



The Right Skills, at the Right Time, *at the Right Price*



Build a RAC Database for Free with VirtualBox A Step-by Step-Guide

By: Chris Ostrowski





Table of Contents

Overview	1
What You'll Need	1
The Steps	3
Oracle VirtualBox	3
CentOS	3
Create Shared Disks	5
Configure the First Virtual Machine	7
Install the Oracle Grid Software	19
Install the Oracle Database	32
Conclusion	34
About the Author	34



Overview

Oracle Corporation has made it incredibly easy for both individuals and organizations to test drive Oracle software before implementing it. Virtually all of their software and documentation is available for download on the Oracle Technology Network website. For DBAs and developers anxious to learn and use new languages, development environments, and software features, the fully functional software is a godsend for those who want to keep their skills up to date.

Perhaps the only real limitation to this bounty provided by Oracle is hardware. Many of the pieces of software are complex and require significant hardware investments, even for just a sandbox environment (i.e., an environment that doesn't require sizing to accommodate many users logging in simultaneously). As an example, a sandbox environment with Oracle SOA Suite running on top of Oracle WebLogic Server driven by an Oracle database requires a significant amount of RAM just to run. While the sizing of said components can be scaled down, it still requires a machine to have pretty significant resources.

While RAM and disk space costs have dropped significantly in the last couple of years, there is still one area where it is very difficult for DBAs to create their own sandbox environment: Oracle Real Application Clusters (RAC). Traditionally, the basic requirements for a RAC system involve two servers with a disk storage array connecting the two. While Network Attached Storage (NAS) systems have dropped in price in the last couple of years, the cost and installation, as well as the cost of investing in hardware with a singular use, are still beyond reach for most DBAs who wish to set up a sandbox environment.

Two years ago, I set a goal for myself to learn about RAC and I went looking for a solution that, in the best scenario, wouldn't cost me anything. There were various resources on the internet with different pieces of information on how to do this. This paper demonstrates how I was able to do it for \$0 and the things I learned since then that have made the process of building your own RAC system much easier.

This paper demonstrates how I was able to set up a sandbox RAC environment for \$0 and the things I learned since then that have made the process of building your own RAC system much easier.

What You'll Need

Please remember that the software you download from Oracle is for evaluation purposes only. Do not use anything you build with these instructions in a production environment!

Hardware

At a minimum, you'll need 8GB of RAM on the server you're planning to build this on. Why 8GB? You'll need 2 virtual machines and the minimum you'll want to create those machines is 2GB of RAM. The virtual machine "grabs" the 2GB of RAM whether you're actively using it or not. (For a DBA analogy, think of the

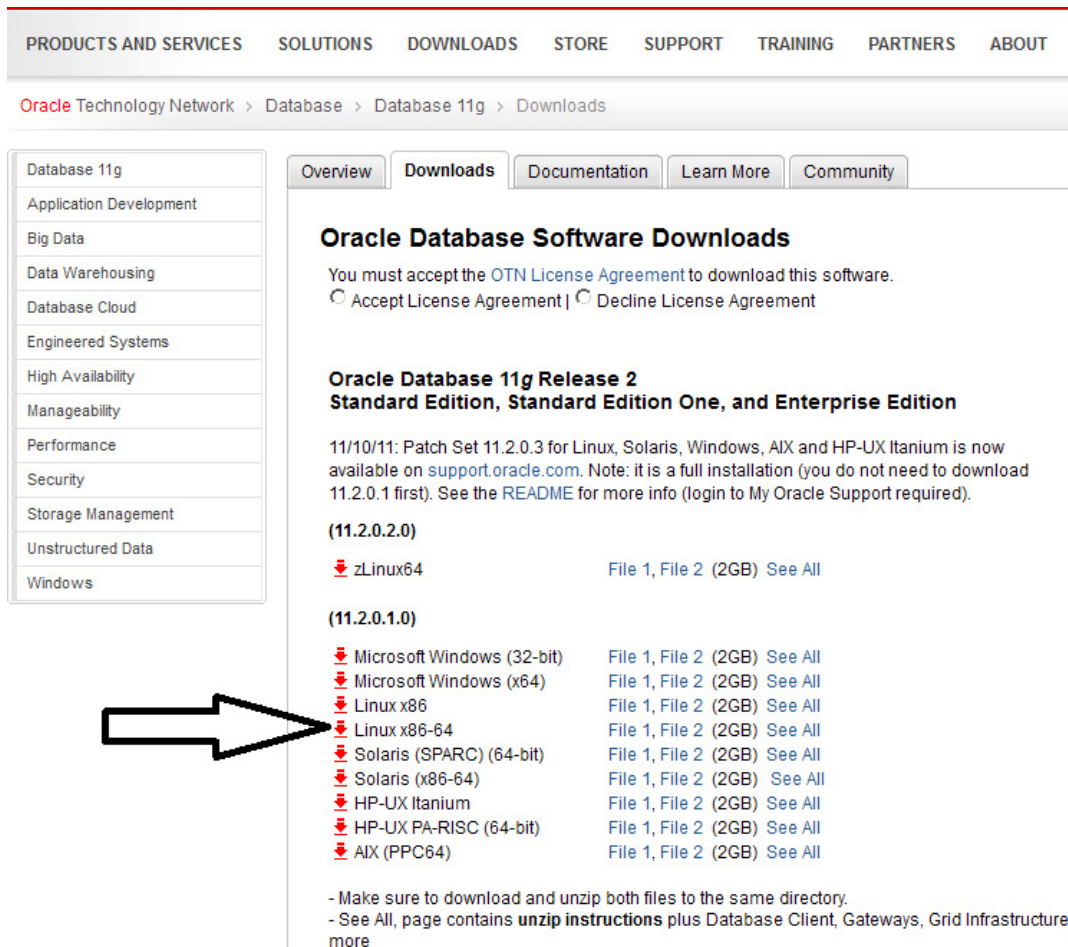
SGA when an Oracle instance starts up – the instance “grabs” the physical memory outlined in your init. ora file and keeps it allocated as long as the instance is running). I know, you’re thinking: 2GB+2GB is 4GB – why do I need 8GB of RAM? It’s never a good idea to use more than 50% of your physical RAM for virtual machines. You certainly CAN do it—it’s very possible, however weird things will start to happen if your VMs use more than 50%.

Disk Space

At a minimum, allocate 20GB for each virtual machine, and at least 30GB for your shared disks, so you’ll need at least 70GB of disk space. As we will see, the virtualization software we’ll use is very efficient at using disk space. The actual disk space used at the host operating system level doesn’t get allocated to the virtual machine until it is needed, but making sure you have at least 70GB of usable disk space will be the minimum to get started.

Software

1. **Oracle Database 11gR2** Available for download at <http://www.oracle.com/technetwork/database/enterprise-edition/downloads/index.html>. As of February 2013, the latest version available is 11.2.0.1.0. Download the two files that make up the Linux-x86-64 link:



PRODUCTS AND SERVICES SOLUTIONS DOWNLOADS STORE SUPPORT TRAINING PARTNERS ABOUT

Oracle Technology Network > Database > Database 11g > Downloads

Database 11g
Application Development
Big Data
Data Warehousing
Database Cloud
Engineered Systems
High Availability
Manageability
Performance
Security
Storage Management
Unstructured Data
Windows

Overview Downloads Documentation Learn More Community

Oracle Database Software Downloads

You must accept the [OTN License Agreement](#) to download this software.
☐ Accept License Agreement | ☐ Decline License Agreement

Oracle Database 11g Release 2 Standard Edition, Standard Edition One, and Enterprise Edition

11/10/11: Patch Set 11.2.0.3 for Linux, Solaris, Windows, AIX and HP-UX Itanium is now available on [support.oracle.com](#). Note: it is a full installation (you do not need to download 11.2.0.1 first). See the [README](#) for more info (login to My Oracle Support required).

(11.2.0.2.0)

Linux64 File 1, File 2 (2GB) See All

(11.2.0.1.0)

Microsoft Windows (32-bit) File 1, File 2 (2GB) See All
 Microsoft Windows (x64) File 1, File 2 (2GB) See All
 Linux x86 File 1, File 2 (2GB) See All
 Linux x86-64 File 1, File 2 (2GB) See All
 Solaris (SPARC) (64-bit) File 1, File 2 (2GB) See All
 Solaris (x86-64) File 1, File 2 (2GB) See All
 HP-UX Itanium File 1, File 2 (2GB) See All
 HP-UX PA-RISC (64-bit) File 1, File 2 (2GB) See All
 AIX (PPC64) File 1, File 2 (2GB) See All

- Make sure to download and unzip both files to the same directory.
 - See All, page contains **unzip instructions** plus Database Client, Gateways, Grid Infrastructure, more

2. **Oracle Grid Software** – the Oracle Grid software is what communicates between your servers and what allows the servers to act as a single entity. The grid software can be downloaded from <http://www.oracle.com/technetwork/products/clusterware/downloads/index.html>. As of February 2013, the latest version of the Grid software is 11.2.0.1.0. Download the Linux x86-64 version. Make sure to also grab the cluvfy utility – this will be used to verify the cluster right before installing.
3. **CentOS Release 5.9 64-bit** – CentOS is a free operating system that is equivalent (with some very minor exceptions) to Red Hat Enterprise Linux. You can find a public mirror to download CentOS from <http://centos.org/modules/tinycontent/index.php?id=15>. From there, click on CentOS Mirror Public List, then select a location close to you (don't click on the actual name of the site – click on either http or ftp to the right side of the page). After that, look for a folder called 5.9 and click that, then click ISOs and x86_64. Click on and download the DVD image (CentOS-5.9-x86_64-bin-DVD-1of2.iso) – don't worry if you don't have a DVD burner; we're not going to actually burn the DVD.
4. **Oracle VirtualBox** Available from <http://www.oracle.com/technetwork/server-storage/virtualbox/downloads/index.html>. Oracle VirtualBox is a free virtualization program from Oracle. It differs from Oracle's other virtualization product (Oracle VM) in the important distinction that it requires an underlying operating system to run on top of. As such, it is not suitable to most virtualized production environments as all system calls (disk reads and writes, memory reads and writes, etc.) have to be "translated" to the native host operating system. This usually causes enough of a performance hit that using VirtualBox in production is not acceptable. For our purposes, however, VirtualBox will do the job.

Believe it or not, that's all the pieces you'll need to build your own sandbox RAC environment.

The Steps

Oracle VirtualBox

First, install Oracle VirtualBox on the machine you wish to use. As mentioned before, make sure you have at least 8GB of RAM and 70GB of disk space on this server. The installation is very straightforward and will not be covered in detail here.

