High Availability DB on the Cheap using Oracle VM, Oracle Linux, ASM, and DataGuard

William Petro

ABSTRACT

This paper will identify the business drivers to implement an Oracle database private cloud using Oracle VM and the technology benefits from Oracle VM implementation

TARGET AUDIENCE

The target audience for this paper is Oracle DBAs, Cloud Architects, Technology Management interested in a combination of High Availability, and Cost Containment.

EXECUTIVE SUMMARY

The presentation will cover Hardware level High Availability on the Cheap (relatively speaking) using Oracle VM, Oracle Linux, ASM, and Oracle Enterprise Edition. It will illustrate a successful enterprise level migration of Oracle databases from Windows-based bare metal servers to fully virtualized Oracle Linux/Oracle VM Server Virtual Machines. The paper is not intended to be an in depth technical review of the technologies. It is a high level use case of the viability of an implementation.

Technical detail is included for two features that were important to the implementation, which may not be apparent to someone new to Oracle VM technology. One is physical disk access bypassing file virtualization for database files to significantly improve performance. The other is CPU Pinning to enable partial server hardware utilization supporting the Oracle CPU based licensing model to contain license costs.

Additionally, the appendices reference a Demo environment build that is available for evaluation purposes.

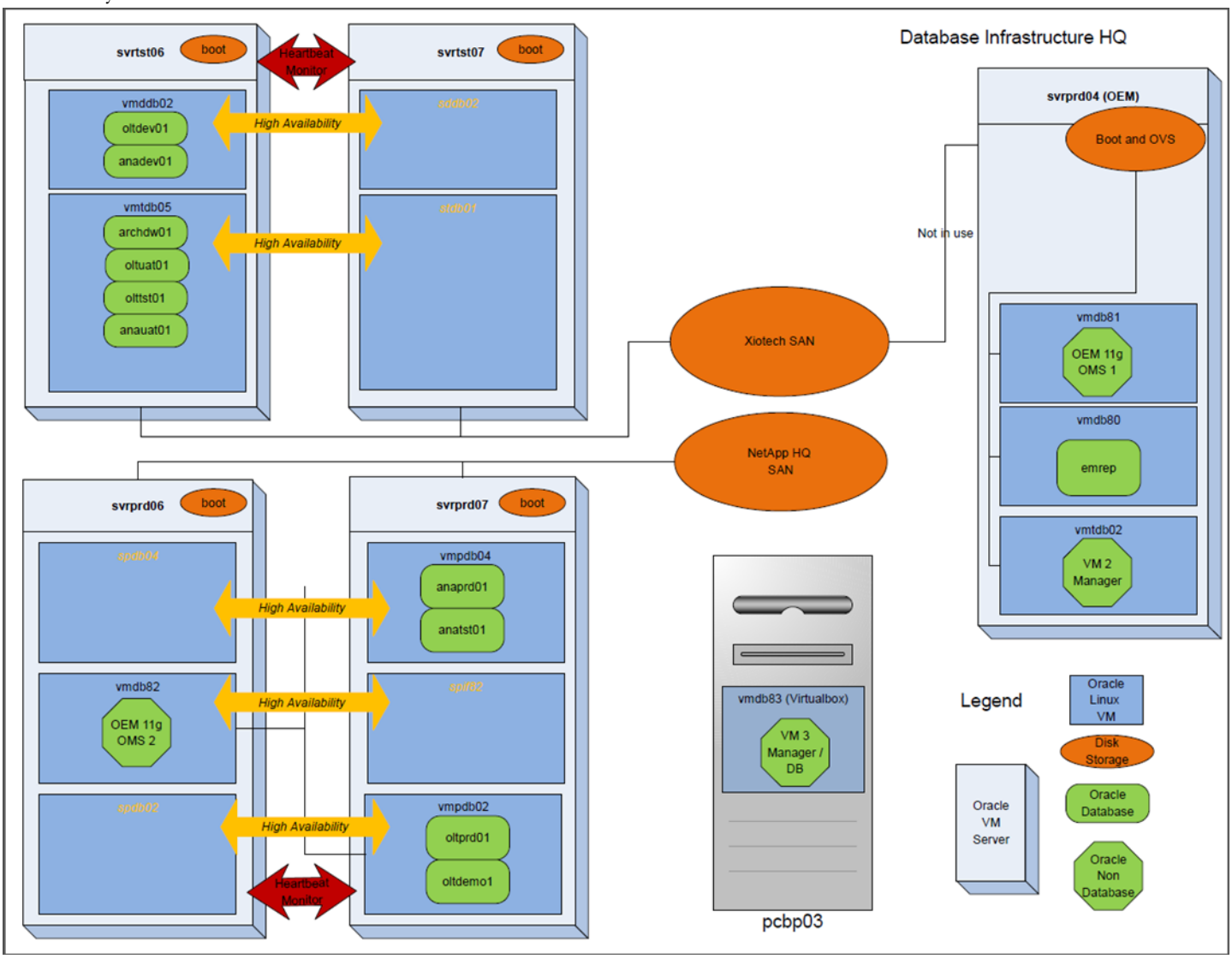
BACKGROUND

The primary drivers for the virtualization effort were Oracle license cost savings, and database availability. The company wanted to defer increased Oracle CPU based license costs, while implementing additional hardware redundancy capabilities during a hardware refresh. The entire full lifecycle database infrastructure was migrated from Windows based Oracle 10g with DataGuard to HA clustered virtualized Linux based Oracle 11g with ASM in one step with minimal downtime. A migration to new servers and SAN hardware was also included in the mix.

