An Oracle Technical White Paper
April 2009

Oracle VM High Availability: Hands-on Guide to Implementing Guest VM HA

Introduction

One of the key benefits of Oracle VM (http://oracle.com/virtualization) server virtualization and management solution is the capability to deploy the technology on industry standard x86/x86-64 servers and support a broad range of network and storage devices. Oracle VM can help customers improve server utilization, achieve higher availability and better performance while reducing costs.

This white paper focuses on best practices of the Oracle VM Guest VM High Availability (HA) design and implementation. It serves as a practical guide to help customers design the HA environment and experience the benefits of Oracle VM. Guest VM HA functionality provides a powerful, easy-to-manage solution for maximizing up-time for virtually any guest VM workload, without requiring any tailoring inside the VM, making it simple to set-up, use, and maintain.

The intended audience should have basic knowledge of the Linux operating system and understand how to set up network and storage configurations in Linux.

Oracle VM HA Overview

Beginning with Oracle VM 2.1.2, Oracle VM has incorporated advanced High Availability (HA) features including:

- Guest VM HA
- Secure Live Migration
- Server Pool Load Balancing
- Oracle VM Manager HA

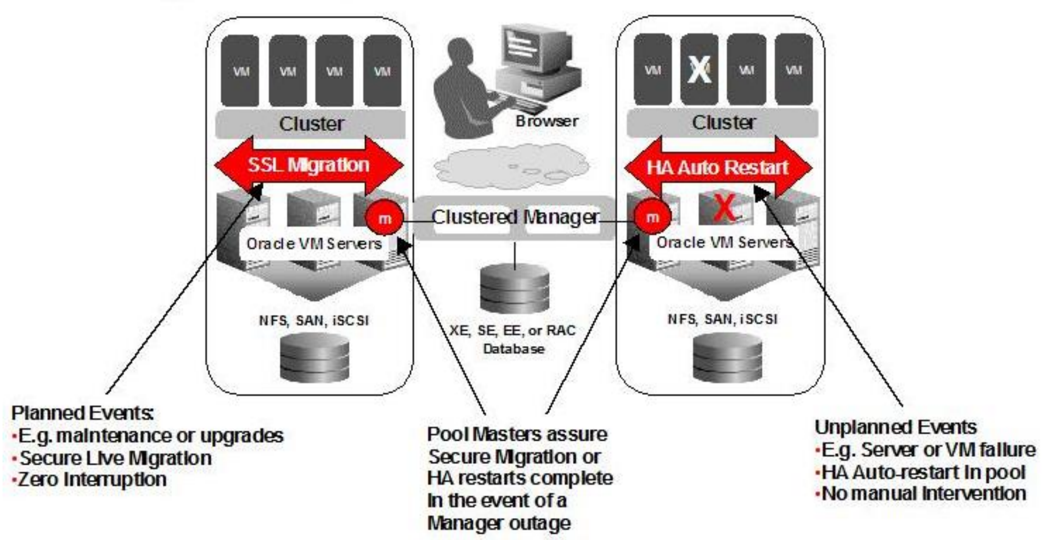


Figure 1: Oracle VM HA – Complete Solution

The above features provide customers an end-to-end HA solution to minimize or eliminate planned and unplanned downtime through advanced functionality included at no extra charge. For details about HA features, benefits and positioning, please refer to Oracle Technical White Paper and Oracle VM Blog:

- Oracle Technical White Paper: Oracle VM – Creating & Maintaining a Highly Available Environments for Guest VMs
  http://www.oracle.com/technologies/virtualization/docs/ovmha-whitepaper.pdf
- Oracle's Virtualization Blog http://blogs.oracle.com/virtualization

Oracle VM Architecture for HA

The components of Oracle VM are Oracle VM Manager and Oracle VM Server. Oracle VM Agent is installed with Oracle VM Server. Oracle VM Manager manages the virtual machines running on Oracle VM Server through the Oracle VM Agent.