CentOS

The process we're going to use to create our virtual machines is as follows: we'll create the first virtual machine, create shared disks, then clone the first virtual machine. After VirtualBox is installed, run it and create a new virtual machine by clicking on the "New" icon in the top left of the screen. Give your new virtual machine a meaningful name (I called mine RAC1), select Linux as the type and "Red Hat (64-bit)" as the version. For memory size, select 2048MB. Note that this is the minimum – if you have more memory you can use on this server, bump up the memory allocation accordingly.

Next, select "Create a virtual hard drive now," then VDI (VirtualBox Disk Image), then "Dynamically Allocated." Specify a location and make sure the disk is at least 30GB (again, you can allocate more if

you have the space). I mentioned earlier that the virtualization software we're going to use is very efficient when it comes to disk space. After creating the virtual machine, we can look at the corresponding file on our base operating system and we'll see that it's much less than 30GB in size – VirtualBox will dynamically allocate space as it's needed, up to 30GB (or more if we specify more in the wizard).

After that last page in the wizard, you'll see the main VirtualBox page listing the virtual machines that have been created. Before we can start up our VM, we need to make a few tweaks to the network options for the VM. Click on the Network link on the right side of the page, then click on the Adapter 1 tab. Make sure "Enable Network Adapter" is checked and "Attached to:" is set to Bridged Adapter, then click Adapter 2. Make sure "Enable Network Adapter" is checked and "Attached to:" is set to Internal Network.

Why do we do this? Oracle RAC needs two network cards attached to each server – one to handle communications with the outside world and one to handle communications between the two servers. This second connection is referred to as interprocess communication and needs to be a direct connection between the two servers – this is why the second network adapter for the virtual machine has a connection type of "Internal Network".

Click on OK to close the wizard, then click "Start" in the top left of the VirtualBox Manager window. Since this is the first time we're starting up the virtual machine, VirtualBox is smart enough to ask where the operating system disk is. Click the folder icon to the left and find where you saved the CentOS ISO file (CentOS-5.9-x86_64-bin-DVD-1of2.iso). Continue through the Oracle Linux 5 installation as you would for a basic server. It should be a server installation with:

- A minimum of 4GB of swap space
- Firewall disabled
- SELinux set to disabled
- Package groups:
 - » Desktop Environments > GNOME Desktop Environment
 - » Applications > Editors and Graphical Internet
 - » Development > Development Libraries and Development Tools
 - » Servers > Server Configuration Tools

On the networking screen, do NOT choose DHCP – the IP addresses need to remain consistent for your server, so pick an IP address for both eth0 (the public interface) and eth1 (the private interface (interconnect)). Make sure both addresses are on a different subnet. As an example, I used the following on my system:

IP Address eth0: 192.168.0.101 (public address)

Default Gateway eth0: 192.168.0.1 (public address)

IP Address eth1: 192.168.1.101 (private address)

Default Gateway eth1: none

Upon completion, shut down your server.



Create Shared Disks

Here's where we get to use the really cool features of VirtualBox. In VirtualBox, we can create network-attached disks just by issuing two commands:

```
VBoxManage createhd --filename c:\VMs\shared\asm1.vdi --size 10240 --format VDI --variant Fixed
```

```
VBoxManage storageattach RAC1 --storagectl "SATA" --port 1 --device 0 --type hdd --medium c:\VMs\shared\asm1.vdi --mttype shareable
```

The first command creates a 10GB disk and makes it available to VirtualBox. The second command attaches the disk to a specific virtual machine. Since we specified `--mttype shareable` at the end, the disk can be attached to more than one virtual machine. After we clone RAC1, we'll attach the disks to the second virtual machine.

Issue the following commands to create four more attached disks:

```
VBoxManage createhd --filename c:\VMs\shared\asm2.vdi --size 10240 --format VDI --variant Fixed
```

```
VBoxManage createhd --filename c:\VMs\shared\asm3.vdi --size 10240 --format VDI --variant Fixed
```

```
VBoxManage createhd --filename c:\VMs\shared\asm4.vdi --size 10240 --format VDI --variant Fixed
```

```
VBoxManage createhd --filename c:\VMs\shared\asm5.vdi --size 10240 --format VDI --variant Fixed
```

And then attach them to the RAC1 virtual machine:

```
VBoxManage storageattach RAC1 --storagectl "SATA" --port 2 --device 0 --type hdd --medium c:\VMs\shared\asm2.vdi --mttype shareable
```

```
VBoxManage storageattach RAC1 --storagectl "SATA" --port 3 --device 0 --type hdd --medium c:\VMs\shared\asm3.vdi --mttype shareable
```

```
VBoxManage storageattach RAC1 --storagectl "SATA" --port 4 --device 0 --type hdd --medium c:\VMs\shared\asm4.vdi --mttype shareable
```

```
VBoxManage storageattach RAC1 --storagectl "SATA" --port 5 --device 0 --type hdd --medium c:\VMs\shared\asm5.vdi --mttype shareable
```

Even though we've defined the disks as sharable, we still need to issue the following commands:

```
VBoxManage modifyhd c:\VMs\shared\asm1.vdi --type shareable
```

```
VBoxManage modifyhd c:\VMs\shared\asm2.vdi --type shareable
```

```
VBoxManage modifyhd c:\VMs\shared\asm3.vdi --type shareable
```

```
VBoxManage modifyhd c:\VMs\shared\asm4.vdi --type shareable
```

```
VBoxManage modifyhd c:\VMs\shared\asm5.vdi --type shareable
```



At the virtual machine operating system level, the new disks will be named:

/dev/sdb

/dev/sdc

/dev/sdd

/dev/sde and

/dev/sdf

Start the RAC1 virtual machine and partition the new disks:

```
# fdisk /dev/sdb
```

```
Command (m for help): n
```

```
Command action
```

```
e extended
```

```
p primary partition (1-4)
```

```
p
```

```
Partition number (1-4): 1
```

```
First cylinder (1-1305, default 1):
```

```
Using default value 1
```

```
Last cylinder or +size or +sizeM or +sizeK (1-1305,default 1305):
```

```
Using default value 1305
```

```
Command (m for help): p
```

```
Disk /dev/sdb: 10.7 GB, 10737418240 bytes
```

```
255 heads, 63 sectors/track, 1305 cylinders
```

```
Units = cylinders of 16065 * 512 = 8225280 bytes
```

```
Device Boot Start End Blocks Id System
```

```
/dev/sdb1 1 1305 10482381 83 Linux
```

```
Command (m for help): w
```

```
The partition table has been altered!
```

```
Calling ioctl() to re-read partition table.
```

```
Syncing disks.
```

```
Repeat the process for disks /dev/sdc through /dev/sdf.
```




Configure the First Virtual Machine

Step 1: Create groups

As root:

```
/usr/sbin/groupadd -g 500 dba
/usr/sbin/groupadd -g 600 oinstall
/usr/sbin/groupadd -g 700 oper
/usr/sbin/groupadd -g 800 asm
cat /etc/group
```

Step 2: Check that user nobody exists

As root:

```
grep nobody /etc/passwd
```

Step 3: Add Oracle user

As root:

```
/usr/sbin/useradd -b /home/local/oracle -d /home/local/oracle -g 500 -m -p oracle -u 500 -s /bin/
bash oracle
grep oracle /etc/passwd
/usr/sbin/usermod -g oinstall oracle
/usr/sbin/usermod -a -G dba oracle
/usr/sbin/usermod -a -G oper oracle
/usr/sbin/usermod -a -G asm oracle
id oracle
uid=500(oracle) gid=600(oinstall) groups=600(oinstall),500(dba),700(oper),800(asm)
```

Step 4: Set up directories

As root, create directories for Oracle grid software (must be outside of Oracle's home directory), change ownership and permission levels.

```
cd /
mkdir oracledb
mkdir oraclegrid
mkdir oraclegridbase
mkdir oraInventory
chown oracle:oinstall oracledb
chown oracle:oinstall oraclegrid
chown oracle:oinstall oraclegridbase
chown oracle:oinstall oraInventory
chmod 777 oracledb
chmod 777 oraclegrid
chmod 777 oraclegridbase
```



```
chmod 777 oraInventory
```

Step 5: Unzip Oracle software

As oracle:

```
[oracle@RAC1 software]$ pwd
/home/local/oracle/software

unzip linux.x64_11gR2_grid.zip
unzip linux.x64_11gR2_database_1of2.zip
unzip linux.x64_11gR2_database_2of2.zip

mkdir cvu
mv cvupack_Linux_x86_64.zip cvu
cd cvu
unzip cvupack_Linux_x86_64.zip
```

Step 6: Test XWindows

On PC:

Start Xming

Modify putty connection: Connection -> SSH -> X11 -> Select "Enable X11 forwarding" and add :0.0 to X Display Location

On server:

Make sure DISPLAY is set to localhost:10.0
xclock &

If you get the error "MIT-MAGIC-COOKIE-1 data did not match":

```
Log in as root
xauth list - put the returned value in your buffer
su - oracle
xauth add <paste from your buffer>
```

Step 7: Verify that the following packages exist

64-bit only:

```
yum install binutils.x86_64 -y
yum install elfutils-libelf.x86_64 -y
yum install elfutils-libelf-devel.x86_64 -y
yum install gcc.x86_64 -y
yum install gcc-c++.x86_64 -y
yum install glibc-common.x86_64 -y
yum install libstdc++-devel.x86_64 -y
yum install make.x86_64 -y
yum install sysstat.x86_64 -y
```

**Both 32 and 64 bit:**

```
yum install compat-libstdc++-33.i386 -y
yum install compat-libstdc++-33.x86_64 -y
yum install glibc.i686 -y
yum install glibc.x86_64 -y
yum install glibc-devel.i386 -y
yum install glibc-devel.x86_64 -y
yum install libaio.i386 -y
yum install libaio.x86_64 -y
yum install libgcc.i386 -y
yum install libgcc.x86_64 -y
yum install libstdc++.i386 -y
yum install libstdc++.x86_64 -y
yum install libaio-devel.x86_64 -y
yum install libaio-devel.i386 -y
yum install unixODBC.x86_64 -y
yum install unixODBC.i386 -y
yum install unixODBC-devel.i386 -y
yum install unixODBC-devel.x86_64 -y
```

Step 8: Change security level

Disable SELinux

As root:

```
selinuxenabled && echo enabled || echo disabled
```

To disable:

```
echo 0 > /selinux/enforce
```

Step 9: Check NTP

```
vi /etc/sysconfig/ntpd
```

Add -x to end of OPTIONS line (inside of quote marks)

```
/sbin/service ntpd stop
/sbin/service ntpd start
/usr/sbin/ntpq
ntpq> peers
```

Make sure at least one entry shows up. If not:

1. copy /etc/ntp.conf from RAC1.
2. /sbin/service ntpd stop
3. /sbin/service ntpd start



4. /usr/sbin/ntpq

5. ntpq> peers

For ntpd reference, see: <http://www.eecis.udel.edu/~mills/ntp/html/ntpd.html>.