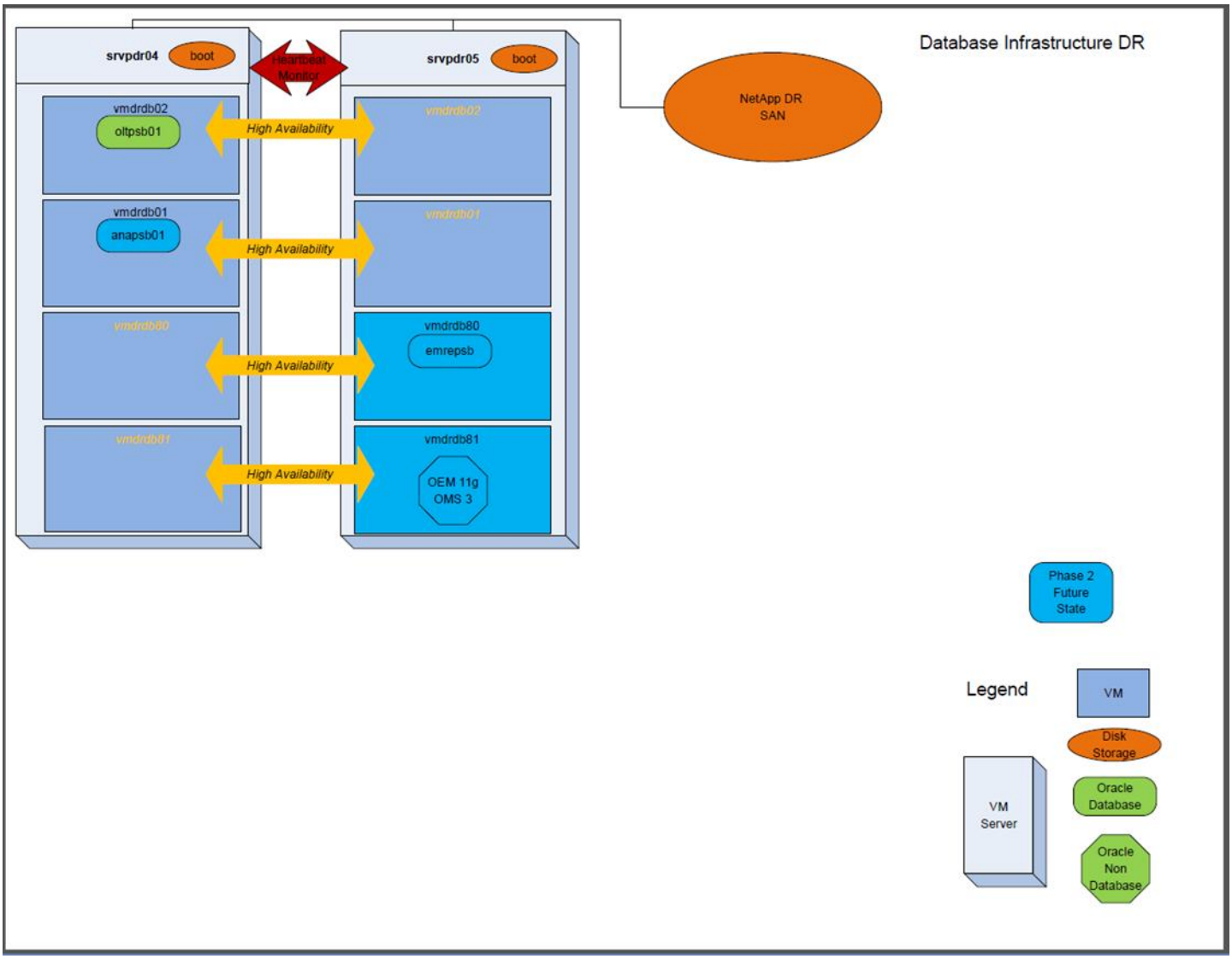
Oracle VM features available include CPU isolation (for Oracle license savings), database/VM live server migration using Oracle VM server clustering (for minimal downtime from hardware failure/maintenance), and VM templating/cloning (for consistent/efficient environment creation).

