



SOLUTION
READINESS

Oracle Databases on vSphere Workload Characterization Study

Conducted by

Cisco, EMC and VMware

December, 2010

© 2010 VMware, Inc. All rights reserved. This product is protected by U.S. and international copyright and intellectual property laws. This product is covered by one or more patents listed at <http://www.vmware.com/download/patents.html>.

VMware is a registered trademark or trademark of VMware, Inc. in the United States and/or other jurisdictions. All other marks and names mentioned herein may be trademarks of their respective companies.

VMware, Inc
3401 Hillview Ave
Palo Alto, CA 94304
www.vmware.com

Contents

1. Executive Summary	4
2. Introduction	5
3. Virtualization of Oracle Databases.....	6
4. Test Methodology.....	7
4.1 Large Scale Order Entry Benchmark Kit –Swingbench.....	7
4.2 Test Setup/Infrastructure	7
4.3 Summary of Test Scenarios.....	9
5. Test Scenarios and Results	10
5.1 Single Virtual Machine – Scale Up Test	10
5.2 Multiple Virtual Machines – Scale Out Test.....	14
5.3 Live Migration of Virtual Machine (vMotion).....	18
5.4 VMware vCPU Hot Add	20
6. Conclusion	22
7. References.....	23
8. Disclaimers.....	24
Appendix A: Virtual Machine Configuration.....	25
Appendix B: 8 vCPU/32GB Virtual Machine Memory Split	28

1. Executive Summary

Oracle database deployments can generate significant server sprawl due to the need to provision separate systems for development, quality assurance, test and production environments. As a result, Oracle database implementations can increase exponentially, and even small implementations can have a relatively large IT footprint. In a typical deployment, the database server environment is hosted on dedicated physical systems that are not fully utilized much of the time. VMware vSphere enables the deployment of multiple virtual machines running different Oracle database solutions on the same physical server, resulting in increased CPU utilization and reduction in overall server requirements.

This paper describes a workload characterization study of virtualized Oracle databases (version 11g Release 2) on VMware ESXi 4.1. Results show that virtualized Oracle databases on vSphere can handle intensive OLTP loads in single and multi-virtual machine scenarios. The findings show:

- A single Oracle database virtual machine linearly scaled from 2 to 8 vCPU - the Transactions Per Minute (TPM) increased in a nearly linear fashion as more vCPUs were added.
- A single ESX server can manage the workload of multiple Oracle databases in separate virtual machines. Consolidation is achieved both in terms of CPU and storage. Total vCPU count can exceed the number of physical cores and multiple databases can share the same VMFS datastores.
- The results were *not* achieved by over-allocating memory resources. The total memory of the virtual machines was less than or equal to the physical memory available on the ESX host machine.
- When running an 8-vCPU virtual machine (which in the test environment here has more vCPUs than available cores on a NUMA node) ESX split the memory across the NUMA nodes. (See Appendix B.)
- A live migration of a virtual machine running an Oracle database OLTP load, using VMware vMotion™, experienced zero transaction errors and no loss of data integrity, though a temporary dip in throughput was observed during the migration.
- Dynamic Hot Add of vCPU helps in dynamically scaling the processing resources, improving overall performance and throughput of a saturated virtual machine running an Oracle Database Server processing an OLTP load and can be accomplished with zero down time.

2. Introduction

The VMware vSphere™ 4 hypervisor contains numerous performance-related enhancements and has improved resource management capabilities that enable effective consolidation of multiple Oracle Database server virtual machines (VMs) on a single host without compromising performance or scalability. Greater consolidation can significantly reduce the cost of physical infrastructure and licensing of Oracle Database server, even in small-scale environments.

This paper describes a workload characterization effort conducted by Cisco, EMC and VMware to profile an OLTP workload against virtualized Oracle databases (version 11gR2) on VMware vSphere ESXi 4.1 in various single and multi-virtual machine configurations. The following tests were conducted:

- Scale up – Scaling up a virtual machine is the process of adding virtual CPUs (vCPUs) to increase a virtual machine's compute power. With vSphere, your virtual machine can easily take advantage of multi-core processors with no special settings or configuration beyond building up the virtual machine with more vCPUs.
- Scale out – Scaling out is the process of adding more virtual machines on the host which enables consolidation of multiple virtual machines, each running a single Oracle database, on a single ESXi host.
- Live migration via VMware vMotion™ and vCPU Hot Add of a virtual machine running an Oracle database while processing an OLTP workload.

OLTP workloads were created using the Oracle SwingBench kit - this is comprised of scripts and load drivers that generate business transactions which simulate large scale order entry OLTP loads that are I/O-intensive.

3. Virtualization of Oracle Databases

VMware vSphere is the industry's most trusted virtualization platform, transforming datacenters into a dramatically simplified cloud infrastructure and enabling the next generation of flexible, reliable IT services. VMware vSphere virtualizes and aggregates the underlying physical hardware resources across multiple systems and provides pools of virtual resources to the datacenter. Oracle databases are excellent candidates to take advantage of these features and they have been shown to run successfully on vSphere, providing significant scalability, availability and performance benefits. Virtualizing database workloads on vSphere significantly reduces the number of physical systems your organization requires while achieving more effective utilization of datacenter resources. Customers realize tangible savings from this consolidation along with operational cost savings from reduced datacenter floor space, power, and cooling requirements.

VMware Virtual Symmetric Multi-Processing (SMP) enhances virtual machine performance by enabling a single virtual machine to use multiple physical processors simultaneously. Virtual SMP enables virtualization of processor and resource-intensive enterprise applications such as databases and ERP. It moves processing tasks between available processors to balance the workload and more fully utilize all processing power.

VMware vMotion technology enables an entire running virtual machine to be instantaneously moved from one server to another. This migration is live, meaning that the guest OS and the Oracle database continue to run without disruption of any batch or online activity. The entire state of a virtual machine is encapsulated by a set of files stored on shared storage, and the VMware VMFS cluster file system allows both the source and the target ESX host to access these virtual machine files concurrently. The active memory and precise execution state of a virtual machine can then be rapidly transmitted over a high speed network. Because the network is also virtualized by ESX, the virtual machine retains its network identity and connections, facilitating a seamless migration process.

The VMware Hot-add vCPU feature enables CPU capacity to be added to virtual machines running Oracle databases when needed without disruption or downtime.

4. Test Methodology

4.1 Large Scale Order Entry Benchmark Kit –Swingbench

The SwingBench kit is comprised of scripts and load drivers that generate business transactions which simulate large scale order entry OLTP loads that are I/O-intensive. It is a TPC-C like workload generator that includes a data generator tool which was used to create larger schemas that generate much higher levels of I/O (larger index lookups). Data was loaded to grow the database size to 1TB. This workload has a read/write ratio of 60/40.

More information about the benchmark kit is available at <http://www.dominicgiles.com/largesoe.html>.

4.2 Test Setup/Infrastructure

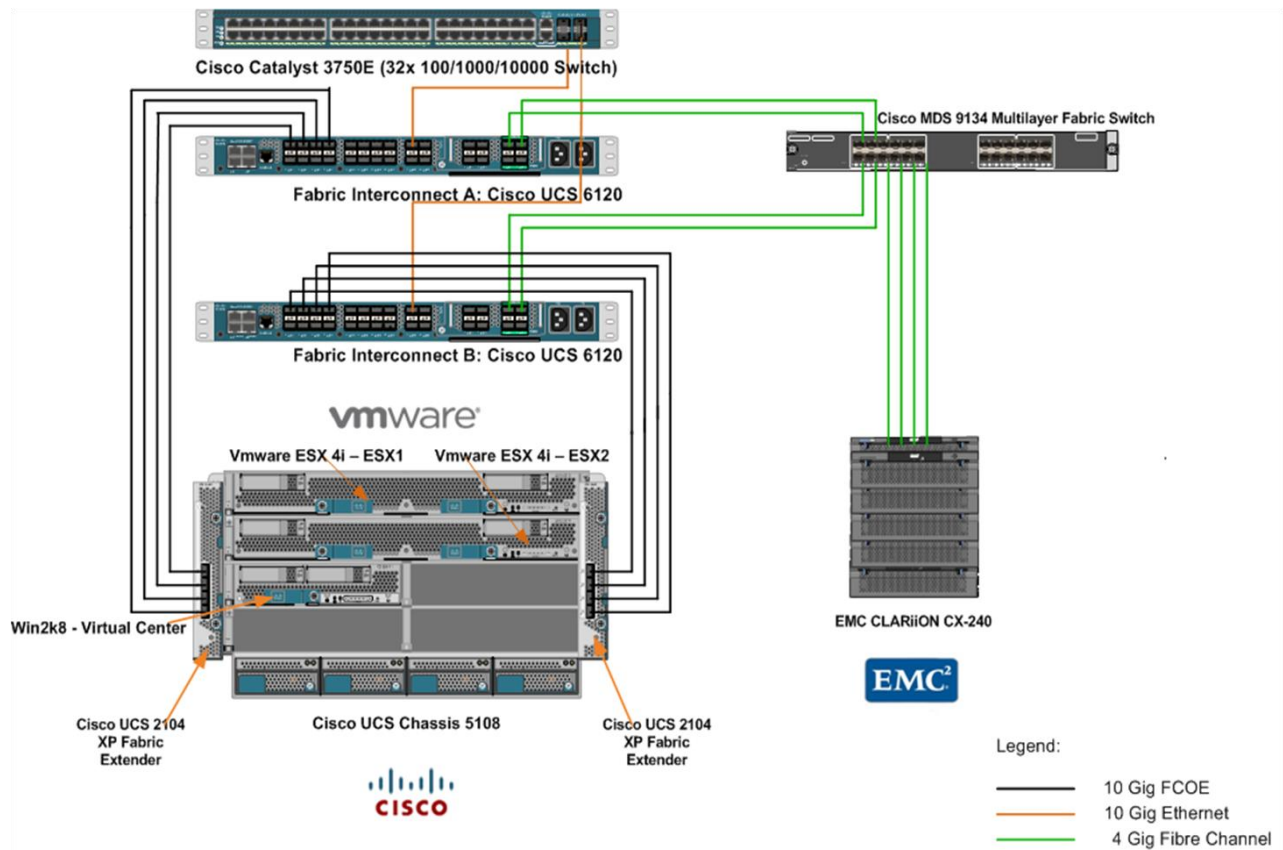
Table 1 summarizes the hardware and software used in the tests.