Step 10: Set kernel parameters

```
vi /etc/sysctl.conf
```

```
kernel.sem=250 32000 100 142
fs.file-max=327679
net.ipv4.ip_local_port_range=1024 65000
net.core.rmem_default=4194304
net.core.rmem_max=4194304
net.core.wmem_default=262144
net.core.wmem_max=262144
net.ipv4.tcp_rmem=4194304 4194304 4194304
net.ipv4.tcp_wmem=262144 262144 262144
```

```
vi /etc/security/limits.conf
```

```
oracle soft nfile 131072
oracle hard nfile 131072
oracle soft nproc 131072
oracle hard nproc 131072
```

```
vi /etc/pam.d/login
```

```
session    required    pam_limits.so
```

Have system changes take effect:

```
sysctl -p
```

Step 11: Configure hangcheck timer

```
/sbin/insmod /lib/modules/2.6.18-308.11.1.el5/kernel/drivers/char/hangcheck-timer.ko hangcheck_
tick=1 hangcheck_margin=10 hangcheck_reboot=1
```

Check that at least 1 row is returned:

```
[root@RAC1 bin]# lsmod | grep -i hang
hangcheck_timer      2526  0
```

Add command to /etc/rc.d/rc.local:

```
vi /etc/rc.d/rc.local
```

```
/sbin/insmod /lib/modules/2.6.18-308.11.1.el5/kernel/drivers/char/hangcheck-timer.ko
hangcheck_tick=1 hangcheck_margin=10 hangcheck_reboot=1
```



Step 12: Configure network

Node 1 Public: 192.168.0.101 (bond0)

Node 1 Private: 192.168.1.101 (bond1)

```
[root@RAC11 ~]# cat /etc/hosts
```

```
127.0.0.1 RAC1 RAC1    localhost localhost.localdomain localhost4 localhost4.localdomain4
::1 localhost localhost.localdomain localhost6 localhost6.localdomain6
```

Add these lines:

Public

```
192.168.0.170      rac1.localdomain rac1          rac2.localdomain rac2
```

Private

```
192.168.2.170      rac1-priv.localdomain rac1-priv
```

```
192.168.2.180      rac2-priv.localdomain rac2-priv
```

Virtual

```
192.168.0.171      rac1-vip.localdomain rac1-vip
```

```
192.168.0.181      rac2-vip.localdomain rac2-vip
```

SCAN

```
192.168.0.190      rac-scan.localdomain rac-scan
```

```
192.168.0.191      rac-scan.localdomain rac-scan
```

```
192.168.0.192      rac-scan.localdomain rac-scan
```

Step 13: Configure ASM support

Step 13.1: Download 3 files based on kernel version

<http://www.oracle.com/technetwork/server-storage/linux/downloads/rhel5-084877.html>

oracleasm-2.6.18-308.11.1.el5-2.0.5-1.el5.x86_64.rpm

oracleasm-lib-2.0.4-1.el5.x86_64.rpm

oracleasm-support-2.1.7-1.el5.x86_64.rpm

Step 13.2: Install ASM RPMs as root

```
rpm -ivf oracleasm-support-2.1.7-1.el5.x86_64.rpm
```

```
rpm -ivf oracleasm-2.6.18-308.11.1.el5-2.0.5-1.el5.x86_64.rpm
```

```
rpm -ivf oracleasm-lib-2.0.4-1.el5.x86_64.rpm
```

Step 13.3: Check that all were installed successfully

```
[root@RAC1 software]# rpm -qav | grep oracleasm
```

```
oracleasm-2.6.18-308.11.1.el5-2.0.5-1.el5
```

```
oracleasm-support-2.1.7-1.el5
```

```
oracleasm-lib-2.0.4-1.el5
```




Step 13.4: Configure ASM

```
[root@RAC1 software]# /etc/init.d/oracleasm configure -i
Configuring the Oracle ASM library driver.
```

This will configure the on-boot properties of the Oracle ASM library driver. The following questions will determine whether the driver is loaded on boot and what permissions it will have. The current values will be shown in brackets ('[]'). Hitting <ENTER> without typing an answer will keep that current value. Ctrl-C will abort.

```
Default user to own the driver interface []: oracle
Default group to own the driver interface []: asm
Start Oracle ASM library driver on boot (y/n) [n]: y
Scan for Oracle ASM disks on boot (y/n) [y]: y
Writing Oracle ASM library driver configuration: done
```

Step 13.5: Initialize ASM

```
[root@RAC1 /]# /etc/init.d/oracleasm stop
Dropping Oracle ASMLib disks:          [ OK ]
Shutting down the Oracle ASMLib driver: [ OK ]

[root@RAC1 /]# /etc/init.d/oracleasm start
Initializing the Oracle ASMLib driver:  [ OK ]
Scanning the system for Oracle ASMLib disks: [ OK ]

[root@RAC1 /]# /etc/init.d/oracleasm status
Checking if ASM is loaded: yes
Checking if /dev/oracleasm is mounted: yes
```

Step 14: Verify Cluster

Step 14.1 Run cluvfy

```
[oracle@RAC1 bin]$ pwd
/home/local/oracle/software/cvu/bin

[oracle@RAC1 bin]$ ./cluvfy comp sys -n RAC11 -p crs -r 11gR2 -osdba dba

Verifying system requirement
Total memory check passed
Available memory check passed
Swap space check passed
Free disk space check passed for "RAC1:/tmp"
Check for multiple users with UID value 500 passed
User existence check passed for "oracle"
Group existence check passed for "oinstall"
Group existence check passed for "dba"
```



Membership check for user "oracle" in group "oinstall" [as Primary] passed

Membership check for user "oracle" in group "dba" passed

Run level check passed

Hard limits check passed for "maximum open file descriptors"

Soft limits check passed for "maximum open file descriptors"

Hard limits check passed for "maximum user processes"

Soft limits check passed for "maximum user processes"

System architecture check passed

Kernel version check passed

Kernel parameter check passed for "semmsl"

Kernel parameter check passed for "semmns"

Kernel parameter check passed for "semopm"

Kernel parameter check passed for "semmni"

Kernel parameter check passed for "shmmax"

Kernel parameter check failed for "shmmni"

Check failed on nodes:

RAC1

Kernel parameter check passed for "shmall"

Kernel parameter check failed for "file-max"

Check failed on nodes:

RAC1

Kernel parameter check passed for "ip_local_port_range"

Kernel parameter check passed for "rmem_default"

Kernel parameter check passed for "rmem_max"

Kernel parameter check passed for "wmem_default"

Kernel parameter check failed for "wmem_max"

Check failed on nodes:

RAC1

Kernel parameter check failed for "aio-max-nr"

Check failed on nodes:

RAC1

Package existence check passed for "make"

Package existence check passed for "binutils"

Package existence check passed for "gcc(x86_64)"

Package existence check passed for "libaio(x86_64)"

Package existence check passed for "glibc(x86_64)"

Package existence check passed for "compat-libstdc++-33(x86_64)"

Package existence check passed for "elfutils-libelf(x86_64)"

Package existence check passed for "elfutils-libelf-devel"

Package existence check passed for "glibc-common"

Package existence check passed for "glibc-devel(x86_64)"

Package existence check passed for "glibc-headers"

Package existence check passed for "gcc-c++(x86_64)"

Package existence check passed for "libaio-devel(x86_64)"



Package existence check passed for "libgcc(x86_64)"
Package existence check passed for "libstdc++(x86_64)"
Package existence check passed for "libstdc++-devel(x86_64)"
Package existence check passed for "sysstat"
Package existence check passed for "ksh"
Check for multiple users with UID value 0 passed

Starting check for consistency of primary group of root user

Check for consistency of root user's primary group passed
Time zone consistency check passed

Verification of system requirement was unsuccessful on all the specified nodes.

Step 14.2 Run cluvfy with -fixup switch

```
./cluvfy comp sys -n RAC1 -p crs -r 11gR2 -osdba dba -fixup -fixupdir /home/local/oracle/  
software/cvu/bin/fixit
```

Log in as root:

```
cd /tmp/CVU_11.2.0.3.0_oracle  
./runfixup.sh
```

Log back in as oracle:

```
su - oracle
```

Step 14.3 Verify Cluster again

```
[oracle@RAC1 bin]$ ./cluvfy comp sys -n RAC1 -p crs -r 11gR2 -osdba dba
```

Verifying system requirement
Total memory check passed
Available memory check passed
Swap space check passed
Free disk space check passed for "RAC1:/tmp"
Check for multiple users with UID value 500 passed
User existence check passed for "oracle"
Group existence check passed for "oinstall"
Group existence check passed for "dba"
Membership check for user "oracle" in group "oinstall" [as Primary] passed
Membership check for user "oracle" in group "dba" passed
Run level check passed
Hard limits check passed for "maximum open file descriptors"
Soft limits check passed for "maximum open file descriptors"



Hard limits check passed for "maximum user processes"
Soft limits check passed for "maximum user processes"
System architecture check passed
Kernel version check passed
Kernel parameter check passed for "semmsl"
Kernel parameter check passed for "semmns"
Kernel parameter check passed for "semopm"
Kernel parameter check passed for "semmni"
Kernel parameter check passed for "shmmax"
Kernel parameter check passed for "shmmni"
Kernel parameter check passed for "shmall"
Kernel parameter check passed for "file-max"
Kernel parameter check passed for "ip_local_port_range"
Kernel parameter check passed for "rmem_default"
Kernel parameter check passed for "rmem_max"
Kernel parameter check passed for "wmem_default"
Kernel parameter check passed for "wmem_max"
Kernel parameter check passed for "aio-max-nr"
Package existence check passed for "make"
Package existence check passed for "binutils"
Package existence check passed for "gcc(x86_64)"
Package existence check passed for "libaio(x86_64)"
Package existence check passed for "glibc(x86_64)"
Package existence check passed for "compat-libstdc++-33(x86_64)"
Package existence check passed for "elfutils-libelf(x86_64)"
Package existence check passed for "elfutils-libelf-devel"
Package existence check passed for "glibc-common"
Package existence check passed for "glibc-devel(x86_64)"
Package existence check passed for "glibc-headers"
Package existence check passed for "gcc-c++(x86_64)"
Package existence check passed for "libaio-devel(x86_64)"
Package existence check passed for "libgcc(x86_64)"
Package existence check passed for "libstdc++(x86_64)"
Package existence check passed for "libstdc++-devel(x86_64)"
Package existence check passed for "sysstat"
Package existence check passed for "ksh"
Check for multiple users with UID value 0 passed

Starting check for consistency of primary group of root user

Check for consistency of root user's primary group passed
Time zone consistency check passed

Verification of system requirement was successful.

**Step 15: Do steps 1-4 and 7-14 for node 2**

On node 2, cut and paste the commands for each of the steps from this document.

