 and Inxwks2.

```
$/usr/sbin/ping Inxwks1-i
$/usr/sbin/ping Inxwks2-i
```

Inxwks1-i and Inxwks2-i will be used as private node names for the cluster interconnects. These interface names will be used for the PrivateNodeNames parameters for Oracle10g RAC installation.

6. Create mountpoints and mount the volumes with the following mount options on all the cluster nodes. Update the /etc/fstab file on all server nodes and add following entries:

```
Data1:/vol/voldb      /voldb      nfs
rsize=32768,wsiz=32768,hard,nointr,rw,bg,vers=3,tcp,actimeo=0,timeo=600,sync
```

```
Data1:/vol/vollog      /vollog      nfs
rsize=32768,wsiz=32768,hard,nointr,rw,bg,vers=3,tcp,actimeo=0,timeo=600
Data1:/vol/volhome    /volhome     nfs
rsize=32768,wsiz=32768,hard,nointr,rw,bg,vers=3,tcp,actimeo=0,timeo=600
```

Where:

Data1 is the name of the filer.

voldb, vollog, and volhome are the mountpoints on the cluster nodes.

The mount options that are required for the Oracle10g RAC are actimeo (use direct I/O) and tcp (mount file system using the tcp option).

Note: actimeo=0 is required for shared files such as data files, log files, cluster registry file, and quorum file.

Note: sync is required with actimeo=0 for the OCRS quorum file. Our quorum file will be in /voldb.

7. Create the following mountpoints on all cluster nodes:

```
#mkdir /voldb
#mkdir /vollog
#mkdir /volhome
```

Mount exported volumes on the mountpoints created above on the cluster nodes and verify mount options in the /var/log/messages file.

7. Installation Procedure

For more information about installing Oracle10g release 1 on Linux, refer to the Oracle10g RAC Installation Guide (Part No: B10766-02) at <http://otn.oracle.com/docs/content.html>.

1. Preparing to install the Oracle10g RAC on cluster nodes:
 - a. This document assumes the Oracle user account and the group to be ora10g and dba, respectively, on both cluster nodes, Inxwks1 and Inxwks2. The user ID and group name for the ora10g account should be the same on both cluster nodes. A sample ora10g user .profile file is provided in the appendix. Make sure the user profile file exports at least ORACLE_BASE, ORACLE_PRODUCT, ORACLE_HOME, ORA_CRS_HOME, ORACLE_SID, LD_LIBRARY_PATH, and PATH entries.
 - b. Grant appropriate permissions to the ora10g user on all shared mounted volumes, /voldb, /volhome, and /vollog as indicated below:

```
#chown -R ora10g:dba /voldb
#chmod -R 755 /voldb
```

Repeat the same for volhome and vollog volumes.

- c. Set up account equivalence between the cluster nodes for the ora10g user account. Add the following entries to the /etc/hosts.equiv file on all cluster nodes:

```
Inxwks1 ora10g
Inxwks2 ora10g
```

```
Inxwks1-i ora10g
Inxwks2-i ora10g
```

- d. Test the ora10g account equivalence using a remote shell utility such as rsh after logging in as the ora10g user from both cluster nodes.

```
Inxwks1:
#su - ora10g
$rsh Inxwks1 pwd
$rsh Inxwks2 pwd
$rsh Inxwks1-i pwd
$rsh Inxwks2-i pwd
```

```
Inxwks2:
#su - ora10g
$rsh Inxwks1 pwd
$rsh Inxwks2 pwd
$rsh Inxwks1-i pwd
$rsh Inxwks2-i pwd
```