The following diagram illustrates the virtualized end state HQ database infrastructure consisting of two physical servers and a SAN for Production. Two older servers and a SAN (formerly Production) were used to support Development and Test environments. OEM is hosted on older OVM 2.x technology as the hardware was too old to support OVM 3.x. This was a legacy first step, moving to Oracle VM from VMWare. VM Manager is also shown running on a PC using an Oracle VM

Manager VM template installation. VM Manager was moved to the clustered Production environment after the initial migration with the intention of doing the same with the OEM environment. So the intended Production end state is the two servers for Production and two for Test. At any point in time all Test VMs are running on one physical server and all Production VMs are also running on a single physical server. This is done to conserve on license costs. The "other" server capacity is intended to support OEM, RMAN Recovery Catalog, and VM Manager, as well as server level hardware redundancy for the database server VMs.



The DR Site consists of two physical Servers and a SAN mirroring the Production Site. The databases in the DR site are all Oracle DataGuard Physical Standby databases. This provides DR coverage for HQ catastrophic failure, a major worry being hurricanes in the central Florida HQ location. As such, the DR facility is hosted geographically far enough away that both locations should not be impacted by a single storm.



TECHNICAL DISCUSSIONS AND EXAMPLES

The business requirementsfor the implementation were that data loss is unacceptable. This is covered by full database redundancy between HQ and DR, which is delivered by Oracle DataGuard Physical Standby database implementation. This is further augmented by daily database backups to tape in HQ with tapes archived offsite.

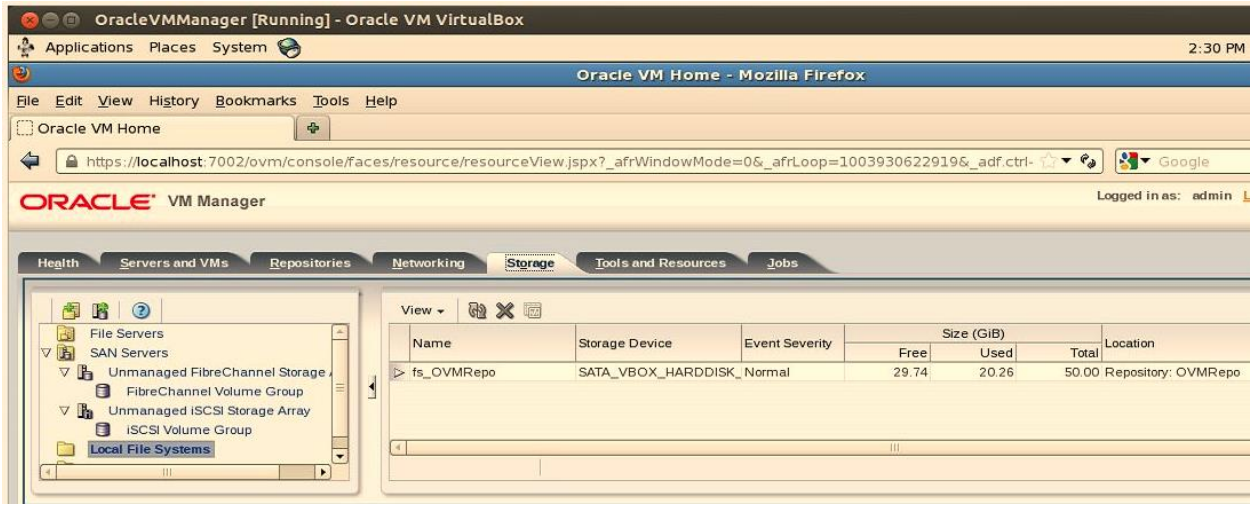
Any unscheduled downtime is bad for client and bank confidence of the company. Unscheduled downtime greater than 20 minutes jeopardizes deliverables and must be avoided if at all possible. Clients can be fined by state governments if payments are not processed the same day; this must be avoided at all possible cost. Full internal hardware redundancy in each server, and a second server clustered, protects against a lingering downtime due to server hardware failure. The one potential point of failure is the single SAN in HQ. This is protected against by the DR Site.

Database and application optimal performance is less important than availability and consistency. A small performance penalty due to virtualization would be acceptable, but was not encountered due to the full hardware refresh in conjunction with the move to virtualization and the utilization of direct physical disk access for database files provided by Oracle VM Server. This physical disk access method avoids what can be a significant performance overhead in place of using virtualized