Step 16: Create ASM disks

On Node 1 as root, reset the headers on the disks:

```
dd if=/dev/zero of=/dev/sdb bs=1024 count=1000
dd if=/dev/zero of=/dev/sdc bs=1024 count=1000
dd if=/dev/zero of=/dev/sdd bs=1024 count=1000
dd if=/dev/zero of=/dev/sde bs=1024 count=1000
dd if=/dev/zero of=/dev/sdf bs=1024 count=1000
```

Make sure ownership and permissions are correct on all 3 nodes:

```
[root@RAC1 etc]# ls -ltr /dev/sd*
brw-rw---- 1 oracle oinstall 253, 3 Aug  9 08:03 /dev/sdb
brw-rw---- 1 oracle oinstall 253, 4 Aug  9 08:03 /dev/sdc
brw-rw---- 1 oracle oinstall 253, 5 Aug  9 08:03 /dev/sdd
brw-rw---- 1 oracle oinstall 253, 6 Aug  9 08:03 /dev/sde
brw-rw---- 1 oracle oinstall 253, 6 Aug  9 08:03 /dev/sdf
```

On node 1 as root:

```
[root@RAC1 ~]# /etc/init.d/oracleasm createdisk data1 /dev/sdb
Writing disk header: done
Instantiating disk: done
```

```
[root@RAC1 ~]# /etc/init.d/oracleasm createdisk data2 /dev/sdc
Writing disk header: done
Instantiating disk: done
```

```
[root@RAC1 ~]# /etc/init.d/oracleasm createdisk data3 /dev/sdd
Writing disk header: done
Instantiating disk: done
```

```
[root@RAC1 ~]# /etc/init.d/oracleasm createdisk data4 /dev/sde
Writing disk header: done
Instantiating disk: done
```

```
[root@RAC1 ~]# /etc/init.d/oracleasm createdisk data5 /dev/sdf
Writing disk header: done
Instantiating disk: done
```

```
[root@RAC1 ~]# /etc/init.d/oracleasm listdisks
DATA1
DATA2
DATA3
```




DATA4
DATA5

On Node 2 as root:

```
[root@RAC2 CVU_11.2.0.3.0_oracle]# /etc/init.d/oracleasm scandisks
Reloading disk partitions: done
Cleaning any stale ASM disks...
Scanning system for ASM disks...
Instantiating disk "DATA1"
Instantiating disk "DATA2"
Instantiating disk "DATA3"
Instantiating disk "DATA4"
Instantiating disk "DATA5"
```

```
[root@RAC2 CVU_11.2.0.3.0_oracle]# /etc/init.d/oracleasm listdisks
DATA1
DATA2
DATA3
DATA4
DATA5
```

Step 17: Clone the VM

Shut down RAC1:

```
# shutdown -h now
```

Clone the RAC.vdi disk:

```
VBoxManage clonehd c:\VMs\RAC1\RAC.vdi c:\VMs\RAC2\RAC.vdi
```

Create the RAC2 virtual machine in VirtualBox in the same way as you did for RAC1, with the exception of using the existing RAC.vdi virtual hard drive.

Add the second network adaptor as you did on RAC1. After the VM is created, attach the shared disks to RAC2:

```
VBoxManage storageattach ol5-112-rac2 --storagectl "SATA" --port 1 --device 0 --type hdd
--medium c:\VMs\shared\asm1.vdi --mttype shareable
```

```
VBoxManage storageattach ol5-112-rac2 --storagectl "SATA" --port 2 --device 0 --type hdd
--medium c:\VMs\shared\asm2.vdi --mttype shareable
```

```
VBoxManage storageattach ol5-112-rac2 --storagectl "SATA" --port 3 --device 0 --type hdd
--medium c:\VMs\shared\asm3.vdi --mttype shareable
```

```
VBoxManage storageattach ol5-112-rac2 --storagectl "SATA" --port 4 --device 0 --type hdd
--medium c:\VMs\shared\asm4.vdi --mttype shareable
```



```
VBoxManage storageattach ol5-112-rac2 --storagectl "SATA" --port 5 --device 0 --type hdd  
--medium c:\VMs\shared\asm5.vdi --mtpe shareable
```

Start RAC2 by clicking the “Start” button on the toolbar. Ignore any network errors during the startup.

Log in to the RAC2 as “root” and reconfigure the network settings:

```
hostname: RAC2  
IP Address eth0: 192.168.0.180 (public address)  
Default Gateway eth0: 192.168.0.1 (public address)  
IP Address eth1: 192.168.2.180 (private address)  
Default Gateway eth1: none
```

Amend the hostname in the “/etc/sysconfig/network” file.

```
NETWORKING=yes  
HOSTNAME=RAC2
```

Remove the current “ifcfg-eth0” and “ifcfg-eth1” scripts and rename the original scripts from the backup names:

```
# cd /etc/sysconfig/network-scripts/  
# rm ifcfg-eth0 ifcfg-eth1  
# mv ifcfg-eth0.bak ifcfg-eth0  
# mv ifcfg-eth1.bak ifcfg-eth1
```

Edit the “/etc/sysconfig/network-scripts/ifcfg-eth0”, amending only the IPADDR and HWADDR settings as follows:

```
HWADDR=08:00:27:95:ED:33  
IPADDR=192.168.0.102
```

Edit the “/etc/sysconfig/network-scripts/ifcfg-eth1”, amending only the IPADDR and HWADDR settings as follows:

```
HWADDR=08:00:27:E3:DA:B6  
IPADDR=192.168.1.112
```

Edit the “/home/oracle/.bash_profile” file on the “ol5-112-rac2” node to correct the ORACLE_SID and

```
ORACLE_HOSTNAME values.  
ORACLE_SID=RAC2; export ORACLE_SID  
ORACLE_HOSTNAME=RAC2; export ORACLE_HOSTNAME
```

Restart RAC2 and start RAC1. When both nodes have started, check they can both ping all the public and private IP addresses using the following commands:

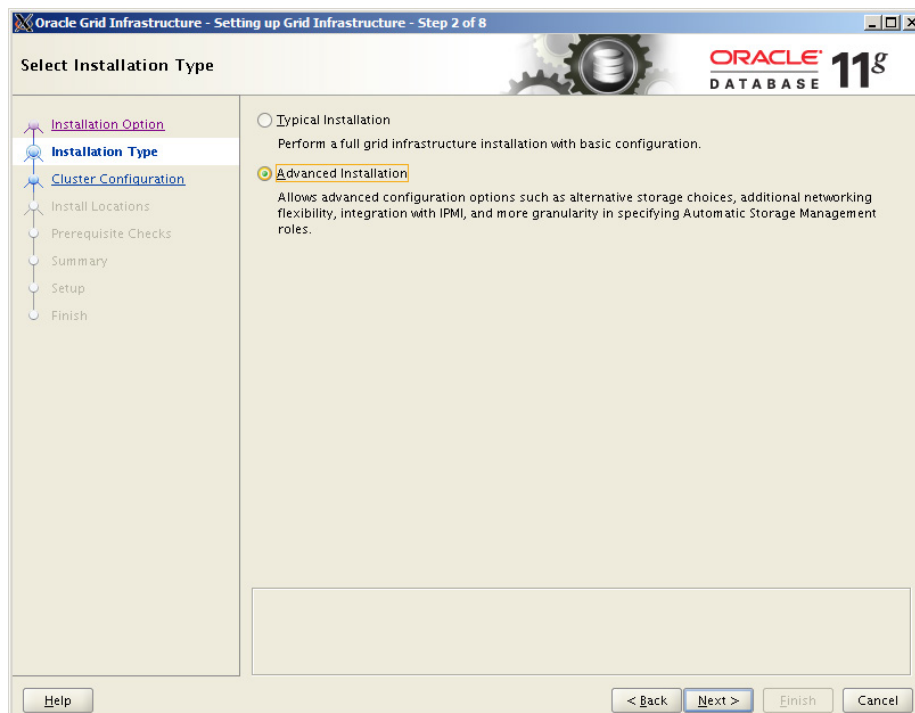
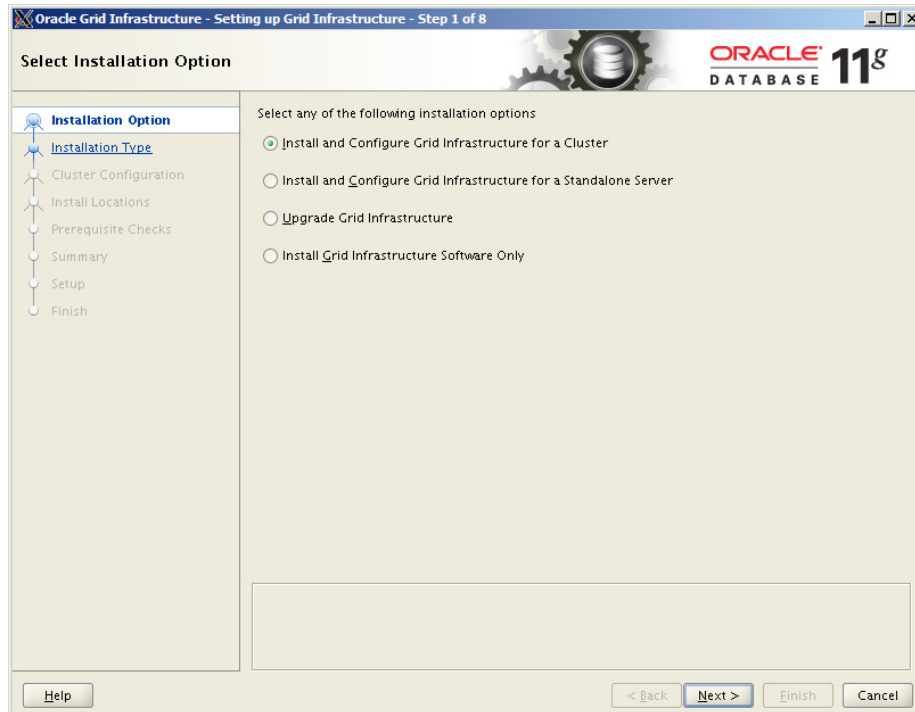
```
ping -c 3 RAC1  
ping -c 3 RAC11-priv  
ping -c 3 RAC2  
ping -c 3 RAC2-priv
```