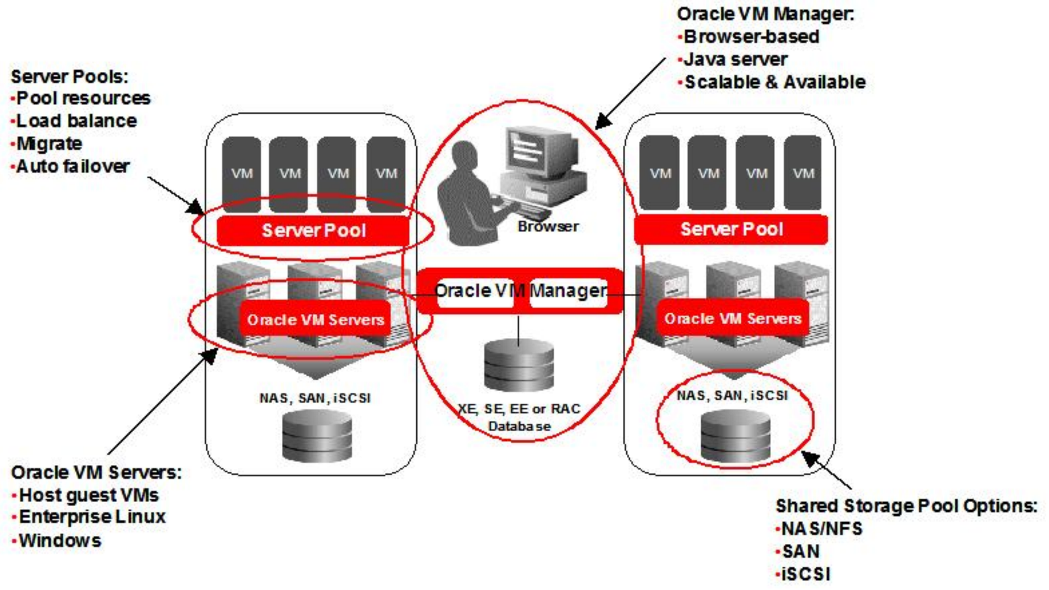


Figure 2: Oacle VM HA – Concept Overview

From a deployment perspective, multiple Oracle VM Servers are grouped into Server Pools in which every server in a given pool has access to shared storage, which can be NFS/NAS, SAN (Fibre Channel) or iSCSI storage. This allows VMs associated with the pool to start and run on any physical server within the pool that is available and has the most resources free. Given the uniform access to shared storage mounted under /OVS directory under which all resources will reside, VMs may also be securely Live Migrated or automatically (re-)started across any servers in the pool.

There are different roles of Oracle VM servers in the pool, and the roles are implemented by the agent running on the Oracle VM server. You can combine multiple roles in a single server, but there is only one Server Pool Master in the server pool.

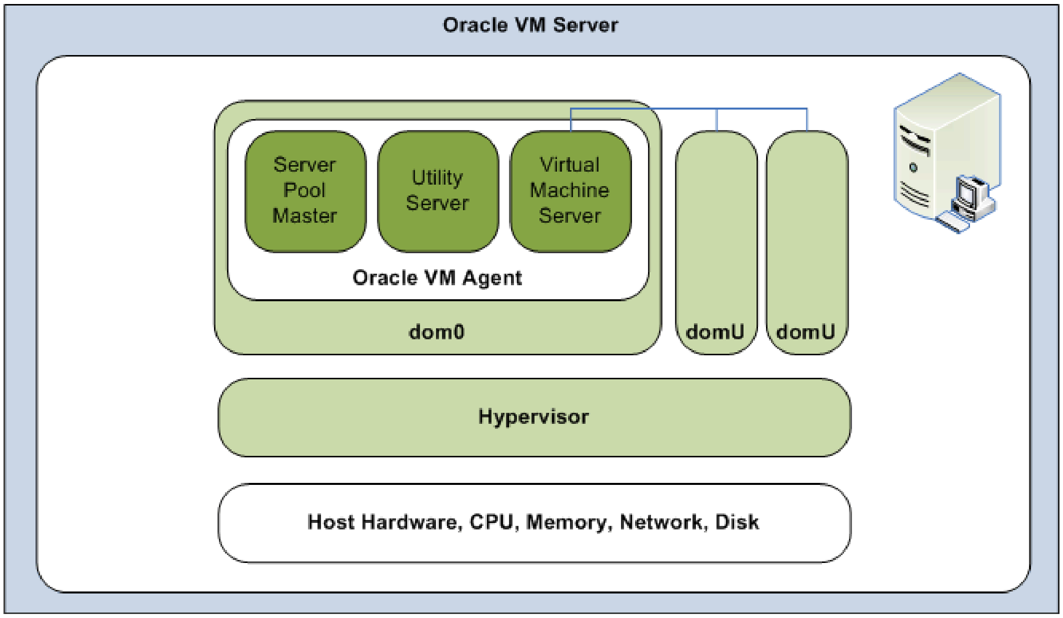


Figure 3: Oracle VM Server Architecture

- **Server Pool Master:** This acts as the contact point to the world outside of Oracle VM Server and dispatches to other Oracle VM Agents. It also provides virtual machine host load-balance, guest VM HA, and local persistence of Oracle VM Server information.
- **Utility Server:** This mainly focuses on creating, removing, migrating, etc. of I/O intensive operations.
- **Virtual Machine Server:** The virtual machine server. This is the daemon for Oracle VM Server virtual machines. Virtual Machine Server can start and stop virtual guests. It also collects performance data for the host and guest operating systems, and acts as a hypervisor for the guest VMs.

VMs are associated with a given Server in the pool dynamically at power-on based on load balancing algorithms or based on a user-defined list of named servers to be used as a host for that specific VM called the Preferred Server list. When VMs are powered-off and not running, they are not associated with any particular physical server since they are simply residing in a powered-off state on shared pool storage.

As a result of this architecture, VMs can easily start-up, power-off, migrate, and/or restart without being blocked by the failure of any individual server or, by the failure of multiple pool servers as long as there are adequate resources in the pool to support requirements of all the VMs running concurrently in the pool.