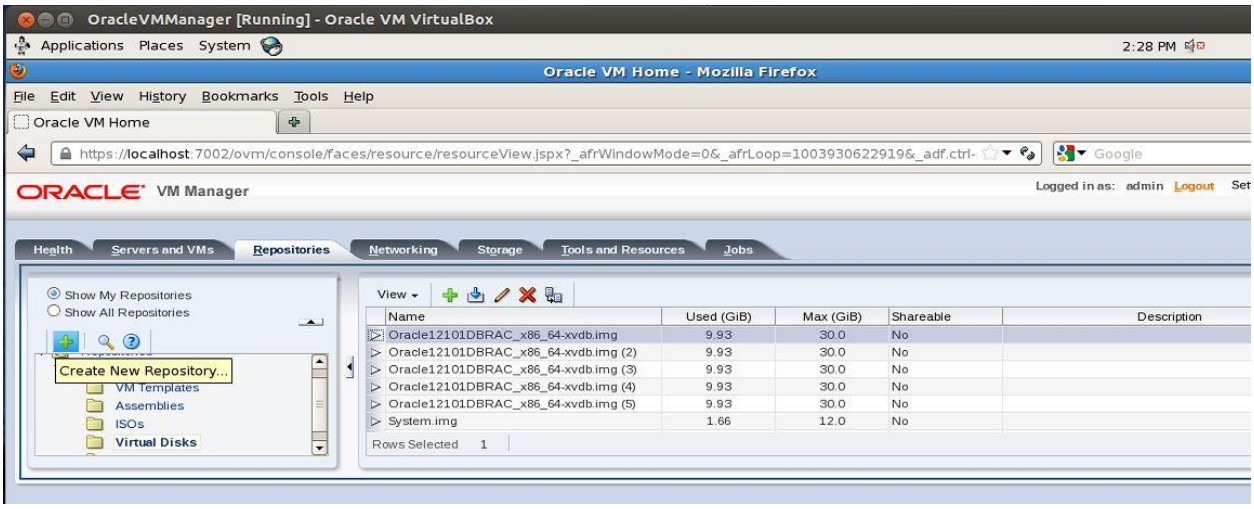
files. Virtualized files are suitable for the OS and Database software locations, but the actual database data when tested with virtualized files ran approximately 50% slower than with physical device access.

The following illustrations will overview the implementation of the physical devices for database data.

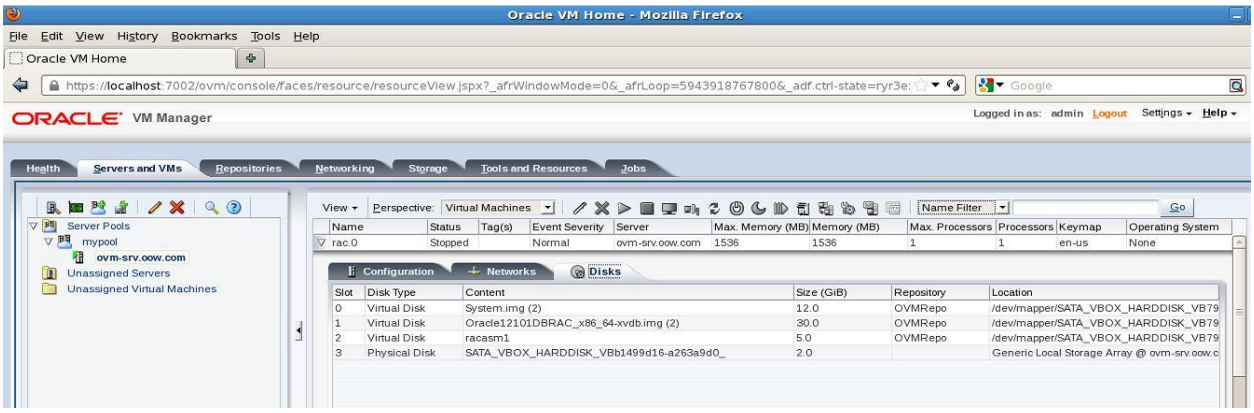
The screen below shows the Oracle VM storage repository where virtualized files are stored. There can be multiple repositories each containing multiple files.



This screen shows virtual disks that are stored in the repository. The sizes are misleading due to sharing and thin provisioning. These files store the Operating System and Oracle Software file systems which are less critical to performance.



This shows what an individual VM sees for storage devices. Notice the last row shows as a Physical disk in the first column and that it is not part of a repository in the 5th column. This means that the device is not managed by the virtualization layer, it is simply passed through to the VM for direct access. It can be shared by multiple VMs, just as a virtual disk can; this makes it very suitable for ASM use.



Another business driver for the virtualization implementation was that Oracle licensing costs must be contained. We were able to upgrade hardware and add additional redundancy without increasing our Oracle license footprint, except for the Oracle VM and Oracle Linux support licenses. These support licenses actually resulted in significant savings over purchasing VM Ware Server license for at least 4 servers, not to mention that the Oracle license costs would have skyrocketed under the VM Ware license model. Additionally, all Oracle technology was now at supported version levels.

The technology in Oracle VM Server that allows for partial licensing of servers with additional CPU is called CPU pinning. It enables the database server VMs to be restricted to using specific CPU cores. This technology does limit Oracle VM capabilities a bit as you are not able to use live migration capabilities moving VMs between clustered servers while they are running, but this is a capability that we could definitely live without. This is what Oracle RAC is designed to cover, but at significant license cost. It is also not really a great idea to try to move a running database instance between servers anyway. Timing issues are likely during the move while memory realms are exchanged, though during evaluation, instance moves did actually work – most times.