Table 1. Hardware and Software Used for Workload Characterization Tests

Hardware	<ul style="list-style-type: none"> • 1 x Cisco UCS B-250 blade server: 2 socket , 6 cores each, Intel Xeon X5670 @ 2.93GHz 192 GB RAM (running ESXi 4.1) • 1 x Cisco UCS B-250 blade server: 2 socket , 6 cores each, Intel Xeon X5670 @ 2.93GHz 192 GB RAM (running ESXi 4.1) • One ESXi host used to run database server virtual machines, the other used in the vMotion scenario and to run Swingbench clients in virtual machines (one Swingbench client for each database server) • 1 x B200 blade server running Windows 2008 and Virtual Center • Network: 10GB Ethernet • Shared storage: EMC Clarrion CX4-240 (Flare 30) with 75 Fibre channel 15K RPM spindles
VMware Software	<ul style="list-style-type: none"> • VMware Hypervisor ESXi 4.1 • vCenter Server 4.1 • vCenter Client 4.1 • vMA – Virtual Management Appliance
Guest OS and Oracle Database Software in VM	<ul style="list-style-type: none"> • Oracle Enterprise Linux 5.4 x86_64 • Oracle 11gR2 Enterprise Edition Single Instance x86_64
Oracle OLTP Workload	<ul style="list-style-type: none"> • Oracle Swingbench Load Generator – Large Scale Order Entry Benchmark

Figure 1 shows the physical architecture used for the tests.

Figure 1. Physical Architecture



4.3 Summary of Test Scenarios

Table 2 summarizes the test scenarios.

Table 2. Summary of Test Scenarios

Scenario	Virtual Machine (vCPU, RAM)	Test Description
Scaling up single Oracle database virtual machine (HT OFF)	2vCPU/8GB 4vCPU/16GB 6vCPU/24GB 8vCPU/32GB	<ul style="list-style-type: none"> • Run Swingbench Large Scale Oracle Entry Benchmark user load against each VM configuration • 4 separate test runs • 300 concurrent users • Min and max think time = 5 and10ms respectively
Scaling out and consolidating multiple virtual machines running an Oracle Database (HT ON)	4vCPU/16GB (1 – 5 VMs)	<ul style="list-style-type: none"> • Run Swingbench Large Scale Oracle Entry Benchmark user load against each VM configuration • 5 separate test runs • 200 concurrent users per VM • Min and max think time = 50 and100ms respectively
Live migration of Oracle database in a virtual machine (vMotion)	4vCPU/16GB	<ul style="list-style-type: none"> • Run Swingbench Large Scale Oracle Entry Benchmark user load • 300 concurrent users • Min and max think time = 5 and10ms respectively
VMware hot add CPU of Oracle database in a virtual machine	2vCPU/8GB (Hot Add 2 vCPU)	<ul style="list-style-type: none"> • Run Swingbench Large Scale Oracle Entry Benchmark user load • 100 concurrent users • Min and max think time = 5 and10ms respectively

5. Test Scenarios and Results

These sections describe the setup and results for all the test scenarios. Performance results were obtained from:

- Performance data captured by the VMware “esxtop” utility.
- Output data from the Oracle Swingbench generator. Performance is indicated by transaction throughput which is measured in Transactions Per Minute (TPM).

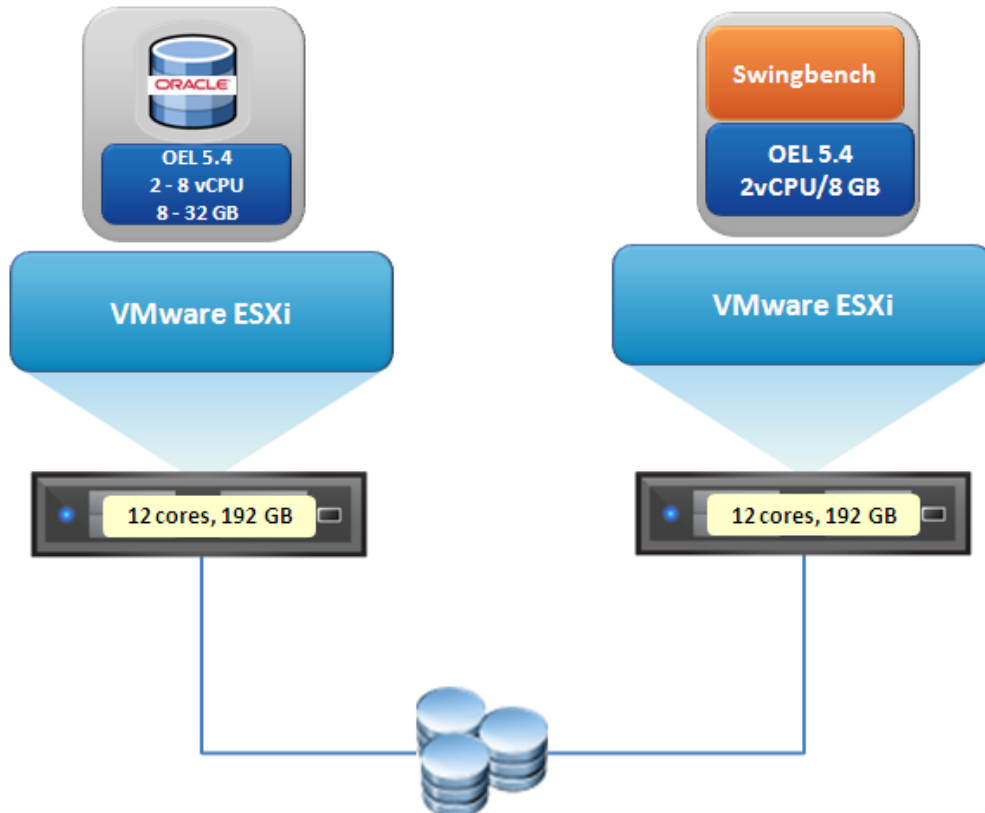
5.1 Single Virtual Machine – Scale Up Test

In the scale up test we ran four separate test runs against a single virtual machine configured with 2, 4, 6 and 8 vCPUs. The goal was to demonstrate increasing transaction throughput with additional processing capability of the virtual machine. The test details are described in Table 3 and Figure 2. Single Virtual Machine – Scale Up Test below. (See Appendix A for virtual machine configuration details and properties).

Table 3. Single Virtual Machine - Test Configurations

No. of vCPUs	Database Size (TB)	Concurrent Users (Min, Max think time - ms)	Virtual Machine Memory	Oracle Server SGA Size
2	1.1	300 (5,10)	8GB	6GB
4	1.1	300 (5,10)	16GB	10GB
6	1.1	300 (5,10)	24GB	16GB
8	1.1	300 (5,10)	32GB	20GB

Figure 2. Single Virtual Machine – Scale Up Test

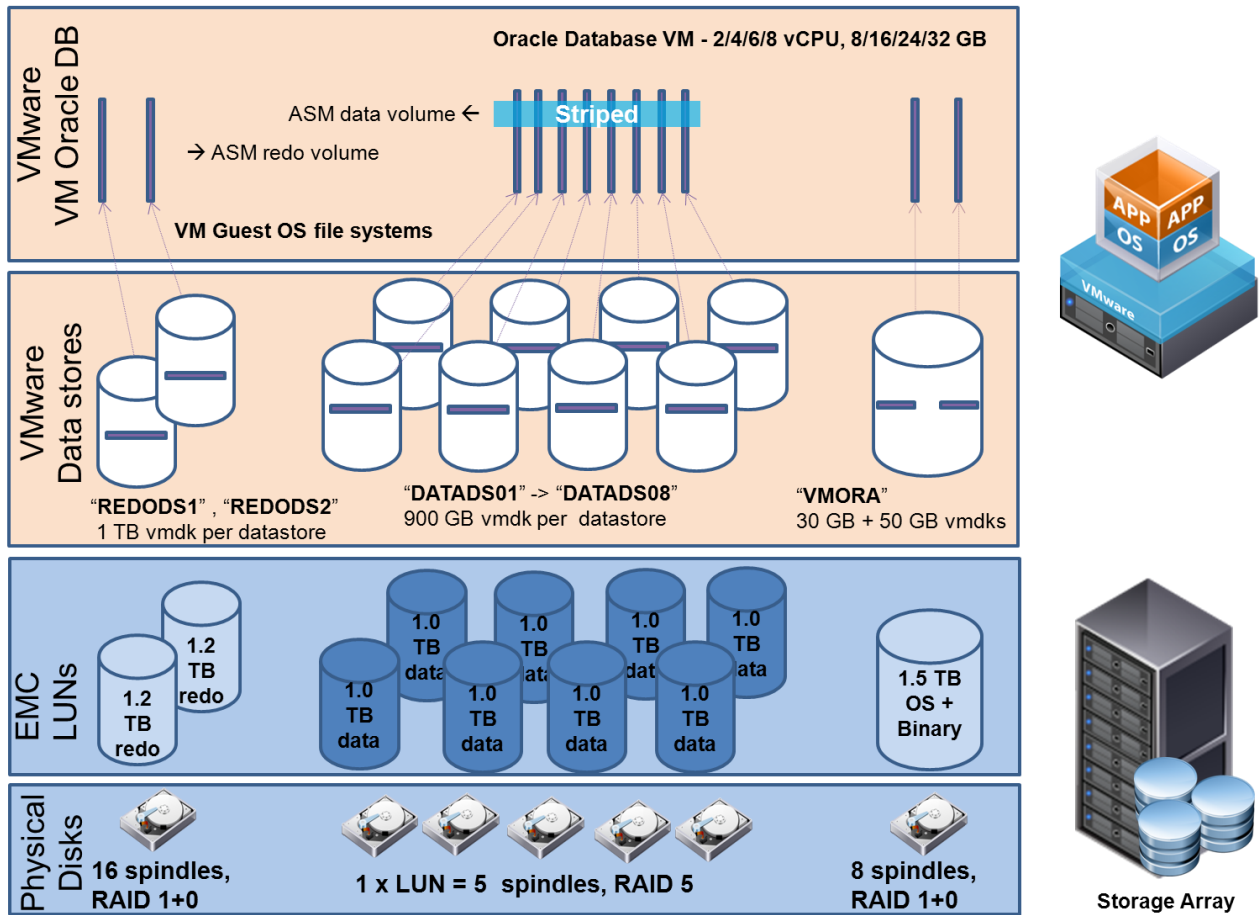


5.1.1 Storage Layout for Single Virtual Machine

Figure 3 represents the storage layout of the Oracle database.

