Last Modified: 24-Jul-2011

Oracle RAC 11g Release 2 (11.2.0.1.4) Oracle VM templates Linux X86 64bit and X86 32bit (OneCommand/Oracle VM) Production Configuration

Table of Contents:

Overview:

This document introduces the Oracle Real Application Clusters 11g Release 2 (11.2.0.1.4) Oracle VM Templates. It explains the concepts of the templates and how to use the templates to build Oracle RAC clusters. Two separate templates are provided:

- Oracle Linux 5 Update 5 32bit Linux
- Oracle Linux 5 Update 5 64bit Linux

For the latest version of this document and more information on the benefits of Oracle RAC, Oracle VM and Templates, see OTN at: http://www.oracle.com/technetwork/server-storage/vm/rac-template-11grel2-166623.html and Note# 1185244.1 on My Oracle Support.

Supported vs. Unsupported implementations:

There are 2 separate scenarios for implementation of an Oracle RAC template

- **Production – Supported**: where the shared disk is configured as 'phy' devices passed through to the guest Oracle VMs. For more details on running Oracle RAC in Oracle VM environment, see: http://www.oracle.com/technetwork/database/clustering/oracle-rac-in-oracle-vm-environment-131948.pdf
- **Test – Unsupported:** where the shared disk is configured as 'file' devices – in this configuration both guests can also run on one Oracle VM server.

*This document covers the **Production (Supported)** configuration.* In production environments you may **not**:

a) Run both machines on the same Oracle VM Server (DOM-0)
b) Use files in DOM-0 to emulate shared disks for the Oracle RAC Node Dom-U

Shared disks must be configured as 'phy' devices.

Naming Convention:

This document describes both **x86** (32bit) and **x86-64** (64bit) environments. Template file & directory names are as follows:

32 bit: OVM_EL5U5_**X86**_11201RAC_PVM
64 bit: OVM_EL5U5_**X86_64**_11201RAC_PVM

Screenshots may refer to templates as: OVM_EL5U4_X86_**11GR2**RAC_PVM, these are generic screenshots, substitute your version number, i.e. 11201 or 11202 based on release used.

Feedback:

Feel free to post feedback at the Oracle VM or Oracle RAC Forums on the Oracle Technology Network:

Oracle VM: http://forums.oracle.com/forums/forum.jspa?forumID=823

Oracle RAC: http://forums.oracle.com/forums/forum.jspa?forumID=62

Or contact Oracle Support.

NOTE: The Oracle RAC build scripts inside the template (under /u01/racovm directory) are only supported inside the Oracle VM RAC Templates. Do not modify or run them in other environments.

Minimum Hardware Requirements:

When running Oracle RAC in a production environment, as described in this document, none of the Oracle RAC VM guest machines can run on the same Oracle VM Server. The physical resources required are as follows:

Minimum Storage Requirements

Filesystems that support SPARSE FILES[1] reduce the required storage footprint, however due to various filesystem settings these numbers might be different (below values are from ext3)

	No Sparse file support	Sparse file support (approx)	
Under /OVS:	**32bit or 64bit**	**32bit**	**64bit**
seed_pool stored template	29GB	9.5GB	11GB
running pool image for RAC node1	29GB	9.5GB	11GB
running pool image for RAC node2	29GB	9.5GB	11GB
Total under /OVS volume (DOM-0)	**87GB**	**28.5GB**	**33GB**
Any other filesystem:			
Temp space to unpack template[2]	**4.2 + 4.2GB**	**4.2 + 4.2GB**	

Additional space on shared disks to hold database, minimum **5GB.**

Minimum CPU Requirements

Each VM requires a minimum of 1 physical core, recommended to run with 2 or more physical cores for the hosting server.

Minimum Memory Requirements

Each VM requires a minimum of 2GB or more of physical RAM.

Minimum Network Requirements

Each Guest requires 2 network devices, which are 'virtualized' from physical cards in the hosting server, therefore a minimum of 2 physical NICS for the hosting server.

A minimum of 5 public IP addresses and 2 private IP addresses. If a DNS Server is available a further 2 IP addresses should be used to provide additional IPs for the SCAN name.

[1] OCFS2 v1.4.4 bundled with Oracle VM Server 2.2 has sparse file support. Oracle VM Agent has sparse file support starting with version 2.3-27 (might need to set disk.sparse.ratio=0 in /etc/ovs-agent/agent.ini). Ext3 filesystem has sparse file support. NFS depends if server filesystem supports sparse files or not. Space consumed by sparse files varries by filesystems's minimum allocation unit often referred to as *cluster size*.

[2] Template stored in a TGZ file inside a ZIP file hence size is listed twice

WHAT'S NEW:

- Applied all current recommended patches on top of 11.2.0.1.4
- Upgraded OS to Oracle Linux 5 Update 5
- Increased OS disk size

Templates Description:

These templates allow building clusters of any number of nodes with an Oracle RAC 11g Release 2 (11.2.0.1.4) configuration, which includes Oracle Clusterware and ASM (11.2.0.1.2), Oracle Database 11g Release 2 (11.2.0.1.4) patched to the latest recommended patches, including Grid Infrastructure Bundle Patch #2 and Database Patchset Update #4 & Critical Patch Update JAN2011 (CPUJAN2011).

Patches applied to both templates: 3934729 8328200 8448079 8730312 8898852 9002336
9233285 9318214 9362218 9594372 9706490[3] 9711859
9893913 10248516 10253888 11064298 11064457
(EBS) 8771297 8772028 8993052 9218789 9081430 9500046
9535951 9644960 9719541
Specific to 32bit template: 8670579 9377144
Specific to 64bit template: 8677763 8880196 8943287 9181300 9024611 9411739 9577583

Full opatch lsinventory output is provided in the zip file so there is no need to boot the VM or build a cluster to see the full patch listings for all homes.

The entire install is automated, with the user only providing the required input parameters (node names, IP addresses, etc.) when guests are booted. Fully automated builds directly from Dom0 are also possible, including automated network setup to all VMs.

- The root password is 'ovsroot', oracle password is 'oracle' and sys/system password is 'oracle', all can be changed during the build.
- The OS image contains a minimal install of Oracle Linux 5. Only basic RPM packages are installed.
- The install is performed using root access. With ssh passwordless login temporarily enabled for root user. At the completion of the install (on both nodes), ssh for root will be disabled.
- Optionally, the entire install could be done as the Oracle user, provided that 'sudo' access is configured on the first node only.
- During the installation process an Oracle RAC database instance is created on both nodes. It is possible to skip this database creation and do software only install.
- The OS and Oracle image disks inside the template are sparse files (non allocated space is not counted against filesystem), meaning the size displayed via 'ls -l' is not the actual space consumed on-disk. To see actual space consumed use 'ls -ls'. If /OVS filesystem has no sparse-file support then both sizes will be the same (all space is allocated upfront).
- Each template (32bits or 64bits) is delivered as 2 ZIP files (~4.2GB in size total), inside each ZIP is a TGZ (Tar gzip file), with very similar sizes, hence you will need ~8.4GB of temporary storage to fully expand each template. Once the template is imported into /OVS/seed_pool, these zip/tgz files can be deleted.

[3] This is the re-issued patch 9706490 from April 2011.

Main steps, using Oracle VM Manager

These steps are using Oracle VM Manager, however all of them could be done from the command line, using Oracle VM Command Line Interface (CLI).

1) Load the template into the /OVS/seed_pool directory, on DOM-0:

```
# cd /tmp
# unzip -q OVM_EL5U5_X86_11201RAC_PVM-1of2.zip &   # or e-delivery equivalent part number
# unzip -q OVM_EL5U5_X86_11201RAC_PVM-2of2.zip &   # or e-delivery equivalent part number
# cd /OVS/seed_pool
# wait
# tar xzf /tmp/OVM_EL5U5_X86_11201RAC_PVM-1of2.tgz &    # or X86_64 for the 64-bit version
# tar xzf /tmp/OVM_EL5U5_X86_11201RAC_PVM-2of2.tgz &    # or X86_64 for the 64-bit version
# wait
```

(When the above commands complete, the ZIP & TGZ files may be deleted from /tmp)

This will create the following, e.g. on 32bit:

```
/OVS/seed_pool/OVM_EL5U5_X86_11201RAC_PVM          (or X86_64 for the 64-bit version)
|- System.img (OS image file)
|- Oracle11201RAC_x86-xvdb.img (database software image file)
|- vm.cfg (VM configuration file)
|- README.txt
|- These PDF documents
|- lsinventory-sample.grid-32bit       lsinventory-sample.racdb-32bit   (or 64bit)
|- utils  (files to help with automated network setup and builds)
```

2) Create 5 shared virtual disks to hold the Database

Shared disks must be configured on DOM-0 of both OVM Servers (create at least one partition on DOM-0; do not use whole disk, e.g. /dev/sdd), these should be persistently named physical devices (multi-pathed or not) accessible from both nodes, e.g.:

```
lrwxrwxrwx 1 root root 4 Dec 28 01:28 /dev/racdevc1 -> sdc1
lrwxrwxrwx 1 root root 4 Dec 28 01:28 /dev/racdevd2 -> sdd2
lrwxrwxrwx 1 root root 4 Dec 28 01:28 /dev/racdeve2 -> sde2
lrwxrwxrwx 1 root root 4 Dec 28 01:28 /dev/racdevf4 -> sdf4
lrwxrwxrwx 1 root root 4 Dec 28 01:28 /dev/racdevg1 -> sdg1
```

The following symlinked devices were created with udev rules similar to this one below, on Dom0, in /etc/udev/rules.d/55-oracle-devices.rules file:

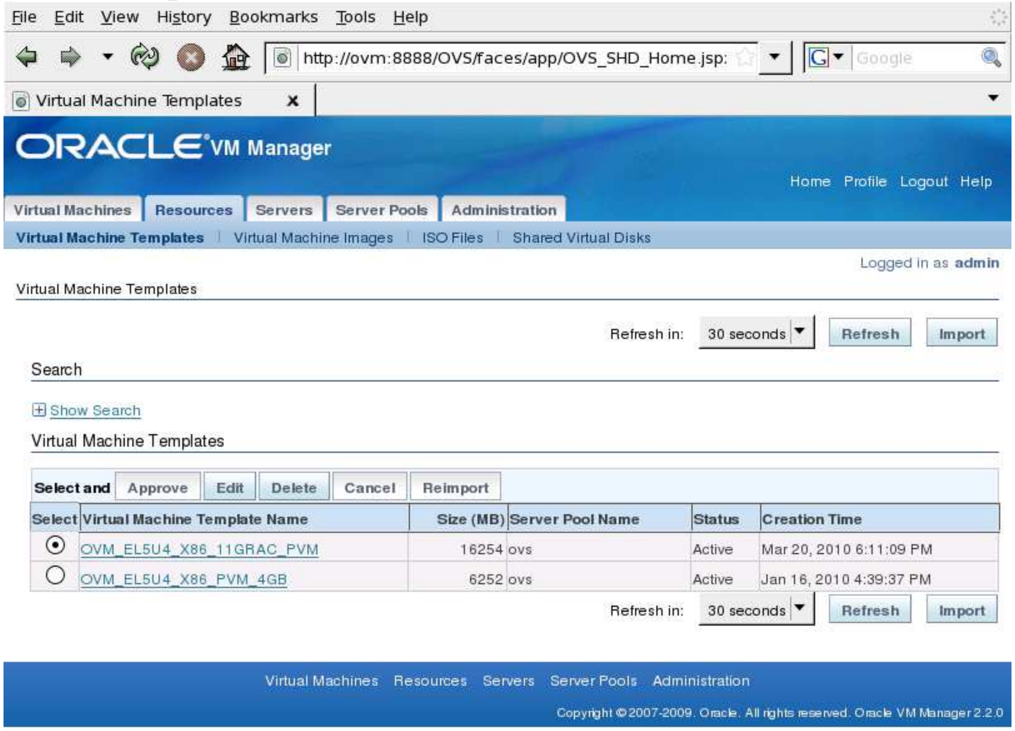
```
KERNEL=="sd*", BUS=="scsi", PROGRAM=="/sbin/scsi_id -g -u -s %p",
RESULT=="360a98000686f61506434386f65663577", SYMLINK+="racdevc%n", OWNER="root",
GROUP="root", MODE="0640"
```

Using SYMLINK+= (instead of NAME= in udev rules) keeps the original kernel device names, e.g. /dev/sd*. Using udev rules ensures persistent device naming across reboots. The OneCommand RAC install requires 5 such devices, minimum size of 1024MB each (See Appendix C). These will hold your database, size them appropriately.