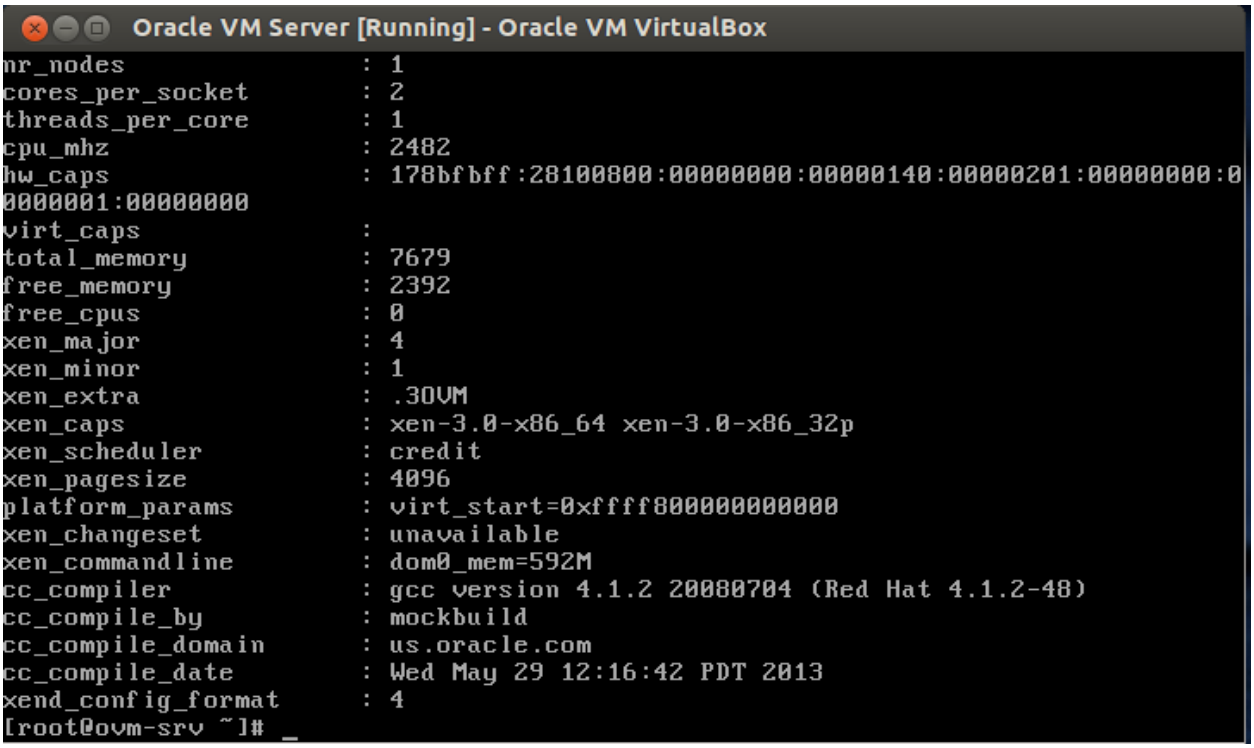
CPU Pinning in OVM 3.1 binds a VM to physical CPUs (actually threads). This supports CPU based licensing models. It also can provide some isolation of VMs from each other at the CPU level. This must be configured for each VM that will be pinned. You can use the Oracle VM command line utilities (ovm_vmcontrol) to set up hard partitioning, which MUST be downloaded separately via Oracle Support patch ID 13602094. Alternately, pinning can be implemented by modifying the vm.cfg manually on the VM Server following syntax:

```
vi /<repository file system path>/VirtualMachines/<virtual machine ID>/vm.cfg
```

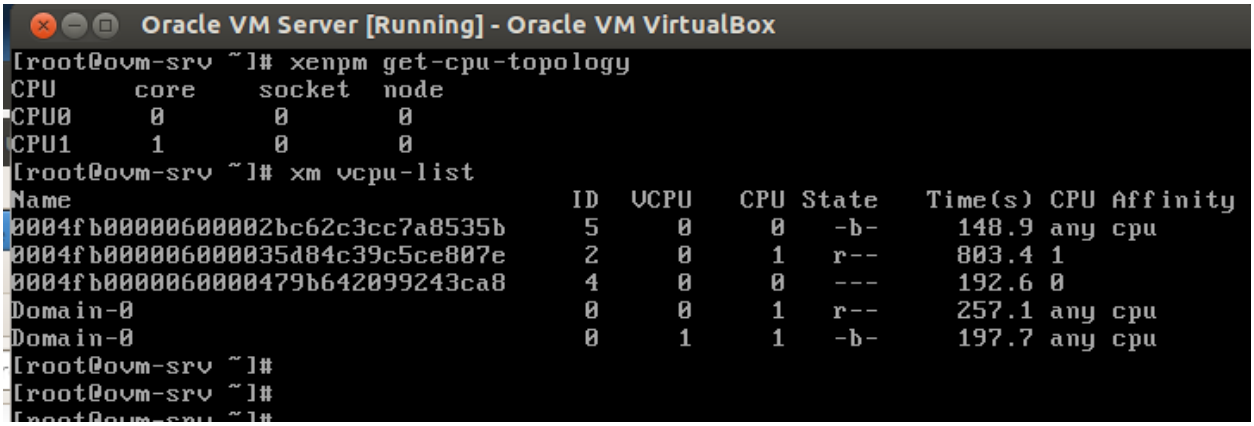
Pinning is disabled during VM migrations between hosts, so it MUST be re-enabled via the VM Utilities or manually, after migration.