Each server pool must have its own shared storage resources accessed by Oracle VM servers within the same pool; a separate server pool must have separate shared storage.

Requirements for HA

To implement HA, you must create a cluster of Virtual Machine Servers in a server pool and have them managed by Oracle VM Manager or Oracle Enterprise Manager. HA cannot be implemented with Oracle VM Servers alone. First, you must first create shared storage for the VM servers in the server pool and enable HA on the server pool, then on the guest VMs, as shown the figure below. If you enable HA in the server pool and then for guests, when a virtual machine server is shut down or fails, the guests are migrated or restarted on another available virtual machine server. HA must be enabled for both the server pool and for guests. If HA is not enabled for both, HA is disabled.

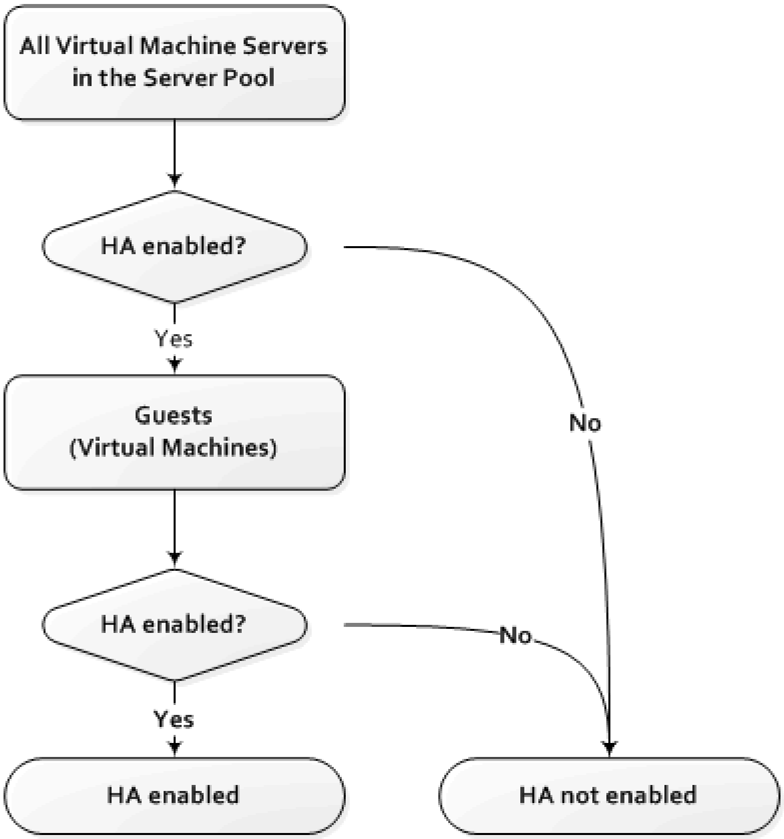

All Virtual Machine Servers in the Server Pool
HA enabled?
Yes
No
Guests (Virtual Machines)
HA enabled?
No
Yes
HA enabled
HA not enabled

Figure 4: Oacle VM HA Checklist

If you are using Oracle VM 2.1.2 as your starting point, assure that it has the most recent updates applied. A working Unbreakable Linux Network (ULN, http://linux.oracle.com) account with Oracle VM activation is required to upgrade the base Oracle VM release media to the latest update, which is critical for a successful implementation (e.g. faster failure detection) of Oracle

VM Guest VM HA function. Please refer to the References on how to update Oracle VM Server and Oracle VM Manager.

Oracle Enterprise Manager Grid Control 10*g* Release 5 (10.2.0.5) introduced the Oracle VM Management Pack which manages Oracle VM resources either by Oracle VM Manager or Oracle Enterprise Manager Grid Control. To ensure resource access integrity, the server pool and associated Oracle VM servers can only be managed by one Oracle VM Manager or one Oracle Enterprise Manager Grid Control exclusively.

Please refer to the release notes of Oracle VM Documentation on specific hardware requirements for Oracle VM Server and Manager.

The Oracle VM Manager is used as the example of setting up guest VM HA feature. Similarly, you can enable the guest VM HA feature from Oracle Enterprise Manager Grid Control 10*g* R5. Please refer to the Oracle Enterprise Manager Documentation.

Oracle VM Server and Manager Installation

Oracle VM Server and Oracle VM Manager installation is a straightforward process, taking a few minutes to complete depending on the hardware configuration and installation methods you are using. Please follow the instructions in the installation guides of Oracle VM Documentation.

Since you install Oracle VM servers which will be placed into a server pool for HA purpose, so you don't need to mount anything over network or SAN during the installation. For example as a standard installation, you just install Oracle VM server onto on local disk (/boot, swap, /, and /OVS).