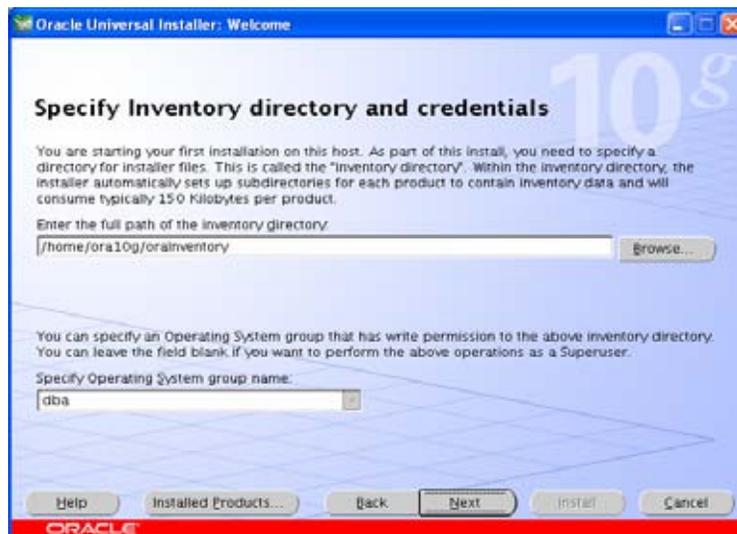
2. Installing the Oracle10g RAC Cluster Ready Services (CRS):

Prior to installing the Oracle10g database, Oracle Cluster Ready Services (CRS) must be installed, configured, and started. Refer to Oracle Real Application Clusters Installation and Configuration Guide 10g release 1 for UNIX[®] Systems (part no. B10766-04) at <http://otn.oracle.com/docs/content.html> for more information on installing Oracle Cluster Ready Services on Linux. This section briefly describes the procedures for using the Oracle Universal Installer (OUI) to install CRS. Note that the CRS home that you identify in this phase of the installation is only for CRS software; this home *cannot* be the same home as the Oracle10g RAC database home. In short, ORACLE_HOME and ORA_CRS_HOME must be different locations.

- a. Run the runInstaller command from the /crs subdirectory on the Oracle Cluster Ready Services release 1 CD-ROM. This is a separate CD that contains the Cluster Ready Services software. This document assumes that the OUI is started from node 1 (Inxwks1). When the OUI displays the Welcome page, click Next.



- b. On the “Specify Inventory...” page, enter a nonshared location for Oracle Inventory. This is the only part of Oracle10g that should not be shared. For this test, we used /home/ora10g/orainventory for the Oracle Inventory information. Click Next.



- c. The OUI dialog indicates that you should run the *oraInventory location/orainstRoot.sh* script. Run the *orainstRoot.sh* script as root user, then click Continue.



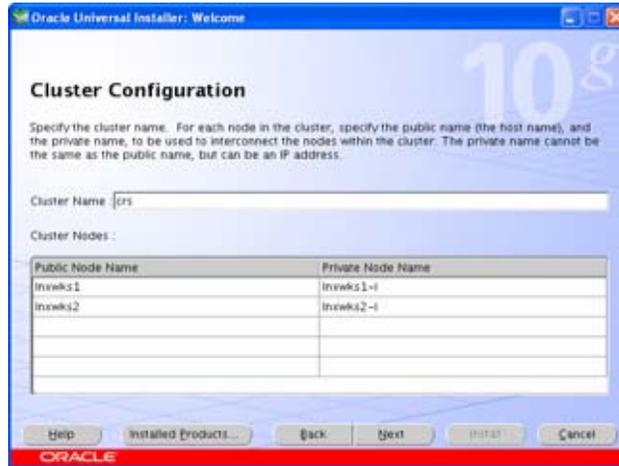
- d. The Specify File Locations page contains predetermined information for the source of the installation files and the target destination information. Specify the destination path for the shared CRS home. The path should be on a shared file system and different from \$ORACLE_HOME. In this exercise, the shared CRS home was /volhome/ora10g/product/10.1.0/crs_1.



- e. Select the language in which your product will run. We used English for our tests.



- f. On the next screen, specify the cluster name, public names (hostnames), and private names to be used for the cluster interconnect. In our case, the public names are Inxwks1 and Inxwks2, and the private names are Inxwks1-i and Inxwks2-i.