The following illustrations will overview CPU Pinning which is documented by Oracle here:
Oracle White Paper - Hard Partitioning With Oracle VM Server for x86 - http://www.oracle.com/technetwork/server-storage/vm/ovm-hardpart-168217.pdf

This screen shows the hardware that Oracle VM Server sees. In the first three lines it shows one CPU socket, with 2 cores single threaded.



The output from the first command below is an alternate view of the CPU configuration presented to the VM Server. The output from the second command show that there are 3 VMs running by looking at the name column. Domain-0 is the VM Server management domain and can be skipped for this discussion. The last column "CPU Affinity" shows where pinning is in effect. The first VM can use all CPU threads. The second VM can use only CPU1 and the third can use only CPU0. Each VM can be assigned more than one CPU thread. It should be kept in mind that for systems supporting two threads per CPU VMs should be configured so that both threads are assigned to a the VM for full CPU utilization.



APPENDICES

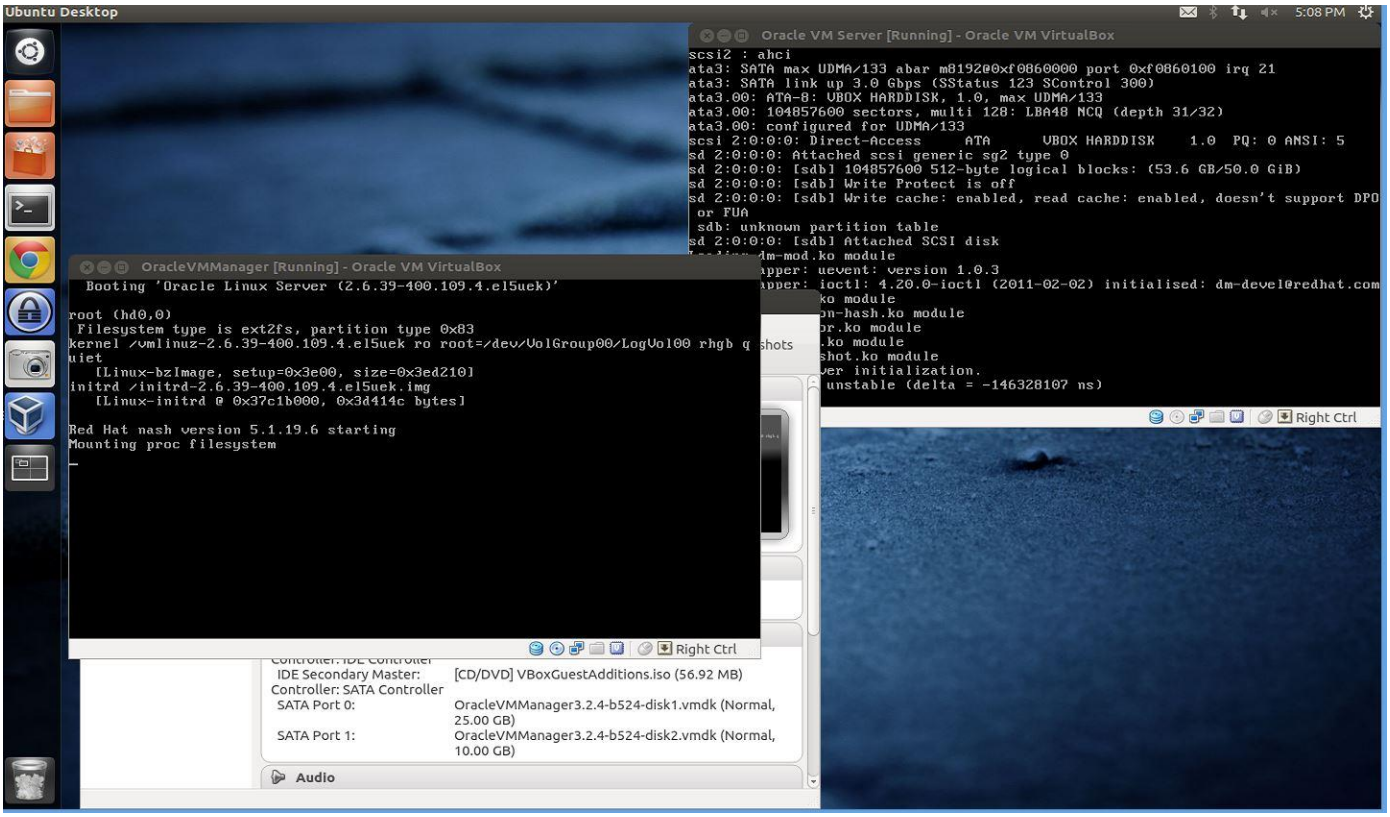
I wanted to have some screen shots available (above) to illustrate several technical points about Oracle VM, and have been curious as to the viability of running Oracle VM Server under Oracle Virtualbox for Demo purposes. I found a Lab targeting installation of a 4 node Oracle 12c environment from Oracle VM Server templates. This is hosted under Oracle VirtualBox using VirtualBox VM templates for the Oracle VM Server and VM Manager.