When asked to provide network configuration, make sure that you give fully qualified domain name for the Oracle VM server (e.g. vmserver1.example.com). The real host name (vmserver1 for example) must not be associated with the loopback address (127.0.0.1). So the /etc/hosts file should look like:

```
127.0.0.1       localhost.localdomain localhost

192.168.100.100 vmserver1.example.com vmserver1
```

Make sure that you have all the server entries in your DNS server; If DNS is not used, make sure the correct setting in /etc/hosts for all the servers in the pool. If you plan to use DNS for all servers, but DNS was not specified during the server installation, please update /etc/resolv.conf file and add your domain name in it as

```
search example.com

domain example.com

nameserver your-dns-ip-address
```

Please note that all the servers in the same pool must have the consistent name resolution, either by DNS or by file (/etc/hosts). You should not have mixed name services for the servers in the same server pool. For example, some have DNS, while others use /etc/hosts to resolve host names.

Once you complete server installation, you should run up2date to connect to ULN and bring the Oracle VM server including Oracle VM agent up to date.

Then you move on to the Oracle VM Manager installation. After the installation is completed, remember to download and install the tightvnc-java rpm (http://oss.oracle.com/oraclevm/manager/RPMS/) onto the host running Oracle VM Manager if you want to access Oracle VM Manager from any Java enabled web browser.

You may also consider running Oracle VM Manager inside a virtual machine. We have a pre-built Oracle VM Template for Oracle VM Manager which you can download from Oracle E-Delivery (http://edelivery.oracle.com/oraclevm). Please refer to the Oracle VM Manager Template white paper for more details on how to use the Oracle VM Manager Template.

Creating Shared Storage for the Server Pool

Creating shared storage is very important and it removes resource access and system-specific dependency limitations. For example, when using a repository on shared storage, in the event of an Oracle VM Server failure, resources are readily accessible to other Oracle VM servers in the server pool, which is critical for HA functionality.

Oracle VM supports the use of various types of shared storage, including SAN, NAS, NFS and iSCSI. All the shared resources must be mounted under /OVS, known as OVS repositories. Repositories are a flexible and extensible way to incrementally add additional storage for Oracle VM resource use. Aside from growing (resizing) the primary repository (/OVS mount point), one can extend a repository by adding additional (sub)repositories beneath it.

You can set up shared storage for the server pool in the following configurations:

- OCFS2 (Oracle Cluster File System) using the iSCSI (Internet SCSI) network protocol
- OCFS2 using SAN (Storage Area Network)
- NFS (Network File System)

The procedures for creating shared storage for HA are essentially the same as what's described in the Oracle VM Server User Guide for creating a shared virtual disk using the above storage configurations for live migration. But you have fewer steps to go through when creating shared storage for HA. For example, you don't need to manually modify /etc/fstab for enabling HA since the configuration files will be handled by Oracle VM server agent automatically when you run /usr/lib/ovs/ovs-makerepo utility. In addition, the startup of related cluster services (o2cb) will also be handled when you run ovs-cluster-configure utility. We'll explain how to enable later in this document.

Creating a Shared Virtual Disk Using OCFS2 on iSCSI

To create a shared virtual disk using OCFS2 on iSCSI:

1. Install the iscsi-initiator-utils RPM in Dom0 of Oracle VM server. The iscsi-initiator-utils RPM is available on the Oracle VM Server CDROM or ISO file.

   ```
   # rpm -Uvh iscsi-initiator-utils-6.2.0.868-0.7.el5.i386.rpm
   ```

2. Start the iSCSI service:

   ```
   # service iscsi start
   ```

3. Run discovery on the iSCSI target. In this example, the target is 10.1.1.1:

   ```
   # iscsiadm -m discovery -t sendtargets -p 10.1.1.1
   ```

   This command returns output similar to:

   ```
   10.1.1.1:3260,5 iqn.1992-04.com.emc:cx.apm00070202838.a2

   10.1.1.1:3260,6 iqn.1992-04.com.emc:cx.apm00070202838.a3

   10.2.1.250:3260,4 iqn.1992-04.com.emc:cx.apm00070202838.b1

   10.1.0.249:3260,1 iqn.1992-04.com.emc:cx.apm00070202838.a0

   10.1.1.249:3260,2 iqn.1992-04.com.emc:cx.apm00070202838.a1

   10.2.0.250:3260,3 iqn.1992-04.com.emc:cx.apm00070202838.b0
   ```

4. Delete entries that you do not want to use for the server pool, for example:

   ```
   # iscsiadm -m node -p 10.2.0.250:3260,3 -T iqn.1992-
   04.com.emc:cx.apm00070202838.b0 -o delete

   # iscsiadm -m node -p 10.1.0.249:3260,1 -T iqn.1992-
   04.com.emc:cx.apm00070202838.a0 -o delete

   # iscsiadm -m node -p 10.2.1.250:3260,4 -T iqn.1992-
   04.com.emc:cx.apm00070202838.b1 -o delete

   # iscsiadm -m node -p 10.1.1.249:3260,2 -T iqn.1992-
   04.com.emc:cx.apm00070202838.a1 -o delete

   # iscsiadm -m node -p 10.0.1.249:3260,5 -T iqn.1992-
   04.com.emc:cx.apm00070202838.a2 -o delete
   ```