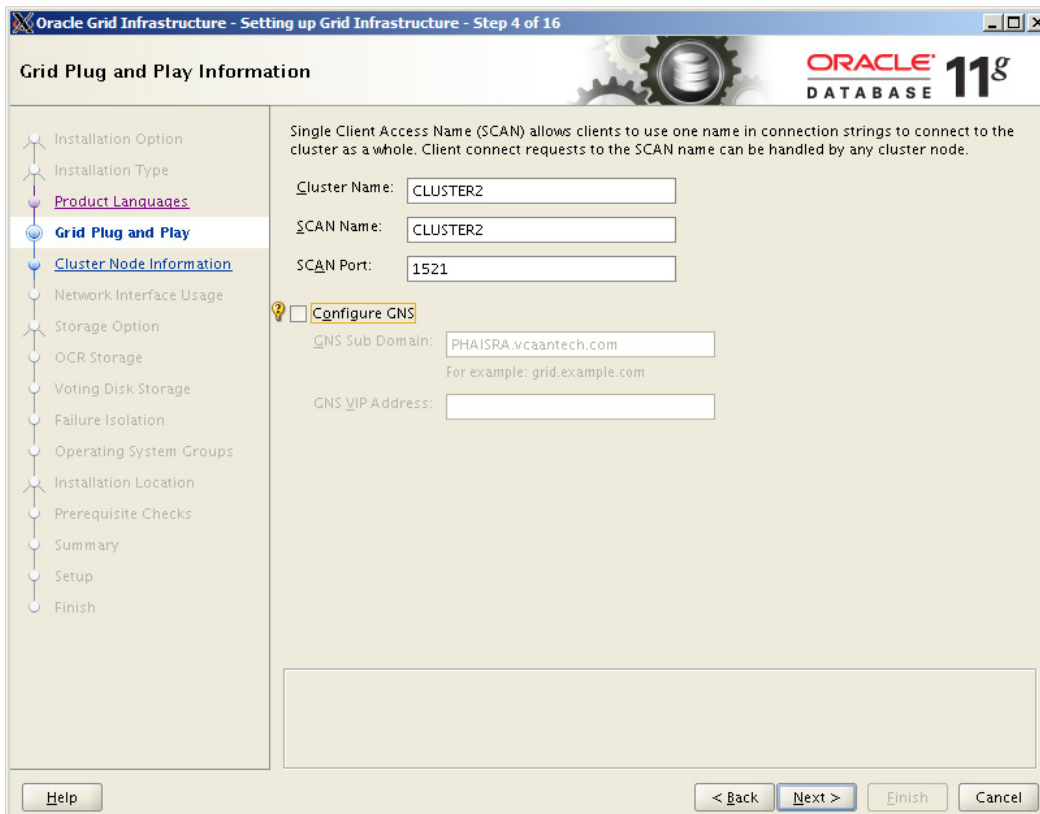
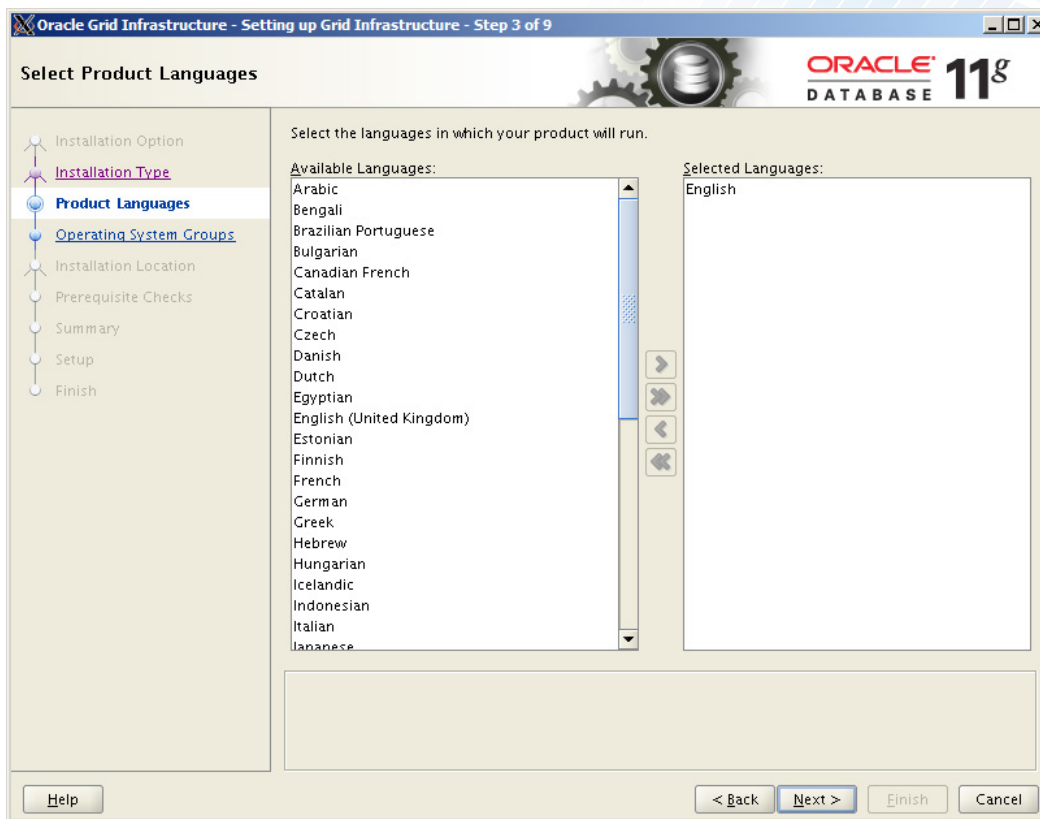
Install the Oracle Grid Software

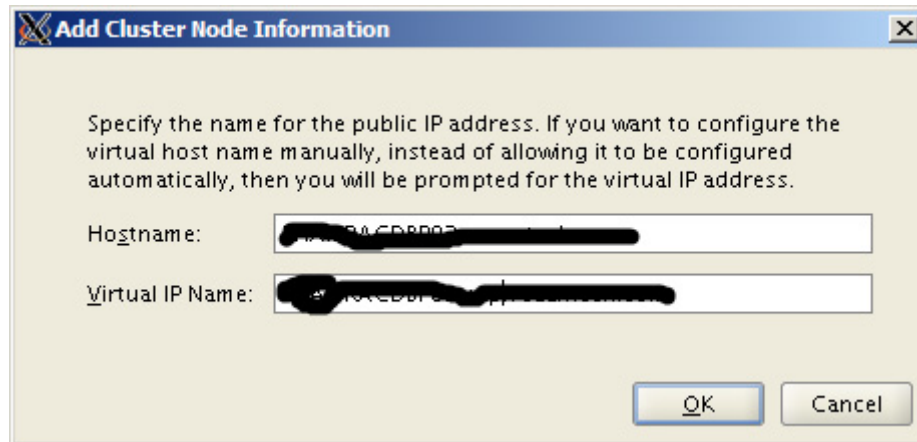
As the oracle user on node 1 (RAC1):

```
cd /home/local/oracle/software/grid
```

```
./runInstaller
```







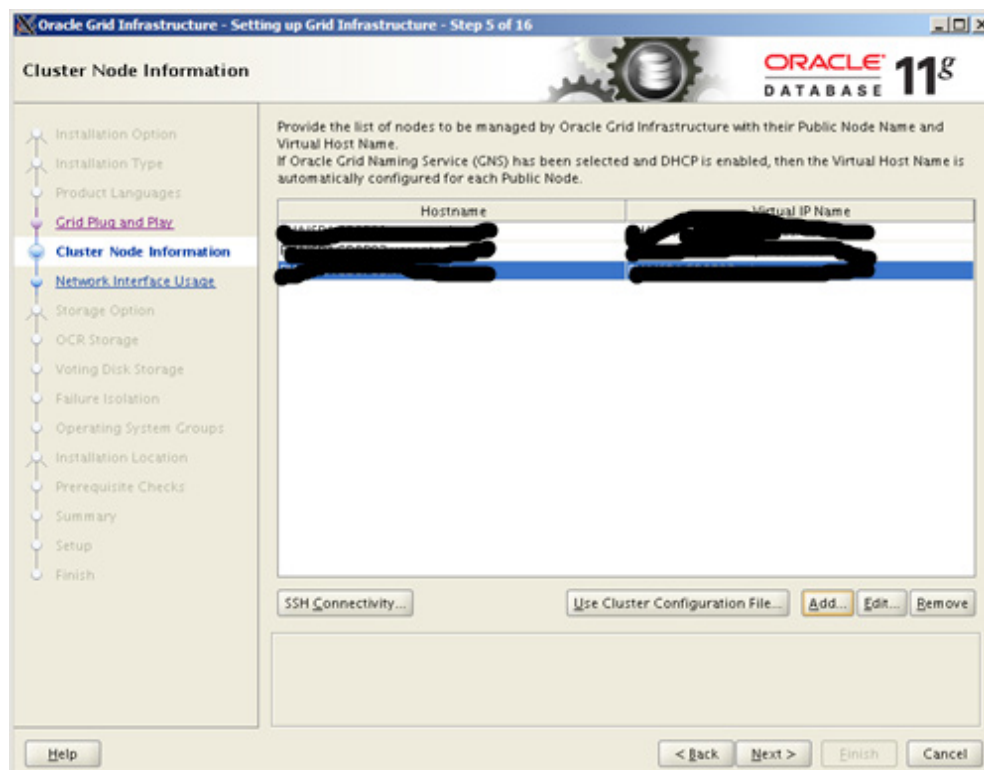
Add Cluster Node Information

Specify the name for the public IP address. If you want to configure the virtual host name manually, instead of allowing it to be configured automatically, then you will be prompted for the virtual IP address.

Hostname:

Virtual IP Name:

OK Cancel



Oracle Grid Infrastructure - Setting up Grid Infrastructure - Step 5 of 16

Cluster Node Information

Provide the list of nodes to be managed by Oracle Grid Infrastructure with their Public Node Name and Virtual Host Name.
If Oracle Grid Naming Service (GNS) has been selected and DHCP is enabled, then the Virtual Host Name is automatically configured for each Public Node.

Hostname	Virtual IP Name
racdb02	racdb02
racdb03	racdb03
racdb04	racdb04

SSH Connectivity... Use Cluster Configuration File... Add... Edit... Remove

Help < Back Next > Finish Cancel

Oracle Grid Infrastructure - Setting up Grid Infrastructure - Step 6 of 16

Specify Network Interface Usage

Identify the planned use for each global interface shown in the box below as Public, Private, or Do Not Use. Private interfaces are used by Oracle Grid Infrastructure for internode traffic.

If there is more than one subnet associated with an interface, then change the interface's attributes to associate the interface name with the additional subnets.

Interface Name	Subnet	Interface Type
bond0		Public
bond1		Private

Help < Back Next > Finish Cancel

Oracle Grid Infrastructure - Setting up Grid Infrastructure - Step 7 of 16

Storage Option Information

You can place Oracle Cluster Registry (OCR) disks and voting disks on ASM storage or on a file system.

☒ Automatic Storage Management (ASM)
Choose this option to configure OCR and voting disks on ASM storage.

☐ Shared File System
Choose this option to configure OCR and voting disks on an existing shared file system.