- g. On the Private Interconnect Enforcement page, specify the private network to be used for the cluster interconnect. This is a very important step. Do not leave it set to the default, which is Do Not Use. In this case, eth1(Inxwks1-i) was used as the private interconnect. Click Next.



- h. On the Oracle Cluster Registry page, specify the OCR (Oracle Cluster Registry) file. Make sure to specify the full path to a shared location along with the name of the file. In our case, we used /voldb/ora10g/ocrfile. Click Next.



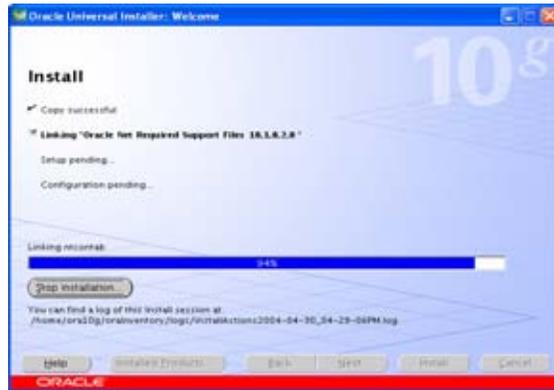
- i. On the Voting Disk page, specify the CSS (Cluster Synchronization Services) voting disk file location. We used /voldb/ora10g/cssfile for CSS services. Click Next.



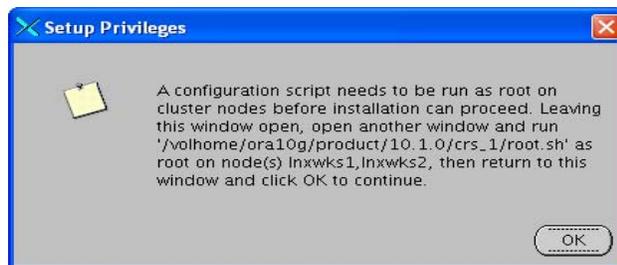
- j. When prompted, execute the oraInstRoot.sh file on all nodes from the /home/ora10g/orainventory directory. Click Continue.



- k. On the Install page, click Install to start the Cluster Ready Services installation.



- I. After installing the CRS, execute the root.sh script on all the cluster nodes. Wait for the script to finish on the first node before running it on the other cluster nodes. After executing root.sh on all the nodes, the CRS process should be started, and the CSS should be active on all nodes.



- m. You can verify your CRS installation by executing the olsnodes command from the *CRS Home/bin* directory. The olsnodes command syntax is:

```
olsnodes [-n] [-l] [-v] [-g]
```

Where:

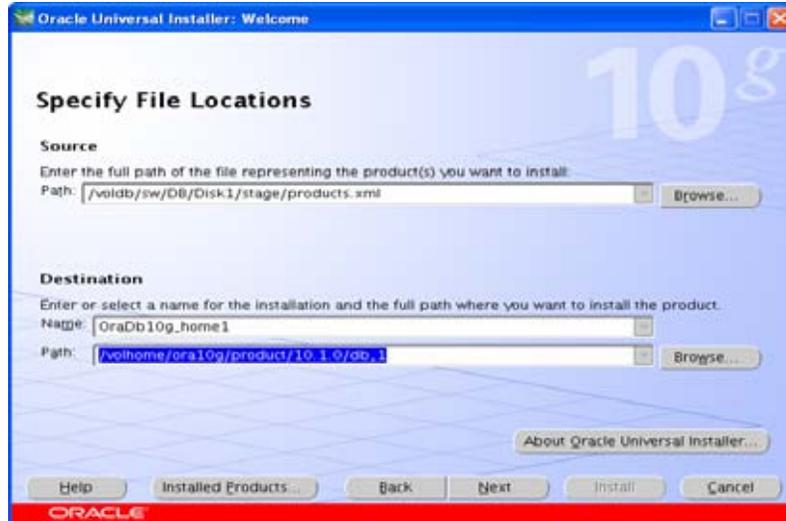
- n displays the member number with the member name
- l displays the local node name
- v activates verbose mode
- g activates logging

The output from this command should be a listing of the nodes on which CRS was installed.