5. Verify that only the iSCSI targets you want to use for the pool are visible:

```
# iscsiadm -m node
```

6. Review the partitions by checking /proc/partitions:

```
# cat /proc/partitions

major minor  #blocks  name

   8     0   71687372 sda

   8     1     104391 sda1

   8     2   71577607 sda2

 253     0   70516736 dm-0

 253     1    1048576 dm-1
```

7. Restart the iSCSI service:

```
# service iscsi restart
```

8. Review the partitions by checking /proc/partitions. A new iSCSI device (e.g. sdb) is listed. Determine the share disk volume you want to use and create desired partitions (e.g. /dev/sdb1) using fdisk command.

```
# cat /proc/partitions

major minor  #blocks  name

   8     0   71687372 sda

   8     1     104391 sda1

   8     2   71577607 sda2

 253     0   70516736 dm-0

 253     1    1048576 dm-1

   8    16    1048576 sdb
```

9. Format the shared virtual disk from any of the servers in the cluster. For example, to create an OCFS2 partition with a 4k block size, 4k cluster size, 64MB journal size, and 16 node slots:

```
# mkfs.ocfs2 -L mylabel -b 4K -C 4K -J size=64M -N 16 /dev/sdb1
```

Creating a Shared Virtual Disk Using OCFS2 on SAN

To create a shared virtual disk using OCFS2 on SAN:

1. Review the partitions by checking /proc/partitions. Determine the share disk volume (e.g. sdb) you want to use and create desired partitions (e.g. /dev/sdb1) using fdisk command.

   ```
   # cat /proc/partitions

   major minor  #blocks  name

      8     0   71687372 sda

      8     1     104391 sda1

      8     2   71577607 sda2

    253     0   70516736 dm-0

    253     1    1048576 dm-1

      8    16    1048576 sdb
   ```

2. Format the shared virtual disk from any of the servers in the cluster. For example, to create an OCFS2 partition with a 4k block size, 4k cluster size, 64MB journal size, and 16 node slots:

   ```
   # mkfs.ocfs2 -L mylabel -b 4K -C 4K -J size=64M -N 16 /dev/sdb1
   ```

Adding a Shared Virtual Disk Using NFS

To add a shared virtual disk using NFS, you just find an NFS mount point to use. This example uses the NFS mount point that will be used when enabling HA for the server pool.

```
fileserver.example.com:/OVS
```

Enabling HA for the Server Pool

Now you've created shared storage with one of the above storage configurations (OCFS2 using iSCSI, OCFS2 using SAN, or NFS). Before moving on to enable HA, you must make sure all Oracle VM servers in the server pool…

- Have the same cluster root, which is specially designated shared storage used for heatbeating in the cluster. For example, the cluster root might be fileserver.example.com:/OVS for an NFS cluster, or /dev/sdb1 for an OCFS2 cluster. The heartbeat file would be located at fileserver.example.com:/OVS/.server_pool_hb for an NFS cluster. There is no heartbeat file for an OCFS2 cluster as one is in-built in the file system.

- Have the cluster root mounted at /OVS. All other storage may be mounted at the standard location of /OVS/uuid.
- Have moved any /OVS mount points (set in /etc/fstab for Releases 2.1 or 2.1.1) to the /etc/ovs/repositories file. All storage, even that mounted at /OVS, must be maintained in the /etc/ovs/repositories file. You can move the mount points to the /etc/ovs/repositories file with the following command:

  ```
  # /usr/lib/ovs/ovs-cluster-check --alter-fstab
  ```

When you create the server pool from Oracle VM Manager, you've specified one server as the server pool master. You can refer to the Metalink Note 467663.1 if you want to re-assign the server pool master role to another server in the pool. The following are the necessary actions to enable HA for the server pool.

1. Run the script on the server pool master, start the cluster service.

   ```
   # /usr/lib/ovs/ovs-cluster-configure
   ```

   This script generates the /etc/ocfs2/cluster.conf file if it does not already exist, and configures and starts the o2cb service.

2. On the server pool master, configure the shared storage with /usr/lib/ovs/ovs-makerepo source C description. The following is an example of shared SAN storage using OCFS2:

   ```
   # /usr/lib/ovs/ovs-makerepo /dev/sdb1 C "cluster root"
   ```

   The following is an example of NFS storage:

   ```
   # /usr/lib/ovs/ovs-makerepo fileserver.example.com:/OVS C "Cluster
   Root"
   ```