3) Import the template (from step 1) using Oracle VM manager

Make sure all unzipping/untarring operations completed before proceeding with these steps.

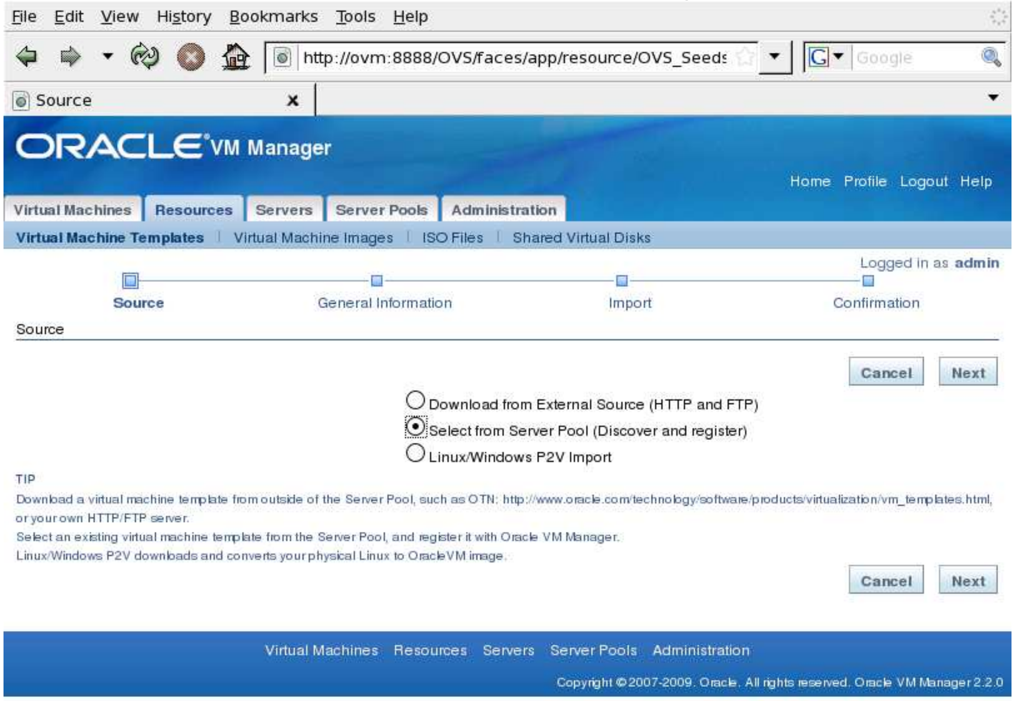
- Click Resources -> Virtual Machine Templates
- Click **Import** button


File Edit View History Bookmarks Tools Help
http://ovm:8888/OVS/faces/app/OVS_SHD_Home.jsp:
Google
Virtual Machine Templates
ORACLE VM Manager
Home Profile Logout Help
Virtual Machines Resources Servers Server Pools Administration
Virtual Machine Templates | Virtual Machine Images | ISO Files | Shared Virtual Disks
Logged in as admin
Virtual Machine Templates
Refresh in: 30 seconds Refresh Import
Search
Show Search
Virtual Machine Templates
Select and Approve Edit Delete Cancel Reimport
Refresh in: 30 seconds Refresh Import
Virtual Machines Resources Servers Server Pools Administration
Copyright © 2007-2009. Oracle. All rights reserved. Oracle VM Manager 2.2.0

Select	Virtual Machine Template Name	Size (MB)	Server Pool Name	Status	Creation Time
●	OVM_EL5U4_X86_11GRAC_PVM	16254	ovs	Active	Mar 20, 2010 6:11:09 PM
○	OVM_EL5U4_X86_PVM_4GB	6252	ovs	Active	Jan 16, 2010 4:39:37 PM

This Oracle VM manager already knows about 2 existing templates. We are going to add a 3rd template.

- Select "Select from Server Pool (Discover and register)", click **Next**

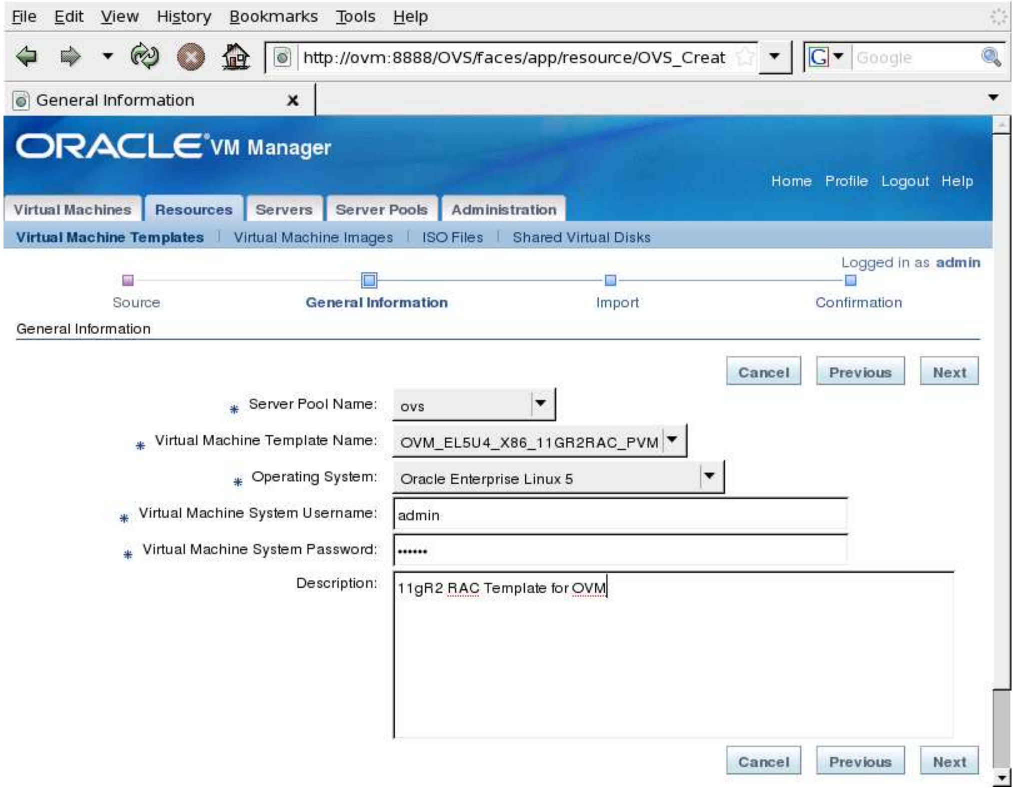

File Edit View History Bookmarks Tools Help
http://ovm:8888/OVS/faces/app/resource/OVS_Seeds
Google
Source
ORACLE VM Manager
Home Profile Logout Help
Virtual Machines Resources Servers Server Pools Administration
Virtual Machine Templates | Virtual Machine Images | ISO Files | Shared Virtual Disks
Logged in as admin
Source
General Information
Import
Confirmation
Source
Cancel Next
Cancel Next
Virtual Machines Resources Servers Server Pools Administration
Copyright © 2007-2009. Oracle. All rights reserved. Oracle VM Manager 2.2.0

○ Download from External Source (HTTP and FTP)
● Select from Server Pool (Discover and register)
○ Linux/Windows P2V Import

TIP
Download a virtual machine template from outside of the Server Pool, such as OTN: http://www.oracle.com/technology/software/products/virtualization/vm_templates.html, or your own HTTP/FTP server.
Select an existing virtual machine template from the Server Pool, and register it with Oracle VM Manager.
Linux/Windows P2V downloads and converts your physical Linux to OracleVM image.

- Pull-down the correct "Server Pool Name"

- Choose the newly untarred template from pulldown “Virtual Machine Template Name” titled OVM_EL5U5_**X86**_11201RAC_PVM or OVM_EL5U5_**X86_64**_11201RAC_PVM.
- Select “Oracle Enterprise Linux 5” or “Oracle Enterprise Linux 5 64-bit” for “Operating System”.
- Optionally enter Description or change username/passwords
- Click **Next** and **Confirm** on the following page.



