Oracle VM 3 Experiences

Overview

"**Virtualization** is the creation of a virtual (rather than actual) version of something, such as an operating system, a server, a storage device or network resources."

At this time, most virtualization effort appears to be computer hardware / OS virtualization – that is the topic of this paper – however, network (SDN), storage (LVM/NFS) and memory (swap and page) are very much in play.

OS Virtualization is the core to 'Cloud Computing'. There are a number of different techniques to implement OS virtualization, including:

- Pure emulation of hardware within a supervising operating system;
- Pure sharing of existing hardware, supporting OS independence;
- Time-sharing of an existing operating system.

Oracle has two major 'VM' implementations, however they encompass all three of these areas:

- Oracle VirtualBox is an emulator that runs inside most popular operating systems;
- Oracle VM for x86 is a hardware sharing through the use of a software hypervisor;
- Oracle VM for SPARC implements the hypervisor in firmware to support multiple Solaris VMs.

OracleVM for x86, version 3.3, is based on XEN 4.3, which is by itself a suitable single-server virtualization platform. Oracle's extensions are very largely around the introduction of a supervisory VM Manager (OVMM).

Using that central OVMM, which includes a repository, it is possible to monitor and manage one or more pools of VM servers. Management includes identifying and allocating CPU, memory, network and storage; creating VMs from scratch or from templates; and migrating those VMs between servers.

Version 3.3 has introduced a radical change in infrastructure, allowing the VM Server some independence from the VM Manager; changing the repository to MySQL EE (only) with integrated backup; and implementing web services for the administration.

Virtualization Techniques

Virtualization of the operating system is actually quite mature. In the 1980s, much of the my computing was performed within the IBM mainframe VM/CMS environment, in which each user had a Virtual Machine which appeared to the user as their own computer.

Hosted VMs

Oracle VirtualBox (http://www.virtualbox.org), VMWare Fusion or VMWare Workstation, and Parallels are examples of hardware emulation. These require a host operating system, and use the operating system to provide services such as storage, memory, CPU and networking.

The benefits of these emulators include:

- They fit within an existing environment;
- They do not require replacing the current OS;
- Configuration of storage and memory is generally easy;
- Multiple operating systems can be run concurrently.

The drawback of these emulators is two-fold: they share the CPU with other operating system applications, and therefore have a significant performance impact; they often require an extra layer of network bridging and accompanied DNS/ DHCP/ gateway emulation, which may introduce complexity.

Firmware hypervisor, chroot, and shared kernel VMs

Linux KVM, Oracle VM for SPARC (formerly LDOM), and containers based on chroot are a very important class of VM. Especially in the case of Oracle VM for SPARC, the big advantage is that virtualization overhead is minimized.

Software Hypervisor

VMWare vSphere, XEN and Oracle VM for x86 are examples of software hypervisor based systems. With these, a very small and tailored kernel, or 'layer', is installed directory on the hardware. Often a separate Virtual Machine is installed specifically used to configure that hypervisor layer; that 'supervisory VM' is started automatically by the hypervisor. Additional VMs are started on demand by the hypervisor through commands from the supervisory VM.

Oracle VM is based on the XEN kernel which itself is based on Linux. For Oracle VM 3.3, Oracle has chosen XEN 4.3, which was released on July, 2013. The beta for

Oracle VM 3.3 was made available before XEN version 4.4 was released in March 2014, and Oracle has chosen to retain XEN 4.3 as the basis for Oracle VM 3.3.

Enterprise operations

Single server considerations

XEN can be used to manage Virtual Machines on a single computer.

It must be aware of the CPU configuration and provide scheduling to allocate time to each VM, including the supervisory VM. Some hypervisors support CPU overcommit, which means allowing more CPU cores to be allocated to VM than are physically available – the implication is that the hypervisor must totally suspend VMs in order to allow others to run. XEN does support overcommit (aka oversell) of CPU core, but this is a poor practice.

It must be aware of the available memory, including total virtual memory (real memory + swap), and must be able to provide that memory to each VM when it is running. Some hypervisors support memory overcommit. With XEN it is possible, although generally considered difficult. The challenge is to keep the total VM 'real memory' requirements to within physical real memory, and the total VM 'virtual memory' requirements within total virtual memory. Improper management here can have some unexpected performance hits when the VM thinks it is getting real memory but that is being served by swap from the hypervisor.

Networking becomes an interesting challenge. The host hardware will often have one or more network cards (NICs), however each VM will need access to the NICs under their own IP address ranges. Techniques such as Virtual IP, VLans, Trunking, and even local switching need to be addressed. XEN 4.3 has support for Open vSwitch, which helps address many of the related issues.

In general, the hypervisor does not present removable storage such as CD, DVD or USB, in prt because coordination of these removable resources becomes a challenge between multiple running machines. As a result, CD or DVD images are generally stored as ISO files made available to the VMs from a storage pool.