Help < Back Next > Finish Cancel

Oracle Grid Infrastructure - Setting up Grid Infrastructure - Step 8 of 15

Create ASM Disk Group

Select Disk Group Characteristics and select disks

Disk Group Name:

Redundancy: ☐ High ☒ Normal ☐ External

Add Disks

☒ Candidate Disks ☐ All Disks

	Disk Path	Size (in MB)	Status
<input checked="" type="checkbox"/>	ORCL:DATA1	768062	Candidate
<input checked="" type="checkbox"/>	ORCL:DATA2	768062	Candidate
<input checked="" type="checkbox"/>	ORCL:DATA3	768062	Candidate
<input checked="" type="checkbox"/>	ORCL:DATA4	768062	Candidate

[Change Discovery Path](#)

Help < Back Next > Finish Cancel

Oracle Grid Infrastructure - Setting up Grid Infrastructure - Step 9 of 15

Specify ASM Password

The new Automatic Storage Management (ASM) instance requires its own SYS user with SYSASM privileges for administration. Oracle recommends that you create a less privileged ASMSNMP user with SYSDBA privileges to monitor the ASM instance.

Specify the password for these user accounts.


☐ Use different passwords for these accounts

Password Confirm Password
 SYS
 ASMSNMP

☒ Use same passwords for these accounts

Specify Password: Confirm Password:

Messages:

 Specify Password:[INS-30011] The password entered does not conform to the Oracle recommended standards.

Help < Back Next > Finish Cancel

Oracle Grid Infrastructure - Setting up Grid Infrastructure - Step 10 of 16

Failure Isolation Support

Choose one of the following Failure Isolation Support options.

☐ Use Intelligent Platform Management Interface (IPMI)

To ensure successful installation with IPMI enabled, ensure your IPMI drivers are properly installed and enabled.

User Name :

Password :

☒ Do not use Intelligent Platform Management Interface (IPMI)

Help < Back Next > Finish Cancel

Oracle Grid Infrastructure - Setting up Grid Infrastructure - Step 11 of 16

Privileged Operating System Groups

Select the name of the operating system group of which you are a member to be used for OS authentication to Automatic Storage Management (ASM).

ASM Database Administrator (OSDBA) Group

ASM Instance Administration Operator (OSOPER) Group

ASM Instance Administrator (OSASM) Group

Help < Back Next > Finish Cancel

Oracle Grid Infrastructure - Setting up Grid Infrastructure - Step 12 of 16

Specify Installation Location

Specify a base location for storing all Oracle software and configuration-related files. This location is the Oracle base directory. Create one Oracle base for each operating system user. By default, software and configuration files are installed by version and database name parallel to the Oracle base directory.

Oracle Base:

Specify a base location for storing Oracle software files separate from database configuration files in the Oracle base directory. This software directory is the Oracle Grid Infrastructure home directory. Change the defaults below either to specify an alternative location, or to select an existing grid infrastructure home.

Software Location:

[Help](#)

Oracle Grid Infrastructure - Setting up Grid Infrastructure - Step 13 of 17

Create Inventory

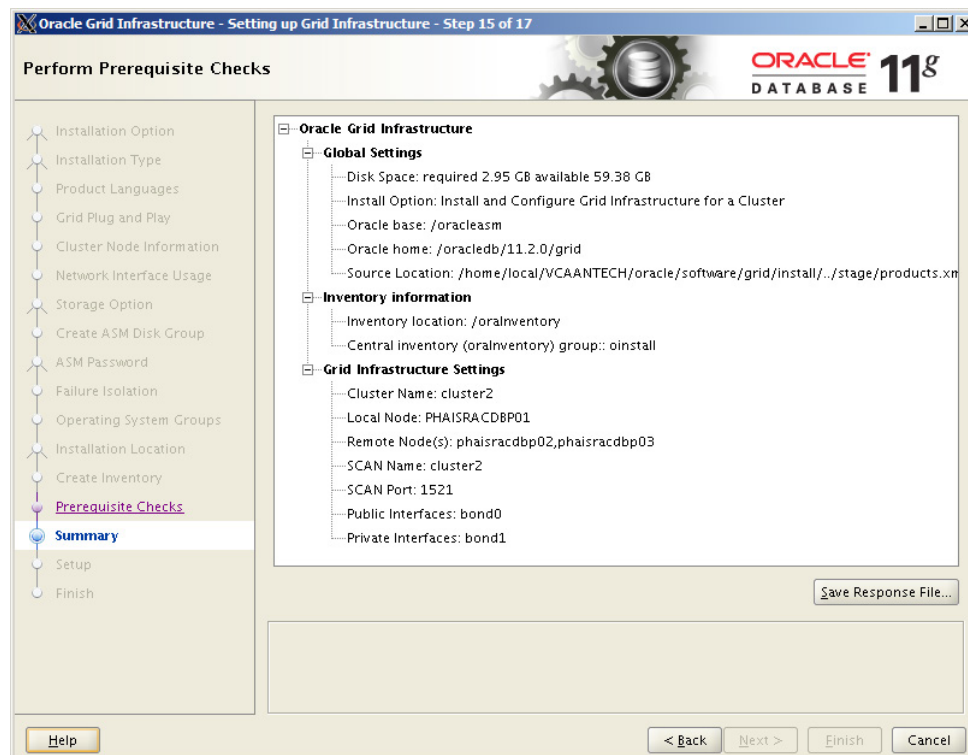
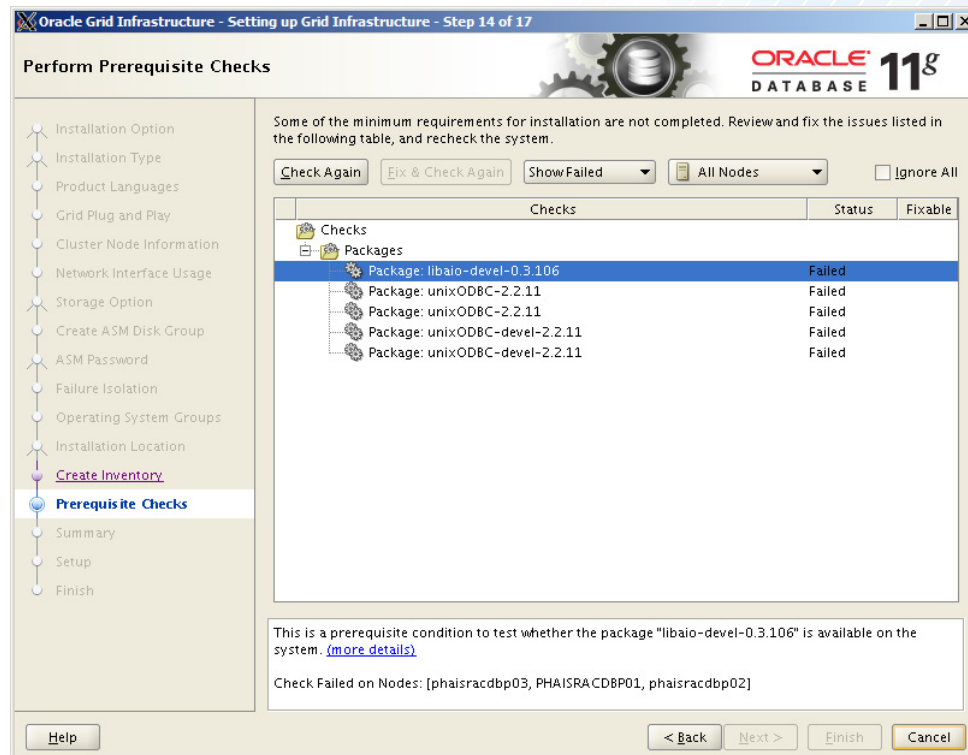
You are starting your first installation on this host. Specify a directory for installation files. This directory is called the "inventory directory". The installer automatically sets up subdirectories for each product to contain inventory data. The subdirectory for each product typically requires 150 kilobytes of disk space.

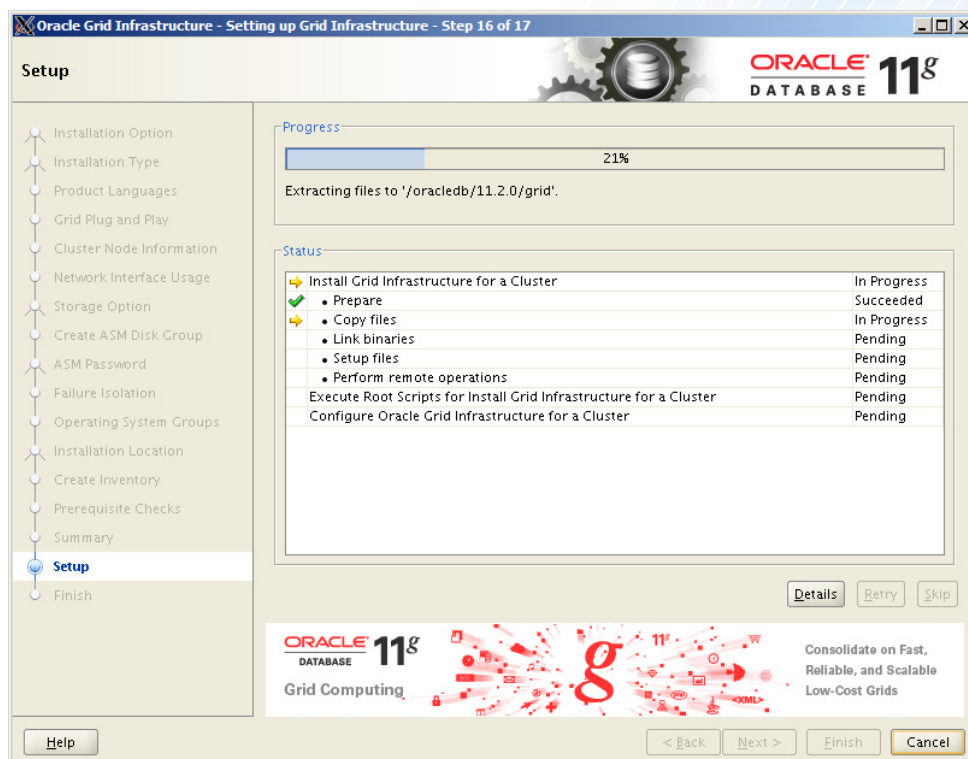
Inventory Directory:

Members of the following operating system group (the primary group) will have write permission to the inventory directory (oralnventory).

oralnventory Group Name: oinstall

[Help](#)





After installation completes, a configuration file called root.sh must be run on all nodes.