3. Installing Oracle10g RAC software:

- a. After making sure that Oracle Cluster Ready Services have started on the cluster nodes, start runInstaller from disk 1 of the Oracle10g release 1 CDs.
- b. On the Specify File Locations screen, enter the destination path for the shared ORACLE_HOME. This should be a different location than the shared CRS

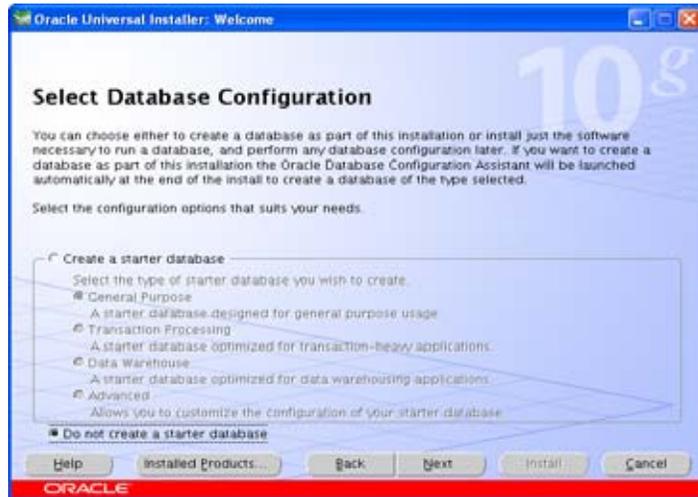
Home. For this exercise, the shared ORACLE_HOME was /volhome/ora10g/product/10.1.0/db_1.



- c. On the next screen, select Cluster Installation and choose all the nodes in the cluster. For our exercise, the two cluster nodes were Inxwks1 and Inxwks2. Click Next. (**Note:** If the nodes are not displayed in the cluster node selection, then Oracle Cluster Ready Services are not configured or started on those cluster nodes.)



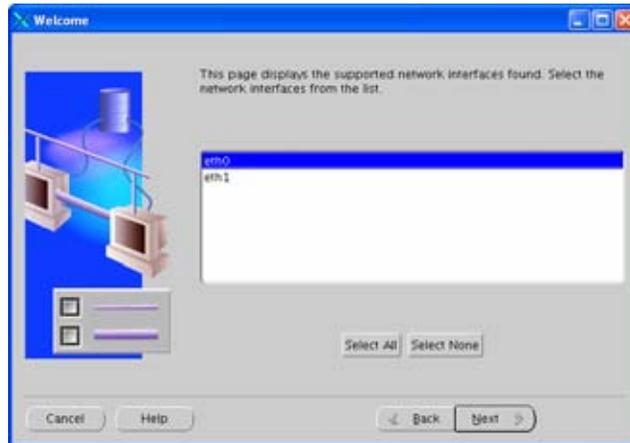
- d. For installation type, select Enterprise Edition and click Next.
- e. On the Select Database Configuration page, choose the "Do not create a starter database" option. We used dbca to create a database later. Click Next to continue further.