- LUN 1–8 (1TB each – RAID 5) -> DATA: datastore 1 – 8 VMFS -> 8 x VMDKs (900GB each) (Managed by Oracle ASM)
- LUN 9-10 (1.5TB each – RAID 10) ->REDO: datastore 9-10 VMFS -> 2 x VMDKs (1TB each) (Managed by Oracle ASM)
- LUN 11 (1.5TB – RAID 10) -> VMORA: datastore 11 VMFS -> 2 VMDK (30GB for OS, 50GB for Binary)

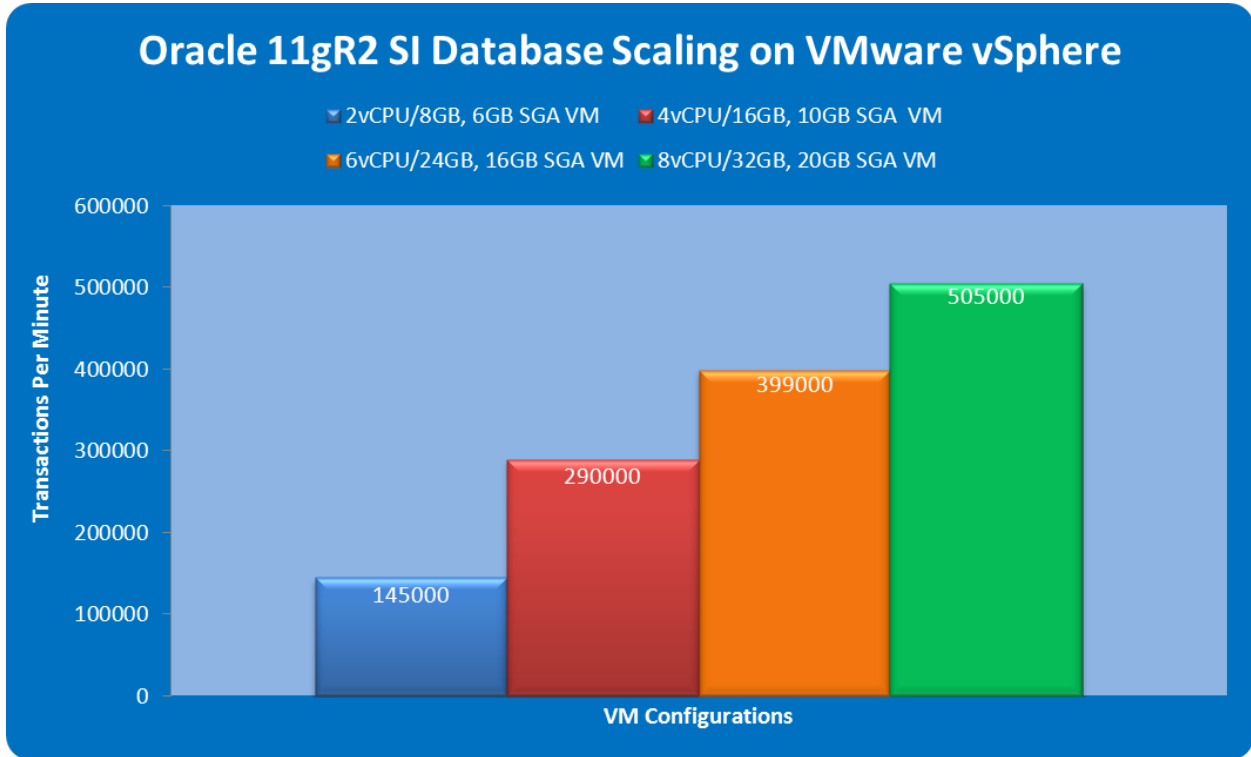
Figure 3. Single Virtual Machine – Storage Layout



5.1.2 Single Virtual Machine Scale Up Results

The results of the test runs are shown below. The chart shows the transaction throughput, measured in TPM, against the size of the virtual machine. In all runs CPU utilization within the virtual machine was near 100%.

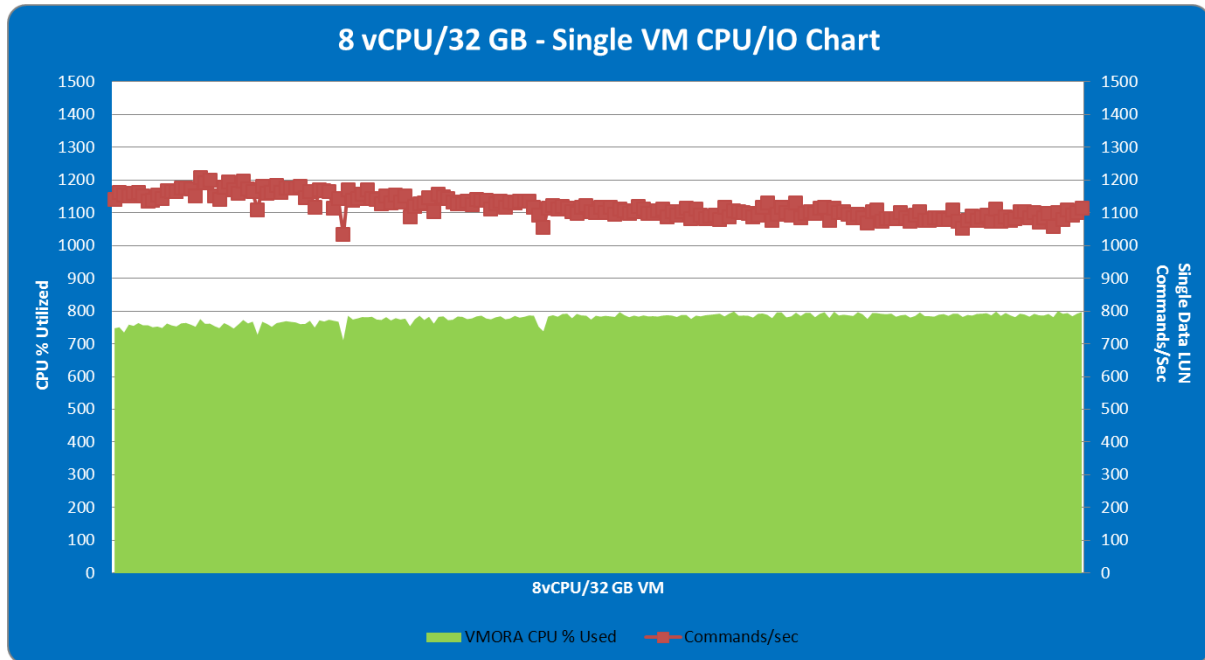
Figure 4. Single Virtual Machine – Scale Up Chart



The above results show that the Oracle database scaled linearly from 2 to 4 vCPUs. It did not scale as well to 8 vCPUs, but still exhibited sizable performance gains. It was observed in the case of 8 vCPUs the virtual machine memory spans NUMA nodes. This is explained in Appendix B.

Figure 5 shows the CPU and IO statistics for the 8-way configuration.

Figure 5. Single Virtual Machine CPU/IO Chart – 8vCPU/32GB



This is a hard working virtual machine producing 505,000 TPM with CPU near 800% (as per standard “esxtop” reporting which corresponds to near 100% CPU utilization within the virtual machine). The disk statistics of one of the data LUNs reported 1120 IOPS (Commands/Sec) with latency of 9.5 ms/cmd.

These findings show that a single database deployed in a virtual machine can be scaled out to handle larger transaction throughput. Customers can size virtual machines for Oracle databases to suit their capacity needs up to a maximum of 8 vCPUs, which is the current maximum for vSphere 4.1.

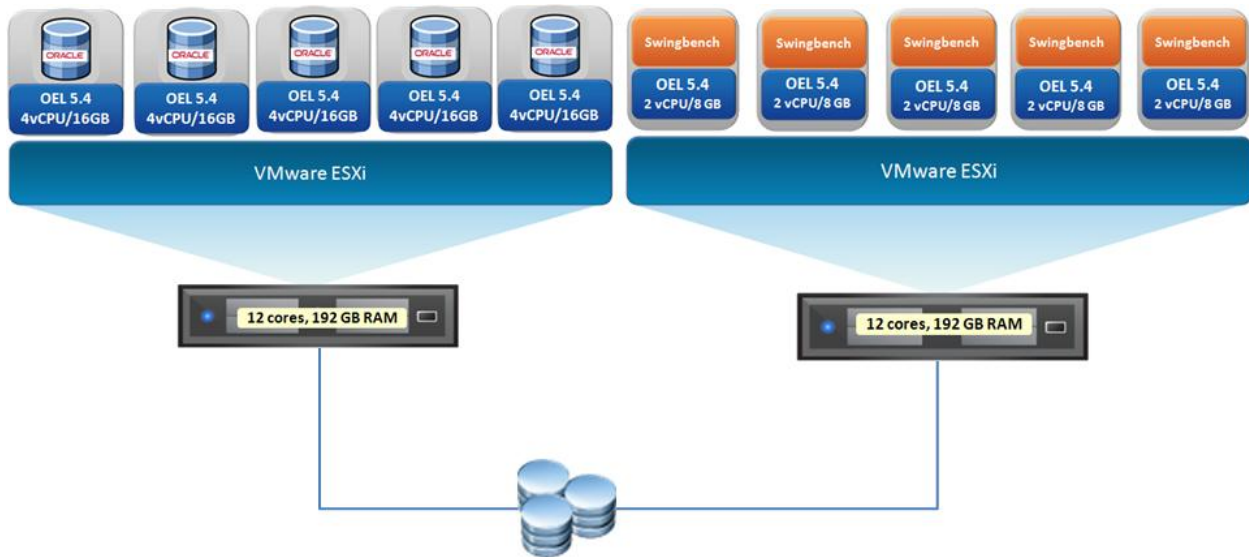
5.2 Multiple Virtual Machines – Scale Out Test

In the scale out test we ran five separate runs from 1 to 5 virtual machines where each virtual machine was configured with the same vCPU and memory. Each virtual machine ran its own individual database. The goal was to demonstrate increasing total transaction throughput with additional virtual machines. The tests are described in Table 4 and Figure 6. All five virtual machines shared the same LUNs/datastores.

