PVSCSI Storage Performance

VMware ESX 4.0

VMware vSphere 4™ offers Paravirtualized SCSI (PVSCSI), a new virtual storage adapter. This document provides a performance comparison of PVSCSI and LSI Logic. The experiment results show that PVSCSI provides better throughput and lower CPU utilization in virtualized environments where guest applications are very I/O intensive. The experiment results also demonstrate that, although PVSCSI provides better CPU efficiency, LSI Logic is able to provide wire-speed throughput, indicating that the storage adapter is not the limiting factor in I/O performance if the system is properly configured.

Introduction

VMware vSphere 4 includes several unique new features that allow IT organizations to leverage the benefit of cloud computing with maximum efficiency, uncompromised control, and flexibility of choice. The new VMware vSphere 4 provides significant performance enhancements that make it easier for organizations to virtualize their most demanding and intense workloads. The storage subsystem is one of the areas where performance has been significantly improved by several architectural enhancements made in vSphere 4.

VMware's virtual platform provides to guest operating systems virtualized versions of hardware storage adapters from BusLogic and LSI Logic, which emulate the physical storage adapters on which they are based. The advantage of this emulation is that most operating systems ship drivers for these devices. However, this design precludes the use of performance optimizations that are possible in virtualized environments.

VMware ESX 4.0™ ships with a new virtual storage adapter—Paravirtualized SCSI (PVSCSI). PVSCSI is a high-performance storage adapter that provides better throughput and lower CPU utilization for virtual machines. It is best suited for environments where guest applications are very I/O intensive.

In this document, we provide a performance comparison of PVSCSI and LSI Logic storage adapters. To evaluate the performance benefit in multiple storage protocols, we select two widely-used storage protocols: Fibre Channel (FC) and software iSCSI (SW iSCSI).

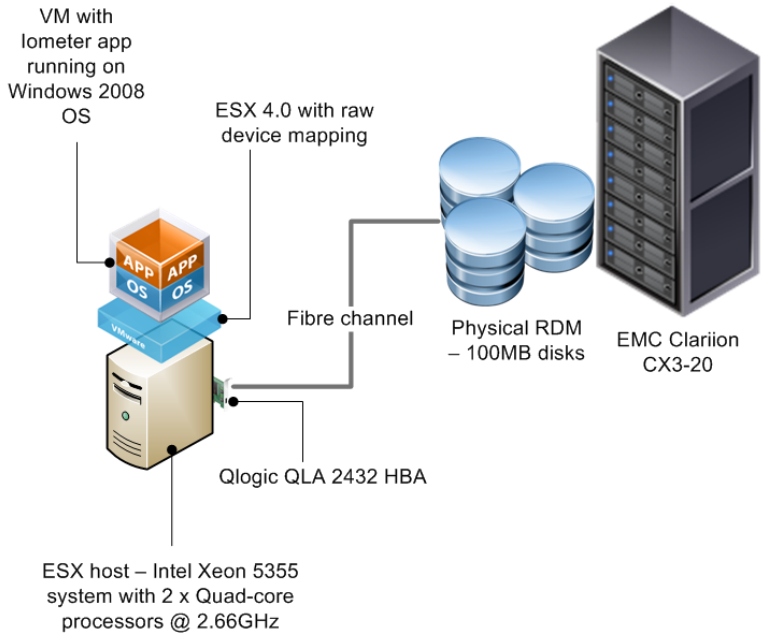
Experimental Environment

System configuration

As an ESX host, we used an Intel Xeon 5355 system which had 2 x Quad-core processors running at 2.66 GHz. We created a single-vCPU virtual machine running a Windows 2008 guest operating system that had 2GB of memory and accessed 100MB RDM-physical (RDMp) disks.

For the FC evaluation, we used an EMC CX3-20 storage connected via a QLogic QLE2432 4Gb HBA.

Figure 1: Fibre Channel configuration



For the SW iSCSI evaluation, we used NetApp FAS6030 filer connected to an Intel Oplin 10Gb NIC.