When using a single standalone server the environment can use local disk or networked disk in the form of NFS or iSCSI block devices presented to the hypervisor and managed through the supervisory VM. The hypervisor needs to present these local or networked disk consistently to the VMs to be used for the installed OS, to store ISOs, and to be used as a pool for the VMs to use for applications.

(In addition, each VM can access iSCSI and NFS storage directly.)

Multi-server considerations

One of the big perceived advantages of virtualization is the ability to move VMs between servers, based on resource utilization as well as to avoid downtime due to hardware maintenance.

In order to achieve this, there are two important changes from the single server considerations mentioned above: the networking must be easily migrated between servers; the disk must be accessible from all servers that may be used to host a VM.

Network migration is accomplished by virtual IPs and, in XEN 4.3, through the use of Open vSwitch (http://en.wikipedia.org/wiki/Open_vSwitch).

Disk accessibility requires that all disks used in the VMs that may be moved are networked disks, either through NFS or iSCSI. Most environments do not support direct SAN access using HBAs.

Oracle VM 3.3 uses a centralized MySQL repository and has a centralized JavaEE-based management utility (with an ADF-based console) that can capture, store and manage both the network and the storage configuration. Some of the changes to the OVMM are described in the document http://www.oracle.com/us/technologies/virtualization/oraclevm/oracle-vm-whats-new-ds-2232913.pdf

Experiences with Oracle VM 3.3

Installation of Oracle VM Server

Installation of Oracle VM Server can be accomplished by downloading the ISO from https://edelivery.oracle.com/linux , creating and inserting the resulting DVD, and following the prompts. Using PXE Boot, or Open Stack, it is also possible to install the system automatically.

Most of the installation is automatic, except for the allocation of IP addresses, and there is very little that can be tuned. NTP is highly recommended.

One network access, the one to be used for VM administration, should have a static IP address.

At the end of the installation, a small Oracle Linux 6u4 JEOS supervisory VM is installed which can be accessed via SSH to manage the hypervisor. The Oracle VM Agent is pre-installed in the supervisory VM.

Installation of Oracle VM Manager

The Oracle VM Manager has many changes with the introduction, most of which are seen post-installation.

Oracle VM Manager is simply an application that is installed into an existing Linux environment. Any Oracle or Red Hat Linux 5u8 or higher or 6u3 or higher seems to work without any issue.

The hosting OS should use the same NTP services as the VM Servers.

The software ZIP file is downloaded from the same location as the VM Server above, and **must** match the VM Server version.

There are significant differences between the VM Server 3.2 and VM Server 3.3 and the agents – the VM Manager 3.3 is capable of communicating with the VM Server 3.2 agent, specifically to assist upgrades but should not be used to manage the VM Server 3.2 for any significant period.

Once downloaded and unzipped, installation is initiated using 'runInstaller'. There is NO GUI installation. A command line menu (install/upgrade/abort) is presented, the administration password is read form the command line, system configuration is checked, and the installation proceeds to install MySQL, Java, ADF and configure and deploy the database schema and application.

At the end of the installation, a summary is presented, which contains very important information.

Configuration of Oracle VM Server, via Manager

Storage

Post installation, any iSCSI storage or NFS storage must be visible to at least one VM Server. The storage does not need to be visible to the VM Manager, and I believe it should NOT be visible to the VM Manager to avoid potential security issues or corruption.

Once verified as visible (for example, the exports file on the NFS server), the storage is discovered through the Oracle VM Manager.

Any VM Server can be designated as the primary handler for that storage, and the Oracle VM Manager will choose that as the preferred path to identifying and reporting on that storage. Servers that can be used to manage the storage are known as 'Admin Servers', and it is permitted (and recommended) to list all servers in that role.

For NFS storage, up to two (and only two) servers can be designated as 'refresh servers' to mount and refresh contents of any file system. For SCSI storage, additional steps to set up and secure the iSCSI initiators, as described at https://docs.oracle.com/cd/E50245_01/E50248/html/vmgsg-storage-san-discover.html, are required.

We have found that once identified by Oracle VM Manager and recorded in the repository, the storage is tagged and locked for a specific purpose. Using that storage in a rebuilt Manager or moving it generally required reformatting (using dd) and potentially searching for the entry in the repository and eliminating it. Hopefully Oracle will provide a 'deallocate' capability at some time.

Network

Oracle VM uses several VLANs to accomplish networking:

- Server Management,
- Live Migrate,
- Cluster Heartbeat,
- Virtual Machine, and
- Storage

The Virtual Machine network is the one for the application and users. It is recommended that this be on a 'user' subnet and if possible a user NIC.

Server Management, Live Migrate, and Cluster Heartbeat are administrative, and should be on separate admin subnet or subnets. Based on experience, Live Migration can seriously load down the subnet and can cause Heartbeat latency issues. If possible these should have their own network cards, however they can be combined if needed.

iSCSI and NFS Storage usually have heavy bandwidth requirements and should be on separate network card and subnet.

Server Pools

When multiple machines are to be used to support migration or VM movement, these machines should be in a single pool. Oracle VM Manager supports the use of many pools, all controlled through one console and recorded in one repository.