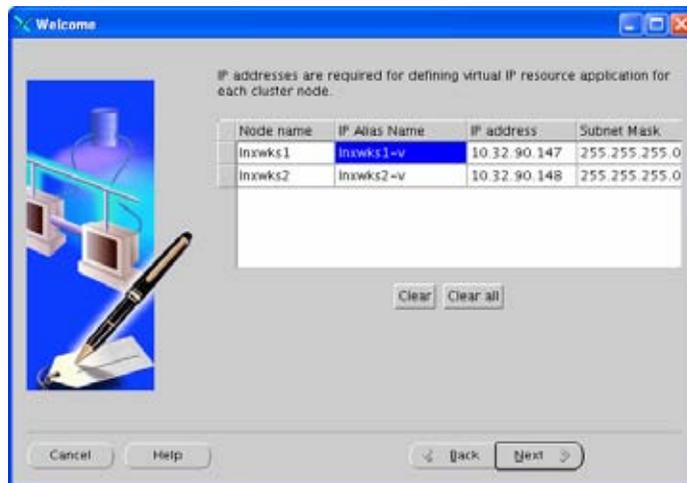
- f. After installing Oracle10g RAC, run root.sh from the ORACLE_HOME directory on all the cluster nodes. Wait for root.sh to finish on the first node before starting on the other cluster nodes. Root.sh must be run from a window that is configured for an xwindows display. Run xclock to test xwindows. If the graphic version of xclock will not run, root.sh will fail due to display issues. The root.sh script will start the Virtual Internet Protocol Configuration Assistant (VIPCA) after finishing the generic part of running root.sh on node 1 of the cluster.



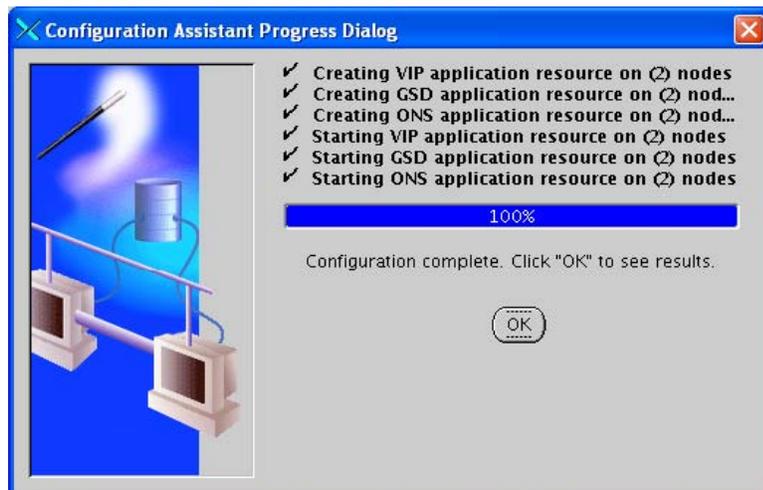
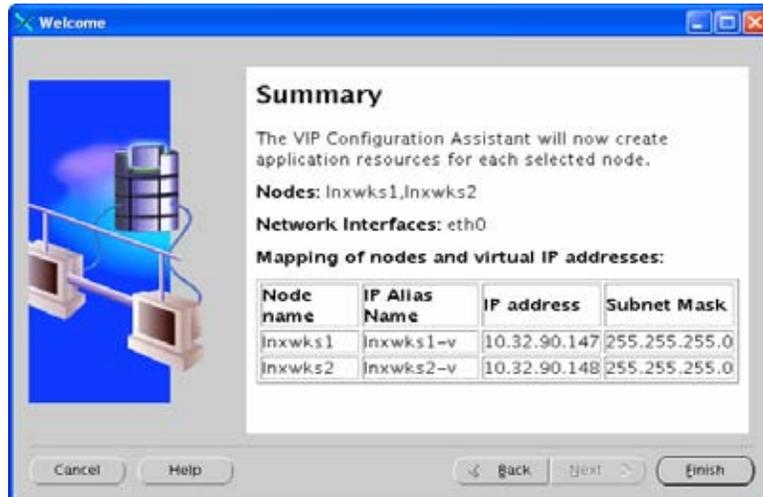
- g. Select the network interface to be used for virtual IP. Do not select the private network interface used by the cluster interconnect. In our case, eth0 (for both nodes) was selected for VIP.



h. Next, enter the IP alias name and the IP address for the virtual IP for each node.



i. Finally, the VIPCA will create the application resources for each node. Click Finish.



4. Running the Oracle Net Configuration Assistant (netca) (no screenshots provided):
 - a. From the first cluster node, execute netca and choose Cluster Configuration.
 - b. Select all nodes in the cluster.
 - c. Select Listener Configuration.
 - d. Choose Add to add a new listener.
 - e. For the listener name, choose the default name, LISTENER.
 - f. Select TCP as the default protocol.