The lab steps are pretty well documented here:
How To Deploy a 4 Node RAC Cluster Using Oracle VM Templates - Olivier Canonge -
http://www.oracle.com/technetwork/systems/hands-on-labs/deploy-rac-ovm-cluster-2101019.html#Prep

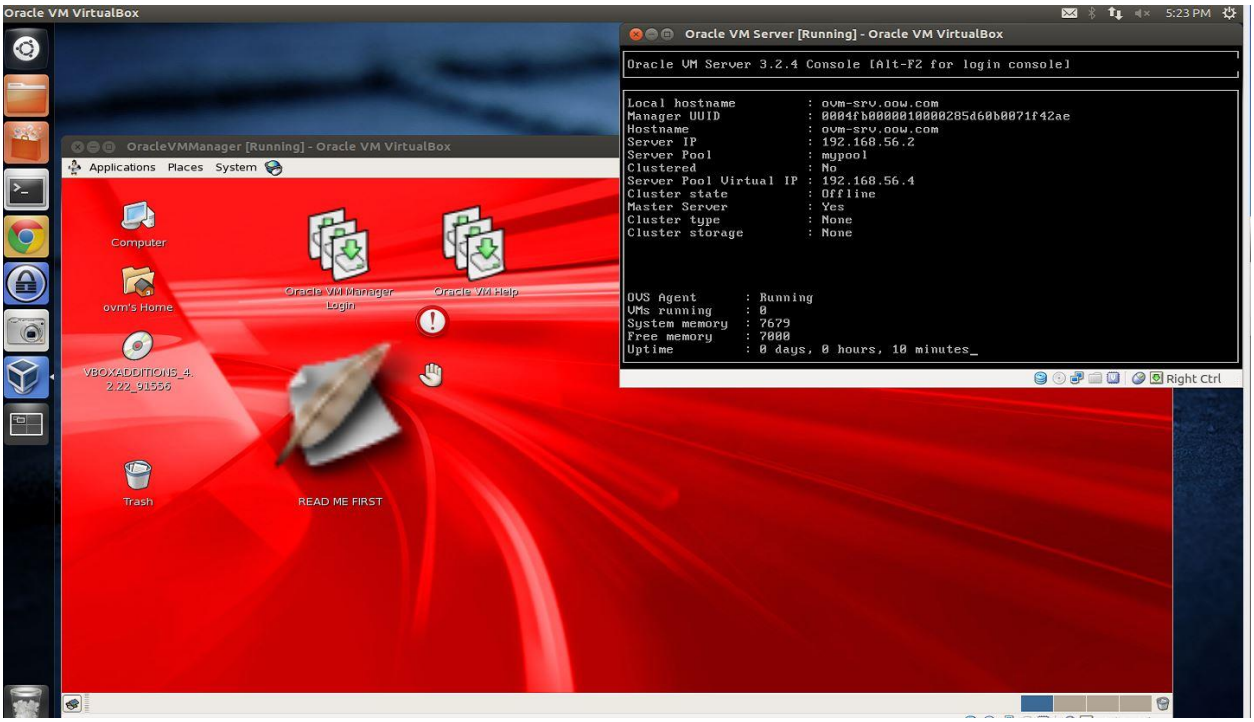
Unfortunately, the only PC available to me with over 8 GB memory is my work laptop that has a 32 Bit Windows Corporate image. which only lets me use less than 4 GB of the memory. So I installed Ubuntu 64 Bit WUBI to host VirtualBox and enable use of the 12 GB of memory. WUBI supplies a full featured Ubuntu dual boot capability and is easily removed in a single step via the Windows installer. The WUBI install is beyond the scope of this paper, but details can easily be found via your favorite web search tool.