If root.sh fails on any node other than the first one, perform the following steps:

On all nodes,

1. Modify the /etc/sysconfig/oracleasm with:

```
ORACLEASM_SCANORDER="dm"
ORACLEASM_SCANEXCLUDE="sd"
```

2. Restart the asm lib (on all nodes except the 1st node):

```
# /etc/init.d/oracleasm restart
```

3. De-configure the root.sh settings on all nodes (except the 1st node):

```
$GRID_HOME/crs/install/rootcrs.pl -verbose -deconfig -force
```

4. Run root.sh again on all nodes except the first

Output of root.sh on node 1:

```
[root@RAC1 grid]# ./root.sh
Running Oracle 11g root.sh script...
```



The following environment variables are set as:

ORACLE_OWNER= oracle

ORACLE_HOME= /oracleasm/11.2.0/grid

Enter the full pathname of the local bin directory: [/usr/local/bin]:

The file "dbhome" already exists in /usr/local/bin. Overwrite it? (y/n)

[n]: y

Copying dbhome to /usr/local/bin ...

The file "oraenv" already exists in /usr/local/bin. Overwrite it? (y/n)

[n]: y

Copying oraenv to /usr/local/bin ...

The file "coraenv" already exists in /usr/local/bin. Overwrite it? (y/n)

[n]: y

Copying coraenv to /usr/local/bin ...

Entries will be added to the /etc/oratab file as needed by

Database Configuration Assistant when a database is created

Finished running generic part of root.sh script.

Now product-specific root actions will be performed.

2012-07-06 00:14:20: Parsing the host name

2012-07-06 00:14:20: Checking for super user privileges

2012-07-06 00:14:20: User has super user privileges

Using configuration parameter file: /oracleasm/11.2.0/grid/crs/install/crsconfig_params

Creating trace directory

LOCAL ADD MODE

Creating OCR keys for user 'root', privgrp 'root'..

Operation successful.

root wallet

root wallet cert

root cert export

peer wallet

profile reader wallet

pa wallet

peer wallet keys

pa wallet keys

peer cert request

pa cert request

peer cert

pa cert

peer root cert TP

profile reader root cert TP

pa root cert TP

peer pa cert TP

pa peer cert TP



```
profile reader pa cert TP
profile reader peer cert TP
peer user cert
pa user cert
```

Adding daemon to inittab

CRS-4123: Oracle High Availability Services has been started.

ohasd is starting

CRS-2672: Attempting to start 'ora.gipcd' on 'RAC1'

CRS-2672: Attempting to start 'ora.mdnsd' on 'RAC1'

CRS-2676: Start of 'ora.gipcd' on 'RAC1' succeeded

CRS-2676: Start of 'ora.mdnsd' on 'RAC1' succeeded

CRS-2672: Attempting to start 'ora.gpnpd' on 'RAC1'

CRS-2676: Start of 'ora.gpnpd' on 'RAC1' succeeded

CRS-2672: Attempting to start 'ora.cssdmonitor' on 'RAC1'

CRS-2676: Start of 'ora.cssdmonitor' on 'RAC1' succeeded

CRS-2672: Attempting to start 'ora.cssd' on 'RAC1'

CRS-2672: Attempting to start 'ora.diskmon' on 'RAC1'

CRS-2676: Start of 'ora.diskmon' on 'RAC1' succeeded

CRS-2676: Start of 'ora.cssd' on 'RAC1' succeeded

CRS-2672: Attempting to start 'ora.ctssd' on 'RAC1'

CRS-2676: Start of 'ora.ctssd' on 'RAC1' succeeded

ASM created and started successfully.

DiskGroup DATA created successfully.

clscfg: -install mode specified

Successfully accumulated necessary OCR keys.

Creating OCR keys for user 'root', privgrp 'root'..

Operation successful.

CRS-2672: Attempting to start 'ora.crsd' on 'RAC1'

CRS-2676: Start of 'ora.crsd' on 'RAC1' succeeded

CRS-4256: Updating the profile

Successful addition of voting disk 4baed8b3ca254f86bf91e6a19ef6aeeb.

Successful addition of voting disk 0e8a2bac79f84fdcbf1a5dcd73fa208e.

Successful addition of voting disk 401dae362bbb4f76bf3bddb8d047a429.

Successfully replaced voting disk group with +DATA.

CRS-4256: Updating the profile

CRS-4266: Voting file(s) successfully replaced

##	STATE	File Universal Id	File Name Disk group
----	-------	-------------------	----------------------

--	----	-----	-----
----	------	-------	-------

- | | | | |
|----|--------|----------------------------------|---------------------|
| 1. | ONLINE | 4baed8b3ca254f86bf91e6a19ef6aeeb | (ORCL:DATA1) [DATA] |
| 2. | ONLINE | 0e8a2bac79f84fdcbf1a5dcd73fa208e | (ORCL:DATA2) [DATA] |
| 3. | ONLINE | 401dae362bbb4f76bf3bddb8d047a429 | (ORCL:DATA3) [DATA] |

Located 3 voting disk(s).



CRS-2673: Attempting to stop 'ora.crsd' on 'RAC1'
CRS-2677: Stop of 'ora.crsd' on 'RAC1' succeeded
CRS-2673: Attempting to stop 'ora.asm' on 'RAC1'
CRS-2677: Stop of 'ora.asm' on 'RAC1' succeeded
CRS-2673: Attempting to stop 'ora.ctssd' on 'RAC1'
CRS-2677: Stop of 'ora.ctssd' on 'RAC1' succeeded
CRS-2673: Attempting to stop 'ora.cssdmonitor' on 'RAC1'
CRS-2677: Stop of 'ora.cssdmonitor' on 'RAC1' succeeded
CRS-2673: Attempting to stop 'ora.cssd' on 'RAC1'
CRS-2677: Stop of 'ora.cssd' on 'RAC1' succeeded
CRS-2673: Attempting to stop 'ora.gpnpd' on 'RAC1'
CRS-2677: Stop of 'ora.gpnpd' on 'RAC1' succeeded
CRS-2673: Attempting to stop 'ora.gipcd' on 'RAC1'
CRS-2677: Stop of 'ora.gipcd' on 'RAC1' succeeded
CRS-2673: Attempting to stop 'ora.mdnsd' on 'RAC1'
CRS-2677: Stop of 'ora.mdnsd' on 'RAC1' succeeded
CRS-2672: Attempting to start 'ora.mdnsd' on 'RAC1'
CRS-2676: Start of 'ora.mdnsd' on 'RAC1' succeeded
CRS-2672: Attempting to start 'ora.gipcd' on 'RAC1'
CRS-2676: Start of 'ora.gipcd' on 'RAC1' succeeded
CRS-2672: Attempting to start 'ora.gpnpd' on 'RAC1'
CRS-2676: Start of 'ora.gpnpd' on 'RAC1' succeeded
CRS-2672: Attempting to start 'ora.cssdmonitor' on 'RAC1'

CRS-2676: Start of 'ora.cssdmonitor' on 'RAC1' succeeded
CRS-2672: Attempting to start 'ora.cssd' on 'RAC1'
CRS-2672: Attempting to start 'ora.diskmon' on 'RAC1'
CRS-2676: Start of 'ora.diskmon' on 'RAC1' succeeded
CRS-2676: Start of 'ora.cssd' on 'RAC1' succeeded
CRS-2672: Attempting to start 'ora.ctssd' on 'RAC1'
CRS-2676: Start of 'ora.ctssd' on 'RAC1' succeeded
CRS-2672: Attempting to start 'ora.asm' on 'RAC1'
CRS-2676: Start of 'ora.asm' on 'RAC1' succeeded
CRS-2672: Attempting to start 'ora.crsd' on 'RAC1'
CRS-2676: Start of 'ora.crsd' on 'RAC1' succeeded
CRS-2672: Attempting to start 'ora.evmd' on 'RAC1'
CRS-2676: Start of 'ora.evmd' on 'RAC1' succeeded
CRS-2672: Attempting to start 'ora.asm' on 'RAC1'
CRS-2676: Start of 'ora.asm' on 'RAC1' succeeded
CRS-2672: Attempting to start 'ora.DATA.dg' on 'RAC1'
CRS-2676: Start of 'ora.DATA.dg' on 'RAC1' succeeded
Updating inventory properties for clusterware
Starting Oracle Universal Installer...

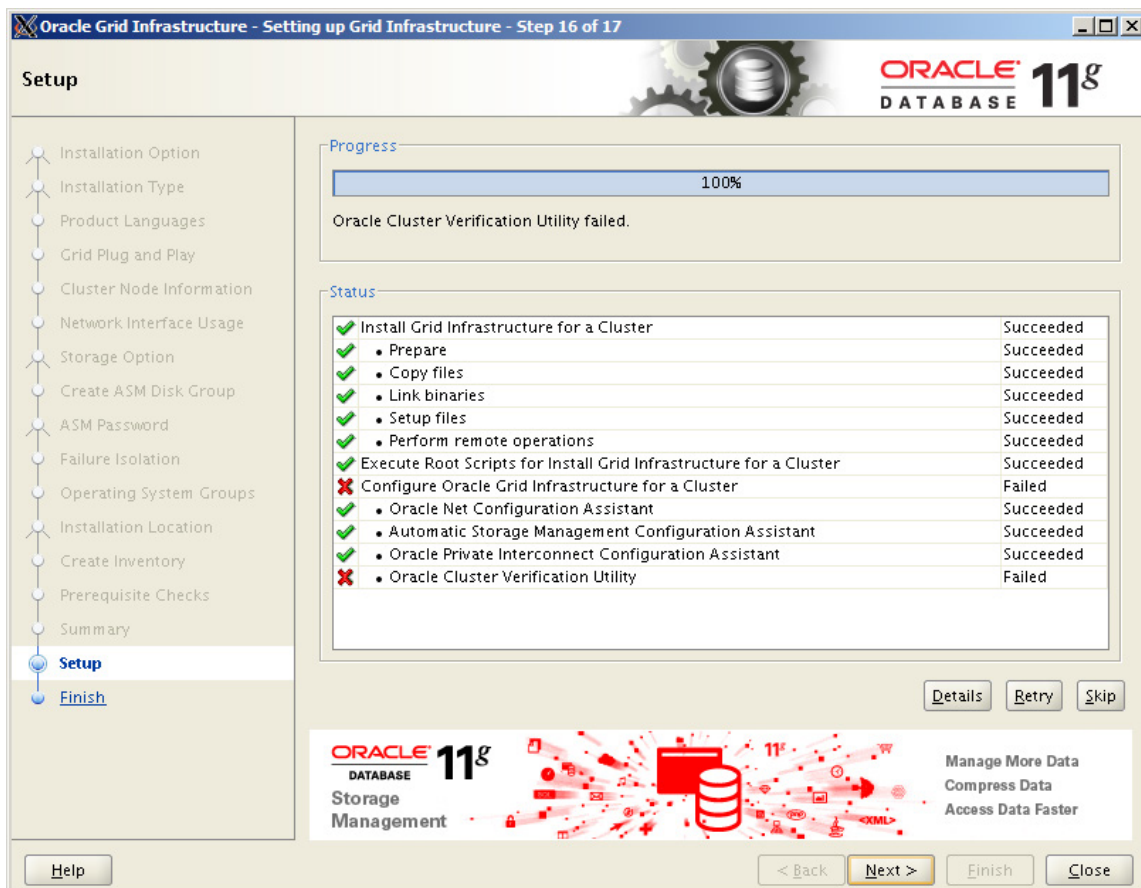
Checking swap space: must be greater than 500 MB. Actual 131071 MB Passed

The inventory pointer is located at /etc/orainst.loc

The inventory is located at /oraInventory

'UpdateNodeList' was successful.