- g. For the TCP port, use the default, 1521. You can specify a different port if you prefer.
- h. After listener configuration is complete, all listeners should be started on respective nodes. Verify that listeners are configured and started on all the nodes. To check the status of the listener, execute the following command on the cluster nodes:

```
$lsnrctl status LISTENER
```

If the listener has not started, start it by executing the following command on all nodes:

```
$lsnrctl start LISTENER
```

- 5. Running DBCA to create an Oracle10g RAC database (no screenshots provided):
 - a. Run the DBCA utility to create an Oracle10g RAC database from Inxwks1.
 - b. On the next screen, select Create a Database.
 - c. Select all nodes in the cluster.
 - d. Select General Purpose.
 - e. Specify Global Database Name and SID Prefix for the database. For our test, we used "grid."
 - f. Accept the default choice for Database Control and Enterprise Manager.
 - g. Specify the password to be used by all the system accounts.
 - h. Select Cluster File System as the storage mechanism for your database.
 - i. Specify the location for the database files to be created. In this example, the location specified is /voldb/ora10g/oradata. Do not specify the shared location used by ORACLE_HOME.
 - j. Choose not to use the Flash Recovery Area.
 - k. Accept the default selection for the next two screens until you arrive at the screen for initialization parameters. Select All Initialization Parameters and click Show Advanced Parameters. You must set the parameter `filesystemio_options=directIO`. This is required and *very important!*

- l. In the next window, specify a different location for Oracle redo log files and one copy of the control file. In our test, the data files and two copies of the control files were stored in the /voldb/ora10g/oradata shared volume, and all the log files and one copy of the control file were stored in /vollog/ora10g/oradata.
- m. Finally, click Finish to start the RAC database creation.
- n. After database creation is complete, select Password Management to unlock the default accounts if needed. The famous scott/tiger account in Oracle10g is locked by default and should be unlocked if needed.
- o. Refer to the Oracle10g Real Application Clusters Installation and Configuration Guide (part no. B10766-04) for more details on using DBCA or creating an Oracle10g database manually using scripts.

8. Known Issues

1. Apply Oracle patch number 3557807:

This applies to Oracle version 10.1.0.2 and is fixed in 10.1.0.3 or later. After the database is installed, please install Oracle patch 3557807 (bug no. 3639444, metalink note 284774.1). This patch fixes an Oracle bug in which IPC uses public Ethernet IP instead of private interconnect. This can result in significant performance issues for some configurations. This patch can be downloaded from <http://metalink.oracle.com>. Patch readme has additional information about patch prerequisites and installation requirements.

2. Cluster nodes may hang during system reboot or shutdown:

Some combinations of Linux version and Oracle RAC database version may cause cluster nodes to hang during system reboot or shutdown. For example this is known to happen with SUSE SLES 9 and Oracle 10.1.0.3 on x86_64 (Opteron). This is expected to be fixed in Oracle 10.1.0.4 or later. There are two known workarounds:

- Cluster daemons can be manually shut down immediately before system shutdown. (Any active RAC database instances should be shut down before shutting down the cluster daemons.)
- ORACLE_HOME and ORA_CRS_HOME can be put on local file systems. (The Oracle cluster registry and quorum file still need to be shared. They can be on a shared NFS file system.) If you have already installed ORACLE_HOME and ORA_CRS_HOME on an NFS file system, you do not necessarily need to reinstall. Do recursive copies of ORACLE_HOME and ORA_CRS_HOME to a local file system, preserving ownership and permissions. Then create symbolic links from the old NFS ORACLE_HOME and ORA_CRS_HOME to the new local ones. If you use this method, remember your cluster registry file and quorum file must remain on a shared NFS file system.