Table 4. Multi-Virtual Machine – Test Configuration

No. of vCPUs per VM	Database Size (TB)	Concurrent Users (Min, Max think time –ms)	Virtual Machine Memory	Oracle Server SGA Size
4 (1 - 5 VMs)	1.1TB	200 (50,100)	16GB	10GB

Figure 6. Multiple Virtual Machines – Scale out Test

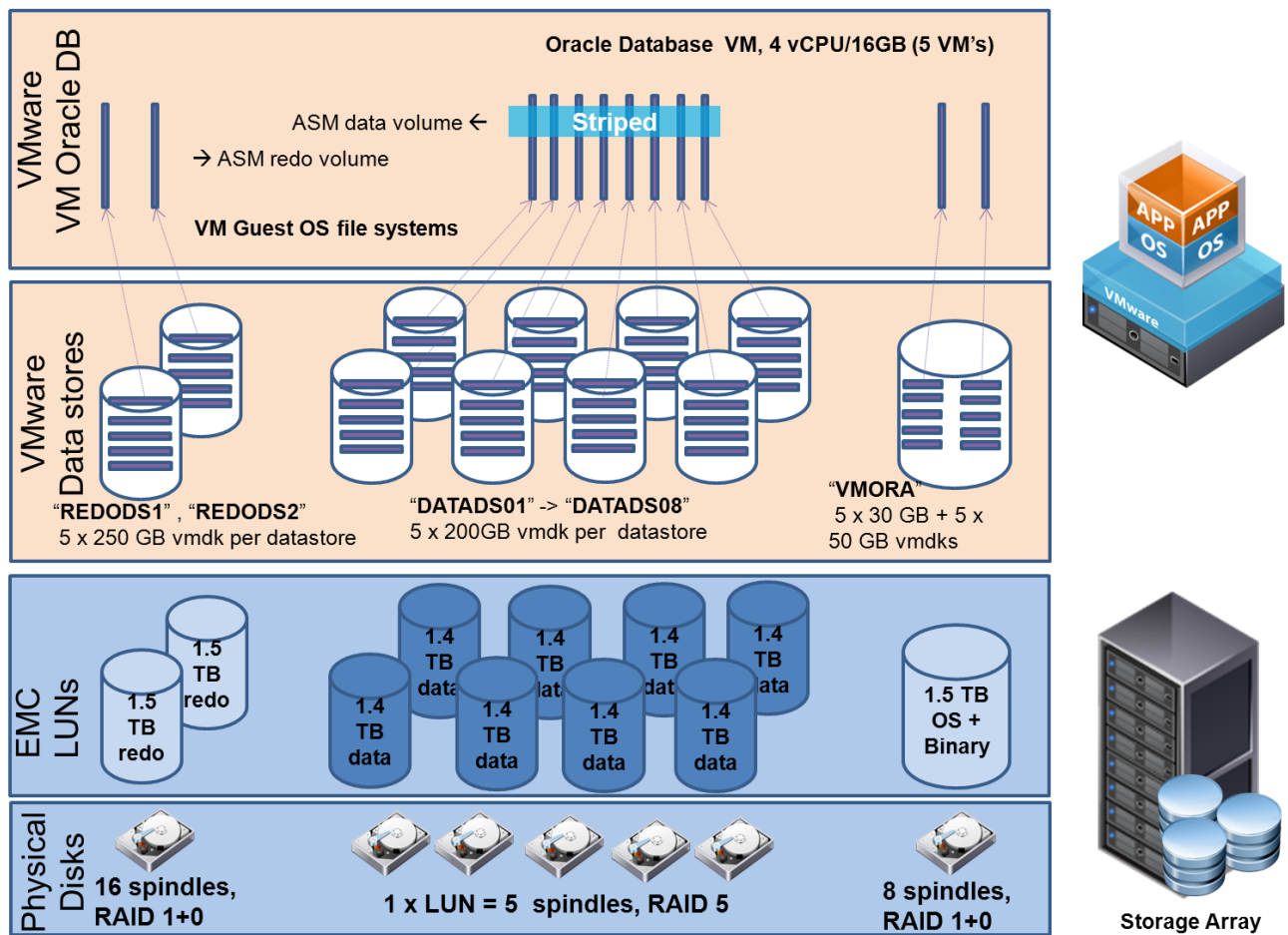


5.2.1 Storage Layout for Multiple Virtual Machines

Figure 7 represents the storage layout for the Oracle databases.

- LUN 1–8 (1TB each – RAID 5) -> DATA: datastore 1 – 8 VMFS -> 8 x VMDKs (200GB each) for each Oracle database (Managed by Oracle ASM) – Each virtual machine has one VMDK from each datastore
- LUN 9-10 (1.5TB each – RAID 10) ->REDO: datastore 9-10 VMFS -> 2 x VMDKs (250TB each) for each Oracle database (Managed by Oracle ASM) – Each virtual machine has one VMDK from each datastore
- LUN 11 (1.5TB – RAID 10) -> VMORA: datastore 11 VMFS -> 2 VMDK (30GB for OS, 50GB for Binary) for each Oracle database – Each virtual machine has two VMDKs from the same datastore

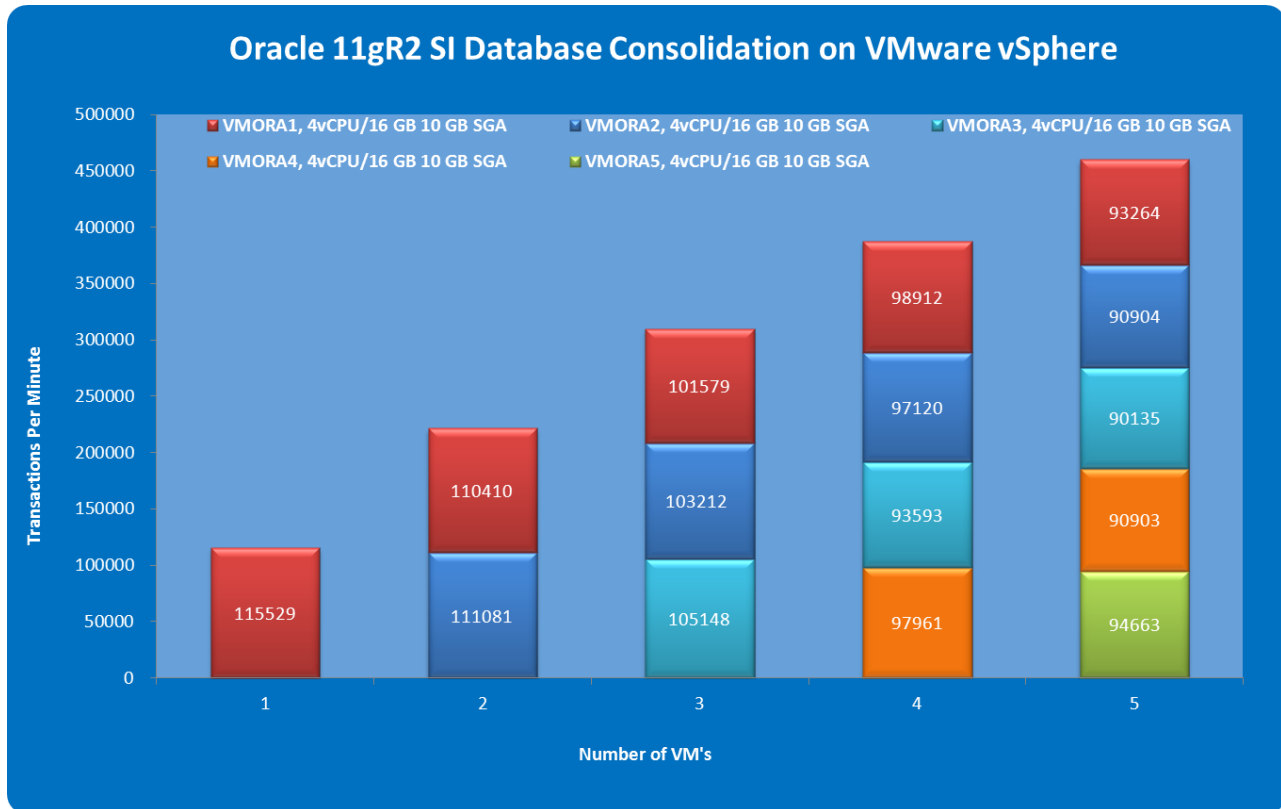
Figure 7. Multi-Virtual Machine – Storage Layout



5.2.2 Multiple Virtual Machines Scale Out Results

Figure 8 shows the results of the five test runs in the scale out test. The chart shows transaction throughput, measured in TPM, against the number of simultaneous virtual machines. Each bar is broken down to show the TPM value of the individual virtual machine.

Figure 8. Multiple Virtual Machines – Scale Out Chart



Note: as storage was limited we could not scale out beyond five virtual machines. With five virtual machines running CPU utilization within each virtual machine was about 45%.

Figure 8 demonstrates that the total throughput increases fairly linearly with additional virtual machines. The results show that ESX is able to fairly distribute CPU resources across the virtual machines. CPU utilization of each virtual machine, when all five virtual machines were running, was similar. When multiple identical virtual machines are running, identical performance is expected from each. This is demonstrated by the similar TPM values processed by each virtual machine and is attributed to the fairness properties of the vSphere scheduler.