* Server Pool Name: ovs

* Virtual Machine Template Name: OVM_EL5U4_X86_11GR2RAC_PVM

* Operating System: Oracle Enterprise Linux 5

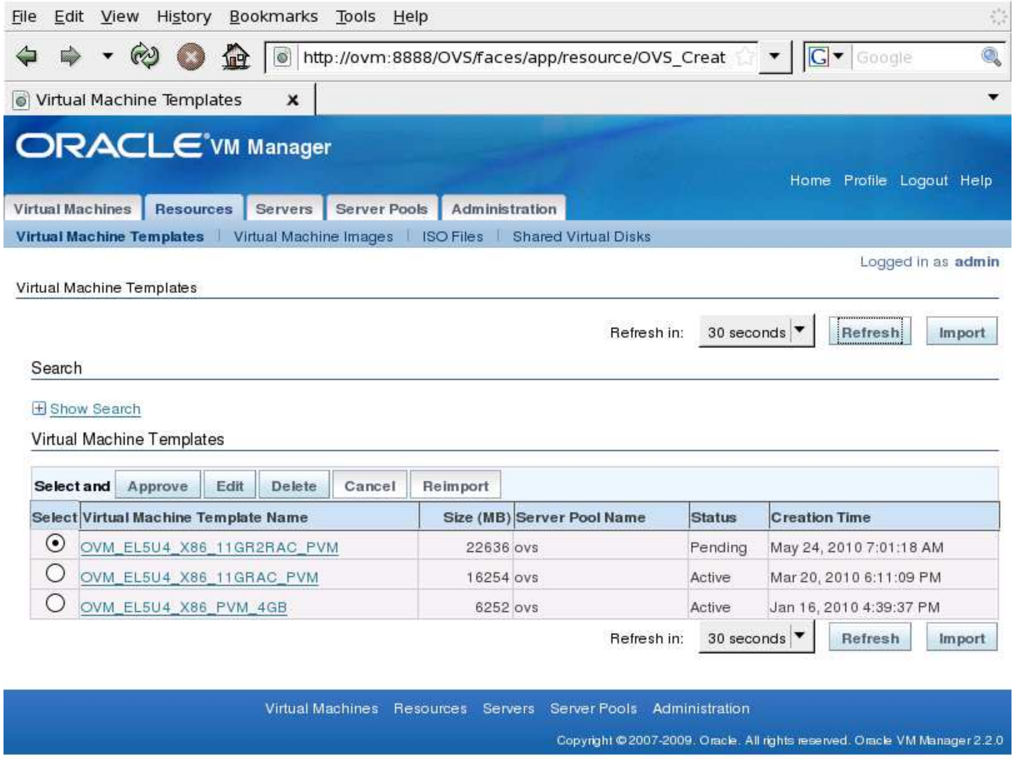
* Virtual Machine System Username: admin

* Virtual Machine System Password: ••••••

Description: 11gR2 RAC Template for OVM

- Template name might be different in screenshot based on release used

- Wait for VM template to change from "Importing" to "Pending" then proceed
- Select Pending VM template from radio box, and Click **Approve** button

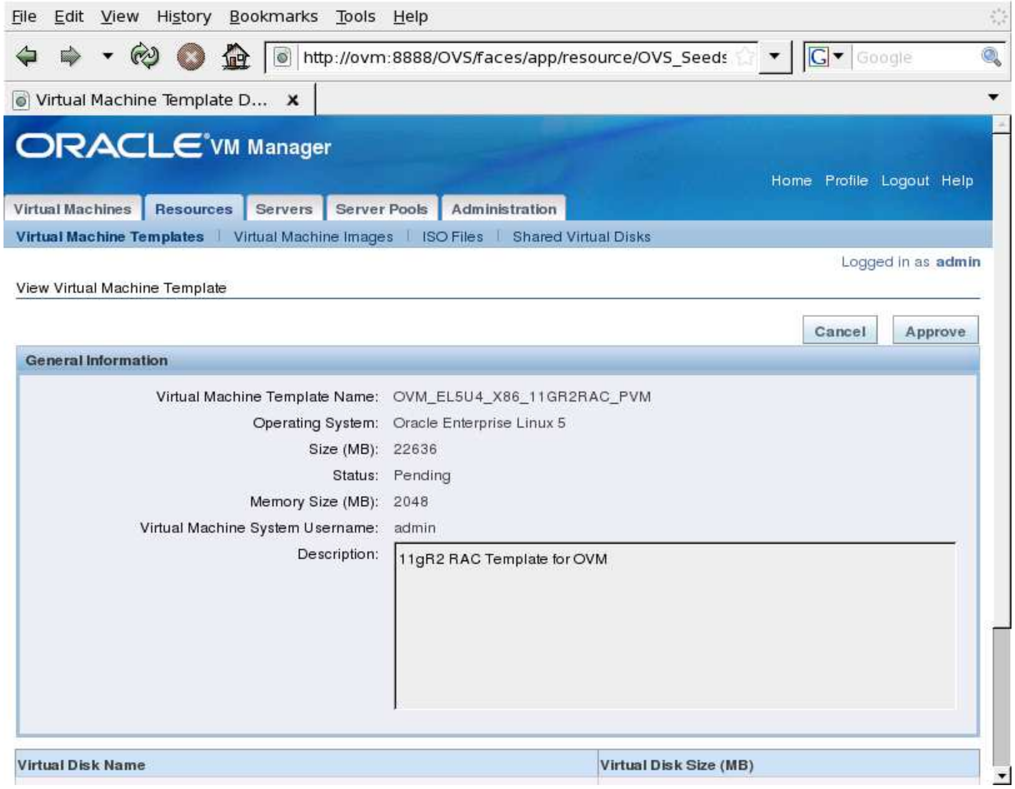

File Edit View History Bookmarks Tools Help
http://ovm:8888/OVS/faces/app/resource/OVS_Creat
Google
Virtual Machine Templates
ORACLE VM Manager
Home Profile Logout Help
Virtual Machines Resources Servers Server Pools Administration
Virtual Machine Templates | Virtual Machine Images | ISO Files | Shared Virtual Disks
Logged in as admin
Virtual Machine Templates
Refresh in: 30 seconds
Refresh
Import
Search
Show Search
Select and Approve Edit Delete Cancel Reimport
Virtual Machines Resources Servers Server Pools Administration
Copyright © 2007-2009. Oracle. All rights reserved. Oracle VM Manager 2.2.0

Virtual Machine Templates

Select	Virtual Machine Template Name	Size (MB)	Server Pool Name	Status	Creation Time
●	OVM_EL5U4_X86_11GR2RAC_PVM	22636	ovs	Pending	May 24, 2010 7:01:18 AM
○	OVM_EL5U4_X86_11GRAC_PVM	16254	ovs	Active	Mar 20, 2010 6:11:09 PM
○	OVM_EL5U4_X86_PVM_4GB	6252	ovs	Active	Jan 16, 2010 4:39:37 PM

  - Template name might be different in screenshot based on release used

- Click **Approve**


File Edit View History Bookmarks Tools Help
http://ovm:8888/OVS/faces/app/resource/OVS_Seeds
Google
Virtual Machine Template D...
ORACLE VM Manager
Home Profile Logout Help
Virtual Machines Resources Servers Server Pools Administration
Virtual Machine Templates | Virtual Machine Images | ISO Files | Shared Virtual Disks
Logged in as admin
View Virtual Machine Template
Cancel
Approve

General Information

Virtual Machine Template Name: OVM_EL5U4_X86_11GR2RAC_PVM
Operating System: Oracle Enterprise Linux 5
Size (MB): 22636
Status: Pending
Memory Size (MB): 2048
Virtual Machine System Username: admin
Description: 11gR2 RAC Template for OVM

Virtual Disk Name	Virtual Disk Size (MB)

  - Template name might be different in screenshot based on release used

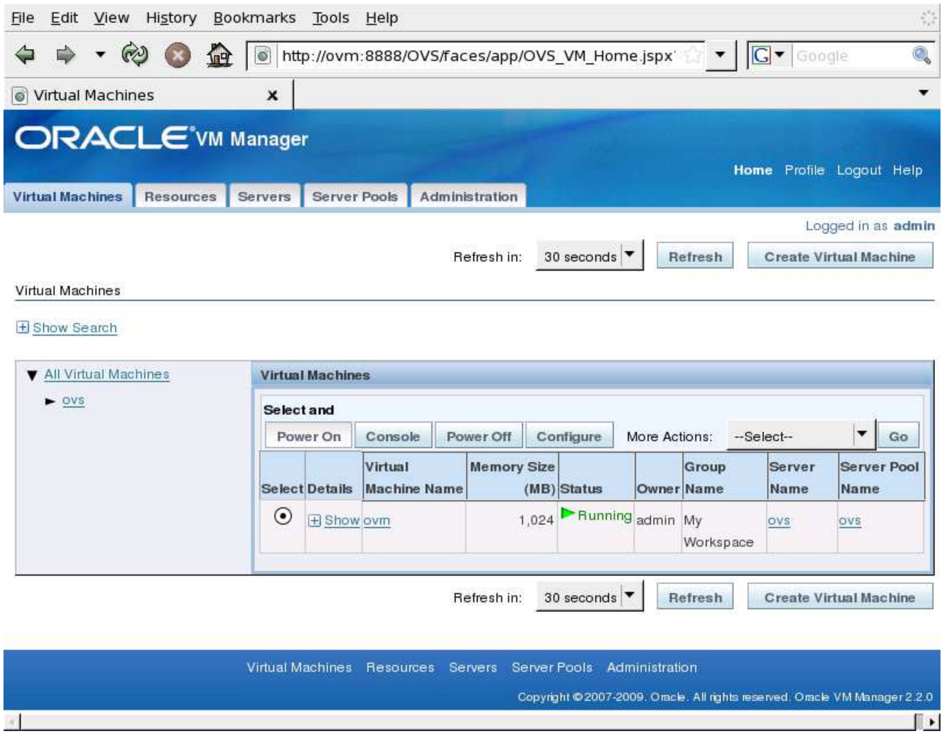
The VM Template is now imported and ready to be deployed as many times as needed.

4) Create 2 Virtual Machines using the imported template

In our example we name the VMs racnode1 & racnode2, you may choose any name you wish.

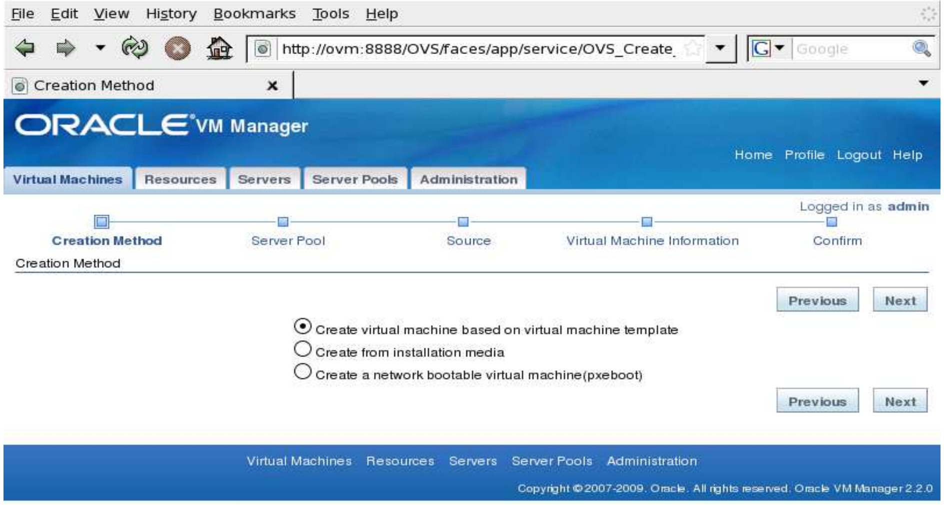
IMPORTANT – The only non-default item you will need is to change the second NIC to xenbr1 (the first NIIC on xenbr0 will automatically be configured, see following page)

- Select Virtual machines
- Click on the **Create Virtual Machine** button


File Edit View History Bookmarks Tools Help
http://ovm:8888/OVS/faces/app/OVS_VM_Home.jspx
Google
Virtual Machines
ORACLE VM Manager
Home Profile Logout Help
Virtual Machines Resources Servers Server Pools Administration
Logged in as admin
Refresh in: 30 seconds
Refresh
Create Virtual Machine
Virtual Machines
Show Search
All Virtual Machines
ovs
Virtual Machines
Select and
Power On Console Power Off Configure
More Actions: --Select-- Go
Virtual Machines Resources Servers Server Pools Administration
Copyright © 2007-2009. Oracle. All rights reserved. Oracle VM Manager 2.2.0