If, **after** running the root.sh script on all nodes, you encounter an error in Grid installation program similar to:



and the error message tells you to search the /oraInventory/logs/installActions<date>.log file and you find an error similar to:

INFO: PRVF-4664 : Found inconsistent name resolution entries for SCAN name "CLUSTER2"

INFO: ERROR:

INFO: PRVF-4657 : Name resolution setup check for "CLUSTER2" (IP address: 10.230.100.82) failed

INFO: ERROR:

INFO: PRVF-4664 : Found inconsistent name resolution entries for SCAN name "CLUSTER2"

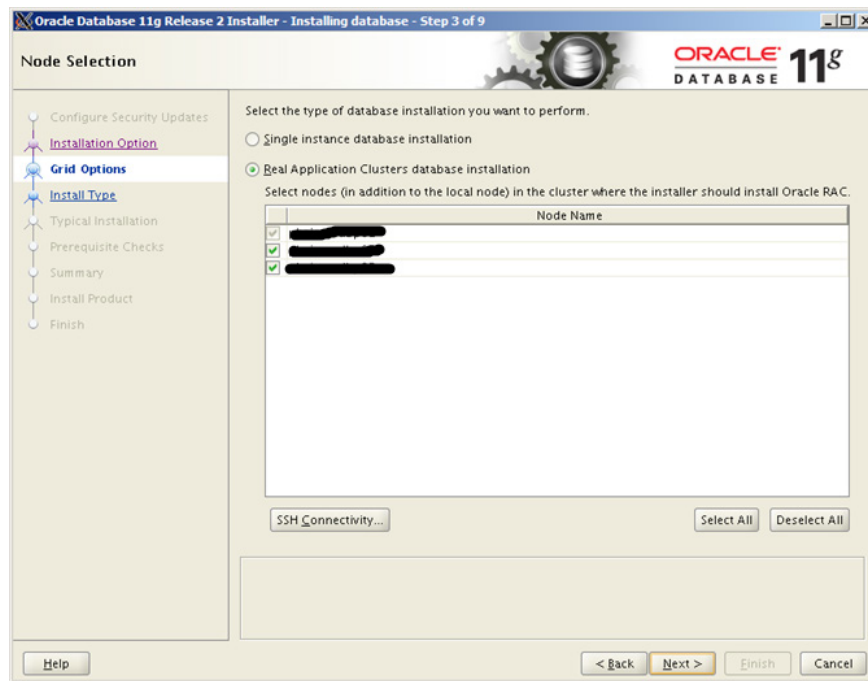
see:

<http://www.oracle-base.com/articles/11g/oracle-db-11gr2-rac-installation-on-ol5-using-vmware-server-2.php>

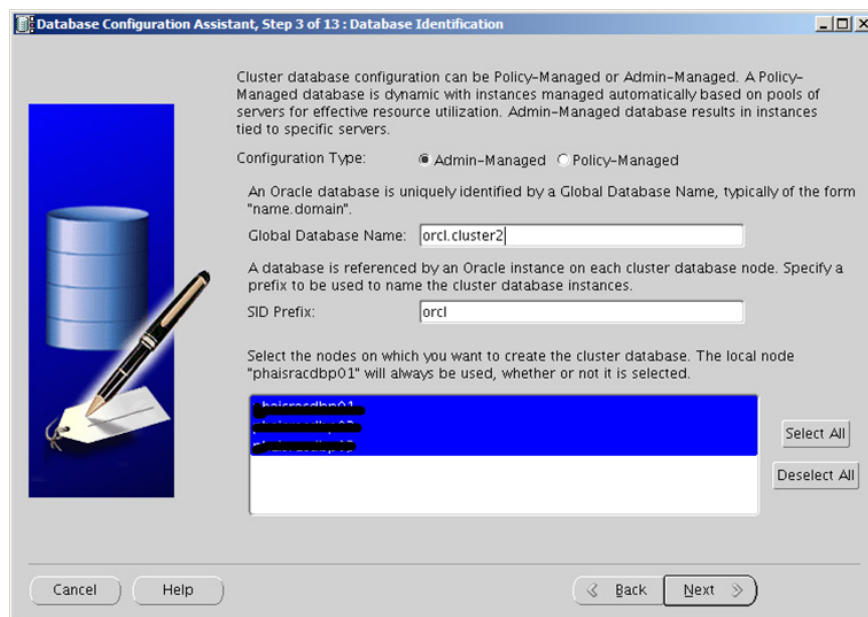
Install the Oracle Database

The installation of the Oracle database is the same as non-RAC instance, with a few exceptions. The screens that are unique to RAC are listed below:

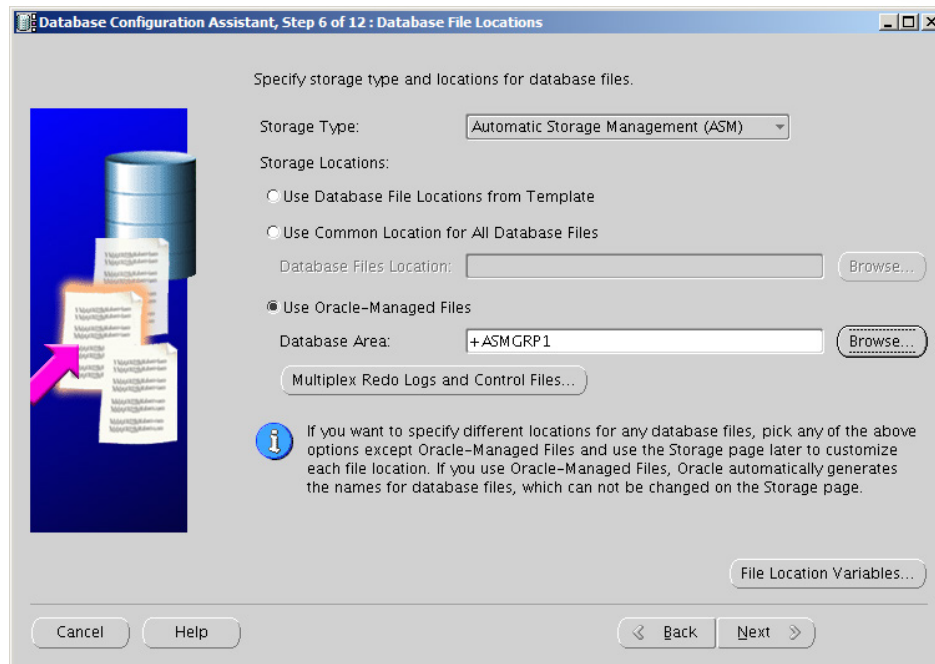
On this screen, you are prompted to enter the nodes in the cluster that the database software will be made aware of:



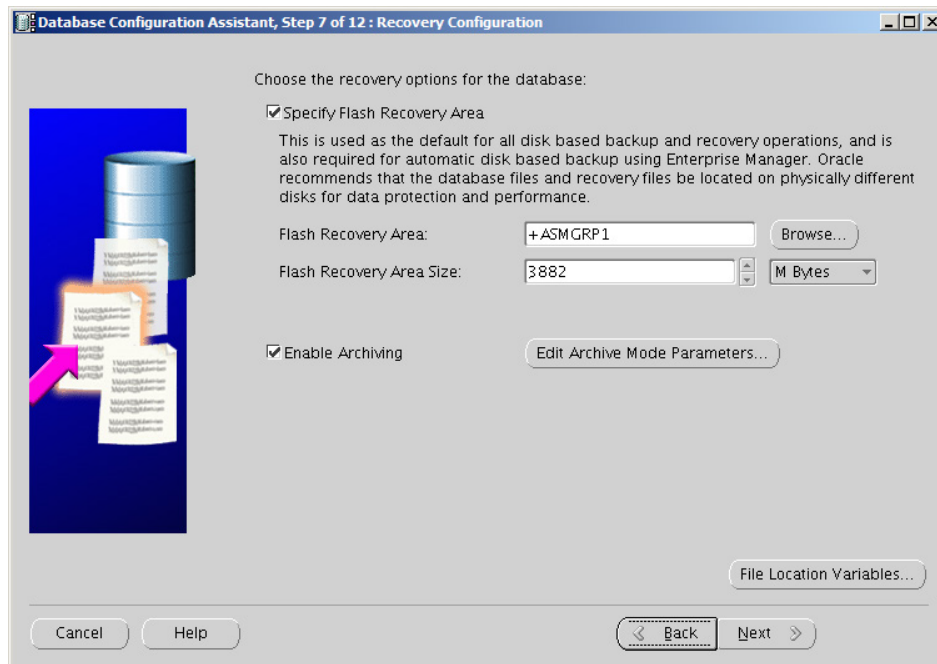
In the database configuration assistant, you can specify what nodes the database will run on:



On this screen of the wizard, ASM is chosen automatically, since RAC was specified. Note that the ASM file group must exist before running the database configuration wizard:



This screen prompts for the Flash Recovery Area. By default the ASM group is selected:





Conclusion

For no money (except for memory and disk requirements) you can build a fully-functional RAC system using Oracle's state-of-the-art software. The sandbox environment can be used to test Oracle software and learn the ins and outs of Oracle premier database product.

About the Author

Chris Ostrowski is the Oracle Solution Architect Director at Avout. He has worked with Oracle technologies for over 20 years as a Developer, DBA, Project Manager, and Enterprise Architect. He is a certified Oracle SOA Implementation Champion and an Oracle ACE, well-versed in assembling complex, end-to-end solutions spanning multiple competencies and platforms.

Recently, Chris has focused his efforts on service oriented architecture (SOA) technologies including Oracle JDeveloper and the Oracle SOA Suite, and enterprise technologies including Oracle Fusion Applications and Oracle's Application Integration Architecture. He is the author of three books from Oracle Press: Oracle Application Server 10g Web Development, Oracle Application Server Portal Handbook, and the soon-to-be-released Migrating to Fusion Applications. He has written articles for technology publications including Update, Select, Oracle Magazine, and the Oracle blog, and frequently presents at Oracle OpenWorld and various Oracle User Groups. Prior to joining Avout, Chris worked as a Director and Oracle Solution Architect for Fujitsu Consulting.

He holds a Bachelor of Arts from Rutgers University. In his spare time he is an avid hockey fan and plays the guitar.

If you have any questions about building a RAC database, please contact Chris.Ostrowski@Avout.com.