The ESX host was able to handle the workloads of the multiple virtual machines:

- With five virtual machines the total number of vCPUs was more than the physical number of server cores. This overcommitment of vCPU was possible as the virtual machines were not saturated. This is common in production scenarios where we typically see average CPU utilization of database servers well below 100% thus allowing more virtual machines to be consolidated on the same ESX host.
- The memory assigned to all the virtual machines was within the memory capacity of the ESX host so that the ESX balloon driver was not invoked during any of the scenarios.
- The multiple databases were able to share the same VMFS LUNs (Figure 7 shows the storage layout and consolidation of the multiple virtual machines in the same redo and data LUNs).

The results highlight that it is possible to consolidate multiple databases into the same VMFS LUNs and still maintain reasonable scalability. While not tested here due to the storage limits of the test environment, performance can be further fine-tuned by adding more disk spindles to the LUNs or rebalancing vmdk disks via storage vMotion to other less utilized LUNs. (VMware Storage vMotion is the live migration of virtual machine disk files within and across storage arrays with no downtime or disruption in service). It should be noted that in cases of heavy I/O workloads, it is wise to separate such workloads out of the shared pool of storage to optimize the performance of those high transactional throughput applications. Since database workloads will vary, when looking at the number of virtual machines per LUN there is no exact rule to determine the limits of performance and scalability. It depends on the combined I/O workload of the virtual machines and storage array specifications, for example the IOPS capabilities of the underlying physical storage.

The scale out capabilities of vSphere enables the consolidation of database virtual machines onto less ESX hosts. Storage consolidation with virtual machine disk files allows storage administrators to maximize the utilization of their storage arrays and still maintain transaction performance. These features allow customers to get more out of their resources and lower their TCO.

5.3 Live Migration of Virtual Machine (vMotion)

In the vMotion test we live migrated a virtual machine running an Oracle database server on ESXi Host 1. The test was executed as follows:

- Virtual machine – VMORA1 running Oracle database server on ESXi Host 1.
- Execute 300-user Swingbench Large Scale Order Entry Benchmark for approximately one hour with minimum and maximum think time of 5 and 10ms respectively.
- Live migrate the virtual machine to target server ESXi Host 2.

Figure 9 below demonstrates the results. The chart shows transaction throughput against time. The first vertical line marks the start of live migration, the second the end of live migration.

Figure 9. Oracle Database Transactions Per Minute During Migration With vMotion

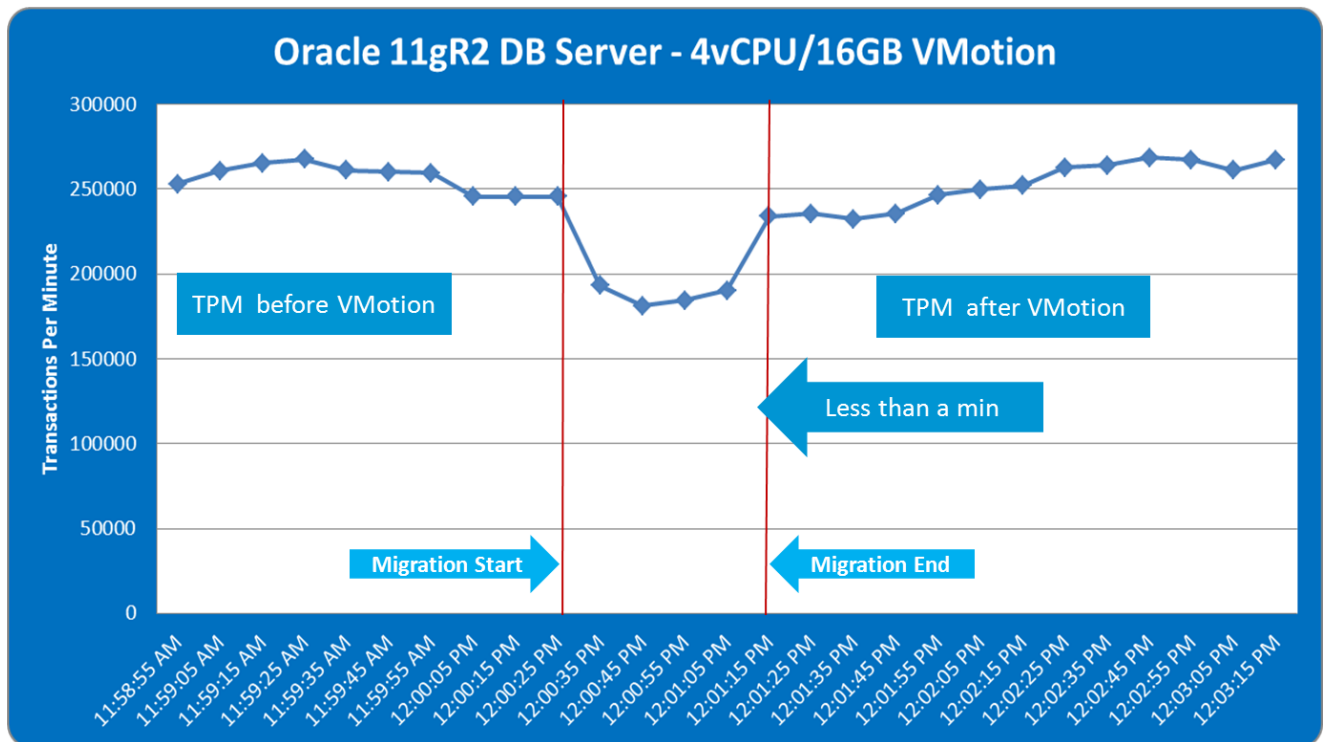


Figure 9 shows a decrease in throughput during migration, but transactions per minute immediately recovers upon completion of the migration. The time taken to complete the migration is less than a minute.

Figure 10 and Figure 11 show the esxtop CPU panel during the migration. vMotion creates a process on the source (vmotion-src) and target (vmotion-dest) servers that consume CPU in addition to the business workload.

VMware vMotion presented minimal impact to the Oracle OLTP workload and no impact to transaction and database consistency. System administrators gain substantial flexibility in being able to balance virtual machines freely across ESX hosts to improve or maintain performance or when they need to perform server maintenance with no impact to end-users.

Figure 10. ESXi Host 1 –Source

```

ESXTOP: 10.1.1.17
4:17:39pm up 28 days 49 min, 224 worlds; CPU load average: 0.39, 0.38, 0.19
PCPU USED(%): 29.1 71.8 20.0 43.6 37.8 70.4 36.9 22.2 54.1 43.8 72.1 23.4 18.1 18.5 19.2 19.1
19.0 17.8 17.7 17.6 18.1 18.1 18.4 19.0 AVG: 31.1
PCPU UTIL(%): 30.0 69.0 21.0 43.0 39.7 69.9 35.8 22.1 50.5 41.2 67.3 22.3 19.2 19.1 19.8 20.1
19.5 18.8 19.0 18.5 19.4 19.3 19.0 19.6 AVG: 31.0
CORE UTIL(%): 92.8 57.8 99.5 51.8 89.8 87.5 34.1 35.4
34.2 33.4 34.3 34.6 AVG: 57.1
    
```

ID	GID NAME	NWLD	%USED	%RUN	%SYS	%WAIT	%RDY	%IDLE	%OVRLP	%CSTP	%MLMTD
1	1 idle	24	499.05	1200.00	0.00	0.00	147.26	0.00	55.27	0.00	0.00
1247396	1247396 VMORa1	7	435.51	399.46	13.91	300.99	0.15	0.00	3.59	0.00	0.0
899	899 hostd.5151	16	123.87	116.57	0.66	1444.79	39.31	0.00	4.29	0.00	0.00
1032	1032 vpxa.5289	18	106.82	91.15	11.68	1676.63	33.60	0.00	3.28	0.00	0.00
1300958	1300958 vmotion-src.407	5	56.90	38.50	27.18	459.55	2.49	0.00	1.36	0.00	0.0
254	254 busybox.4487	1	2.93	3.02	0.06	96.49	0.59	0.00	0.11	0.00	0.00
7	7 helper	69	1.35	0.71	0.00	6900.00	0.13	0.00	0.03	0.00	0.00
2	2 system	8	0.35	0.31	0.00	799.97	0.01	0.00	0.00	0.00	0.00
1209	1209 sh.5481	1	0.16	0.17	0.00	99.92	0.01	0.00	0.00	0.00	0.00
1585	1585 sfcb-sfcb.5855	1	0.08	0.08	0.00	100.00	0.00	0.00	0.00	0.00	0.00
1586	1586 sfcb-sfcb.5856	1	0.06	0.09	0.00	100.00	0.00	0.00	0.02	0.00	0.00
1587	1587 sfcb-ProviderMa	1	0.02	0.01	0.00	100.00	0.00	0.00	0.00	0.00	0.00
1200	1200 vmware-usbarbit	2	0.02	0.02	0.00	200.00	0.00	0.00	0.00	0.00	0.00
1606	1606 sfcb-ProviderMa	6	0.01	0.01	0.00	600.00	0.00	0.00	0.00	0.00	0.00
8	8 drivers	10	0.01	0.01	0.00	1000.00	0.00	0.00	0.00	0.00	0.00
839	839 storageRM.5092	1	0.01	0.00	0.00	100.00	0.00	0.00	0.00	0.00	0.00
1005	1005 opensmand.5259	3	0.00	0.00	0.00	300.00	0.00	0.00	0.00	0.00	0.00
1581	1581 sfcbd.5851	1	0.00	0.00	0.00	100.00	0.00	0.00	0.00	0.00	0.00
795	795 vprobed.5043	1	0.00	0.00	0.00	100.00	0.00	0.00	0.00	0.00	0.00
752	752 ntpd.5000	1	0.00	0.00	0.00	100.00	0.00	0.00	0.00	0.00	0.00