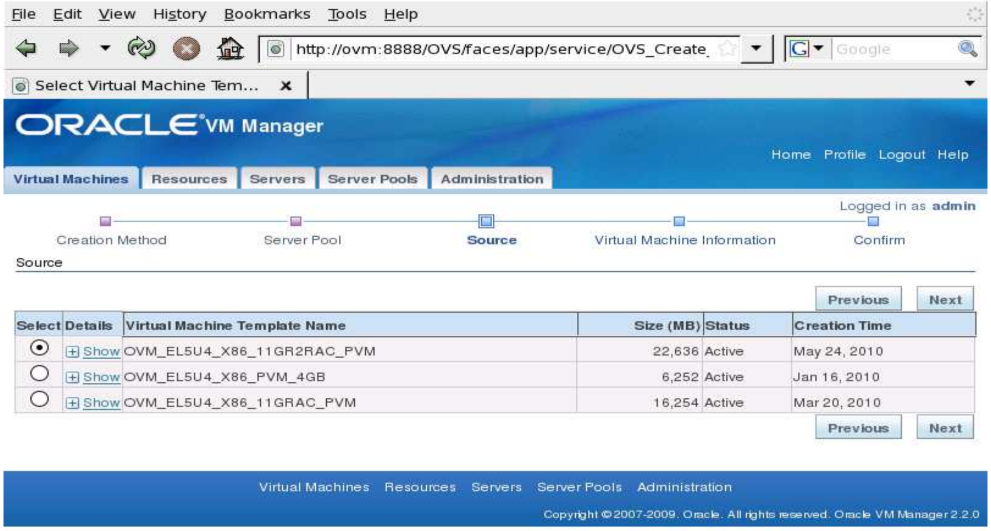
Select	Details	Virtual Machine Name	Memory Size (MB)	Status	Owner	Group Name	Server Name	Server Pool Name
●	Show	ovm	1,024	Running	admin	My Workspace	ovs	ovs

- Select the "Create virtual machine based on… template" radio button
- Click **Next**, and select the correct Server Pool on the following page.


File Edit View History Bookmarks Tools Help
http://ovm:8888/OVS/faces/app/service/OVS_Create_
Google
Creation Method
ORACLE VM Manager
Home Profile Logout Help
Virtual Machines Resources Servers Server Pools Administration
Logged in as admin
Creation Method
Server Pool
Source
Virtual Machine Information
Confirm
Creation Method
Previous
Next
Virtual Machines Resources Servers Server Pools Administration
Copyright © 2007-2009. Oracle. All rights reserved. Oracle VM Manager 2.2.0

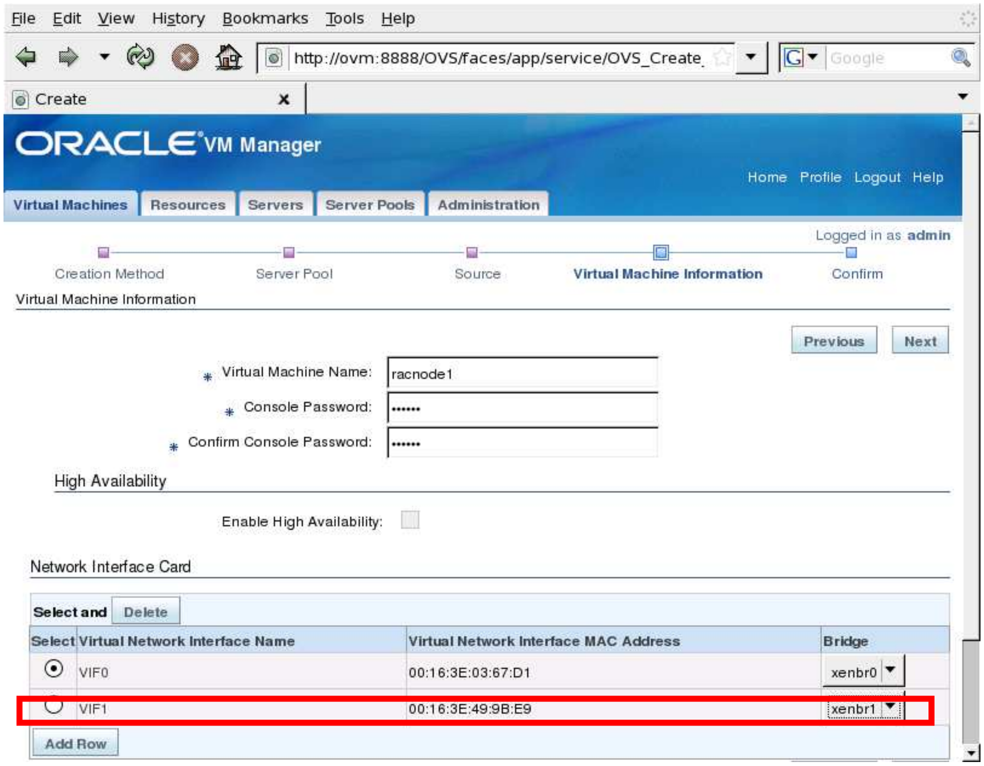
● Create virtual machine based on virtual machine template
○ Create from installation media
○ Create a network bootable virtual machine(pxeboot)

- Select the Oracle RAC 11g Release 2 (11.2.0.1) template (32bit or 64bit)
- Click **Next**
- Note: Template name might be different in screenshot based on release used


File Edit View History Bookmarks Tools Help
http://ovm:8888/OVS/faces/app/service/OVS_Create
Google
Select Virtual Machine Tem...
ORACLE VM Manager
Home Profile Logout Help
Virtual Machines Resources Servers Server Pools Administration
Logged in as admin
Creation Method
Server Pool
Source
Virtual Machine Information
Confirm
Source
Previous Next
Virtual Machines Resources Servers Server Pools Administration
Copyright © 2007-2009. Oracle. All rights reserved. Oracle VM Manager 2.2.0

Select	Details	Virtual Machine Template Name	Size (MB)	Status	Creation Time
●	⊞ Show	OVM_EL5U4_X86_11GR2RAC_PVM	22,636	Active	May 24, 2010
○	⊞ Show	OVM_EL5U4_X86_PVM_4GB	6,252	Active	Jan 16, 2010
○	⊞ Show	OVM_EL5U4_X86_11GRAC_PVM	16,254	Active	Mar 20, 2010

- **IMPORTANT:** Assign the second network adapter to **xenbr1**
- Enter the Virtual Machine Name: racnode1 (or racnode2 for second node)
- Enter the password twice – this will be the VNC password for the console
- Click **Next** and **Confirm** on the following page.


File Edit View History Bookmarks Tools Help
http://ovm:8888/OVS/faces/app/service/OVS_Create
Google
Create
ORACLE VM Manager
Home Profile Logout Help
Virtual Machines Resources Servers Server Pools Administration
Logged in as admin
Creation Method
Server Pool
Source
Virtual Machine Information
Confirm
Virtual Machine Information
Previous Next
High Availability
Network Interface Card
Select and Delete
Add Row

* Virtual Machine Name:	racnode1
* Console Password:	••••••
* Confirm Console Password:	••••••
Enable High Availability:	□

Select	Virtual Network Interface Name	Virtual Network Interface MAC Address	Bridge
●	VIF0	00:16:3E:03:67:D1	xenbr0
○	VIF1	00:16:3E:49:9B:E9	xenbr1

The VM will be built – This will take some time. Repeat this step for the second node.

*** Do not power up the VMs yet ***

5) Add the shared disks to both VMs.

Edit the vm.cfg for both VM's. These can be found on the Oracle VM Server in /OVS/running_pool/*racnode1/vm.cfg and /OVS/running_pool/*racnode2/vm.cfg, add:

```
'phy:/dev/racdevc1,xvdc,w!',
'phy:/dev/racdevd2,xvdd,w!',
'phy:/dev/racdeve2,xvde,w!',
'phy:/dev/racdevf4,xvdf,w!',
'phy:/dev/racdevg1,xvdg,w!',
```

to the 'disk' section, see below for sample vm.cfg.

The OneCommand for Oracle VM build engine, Clusterware, Oracle RAC or ASM do not require that disks have identical names or ordering on all nodes, however, it might be easier for humans to deal with identical names and ordering. Therefore it is recommended to add disks in same order with same disk names to all VMs.

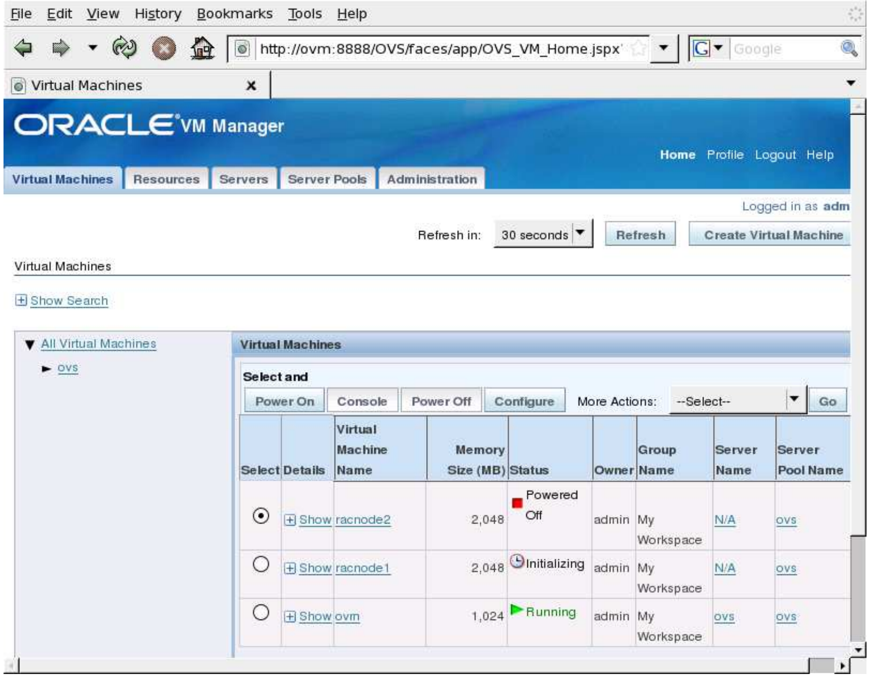
6) When complete, inspect the VM's vm.cfg configuration files.