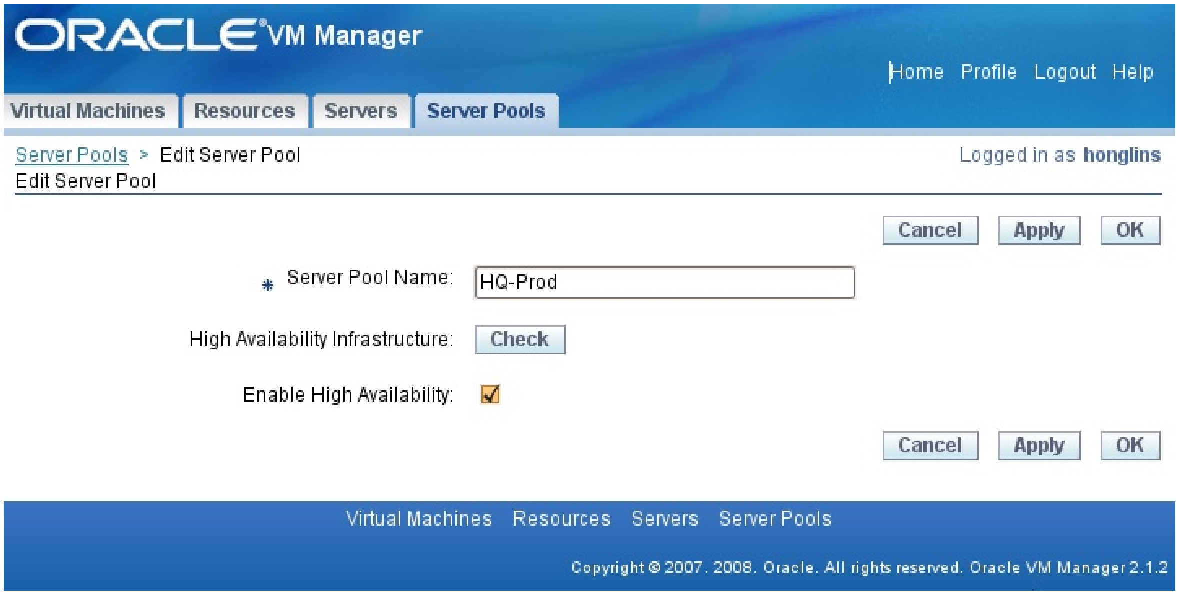
3. Check the cluster has been configured correctly and is running:

   ```
    # /usr/lib/ovs/ovs-cluster-check --master --alter-fstab
   ```

   The --master parameter should only be used on the Server Pool Master. The --alter-fstab parameter modifies the /etc/fstab file. A backup of the original /etc/fstab file is created in /tmp.

4. Log into Oracle VM Manager, enable HA for the Server Pool:

   - Click the "Server Pools" tab
   - Click the "Edit" button
   - Click the "Check" button to ensure that the HA infrastructure is working
   - Check "Enable High Availability" checkbox
   - Click "OK"



Server Pools > Edit Server Pool

Logged in as **honglins**

Edit Server Pool

* Server Pool Name: HQ-Prod

High Availability Infrastructure: Check

Enable High Availability: ☑

Copyright © 2007, 2008, Oracle. All rights reserved. Oracle VM Manager 2.1.2

Figure 5: Oracle VM Manager – Enabling HA for the Server Pool

See the Oracle VM Manager User Guide for additional information on creating an HA-enabled server pool.

Adding a new Server to the pool

You can add the newly installed Oracle VM server to the server pool from Oracle VM Manager. When the server pool is HA enabled, the Oracle VM Manager will automatically communicate with the master server agent to propagate the cluster configuration (e.g. /etc/ocfs2/cluster.conf) to each node of the server pool.

Optionally, you can run the command from the newly added Oracle VM server:

```
# /usr/lib/ovs/ovs-makerepo /dev/sdb1 C "cluster root"
```

or you run the command to propagate the cluster configuration from the master server:

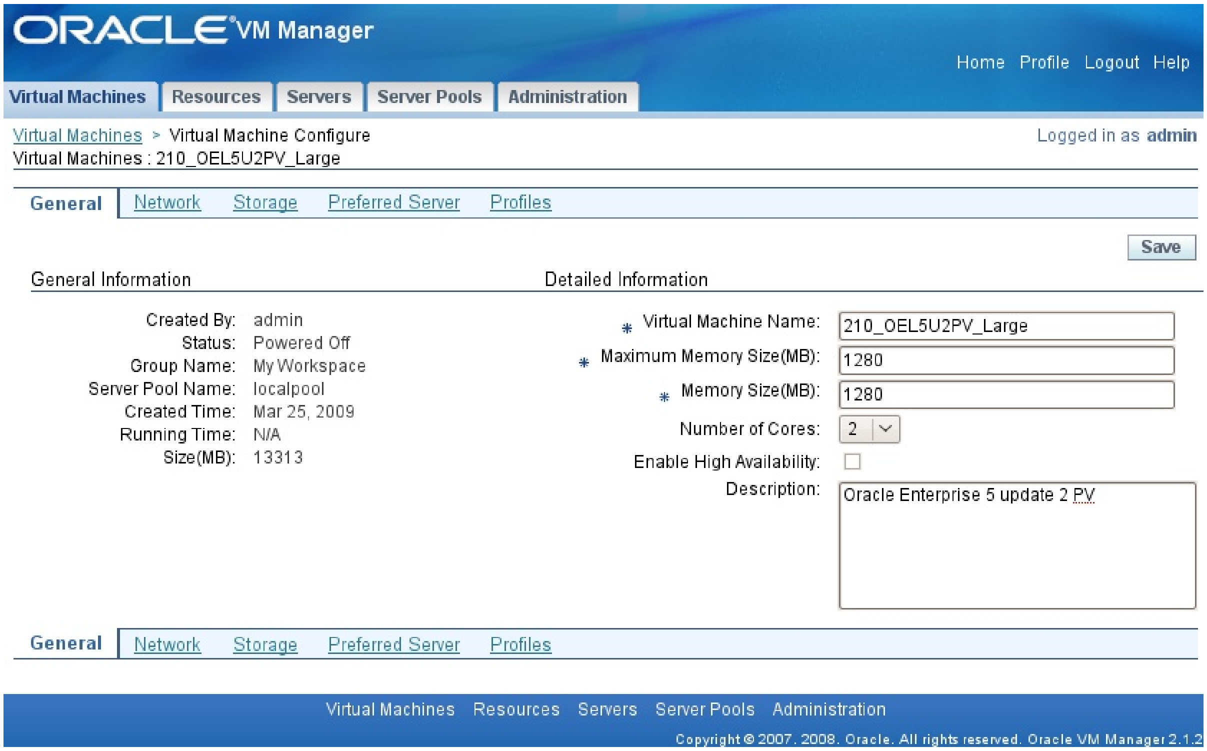
```
# /usr/lib/ovs/ovs-cluster-check --master --alter-fstab –-propagate
```