Figure 11. ESXi Host 2 – Target

```

ESXTOP: 10.1.1.18
4:29:31pm up 43 days 4:37, 231 worlds; CPU load average: 0.09, 0.07, 0.16
PCPU USED(%): 18.0 14.6 18.3 13.8 16.5 17.3 18.2 17.3 17.5 18.8 16.5 17.2 21.5 13.6 16.8 11.
2 13.7 13.8 23.1 9.0 12.9 11.4 18.0 10.4 AVG: 15.8
PCPU UTIL(%): 18.3 15.0 18.5 13.9 17.4 17.6 18.8 18.1 17.9 18.7 16.7 17.8 21.7 13.7 17.9 11.
2 13.6 13.8 23.4 9.8 12.6 10.4 19.9 11.4 AVG: 16.2
CORE UTIL(%): 31.5 30.7 32.8 34.4 34.3 32.2 32.1 26.4
24.9 29.9 21.4 28.3 AVG: 29.9
    
```

ID	GID NAME	NWLD	%USED	%RUN	%SYS	%WAIT	%RDY	%IDLE	%OVRLP	%CSTP	%MLM
1	1 idle	24	831.38	1200.00	0.00	0.00	1200.00	0.00	103.69	0.00	0.
2002485	2002485 vmotion-dest.65	5	203.98	125.00	94.35	374.49	0.77	0.00	16.39	0.00	0
1948658	1948658 SwingBench1	5	50.18	48.71	2.60	451.38	0.26	151.13	0.63	0.00	0
805	805 hostd.5052	16	47.14	42.46	0.14	1548.24	10.11	0.00	0.79	0.00	0.
1048	1048 vpxa.5305	18	34.84	33.02	0.16	1760.39	7.38	0.00	0.69	0.00	0.
982539	982539 vMA	4	9.67	10.62	0.04	389.37	0.16	89.21	0.09	0.00	0.
7	7 helper	69	2.93	0.76	0.00	6900.00	0.13	0.00	0.05	0.00	0.
17	17 vmkapimod	6	1.70	1.57	0.00	598.67	0.00	0.00	0.00	0.00	0.
254	254 busybox.4487	1	1.33	1.44	0.05	98.43	0.20	0.00	0.04	0.00	0.
2002484	2002484 VMORa1	2	1.11	1.19	0.00	198.87	0.04	0.00	0.01	0.00	0
2	2 system	8	0.19	0.18	0.00	799.88	0.01	0.00	0.00	0.00	0.
1022	1022 opensmand.5277	3	0.03	0.01	0.00	300.00	0.00	0.00	0.00	0.00	0.
1226	1226 sh.5498	1	0.03	0.03	0.00	100.00	0.00	0.00	0.00	0.00	0.
1216	1216 vmware-usbarbit	2	0.02	0.02	0.00	200.00	0.00	0.00	0.00	0.00	0.
8	8 drivers	10	0.00	0.01	0.00	1000.00	0.00	0.00	0.00	0.00	0.
769	769 ntpd.5017	1	0.00	0.00	0.00	100.00	0.00	0.00	0.00	0.00	0.
916	916 vprobed.5171	1	0.00	0.00	0.00	100.00	0.00	0.00	0.00	0.00	0.
1615	1615 sfcb-ProviderMa	6	0.00	0.00	0.00	600.00	0.00	0.00	0.00	0.00	0.
872	872 storageRM.5122	1	0.00	0.00	0.00	100.00	0.00	0.00	0.00	0.00	0.
258	258 vmklogger.4491	1	0.00	0.00	0.00	100.00	0.00	0.00	0.00	0.00	0.

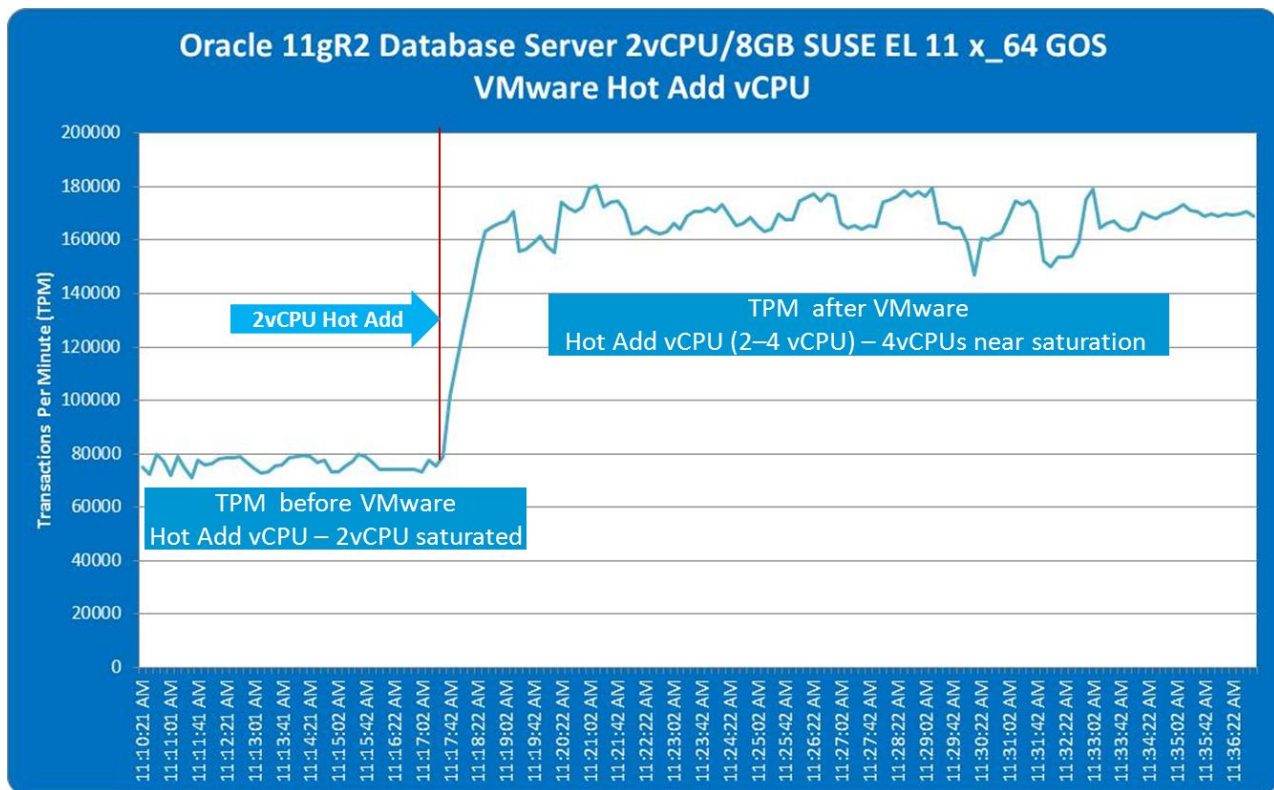
5.4 VMware vCPU Hot Add

In the hot add scenario we tested the vSphere hot plug feature for virtual devices which supports addition of virtual CPUs to a virtual machine without powering off the virtual machine. (Consult the [VMware Compatibility Guide](#) to determine support for hot add CPU for a specific guest OS.) The test was executed as follows:

- Virtual machine – VMSUSE (Oracle database 11gR2 with SUSE Enterprise Linux 11 x_64 – VMware Hot Add vCPU supported Guest Operation System) running on ESXi Host 1 with CPU saturated near 100%.
- Execute 100 users Swingbench Large Scale Order Entry Benchmark load against 2 vCPU/8GB virtual machine for approximately one hour with minimum and maximum think time of 5 and 10ms respectively. Hot Add vCPU to VMSUSE (2 vCPU to 4 vCPU).

Figure 12 shows transaction throughput, measured in TPM against time. It shows the TPM values of the virtual machine before and after the Hot Add vCPU operation. The vertical line marks the point in time the hot add operation was initiated in the vSphere client.

Figure 12. Oracle Database Transactions Per Minute with VMware Hot Add vCPU



The chart shows the virtual machine initially delivered performance at just below 80,000 TPM. After hot add of 2 vCPUs transaction throughput approximately doubles.

The Oracle database is aware of the extra vCPU. Figure 13 shows detection of the change of CPU count in the Oracle alert log while Figure 14 shows that the Oracle database parameter CPU_COUNT changed from 2 to 4.

This test shows that Oracle database administrators can take advantage of the VMware Hot Add CPU feature to add vCPU on the fly without restarting the database or the virtual machine. Databases can be sized in a smaller virtual machine, but as workload increases beyond initial estimates no downtime is required to add more processing capability.

Figure 13. Oracle Database Server Alert Log

```

10.1.1.31 - PuTTY
oracle@vmsuse:/database/u01/app/oracle/diag/rdbms/vmsuse/vmsuse/trace> tail -f
Tue Nov 16 11:31:31 2010
QMNC started with pid=20, OS id=4283
Completed: ALTER DATABASE OPEN
Starting background process CJQO
Tue Nov 16 11:31:33 2010
CJQO started with pid=24, OS id=4313
Tue Nov 16 11:36:34 2010
Starting background process SMCO
Tue Nov 16 11:36:34 2010
SMCO started with pid=21, OS id=4400
Tue Nov 16 11:43:42 2010
Detected change in CPU count to 4

```

Figure 14. Oracle Database Server CPU Count

```

10.1.1.31 - PuTTY
SQL> show parameter CPU_COUNT

NAME                                TYPE        VALUE
-----
cpu count                            integer     2
SQL> show parameter CPU_COUNT

NAME                                TYPE        VALUE
-----
cpu count                            integer     4
SQL>

```

6. Conclusion

The experiments described in this paper demonstrate that out-of-the-box VMware vSphere can handle demanding Oracle workloads.

Results show that a single database deployed in a virtual machine can be scaled up to handle larger transaction throughput. Customers can size virtual machines for Oracle databases to suit their capacity needs up to a maximum of 8 vCPUs, which is the current maximum for vSphere 4.1.

The scale out (consolidation) tests show that a single ESX host can manage with fairness the workload of multiple Oracle databases in separate virtual machines. Consolidation is achieved both in terms of CPU and storage - multiple databases can share the same VMFS datastores. Such consolidation allows customers to get more out of their resources and lower their TCO.