These can be found on the OVM Server in /OVS/running_pool/*racnode1/vm.cfg and /OVS/running_pool/*racnode2/vm.cfg. You should see the 5 shared disks which will be presented to the guests as /dev/xvdc /dev/xvdd /dev/xvde /dev/xvdf /dev/xvdg. They can be in any order on both nodes as described in the previous section. You should also be able to spot the 2 NICs, xenbr0 and xenbr1. This is a **sample** vm.cfg file:

```
bootloader = '/usr/bin/pygrub'
disk = ['file:/OVS/running_pool/3_racnode1/System.img,xvda,w',
'file:/OVS/running_pool/3_racnode1/Oracle11201RAC_x86-xvdb.img,xvdb,w',
'phy:/dev/racdevc1,xvdc,w!',
'phy:/dev/racdevd2,xvdd,w!',
'phy:/dev/racdeve2,xvde,w!',
'phy:/dev/racdevf4,xvdf,w!',
'phy:/dev/racdevg1,xvdg,w!',
]
memory = '2048'
name = '3_racnode1'
on_crash = 'restart'
on_reboot = 'restart'
uuid = 'd98efba4-7445-8459-9326-adae28249200'
vcpus = 2
vfb = ['type=vnc,vncunused=1,vnclisten=0.0.0.0,vncpasswd=oracle']
vif = ['bridge=xenbr0,mac=00:16:3E:5C:AE:24,type=netfront',
'bridge=xenbr1,mac=00:16:3E:24:D3:34,type=netfront',
]
vif_other_config = []
```

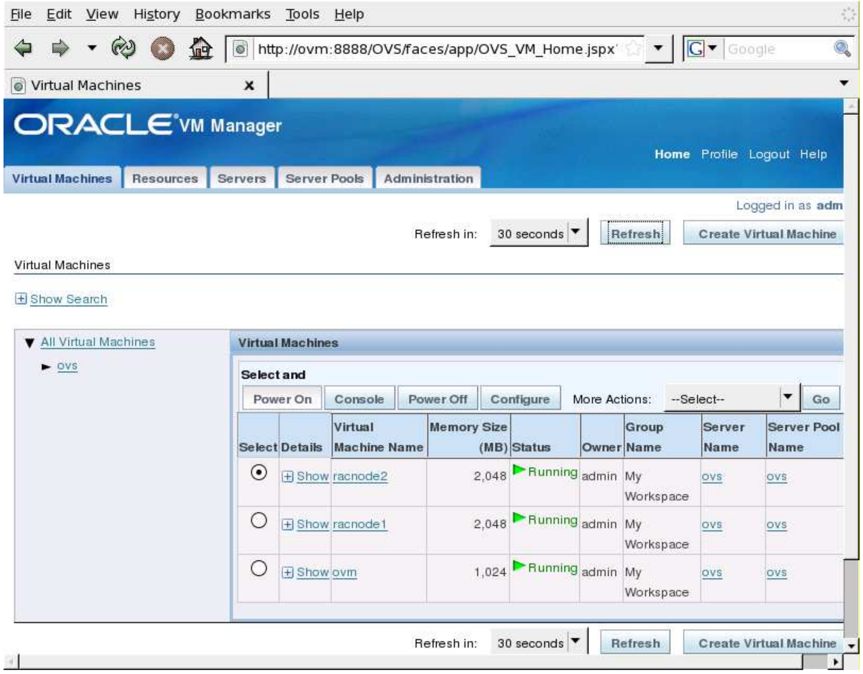
7) Power ON both virtual machines

- Select the Virtual Machines tab
- Select the radio button for the first node
- Click the **Power On** button, repeat for second node



Select	Details	Virtual Machine Name	Memory Size (MB)	Status	Owner	Group Name	Server Name	Server Pool Name
●	Show	racnode2	2,048	🟥 Powered Off	admin	My Workspace	N/A	ovs
○	Show	racnode1	2,048	Initializing	admin	My Workspace	N/A	ovs
○	Show	ovm	1,024	▶ Running	admin	My Workspace	ovs	ovs

- If the VNC plugin has been configured for Oracle VM manager you can connect to the console by selecting the radio button next to the node and clicking the **Console** button. Open both consoles.



Select	Details	Virtual Machine Name	Memory Size (MB)	Status	Owner	Group Name	Server Name	Server Pool Name
●	Show	racnode2	2,048	▶ Running	admin	My Workspace	ovs	ovs
○	Show	racnode1	2,048	▶ Running	admin	My Workspace	ovs	ovs
○	Show	ovm	1,024	▶ Running	admin	My Workspace	ovs	ovs

Alternatively, use vncviewer from any PC; First get the ports from Dom0:

Dom0 # **xm list**

```
Name                        ID    Mem   VCPUs     State    Time(s)
3_racnode1                   1    2048      2     -b----      72.4
4_racnode2                   2    2048      2     r-----      68.1
```

Dom0 # **xm list -l** <VM-full name or ID> **| grep location**

Use the port that appears on line as “location 0.0.0.0:**5901**” e.g.:

$ vncviewer <ovm_server_host>:5901   (substitute your OVM Server node name)

A third alternative, for advanced users is described in APPENDIX D: Using Text Mode Console

8) Complete the First Boot interview:

First Boot is the time when a newly instantiated Guest VM is started for the first time.
At this point in time the guest machines are identical – they have no personality e.g. no hostname, no IP address etc. First boot adds this personality.

The following is displayed on the console of both guests. You must identify which node is going to be the master; the interactive interview will run on that node.

```
  == Oracle RAC 11.2.0.1.4 OneCommand for Oracle VM - template configuration (i686) ==

     NODE SELECTION
     ==============

     Is this the first node in the cluster  (YES/NO): [   ]
```

Figure 1

On **Node1** Enter: **YES** (On **Node2** Enter: **NO**)

The configuration utility then waits until the **NO** response has been given on the second node.

Node1

```
  == Oracle RAC 11.2.0.1.4 OneCommand for Oracle VM - template configuration (i686) ==

     NODE SELECTION
     ==============

     Is this the first node in the cluster  (YES/NO): [YES]

     IMPORTANT
     =========

     Please answer this question on the other node in the cluster
     WAITING
........
```

Figure 2

Go to the Node2 console screen

Node2

```
 == Oracle RAC 11.2.0.1.4 OneCommand for Oracle VM - template configuration (i686) ==

     NODE SELECTION
     ==============

     Is this the first node in the cluster  (YES/NO): [NO ]

WAITING for NODE1 interview phase to be complete!
...
```

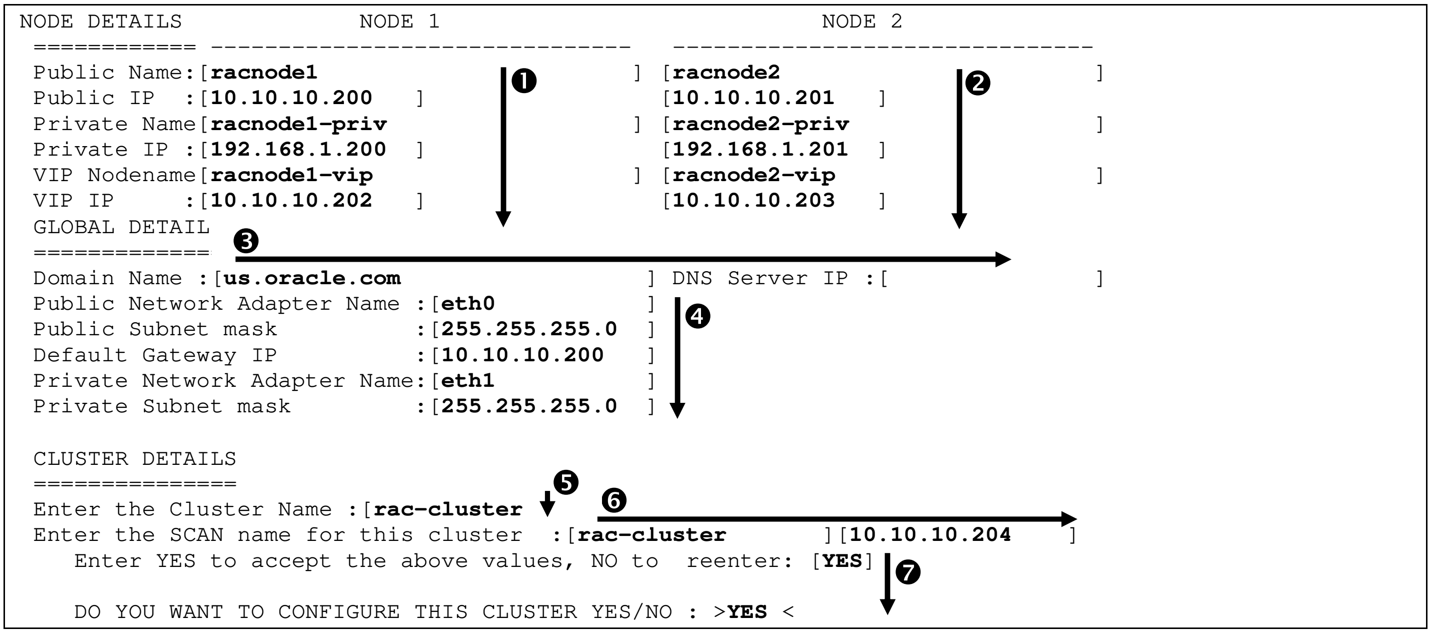
Figure 3

Enter: **NO**

You may enter the responses in any order. This is all the interaction required on the second node. All subsequent configuration steps now take place on the first node.

After completing the node identification steps you must now provide information for first boot to complete. A simple data entry screen is displayed on Node1.

Node1



```
NODE DETAILS             NODE 1                                    NODE 2
 ============ ------------------------------   ------------------------------
 Public Name:[racnode1                       ] [racnode2                      ]
 Public IP  :[10.10.10.200   ]                 [10.10.10.201   ]
 Private Name[racnode1-priv                  ] [racnode2-priv                 ]
 Private IP :[192.168.1.200  ]                 [192.168.1.201  ]
 VIP Nodename[racnode1-vip                   ] [racnode2-vip                  ]
 VIP IP     :[10.10.10.202   ]                 [10.10.10.203   ]
 GLOBAL DETAIL
 =============
 Domain Name :[us.oracle.com                 ] DNS Server IP :[                ]
 Public Network Adapter Name :[eth0          ]
 Public Subnet mask          :[255.255.255.0 ]
 Default Gateway IP          :[10.10.10.200  ]
 Private Network Adapter Name:[eth1          ]
 Private Subnet mask         :[255.255.255.0 ]

 CLUSTER DETAILS
 ===============
 Enter the Cluster Name :[rac-cluster
 Enter the SCAN name for this cluster  :[rac-cluster       ][10.10.10.204    ]
    Enter YES to accept the above values, NO to  reenter: [YES]

    DO YOU WANT TO CONFIGURE THIS CLUSTER YES/NO : >YES <
```

Figure 4

Enter the relevant values, basic user input validation is performed, if you make a mistake – at the reenter prompt enter 'NO' to cycle through the interview process.

Step 1 & 2: Node1&2 details, use your own machines names / IP addresses
Basic IP address and name validation is performed. Node names cannot contain dots (use domain name for that). All Public and VIP IPs must be on same subnet; similarly the Private IPs should be on the same subnet, no duplicate IPs or node names allowed.