Please refer to the Oracle VM Manager User Guide for additional information to add a server into the server pool.

Enabling HA for the Virtual Machines

Once you have HA feature setup for the server pool, you simply mark the HA checkbox of the virtual machines from Oracle VM Manager.

- Click the "Virtual Machines" tab
- Select the VM and click "Configure"
- Check the "Enable High Availability" checkbox
- Click "OK"



Virtual Machines : 210_OEL5U2PV_Large

General Information

Created By:	admin
Status:	Powered Off
Group Name:	My Workspace
Server Pool Name:	localpool
Created Time:	Mar 25, 2009
Running Time:	N/A
Size(MB):	13313

Detailed Information

* Virtual Machine Name:	210_OEL5U2PV_Large
* Maximum Memory Size(MB):	1280
* Memory Size(MB):	1280
Number of Cores:	2
Enable High Availability:	□
Description:	Oracle Enterprise 5 update 2 PV

Figure 6: Oracle VM Manager - Enabling HA for the Virtual Machines

See Oracle VM Manager User Guide for additional information on configuring HA for virtual machines.

Special Considerations

- If the Oracle VM server has a local /OVS repository, you need to unmount it in order for the HA configuration to succeed.

  ```
  # umount /OVS
  ```

  If your local /OVS repository has resources (VM images, Templates, etc.) you want to use for the pool, you can refer to Metalink Note 756968.1 to convert the local /OVS repository to shared storage.

- When the network is not configured properly, the cluster configuration files such as /etc/ocfs2/cluster.conf won't be propagated correctly to each server of the server pool. For example, the loopback address (127.0.0.1) may show up in the /etc/ocfs2/cluster.conf for some servers. You should verify your network settings (DNS, routing table, etc.), replace the loopback address with the public IP address for each server and make sure that the ocsf2 cluster configuration file (/etc/ocfs2/cluster.conf) be the same across all the servers within the same pool. The following is a sample cluster.conf file:

```
node:

        ip_port = 7777

        ip_address = 192.168.100.100

        number = 0

        name = vmserver1.example.com

        cluster = ocfs2

node:

        ip_port = 7777

        ip_address = 192.168.100.101

        number = 1

        name = vmserver2.example.com

        cluster = ocfs2

cluster:

        node_count = 2

        name = ocfs2
```

- When troubleshooting HA issues for a VM server in the pool, you may have to stop and start related services. The sequence of the services should be strictly followed –

  When starting the services…

```
# service start o2cb

# service start ocfs2

# service start ovsrepositories

# service start ovs-agent
```

  When stopping the services, the sequence should be the opposite…

```
# service stop ovs-agent

# service stop ovsrepositories

# service stop ocfs2

# service stop o2cb
```

- If at any point of time you get heartbeat error, you can use the following command to manually stop the cluster services on the master as well as other servers in the pool. You can kill the heartbeat process, then manually start the cluster services in the above sequence.

  ```
  # ocfs2_hb_ctl -K -d /dev/sdb1
  ```

- When you enable HA for server pool using NFS as shared storage, the ovs-makerepo utility will create an OCFS2 heartbeat file for you. In case you need to create the heartbeat file manually for NFS shared storage, here's an example:

  ```
  # dd if=/dev/zero of=/OVS/.server_pool_hb bs=1M count=300
  ```

  Format the heartbeat file with ocfs2:

  ```
  # mkfs.ocfs2 -L server_pool_hb -N 16 -b 2k -C 4k --force
  /OVS/.server_pool_hb
  ```