3. On 64-bit systems LD_LIBRARY_PATH should include paths to both 64-bit and 32-bit libraries.

For 64-bit systems make sure LD_LIBRARY_PATH includes paths to 64-bit and 32-bit libraries; 64-bit libraries are in ORACLE_HOME/lib, and 32-bit libraries are in ORACLE_HOME/lib32. Be aware that 64-bit systems may also use some 32-bit libraries.

4. “lsnrctl stop” hangs:

This is an issue in Oracle 10.1.0.3, but is believed to be fixed in 10gR2.

There is an ONS bug in 10.1.0.3.0. If RAC is installed on a shared file system (such as NFS), the \$O_H/opmn/conf directory must be installed locally on each node. This is Oracle bug 3406450 (or Oracle bug 3881276). A symbolic link can be created from \$O_H/opmn/conf to a local directory. In the following example \$HOME is local to each node:

```
$ mv ORACLE_HOME/opmn/conf $ORACLE_HOME/opmn/conf.orig
$ mkdir $HOME/opmn/conf
```

(Do this on each node.)

```
$ cp ORACLE_HOME/opmn/conf.orig/* $HOME/opmn/conf
$ ln -s $HOME/opmn/conf ORACLE_HOME/opmn/conf
```

Refer to Oracle bug 3881276 for more information.

5. CSS miscount computation:

CSS miscount is the number of heartbeats missed before CSS evicts a node. If you are using NetApp cluster failover (CFO), we highly recommend increasing CSS miscount value to 160 seconds to avoid node reboot issues. Since the NetApp cluster failover process can take up to 120 seconds, it is important to increase css timeout value to accommodate storage failover. You can set the CSS timeout value as indicated below.

```
ORA_CRS_HOME/bin/crsctl set css miscount 160
```

Please make sure to reboot all the nodes after increasing css timeout value. Refer to Oracle metalink note 294430.1 for more information about CSS miscount computation.

9. Best Practices and Recommendations

1. Using async I/O:

Currently NetApp does not support running an Oracle10g RAC with async I/O when using Linux database servers and storing Oracle Database files on NFS-mounted volumes. By default, an Oracle10g RAC on a Linux installation disables async I/O. To use async I/O, the Oracle binaries have to be relinked, and the initORA parameter, filesystemio_options, has to be set to asynch. Make sure you DO NOT relink Oracle binaries to enable async I/O.

2. Using uncached I/O for an Oracle Database:

In order to use directio, the volumes used for storing Oracle Database files should be

mounted with the `actimeo=0` mount option. The `actimeo=0` option was validated with Oracle in conjunction with using `directio`.

3. Using NetApp cluster, F7xxc, F8xxc, and FAS9xxc:

NetApp highly recommends running Oracle10g RAC on a highly available and scalable NetApp cluster solution.

10. Appendix

Sample `.profile` file for the `ora10g` user:

```
ORACLE_BASE=/volhome/ora10g; export ORACLE_BASE
ORACLE_PRODUCT=$ORACLE_BASE/product; export ORACLE_PRODUCT
ORACLE_HOME=$ORACLE_PRODUCT/10.1.0/db_1; export ORACLE_HOME
ORA_CRS_HOME=$ORACLE_PRODUCT/10.1.0/crs_1; export ORA_CRS_HOME
ORACLE_SID=rac1; export ORACLE_SID
LD_LIBRARY_PATH=$ORACLE_HOME/lib:$ORACLE_HOME/lib32; export LD_LIBRARY_PATH
# $ORACLE_HOME/lib32 is required for 64 bit versions
PATH=$PATH:$ORACLE_HOME/bin:$ORACLE_HOME:$ORACLE_CRS/bin; export PATH
```

11. Disclaimer

Each environment has its own specific set of requirements, and no guarantees can be given that the results presented in this report will work as expected on other platforms. This paper should assist in the research and troubleshooting that may be required in a particular case and serve as a checklist of items to be aware of. Please forward any errors, omissions, differences, new discoveries, and comments about this paper to afirebau@netapp.com.