The combination of scale up with scale out on vSphere allows customers to achieve the optimum performance for their database virtual machines and provides more flexibility to right-size system deployment.

VMware vMotion presented minimal impact to an Oracle OLTP load and no impact to transaction and database consistency. System administrators gain substantial flexibility in being able to balance virtual machines freely across ESX hosts to improve or maintain performance or when they need to perform server maintenance with no impact to end-users.

Oracle database administrators can take advantage of the VMware Hot Add CPU feature to add vCPU on the fly without restarting the virtual machine. Databases can be sized in a smaller virtual machine, but as workload increases beyond initial estimates no downtime is required to add more processing capability.

7. References

The following are resources and references for Oracle and VMware vSphere:

- Performance Best Practices for VMware vSphere 4:
http://www.vmware.com/pdf/Perf_Best_Practices_vSphere4.0.pdf
- VMware Systems Compatibility Guide:
<http://www.vmware.com/resources/compatibility/search.php>
- VMware Resource Management Guide:
http://www.vmware.com/files/pdf/techpaper/vsp_41_perf_memory_mgmt.pdf
- VMware vSphere: The CPU Scheduler in VMware ESX 4.1:
http://www.vmware.com/files/pdf/techpaper/VMW_vSphere41_cpu_schedule_ESX.pdf
- *Performance Troubleshooting for VMware vSphere 4*:
<http://communities.vmware.com/docs/DOC-10352>
- Comparison of Storage Protocol Performance in VMware vSphere 4
http://www.vmware.com/files/pdf/perf_vsphere_storage_protocols.pdf
- VMware SAN System Design and Deployment Guide
http://www.vmware.com/pdf/vsp_4_san_design_deploy.pdf
- Oracle on VMware vSphere Essential Database Deployment Tips
http://www.vmware.com/files/pdf/Oracle_Databases_on_vSphere_Deployment_Tips.pdf
- Virtualizing Performance-Critical Database Applications in VMware vSphere
http://www.vmware.com/pdf/Perf_ESX40_Oracle-eval.pdf
- *Oracle Database Installation and Configuration Guide*
http://www.oracle.com/technet/middleware/11g/portal.portal_db?selected=11
- *VMware Compatibility Guide*
<http://www.vmware.com/resources/compatibility/search.php> ► Guest OS tab
(Use the online compatibility guide to determine Hot Add CPU support for a specific guest OS)

8. Disclaimers

All data is based on in-lab results with the vSphere 4.1. Our workload was a fair-use implementation of the TPC-C business model; these results are not TPC-C compliant and are not comparable to official TPC-E results. TPC Benchmark and TPC-C are trademarks of the Transaction Processing Performance Council.

The throughput here is not meant to indicate the absolute performance of Oracle Database 11gR2, or to compare its performance to another DBMS. Oracle database was used to place a DBMS workload on VMware ESX, and observe and optimize the performance of ESXi.

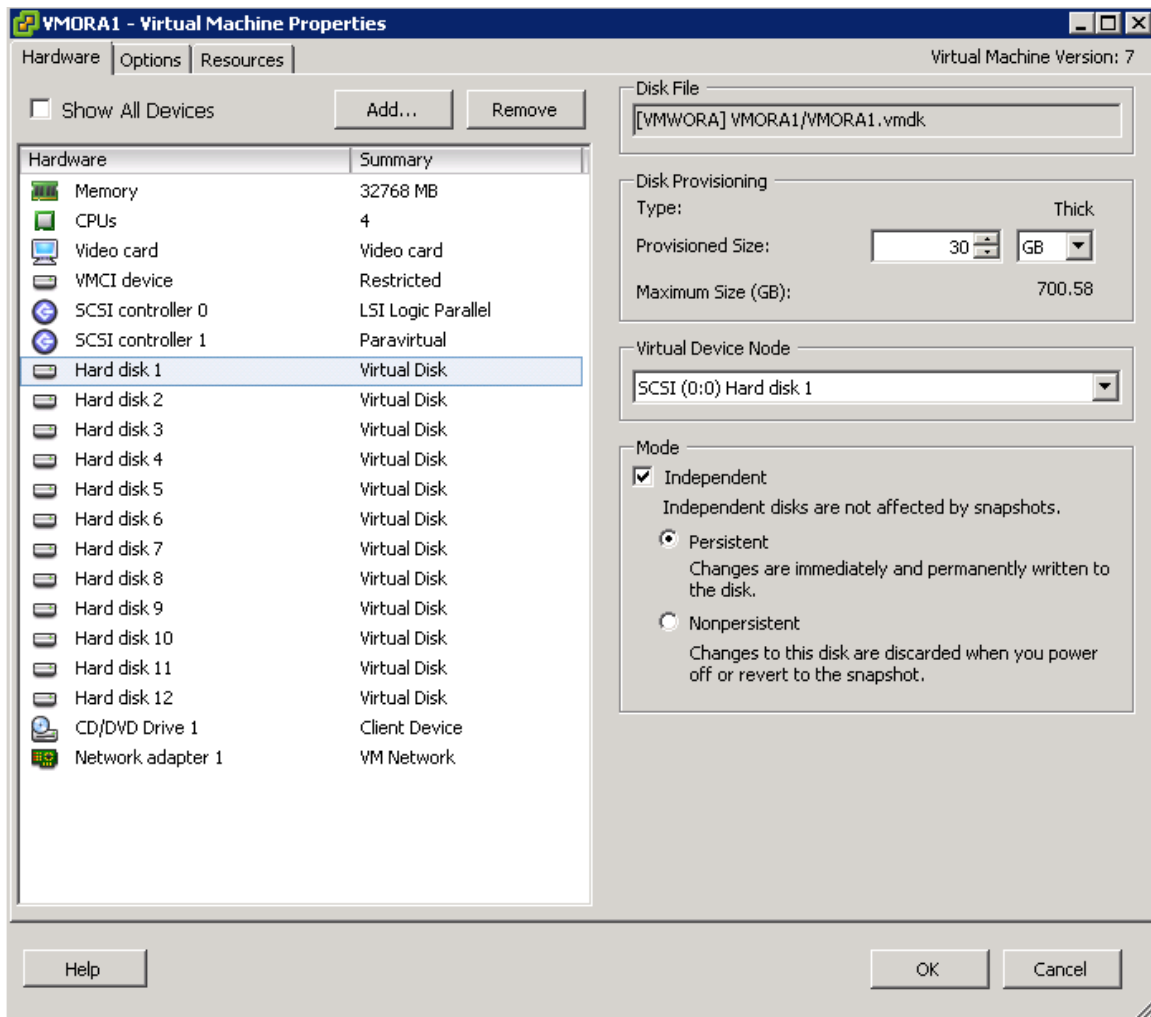
The goal of the experiment was to show the Oracle database virtual machine workload characterization on VMware, and its ability to handle heavy database workloads. It was not meant to measure the absolute performance of the hardware and software components used in the study.

The throughput of the workload used does not constitute a TPC benchmark result.

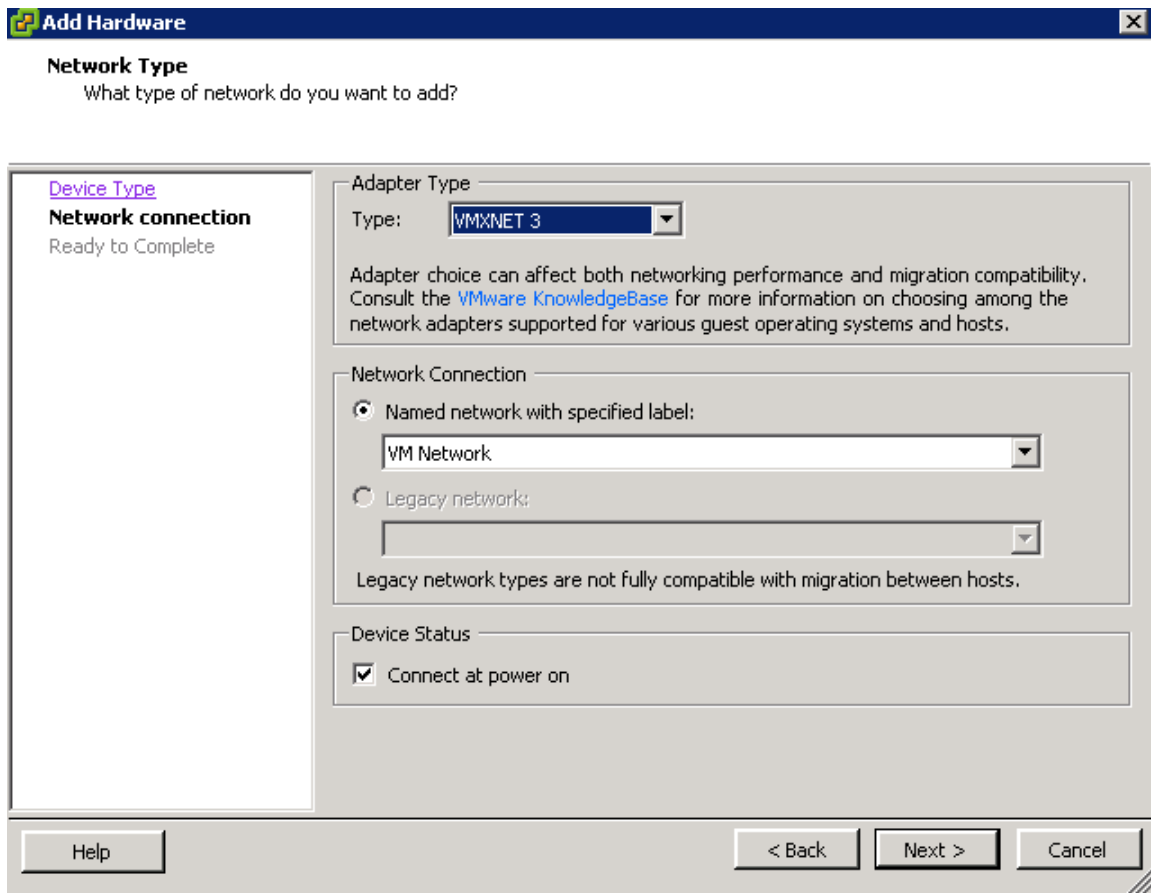
Appendix A: Virtual Machine Configuration

This Appendix describes some of the virtual machine configuration steps and properties.