Step 3: Optionally enter Domain name and DNS Server IP. If no domain name is entered: "localdomain" is used. DNS Server IP is optional. If a DNS Server is specified you must configure the Grid Infrastructure SCAN name/IP Addresses within the DNS Server

Step 4: NIC & general network details, use your own network details
Adapter names must exist and be different from one another, typically they will be eth0 and eth1. Network masks must be in dot notation format, e.g. 255.255.252.0

Step 5: Unique Clustername - 1 to 15 alphanumeric or hyphens – no spaces

Step 6: The SCAN Name. If a DNS Server IP address was NOT entered previously then an IP Address for the SCAN will be requested – this should be in the same subnet as the public/VIP network

Step 7: Opportunity to reenter the configuration data and a final chance to exit without saving

When you select YES at the final confirmation prompt you will see this on both VMs as the network fully configures itself as per interview:

Node1 and Node2

```
    Generating Network Configuration files...
    Modifying /etc/hosts
    Modifying /etc/resolv.conf
    Modifying /etc/sysconfig/network
    Modifying /etc/sysconfig/network-scripts/ifcfg-eth0
    Modifying /etc/sysconfig/network-scripts/ifcfg-eth1
Shutting down loopback interface:                          [  OK  ]
Bringing up loopback interface:                            [  OK  ]
Bringing up interface eth0:                                [  OK  ]
Bringing up interface eth1:                                [  OK  ]
Shutting down kernel logger:                               [  OK  ]
Shutting down system logger:                               [  OK  ]
Starting system logger:                                    [  OK  ]
Starting kernel logger:                                    [  OK  ]
Shutting down Avahi daemon:                                [  OK  ]
Starting Avahi daemon...                                   [  OK  ]
Starting nscd:                                             [  OK  ]
Reloading nscd:                                            [  OK  ]
Template configuration disabled.

INFO (node:racnode1): Network configuration completed...

Enterprise Linux Enterprise Linux Server release 5.5 (Carthage)
Kernel 2.6.18-194.0.0.0.3.el5xen on an i686

racnode1 login: root
Password: ovsroot
```

Figure 5

Now login as **root** with the default password as **ovsroot** on node 1 only.

If for some reason you entered incorrect data (no network between the two nodes) you may redo just this interview phase by logging in as root to both machines and invoking: **/u01/racovm/netconfig.sh.** Do not run this after the installation of RAC software is complete.

Install Grid Infrastructure and Oracle RAC

You have now completed the node preparation phase and the nodes are ready to install Oracle RAC as they have network access and shared storage configured.

Standard Build:

By default, the install will create an Oracle RAC database called 'ORCL' (sid ORCL1 & ORCL2) all database passwords will be 'oracle'. To proceed with the default build (for custom build see Appendix A: Build Options), on node1 only, login as root (default root password is ovsroot), and run the commands:

```
[root@racnode1 ~]# cd /u01/racovm
[root@racnode1 racovm]# ./buildcluster.sh
Are you sure you want to install RAC?
Do not run if software is already installed and/or running..  [yes|no]? yes
```

If you answer anything other than "yes" the installation will not continue. You do not need to run anything on the second node.

Monitoring the Build:

There is an install log in the current directory named: buildcluster1.log (each run will get the lowest free numbered file). By default the progress log is located at /tmp/progress-racovm.out.

Typically the full install will take ~30-45 minutes although this is dependant on the hardware used to host the VMs and the build options. You can watch the progress of the install by viewing progress logfile on node1:

```
[root@racnode1 racovm]# cat /tmp/progress-racovm.out
2010-06-17 14:12:52:[buildcluster:Start:racnode1] Create 11gR2 RAC Cluster
2010-06-17 14:12:52:[setsshroot:Start:racnode1] SSH Setup for the root user...
2010-06-17 14:12:52:[setsshroot:Done :racnode1] SSH Setup for the root user completed successfully
2010-06-17 14:12:52:[setsshroot:Time :racnode1] Completed successfully in 0 seconds (0h:00m:00s)
2010-06-17 14:12:52:[copykit:Start:racnode1] Copy kit files to remote nodes
2010-06-17 14:12:52:[copykit:Done :racnode1] Copy kit files to remote nodes
2010-06-17 14:12:52:[copykit:Time :racnode1] Completed successfully in 0 seconds (0h:00m:00s)
2010-06-17 14:12:52:[usrsgrpslocal:Start:racnode1] Verify Oracle users & groups (create/modify mode)..
2010-06-17 14:12:53:[usrsgrpslocal:Done :racnode1] Verify Oracle users & groups (create/modify mode)..
2010-06-17 14:12:53:[usrsgrpslocal:Time :racnode1] Completed successfully in 1 seconds (0h:00m:01s)
2010-06-17 14:12:54:[usrsgrps:Time :racnode1] Completed successfully in 2 seconds (0h:00m:02s)
2010-06-17 14:12:54:[printparams:Time :racnode1] Completed successfully in 0 seconds (0h:00m:00s)
```

The install log printed to the screen will have all commands executed in verbose mode, so you can see as the various tools, like clone.pl, netca, dbca, emca are executed along with their output.

Default Installation Specifications:
SID: ORCL1 & ORCL2
DB name: ORCL
Grid Infrastructure Home: /u01/app/11.2.0/grid
Oracle RAC Home: /u01/app/oracle/product/11.2.0/dbhome_1
ORACLE_BASE: /u01/app/oracle
Central Inventory: /u01/app/oraInventory
Root OS Password: ovsroot
All other passwords: oracle

APPENDIX A: Build Options:

Before invoking **/u01/racovm/buildcluster.sh** you may edit **/u01/racovm/params.ini** to modify some build options (bottom part of the file). The top part of params.ini should not be modified unless instructed to by Oracle Support. *Some* of the options that may be modified are:

```
# The name of the ASM diskgroup, default "DATA"
#
RACASMGROUPNAME="DATA"

# The ASM Redundancy for the diskgroup above
# Valid values are EXTERNAL, NORMAL or HIGH
#
RACASMREDUNDANCY="EXTERNAL"

# Build Database?
# Character set maybe be selected using DATABASE_CHARACTERSET (defaults to AL32UTF8)
#
BUILD_RAC_DATABASE=yes

# The Database Name
#
DBNAME=ORCL

#
# The Instance name, may be different than database name,
# limited to 1 to 8 alphanumeric characters
#
SIDNAME=ORCL

#
# Configure EM DB Console
#
CONFIGURE_DBCONSOLE=no

#
# The password that will be set for the ASM and RAC databases
# as well as EM DB Console and the oracle OS user.
# If not defined here you will be prompted for it (only once)
# at the start of the build. Required to be set here or environment
# for silent mode.
# Use single quote to prevent shell parsing.
#
RACPASSWORD='oracle'

#
# Password for 'root' user. If not defined here you will be prompted
# for it (only once) at the start of the build.
# Assumed to be same on both nodes and required to be set here or
# envronment for silent mode.
# Use single quote to prevent shell parsing.
ROOTUSERPASSWORD='ovsroot'
```

If you do not wish to store the passwords for the root or Oracle user in the configuration file, remove or comment them, and they will be prompted for at the start of the build.
If you select to configure Enterprise Manager DB Console (`CONFIGURE_DBCONSOLE=yes)` then it will start automatically on both nodes out of /etc/rc.local and listen to port 1158 (by default).

DB Console URLs are https://racnode1:1158/em and https://racnode2:1158/em

APPENDIX B: Adding More Nodes:

It is possible to add nodes to the 2-node cluster (after above build is done) using the kit as follows:

1. Follow the steps in this document (step 4 on page 10) to create a 3rd or 4th VM, using Oracle VM Manager or Oracle VM Command Line Interface (CLI).

2. Attach the same shared storage as the existing cluster nodes have (step 5 on page 12).

3. Boot the additional VM(s) and connect to console via VNC or text mode console (See Appendix D: Using Text Mode Console)

4. At the node selection prompt type "OUT":

```
 == Oracle RAC 11.2.0.1.4 OneCommand for Oracle VM - template configuration (i686) ==

   NODE SELECTION
   ==============
   Is this the first node in the cluster  (YES/NO): [OUT]
```

5. There will be no interview. The VM will be called "localhost" and will have no network configured.

6. From one of the existing VMs copy & paste (available only via text mode console, see Appendix D: Using Text Mode Console) the contents of the file **/u01/racovm/netconfig.ini** as **/u01/racovm/netconfig.ini** in the new VM (overwriting the current file). If you couldn't get the copy & paste to work you can use a normal text editor to type that information in the new VM.

7. Edit **/u01/ racovm/netconfig.ini** on the new VM and add the information about the new VMs, e.g:

```
NODE3=racnode3
NODE3IP=192.168.1.205
NODE3PRIV=racnode3-priv
NODE3PRIVIP=10.10.10.205
NODE3VIP=racnode3-vip
NODE3VIPIP=192.168.1.206
```

8. Invoke the network configuring script in silent mode as root on the new VM. Be sure that you are connected via the console (either VNC or text mode) as this will establish/restart the network:

   ```
   # /u01/racovm/netconfig.sh -n3
   ```

   Flag **-n3** means 3rd node, **-n4** would be 4th node, etc. Do not run this procedure on a node with Oracle RAC software already configured. This will configure the network in the VM to the values of NODE**3*** listed in netconfig.ini. You may add as many nodes as you wish in this fashion. If you enter information about other nodes in netconfig.ini (e.g. NODE4* or NODE5*) the generated /etc/hosts file will contain all their details. Note that silent mode network configuration allows longer node or domain names compared to the text mode interview.

   The add node procedure will automatically propagate this more complete /etc/hosts to the existing nodes, so that all nodes know about the others. If the hosts are registered in DNS, the /etc/hosts file may be skipped by setting resolver order in /etc/nsswitch.conf.

9. Now that network is up on the new node, copy the more up-todate **/u01/racovm/netconfig.ini** from the new node (it should contain all newly added nodes as well as existing nodes) to any existing cluster member (e.g. racnode1) where you plan to run the addnode(s) procedure from.

   ```
   # scp /u01/racovm/netconfig.ini  racnode1:/u01/racovm
   ```

10. Finally, run the addnode(s) procedure from the cluster member (e.g. racnode1) you copied the updated netconfig.ini to:

    ```
    # cd /u01/racovm
    # ./racovm.sh -S addnodes -N racnode3   2>&1 | tee addnode3.log
    ```

The “2>&1 | tee “ means save stdout and stderr to addnode3.log, this is useful since errors are printed to stderr, so using only “tee” or “>” will only capture stdout without any possible errors. If you do not need logging you may omit all of that.

It is possible to add several nodes in one command invocation, just separate them with a comma. By default, new database instance(s) will not be created on the new node(s), if that is required, add the step “addinstances” to the above command, or run it separately at a later time. Here is the sample command to add 2 nodes with their instances:

```
# ./racovm.sh -S addnodes,addinstances -N racnode3,racnode4   2>&1 | tee addnode-inst-node3-4.log
```

Or, to just add instances on a new node (assuming it was already added using addnodes as described above):

```
# ./racovm.sh -S addinstances -N racnode3   2>&1 | tee addinstances-node3.log
```

APPENDIX C: Using More (or less) Disks:

Advanced users may want to use differently named or more than 5 disks for database/ASM.