This laptop also runs on SSD storage for a significant performance boost, but still strained significantly running the VM Manager and VM Server, especially with the 4 VM Server VMs running. I had limited success with the actual Oracle 12c RAC install which ran for 8 hours with very high CPU utilization, resulting in a very hot laptop. I probably should have just went with a 2 node RAC build as Oracle states a 16 GB memory minimum requirement for this exercise. Your mileage may vary. For the purposes of this paper and associated presentation I would call the Demo installation a success though. I have supplied some screen shots of my working system for illustrative purposes below.

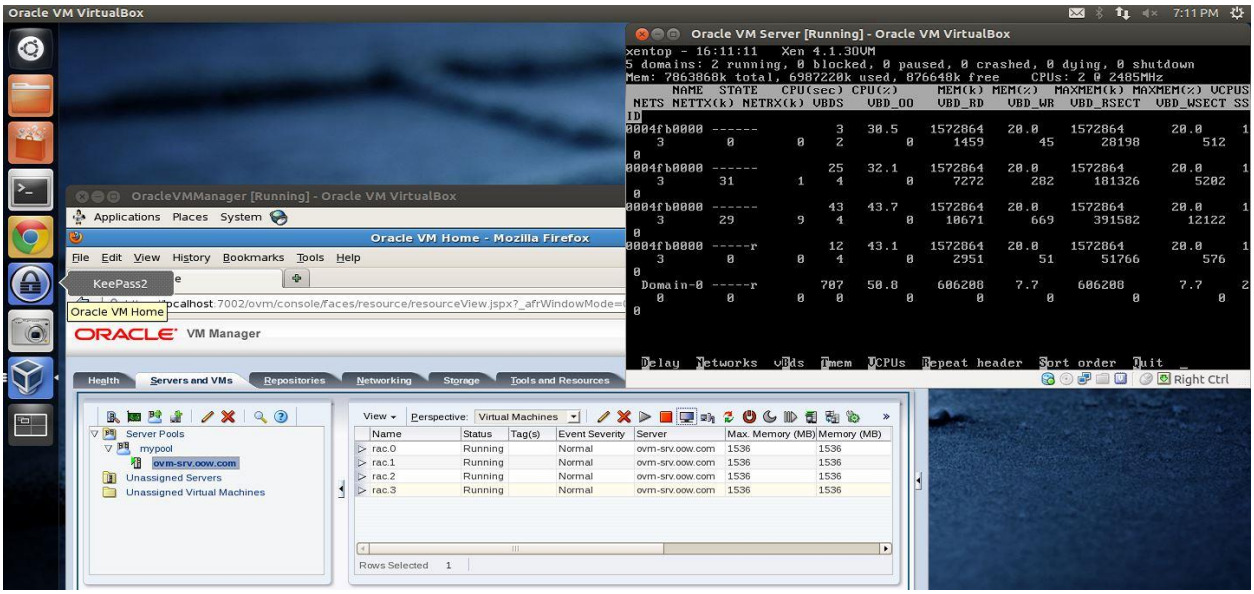
This illustrates the startup of the Oracle VM Manager and VM Server demo templates under Ubuntu hosted Virtualbox



This is the Oracle VM Manager and VM Server VirtualBox templates after startup



This is the actual VM Manager interface and 4 Oracle 12c RAC VMs running. This is the basis for screen captures elsewhere in the paper and presentation.



REFERENCES

Virtualizing Oracle Oracle VM and Vmware A Technical Deep Dive - George Trujillo, Charles Kim – IOUG Collaborate 2013

BUILD A RAC DATABASE FOR FREE WITH VIRTUALBOX - A STEP BY STEP GUIDE - Chris Ostrowski, Avout - IOUG Collaborate 2013

How To Deploy a 4 Node RAC Cluster Using Oracle VM Templates - Olivier Canonge - http://www.oracle.com/technetwork/systems/hands-on-labs/deploy-rac-ovm-cluster-2101019.html#Prep

Oracle Data Sheet - ORACLE VM SERVER FOR X86 VIRTUALIZATION AND MANAGEMENT - http://www.oracle.com/us/technologies/virtualization/ovm-server-for-x86-459312.pdf

Oracle White Paper August 2011 - Oracle VM 3: Application-Driven Virtualization - http://www.oracle.com/us/technologies/virtualization/ovm3-app-driven-459334.pdf

Oracle White Paper - Hard Partitioning With Oracle VM Server for x86 - http://www.oracle.com/technetwork/server-storage/vm/ovm-hardpart-168217.pdf

Oracle VM Server for x86 FAQ - http://www.oracle.com/us/026952.pdf

Oracle VM Users Guide - http://docs.oracle.com/cd/E35328_01/E35332/html/index.html

Oracle VM Templates - http://www.oracle.com/technetwork/server-storage/vm/templates-101937.html

Oracle VM Cost Calculator - http://www.oracle.com/us/media/calculator/vm/vm-home-2132015.html

Oracle Exalytics Administration Seminar – Oracle Education Class Notes