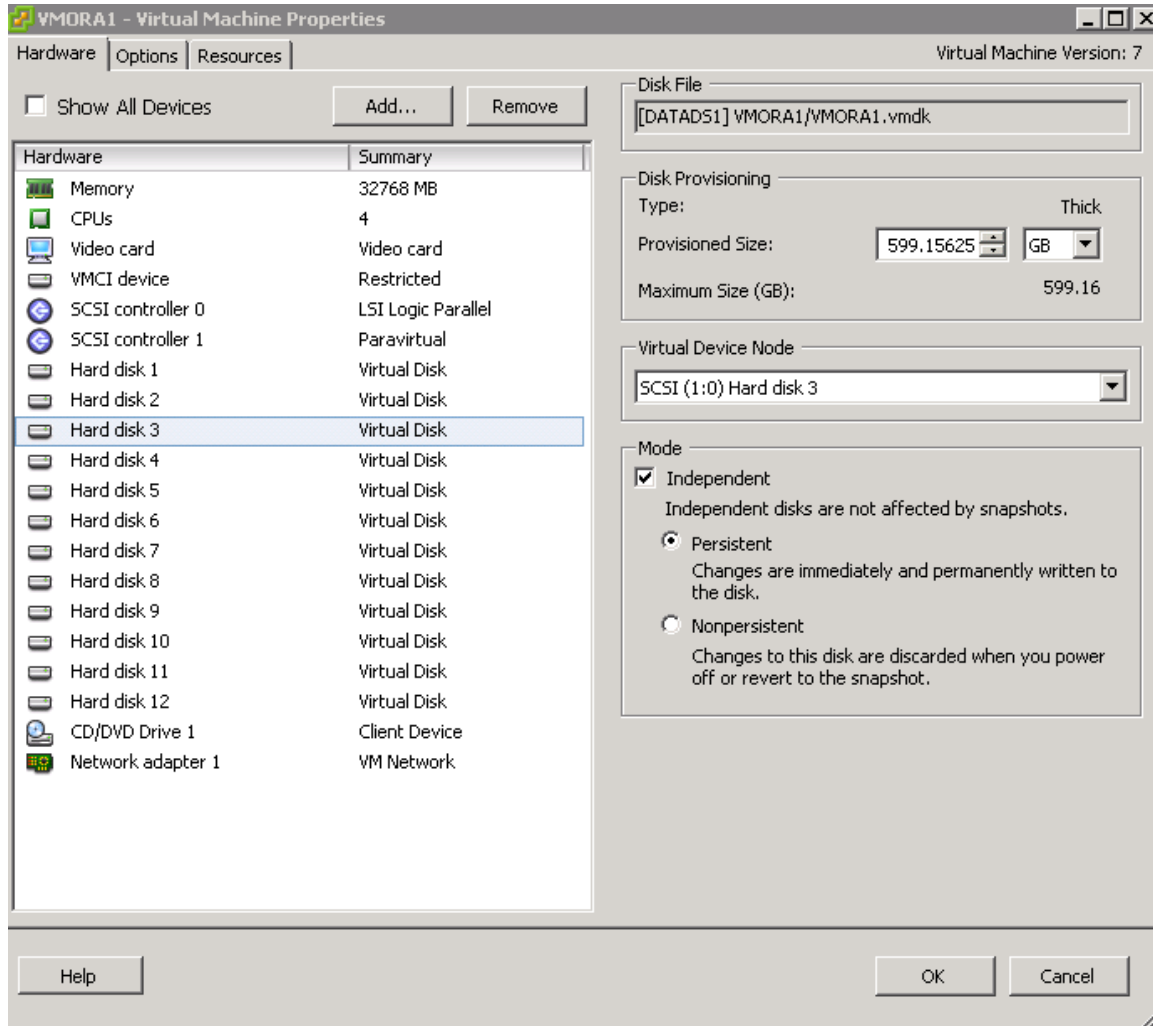
1. Create the Oracle database virtual machine with the following properties.



2. Network: Create or add network adapter and select **VMXNET3** for better performance. VMware Tools must be installed for the VMXNET3 adapter type to be available.



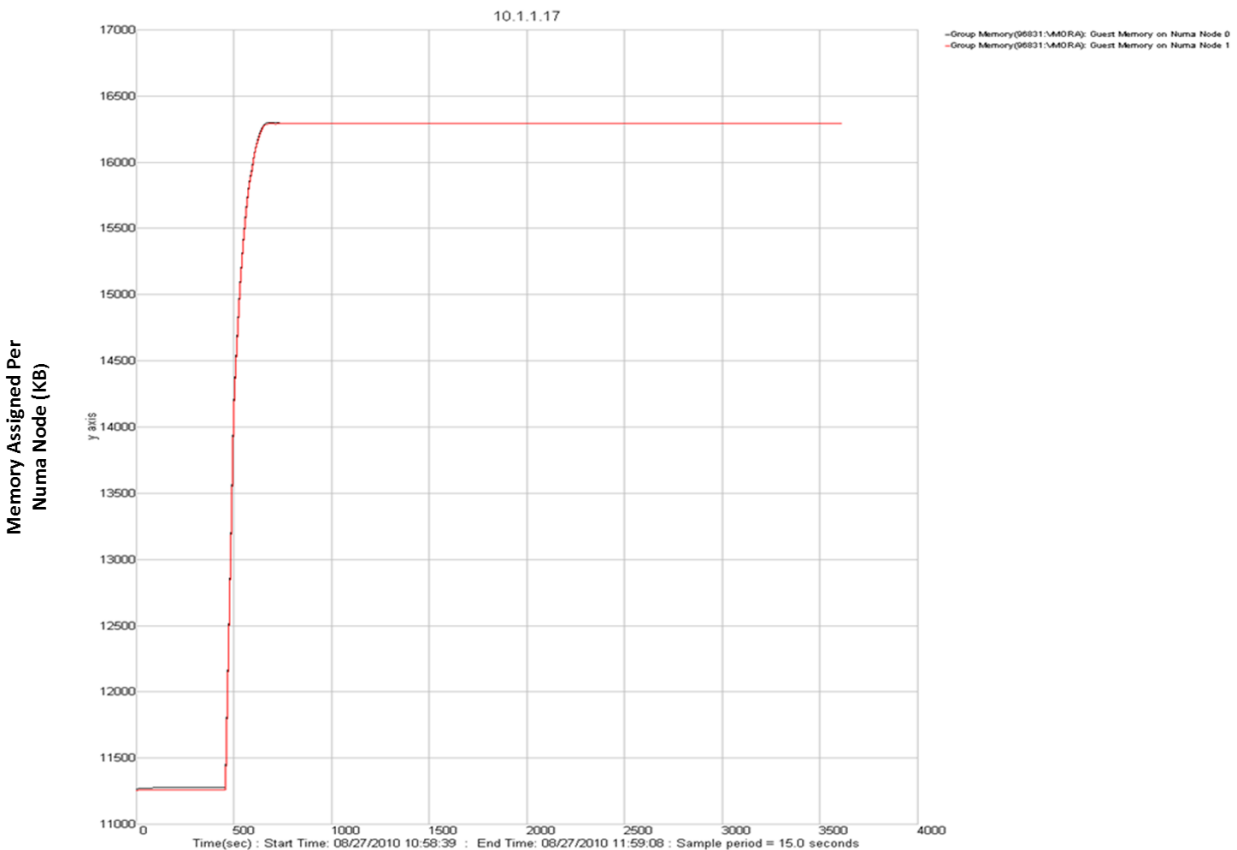
3. Hard Disk: Create Hard Disk 1 and 2 with default SCSI (LSI Logic Parallel) adapter for both VM files and Oracle Binary.
4. Create Hard disk 3 – 10 for Oracle Data and Hard disk 11 – 12 for Oracle Redo logs with a VMware paravirtual adapter for better performance.
5. Select SCSI 1:0 for virtual device node to create a new SCSI controller and change the controller type to **VMware Paravirtual**. VMware Tools must be installed for the paravirtual option to be available. Follow [VMware KB 1010398](http://www.vmware.com/kb/1010398) to configuring disks to use VMware Paravirtual SCSI (PVSCSI) adapters



6. Install Oracle Enterprise Linux 5.4 64-bit and install and configure Oracle Database 11gr2, ASM with database size 1TB. Database disk layouts are described in each tested scenario.

Appendix B: 8 vCPU/32GB Virtual Machine Memory Split

This Appendix shows the NUMA memory statistics captured during the single virtual machine scale up run where the virtual machine was configured with 8 vCPU and 32 GB RAM. The following chart was obtained from “esxtop” data and extracted using “esxplot”. It shows that the virtual machine memory of 32 GB was evenly split across two NUMA nodes.



The above behavior is expected in vSphere 4.1. A Wide-VM is defined as a virtual machine that has more vCPUs than the available cores on a NUMA node – this is the case in the above example where the ESX server consisted of two 6-core sockets, ESX 4.1 allows wide-VMs to take advantage of NUMA management. NUMA management means that a virtual machine is assigned a home node where memory is allocated and vCPUs are scheduled. By scheduling vCPUs on a NUMA node where memory is allocated, the memory accesses become local, which is faster than remote accesses.

Wide-VM NUMA support is accomplished by splitting a wide-VM into smaller NUMA clients whose vCPU count does not exceed the number of cores per NUMA node, and assigning a home node to each client. The 8-vCPU virtual machine on the ESX server has two 4-vCPU NUMA clients. The wide-VM has multiple home nodes because it consists of multiple clients, each with its own home node. This is implemented in a manner to improve performance of memory access – for details see the whitepaper [VMware vSphere: The CPU Scheduler in VMware ESX 4.1](#).