- When configuring the VMs, you can set the preferred server policy as Manual or Auto. The default is Auto if no option is specified. In the Auto mode, when the virtual machine starts, Oracle VM Manager assigns a VM server with maximum resources available automatically to run the virtual machine. While you specify the Manual mode, you select one or more Virtual Machine Servers to include on the preferred servers list. The virtual machine then starts from and runs on the preferred server with the maximum resources available. Either Auto or Manual is fine. When choosing Manual and selecting a list of servers from the pool, you have the same flexibility as Auto, but you also have control on which server your VMs are moving to. However, extensive use of Preferred Server policies may also require you to have additional total capacity (more servers) in the pool to support all of your VMs since, in effect, the Preferred Server Policies reduce the extent to which all of the servers can be shared as hosts for all the VMs.
- Multiple Oracle VM Servers that share a common /OVS repository/filesystem may quickly result in insufficient disk space being available under /OVS. In that case, where possible, either grow the underlying /OVS partition, or use the /usr/lib/ovs/ovs-makerepo utility to incrementally create/add additional, shared repositories beneath the /OVS repository directory itself. For further details of Oracle VM Server repositories, please refer to the Metalink Note 794198.1 - Oracle VM: OVS Repository Design and Storage Considerations.

Conclusions

We've provided a step-by-step guide to plan and set up the Oracle VM environment so you can implement the guest VM HA feature to assure predictable, reliable, and accurate restarting of failed VM and Servers.

Leveraging its deep expertise with OCFS (Oracle Cluster File System), Oracle Real Application Clusters (RAC), and the associated Oracle Clusterware, Oracle has incorporated the OCFS2 cluster stack into Oracle VM as part of its infrastructure to, in effect, transform server pools into clusters from a high availability perspective; but the installation is transparent to the user and configuration is simple.

Furthermore, Oracle VM has a very tightly integrated HA management system to orchestrate everything from the VM failure detection all the way through to the successful restart of the guest VM, which is more reliable than competitive HA schemes which rely on simplistic network pings combined with time-outs to determine if a VM or Server has failed.

In summary, Oracle VM Guest VM HA functionality provides the following benefits:

- Auto-restart unexpectedly failed individual VMs on other servers in the server pool;
- Auto-restart all the guest VMs on another server in the server pool when an unexpected physical server failure occurs;
- Powerful cluster-based network- and storage heartbeat algorithms quickly and deterministically identify failed and/or isolated servers in the server pool to ensure rapid, accurate recovery;
- Sophisticated distributed lock management functionality for SAN, NFS, NAS, and iSCSI storage ensures VMs or entire servers can be rapidly restarted with no risk of data corruption.

For more information about Oracle VM and how customers are deploying it, please visit http://oracle.com/virtualization

References

- Oracle VM wiki http://wiki.oracle.com/page/Oracle+VM
- Oracle's Virtualization Blog http://blogs.oracle.com/virtualization
- Oracle VM Management Pack of Oracle Enterprise Manager

http://www.oracle.com/technology/products/oem/prod_focus/virtualization_mgmt.html

- Oracle VM Documentation http://www.oracle.com/technology/documentation/vm.html

- OCFS2 http://www.oracle.com/technologies/linux/ocfs2.html
- Oracle Technical White Paper: Oracle VM – Creating & Maintaining a Highly Available Environments for Guest VMs http://www.oracle.com/technologies/virtualization/docs/ovmha-whitepaper.pdf
- Oracle Technical White Paper: Oracle VM Manager Template http://oss.oracle.com/~wcoekaer/mgrwp.pdf
- Metalink Note 756968.1, Oracle VM: Converting from a Local to Shared OVS Repository
- Metalink Note 758313.1, Oracle VM: How to Check whether High Availability (HA) is enabled
- Metalink Note 740272.1, How to Enable Oracle VM 2.1.2 High Availability (HA) on an Existing Server Pool using NFS Storage
- Metalink Note 794198.1, Oracle VM: OVS Repository Design and Storage Considerations
- Metalink Note 757144.1, Oracle VM: The "Server Pool Master" Functionality / Role
- Metalink Note 467663.1, Oracle VM 2.1.x: Procedure for Reassignment of Server Pool Master Role to Another Server
- Metalink Note 781659.1, "/OVS" Filesystem is not Mounted after Rebooting Oracle VM Server
- Oracle VM: How to update an Oracle VM Manager,

  http://itnewscast.com/node/56
- Oracle VM: How to update an Oracle VM Server,

  http://itnewscast.com/node/55

White Paper Title
April 2009
Author: Honglin Su
Contributing Authors: Adam Hawley,
Kurt Hackel, Wiekus Beukes

Oracle Corporation
World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065
U.S.A.

Worldwide Inquiries:
Phone: +1.650.506.7000
Fax: +1.650.506.7200
oracle.com

Oracle is committed to developing practices and products that help protect the environment

Copyright © 2009, Oracle and/or its affiliates. All rights reserved. This document is provided for information purposes only and the contents hereof are subject to change without notice. This document is not warranted to be error-free, nor subject to any other warranties or conditions, whether expressed orally or implied in law, including implied warranties and conditions of merchantability or fitness for a particular purpose. We specifically disclaim any liability with respect to this document and no contractual obligations are formed either directly or indirectly by this document. This document may not be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without our prior written permission.

Oracle is a registered trademark of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.