Before the Build:

There is an option to use differently named or more than 5 disks before the build. If you attached more than 5 shared devices to the VMs, before you run **buildcluster.sh**, edit **/u01/racovm/params.ini** and modify **ALLDISKS** to contain the list of all devices or partitions that should be used by the database/ASM. The **params.ini** file describes the rules for setting this parameter.
Remember that whatever disks/partitions are specified should be discoverable by **RACASMDISKSTRING** in the **params.ini** file who's default is "/dev/xvd[c-g]1". Do not set the discovery string to a too wide value, e.g. /dev/*, the udev rule is written based on this string, and so might affect devices you did not intend on affecting.

As an example – if you add a 6th device: /dev/xvdh.

ALLDISKS="/dev/xvdc /dev/xvdd /dev/xvde /dev/xvdf /dev/xvdg /dev/xvdh"

The **RACASMDISKSTRING** should be:

RACASMDISKSTRING="/dev/xvd[c-h]1" (Discovers xvdc1, xvdd1, through xvdh1)

WRONG: RACASMDISKSTRING="/dev/xvd?1" (Since it also discovers xvda1, xvdd1 which are the /u01 and /boot mount points by default)

The discovery string may have multiple entries separated by space.

By default, the build procedure will automatically partition all the disks with 1 partition each. An already partitioned disk, e.g. /dev/xvdc3 will be used as-is. It will also write a new UDEV rule, default rule file is: /etc/udev/rules.d/99-oracle.rules.

If you want to use less than the default and recommended 5 disks (for example 3), set ASM_MIN_DISKS=3 in params.ini.

After the build:

After the build, simply add more disks to the VMs, partition them correctly and add them to ASM as you normally would. If you follow this route remember to modify the UDEV rules file on all nodes to give the new disk the correct permissions. Default rules file is: /etc/udev/rules.d/99-oracle.rules.

APPENDIX D: Using Text Mode Console

Advanced users may wish to connect to console during boot of the VM, useful to allow copy/paste when no network is setup. From Dom0, boot the VMs using the following command; it will append the arguments in “extra” to the kernel boot line:

Dom0# **cd /OVS/running_pool/<name-of-VM>**
Dom0# **xm create -c vm.cfg extra="console=hvc0"**

This will boot the VM and connect the console to your terminal.

Or, if you prefer to do it manually:

Dom0# **cd /OVS/running_pool/<name-of-VM>**
Dom0# **xm create -c vm.cfg**

Then on the grub menu (first screen), quickly type **e** (for “edit”), you will see this screen:

```
   pyGRUB  version 0.6
   root (hd0,0)
   kernel /vmlinuz-2.6.18-194.0.0.0.3.el5xen ro root=LABEL=/ numa=off
   initrd /initrd-2.6.18-194.0.0.0.3.el5xen.img
```

Scroll to second line (**kernel** boot line), type **e** (for “edit”) and add a space followed by **console=hvc0** to the end of that line. Press **enter** then **b** (to boot). The console is now connected to your terminal.

Note: In grub edit mode BACKSPACE key may not work, use CTRL-H instead.

Following either of the above methods will allow copy and paste into/from the VM’s console; this is only needed when network is not yet setup.

APPENDIX E: Building Clusters with more than 2 Nodes

In order to build larger clusters, the network needs to be configured on all nodes. There are two methods of setting up the network on multiple nodes. Assume we want to build a 4 node cluster here are the two methods to setup the network and build the Orcale RAC cluster:

Partial Interview:

1. Instantiate the 4 VM's including adding the shared storage to all 4 VMs

2. Boot all 4 VMs. They will all pause at the 'Is this the first node in the cluster [ ]' question

   On Node1 answer YES
   On Node2 answer NO
   One the other node in the cluster enter the word OUT

3. On node1 complete the network config screen for the first 2 nodes in the cluster as a normal 2-node cluster, and let them continue to configure the network. The hostnames on nodes1 and node2 will be correct. On the other nodes it will show localhost – and network will not be configured.

4. On Node1 login as root (default password ovsroot) and edit the /u01/racovm/netconfig.ini file. You will see that for each node 6 lines have been written. Copy the 6 lines and paste for each new node, modify the node number as well as network information, e.g:

   ```
   NODE3=racnode3
   NODE3IP=192.168.1.205
   NODE3PRIV=racnode3-priv
   NODE3PRIVIP=10.10.10.205
   NODE3VIP=racnode3-vip
   NODE3VIPIP=192.168.1.206
   ```

   Add another such block for node4, all variables will start with NODE4, e.g. NODE4IP=…

5. Update the /etc/hosts file on node1 & node2 (since their network is already configured):

   On node1:
   # cd /u01/racovm
   # ./netconfig.sh -n1 -H (-H means to only update /etc/hosts based on content of netconfig.ini)
   # scp netconfig.ini racnode2:/u01/racovm
   # ssh racnode2 /u01/racovm/netconfig.sh -n2 -H (will update /etc/hosts on node2)

6. **CHOOSE ONE METHOD 6.1 or 6.2:** Create an exact copy of the updated file in the /u01/racovm directory on node3 & node4 and use it to configure their network. This can be done in one of these two methods:

   6.1 Copy & paste the entire netconfig.ini file using the console in text mode, see Appendix D Using Text Mode Console for further information on accessing the console in text mode. Once the file is on node3 & node4 in /u01/racovm directory, run:

On node3:	On node4:
# cd /u01/racovm	**# cd /u01/racovm**
# **./netconfig.sh -n3**	# **./netconfig.sh -n4**

The network is now setup on node3 & node4, go to step 7 to build the cluster.

6.2 On node1 stamp the network information (netconfig.ini) on to the still empty shared disk (which is shared amongst all VMs), as follows:

# **cd /u01/racovm**
# **./netconfig.sh -W** **(Writes the network information from netconfig.ini to shared storage)**

```
INFO (node:racnode1): Network configuration file netconfig.ini written to
/dev/xvdc
Run: netconfig.sh -R -c /dev/xvdc (assuming identical device name on remote
nodes) to read the file on the remote nodes
```

By default /dev/xvdc is used but this can be overridden using netconfig.sh's -c flag.

On node3 and node4 read and apply the network configuration as follows:

On node3:	On node4:
# **cd /u01/racovm**	# **cd /u01/racovm**
# **./netconfig.sh -R -n3**	# **./netconfig.sh -R -n4**

If you wish, you may break the above command into two steps, first reading the netconfig.ini file, which allows you to inspect it for correctness, then apply the network configuration, the commands would be:

# **./netconfig.sh -R** **(Reads the network information from shared storage to netconfig.ini)**
# **./netconfig.sh -n3** **(use -n4 on node4)**

The network is now setup on node3 & node4. Proceed to build the cluster.

Network setup from Dom0/grub:

There is an easier way to configure all nodes at once, directly from Dom 0 as follows:

1. Instantiate the 4 VM's including adding the shared storage to all 4 VMs, do not boot them yet.

2. Prepare a netconfig.ini file with network information for all 4 nodes using any text editor.

3. From Dom0 stamp the network information (netconfig.ini) on to the still empty shared disk (which is shared amongst all VMs). Stamp the disk that will be visible to the VM as /dev/xvdc (using the -c flag, it is possible to use a any other shared disk). You can find netconfig.sh (and sample netconfig.ini) on Dom0 in the "utils" directory under /OVS/running_pool/<vmname>**/utils.**

3.1 If the shared disk is a file based disk it would need to be stamped via the loopback device, as follows **(Look in vm.cfg for the device mapped to xvdc)**:

Dom0 # **losetup -vf /OVS/sharedDisk/shared-diskname.img**

```
Loop device is /dev/loop3
```

Dom0 # **./netconfig.sh -W -c /dev/loop3** (you may ignore the feedback from the command)

```
 INFO (node:dom0): Network configuration file netconfig.ini written to /dev/loop3
Run: netconfig.sh -R -c /dev/loop3 (assuming identical device name on remote
nodes) to read the file on the remote nodes
```

Dom0 # **losetup -d /dev/loop3** **(remember to delete the loop device)**

3.2 If the shared disk is a physical disk on Dom0 that is passed to the VMs as ‘phy:/dev/racdevc1’ for example, then it should be stamped directly as follows:

Dom0 # **./netconfig.sh -W -c /dev/racdevc1** (you may ignore the feedback from the command)

```
 INFO (node:dom0): Network configuration file netconfig.ini written to
/dev/racdevc1
Run: netconfig.sh -R -c /dev/racdevc1 (assuming identical device name on remote
nodes) to read the file on the remote nodes
```

4. Boot the VMs using the following command, which will append the arguments in “extra” to the kernel boot line. These arguments will be processed by the template first-boot module, which will bypass the normal interview and directly read the network information from the shared disk and setup the network on that VM (based on the node number passed in the **-n** flag):

Dom0 # **cd /OVS/running_pool/<name-of-VM1>**

Dom0 # **xm create -c vm.cfg extra="console=hvc0 template-reconfig-args=’-R -n1’"**

For VM’s 2-4 just change the number following the -n flag, e.g. -n2 for VM number 2.

Or, if you prefer to do it manually on the grub menu, boot each VM and at the grub menu (first screen), quickly type **e** (for “edit”), you will see this screen:

```
pyGRUB  version 0.6
root (hd0,0)
kernel /vmlinuz-2.6.18-194.0.0.0.3.el5xen ro root=LABEL=/ numa=off
console=hvc0 template-reconfig-args='-R -n1'
initrd /initrd-2.6.18-194.0.0.0.3.el5xen.img
```

Scroll to second line (**kernel** boot line), type **e** (for “edit”) and add a space followed by `console=hvc0 template-reconfig-args='-R -n1'` to the end of that line. Press **enter** then **b** (to boot). The VM will boot and read it’s network configuration from the shared disk, and automatically configure itself as node1.

Follow the same procedure for node2, node3 and node4, use -n2, -n3 and -n4 respectively.

Note: Use **single** quotes (‘) to enclose the arguments ‘-R -n1’; do not use double quotes. The grub example above wraps the kernel line due to page layout, in real life it would all appear on one line.

If you attached the same shared disks to all 4 VMs and stamped the right disk (/dev/xvdc) with the netconfig.ini file, then all 4 VMs have their network fully configured.

Fully automated & scripted cluster creation:

It is possible to add the ‘**-b**’ flag to any one of the above VMs (e.g**.** template-reconfig-args=’-R -n1 **-b’)**, which will cause the buildcluster.sh to be called when network is setup on all VMs. If you didn’t supply the ‘**-b**’ flag then proceed to build the cluster in the section below.

Build the cluster:

From any node (as root) run:

# /u01/racovm/buildcluster.sh

This will build a 4-node cluster and a 4 instance Oracle RAC database.

APPENDIX F: Troubleshooting / Frequently Asked Questions

1) If upon first boot after answering the node selection prompt:

```
== Oracle RAC 11.2.0.1.4 OneCommand for Oracle VM - template configuration (i686) ==
   NODE SELECTION
   ==============
   Is this the first node in the cluster  (YES/NO): [YES]
```

You see this error:

```
ERROR - SHARED DISKS INCORRECTLY CONFIGURED
For the Oracle RAC OneCommand on Oracle VM install to work,
a minimum of 5 shared disks must be configured with the following
names: /dev/xvdc /dev/xvdd /dev/xvde /dev/xvdf /dev/xvdg.

Power OFF the VM & configure shared disks correctly using Oracle VM Manager,
OVM CLI, or manually editing the vm.cfg configuration file.

Refer to included PDF file for more details.
```

This means that the shared disks were not attached correctly to that VM. Make sure the vm.cfg has the shared disks listed on both VMs and the shared disks are presented to the VMs as /dev/xvdc,d,e,f,g. The automated install relies on these exact names in order to work correctly. Advanced users should refer to Appendix C if they wish to deviate from this naming convention.

Or this error:

```
ERROR - NETWORK CARDS INCORRECTLY CONFIGURED
For the Oracle RAC OneCommand on Oracle VM install to work,
a minimum of 2 NICS must be configured as eth0 & eth1 (xenbr0 & xenbr1).
This should appear in the vm.cfg config file

You may add the NICs dynamically now via Oracle VM Manager,
OVM CLI, or Power OFF the VM and manually edit the vm.cfg configuration file.

Refer to included PDF file for more details.

Press ANY key to recheck  (Last check: <date>)
```

This means that the network cards were incorrectly configured. The VM requires two NICs, named eth0 connected to xenbr0 and eth1 connected to xebbr1. You may hotplug a NIC and hit enter to recheck, or you may power off and add the NICs.

2) If you experiment with the build process by editing **/u01/racovm/params.ini** and incorrectly entered data causes the build to fail or you stop the build before it finishes, you may clean the environment in preparation for another build. This also clears **the shared disks** (if executed on node1), extreme caution should be taken when running clean. If cleanup only on local node is needed, exchange "clean" with "cleanlocal":

   # /u01/racovm/racovm.sh -S clean

   Note that by default, as a safeguard to accidental removal of the environment, the above command will fail if there are any running processes on a VM. To bypass that, change CLONE_CLEAN_CONFIRM_WHEN_RUNNING to **no** in params.ini, then issue "./racovm.sh -S copykit", (will copy kit files+params.ini to all nodes) now you can clean all VMs despite running processes.

3) Similar safeguards are built for other operations. For example, this removenodes request:

   # /u01/racovm/racovm.sh -S removenodes –N racnode2

   Will fail if processes are still running on the node(s) to be removed with the following messages:

```
WARNING (node:racnode2): Above Oracle processes are currently running!!!!

ERROR (node:racnode1): It is not possible to remove nodes from the cluster with
existing software running on them, shutdown the above processes and re-run
'removenodes'. If you are absolutely certain it is safe to clean the nodes before
removing them from this cluster, set CLONE_CLEAN_ON_REMNODES=yes and re-run the
'removenodes' operation.
```

   Either shutdown these processes on racnode2, or change CLONE_CLEAN_ON_REMNODES to yes in params.ini (or set in the environment of the shell), then re-issue the command.

4) If you try to stamp a shared disk with the network information, and the disk is busy or held by any process, an error similar to this one will be displayed:

   [root@racnode1 racovm] # ./netconfig.sh -W

```
ERROR (node:racnode1): Device (/dev/xvdc) used to transfer network configuration to
remote node(s) is incorrect or busy (blockdev) (see output below):
BLKRRPART: Device or resource busy
```

   This is a safeguard measure not to overwrite possible user data. The stamping of shared storage should be used only when there is no user data on the disks. To transfer the netconfig.ini to the new node, try one of these options:

   a) Copy/paste the netconfig.ini using console in text mode, see Appendix D Using Text Mode Console for further information on accessing the console in text mode.
   b) If there is an extra non-used disk shared by the added VMs and the current VM, you may stamp to that disk, simply add the '-c /dev/xvdx' assuming /dev/xvdx is that spare disk. Commands will be: ./netconfig.sh –c /dev/xvdx -W to write, and -R to read on the new node.

5) If your hosts are registered in DNS, the /etc/hosts file may be skipped by setting resolver order in /etc/nsswitch.conf. Change this line "hosts: files dns" to "hosts: dns files" thus making DNS be the first option for resolving names, then if "nscd" (by default it's installed and turned ON in the Oracle RAC OVM templates) is used issue:

# service nscd reload

6) It is a best practice to relocate /tmp to a separate disk, so that an accidental fillup of /tmp doesn't also fillup the root (/) filesystem. This is not done in the templates to reduce footprint and allow easier /tmp setting at deployment time. It is extremely easy to do that using the *modifyjeos* tool (included in the *ovm-modify-jeos* rpm from http://edelivery.oracle.com/linux). When guest is shutdown issue:

```
#  modifyjeos -f System.img -P tmp.img 5000 /tmp
Starting VM image reconfiguration...
Processing tmp.img product image...
Mounting VM image...
Mounting product image tmp.img...
Persisting mount point /tmp ...
Unmount VM image...
Successfully created JeOS image files...
```

The above command creates a 5GB tmp.img disk and will add the following line to /etc/fstab inside the guest, so that next time that guest boots it will use /tmp as a dedicated mount point on the newly created tmp.img disk.

```
LABEL=tmp     /tmp        ext3    defaults        1 2
```

7) In 11.2 NETCA no longer writes individual instance's connect data. If for some reason you need to connect to a specific instance, use EZConnect as follows:

```
[oracle@racnode1 ~]$ sqlplus system/oracle@racnode1:1521/ORCL/ORCL1
SQL*Plus: Release 11.2.0.1.0 Production on Thu Jul 22 05:34:39 2010
Copyright (c) 1982, 2009, Oracle.  All rights reserved.
Connected to:
Oracle Database 11g Enterprise Edition Release 11.2.0.1.0 - Production
With the Partitioning, Real Application Clusters, Automatic Storage
Management, OLAP, Data Mining and Real Application Testing options
```

Note the node number followed by port, database and instance names.

8) By default, OneCommand (**racovm.sh**) operates on the database defined in DBNAME & SIDNAME in params.ini, if you want to create/operate on other databases, you may change DBNAME & SIDNAME in params.ini or comment them in the file, and specify on the command line, e.g.:

[root@racnode1 racovm] # DBNAME=MYORCL ./racovm.sh -sS createdb

The above command will create a new database called MYORCL (SIDNAME will default to MYORCL, e.g. MYORCL1, MYORCL2, etc.)

9) By default the templates create an Aministrator Managed Database. It takes two commands to convert it to Policy Managed, as follows:

```
[oracle@racnode1 bin]$ /u01/app/11.2.0/grid/bin/srvctl config db -d orcl
Database unique name: ORCL
Database name: ORCL
Oracle home: /u01/app/oracle/product/11.2.0/dbhome_1
Oracle user: oracle
  <snip>
Management policy: AUTOMATIC
```

```
Server pools: ORCL
Database instances: ORCL1,ORCL2
 <snip>
Type: RAC
Database is administrator managed

[oracle@racnode1 bin]$ /u01/app/11.2.0/grid/bin/srvctl add srvpool -g prod01 -l 2 -u 2
[oracle@racnode1 bin]$ /u01/app/11.2.0/grid/bin/srvctl status srvpool
Server pool name: Free
Active servers count: 0
Server pool name: Generic
Active servers count: 2
Server pool name: prod01
Active servers count: 0
[oracle@racnode1 bin]$ /u01/app/11.2.0/grid/bin/srvctl modify db -d orcl -g prod01
[oracle@racnode1 bin]$ /u01/app/11.2.0/grid/bin/srvctl config db -d orcl
Database unique name: ORCL
Database name: ORCL
Oracle home: /u01/app/oracle/product/11.2.0/dbhome_1
Oracle user: oracle
 <snip>
Management policy: AUTOMATIC
Server pools: prod01
Database instances:
 <snip>
Type: RAC
Database is policy managed
```

10) It is generally possible to “mix and match” the OS disk from one RAC OVM template to another. For example, if we assume that:

    Oracle 11.2.0.1.4 was released with OS disk of Oracle Linux 5 Update 5
    Oracle 11.2.0.2.2 was released with OS disk of Oracle Linux 5 Update 6

    If there is a need to have 11.2.0.1.4 run with OL5U6 then simply extract only the OS disk (“System.img”) from the 11.2.0.2.2 released template (the OS disk is inside the smaller zip file) and unzip it into the same folder as the Oracle disk of the 11.2.0.1.4 release. If this change is needed after the software is installed, then follow a ‘removenodes’ followed by ‘addnodes’ procedure to add the new node(s) with the modified OS version. Anytime such “mixing” is done, it is highly recommended to set the parameter CLONE_RELINK=yes in params.ini so that relinking takes place as part of the initial OneCommand build cluster or add/remove node(s). After the initial build is over, it is safe to set CLONE_RELINK=no since re-linking is only needed once. It is not recommended to run with mixed OS versions on a cluster for prolonged periods of time; ensure that all cluster nodes are upgraded to the same OS level as described in My Oracle Support Note# 220970.1.

11) Can oneoff patches be added or removed before and/or after the template is fully built?
YES! Patches can be added or removed at any time as follows:

After the cluster is fully built: As any patch is applied or rolled back; follow the README supplied with the oneoff patch, e.g. shutdown any databases, services, etc. apply or rollback the patch using **opatch** and restart services, etc.

After the template's first boot, but before the cluster is built: The templates are delivered as 'software-only' installs, without any central inventory or configuration. In order to apply or rollback patches in this configuration follow these steps, only one node is needed:

- Make sure network is fully configured on a single node (to be patched). At the boot prompt 'Is this the first node in the cluster' type OUT, then edit **/u01/racovm/netconfig.ini** and run **/u01/racovm/netconfig.sh –n1** as needed to configure the network (described in prior sections of this document). Alternatively, setup 2 VMs using normal screen interview and only use one of the VMs.
- Attach both the Grid home and the RAC home to the central inventory by running:

```
[root@racnode1 racovm] # /u01/racovm/racovm.sh -sS giattachlocal,racattachlocal
```

- Switch to the oracle user and apply or rollback the patches as they normally would be using **opatch**. Follow the README supplied with the oneoff patch skipping any instructions to shutdown databases or services, since none are yet defined.
- Once all the patches are applied or rolled back correctly, clean up the disk using:

```
[root@racnode1 racovm] # /u01/racovm/racovm.sh -sS cleanlocal
```

- Shutdown the VM, and use only the Oracle disk, the OS disk was modified (network configured, etc.) and can therefor be discarded. Combine the Oracle disk with an unmodified OS disk, and that becomes the new template 'golden image' that is placed in seed_pool, which is used to create as many VMs as needed.

> Tip: To apply multiple fixes using opatch use the 'napply' option, to rollback multiple fixes use the 'nrollback' option.