Figure 2: Software iSCSI configuration

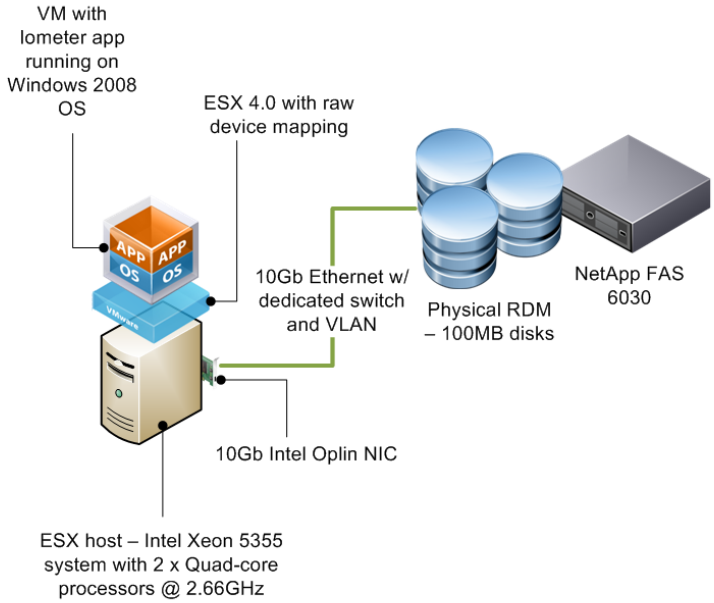


Table 1. ESX host and storage system configuration information

Component	Details
Hypervisor	VMware ESX 4.0
Processors	Two 2.66GHz Quad Core Intel Xeon Processors
Memory	16 GB
NICs for SW iSCSI	10 Gb Intel Oplin
Fibre Channel HBA	QLogic QLA2432
IP Network for SW iSCSI	10 Gb Ethernet with a dedicated switch and VLAN
File System	None, RDM-physical was used
Storage Server/Array	EMC Clariion CX3-20 for Fibre Channel NetApp FAS 6030 for SW iSCSI

Note that no buffer caching is involved in the guest operating system because there's no file system used in the data path of the virtual machine. As such, the amount of memory configured for the virtual machine has no impact on the performance in the experiment.

In the experiment, RDM-physical disks were used to maximize the performance difference between the essential data path of LSI Logic and PVSCSI adapter layers. Note that VMFS and RDM yield very similar performance especially for random reads and writes as described in the previous study (refer to "Performance Characterization of VMFS and RDM Using a SAN" available at http://www.vmware.com/files/pdf/performance_char_vmfs_rdm.pdf). Using VMFS disks would show the similar performance difference between LSI Logic and PVSCSI as observed in the experiment using RDM-physical disks.

Table 2. Virtual machine configuration information

Component	Details
Guest Operating System	Windows Server 2008 RTM 64bit
Number of Virtual Processors	1
Memory	2048 MB
Virtual Disk	100 MB RDM-physical disks are used to achieve "cached runs" effect
File System	None (Physical drives were used)

Workload

We used the latest version of Iometer (version 2006.07.27) as a benchmark. Iometer is a well-known I/O benchmarking tool originally developed at Intel, and distributed under Intel Open Source License (http://sourceforge.net/projects/iometer). The parameters of Iometer were configured such that each test ran for 2 minutes with varying I/O block sizes from 1 KB to 512 KB. A worker thread was created to generate 100% random read operations with 32 outstanding I/Os.

In the experiments, we used two performance metrics: I/O throughput and CPU efficiency. I/O throughput measures how much data can be transferred from and to the physical storage, which is measured in a unit of

megabytes per second (MBps). CPU efficiency measures how many cycles the CPU has to spend to process a single I/O request, which is measured in a unit of cycles per I/O (CPIO). Generally, higher throughput and lower CPIO are considered to be better for a given system configuration.

To focus on the performance difference of two storage adapters, we adopted the "cached run" approach, which is often used when there's a need to minimize latency effects from the physical disk. In such an experimental setup, the entire I/O working set resides in the cache of the storage server or array. Since no mechanical movement is involved in serving a read request, maximum possible read performance can be achieved from the storage device. This also ensures that the degree of randomness in the workload has nearly no effect on throughput or response time and run-to-run variance becomes extremely low.

However, even in cached runs, write performance can still depend on the available disk bandwidth of the storage device. If the incoming rate of write requests outpaces the server's or array's ability to write the dirty blocks to disks, once the write cache is filled and steady state reached, a new write request can be completed only if certain blocks in the cache are physically written to disk and marked as free. For this reason, read performance in cached runs better represents the true performance capability of the host and the storage protocol regardless of the type of storage server or array used.

Results

Fibre Channel

This section compares the performance of PVSCSI and LSI Logic with Fibre Channel. To achieve maximum throughput, we created two worker threads. Each thread issued 100% random read operations with 32 outstanding I/Os.

Figure 3 illustrates the read throughput in megabytes per second. Both PVSCSI and LSI Logic throughputs are increasing as the I/O block size becomes larger until the storage and wire bandwidth limits are reached at or above 16KB block size. Figure 1 shows that PVSCSI is 18%, 13%, and 7% better than LSI Logic in throughput with 1KB, 2KB, and 4KB I/O block sizes, respectively.

Figure 3. Random read throughput for Fibre Channel (higher is better)

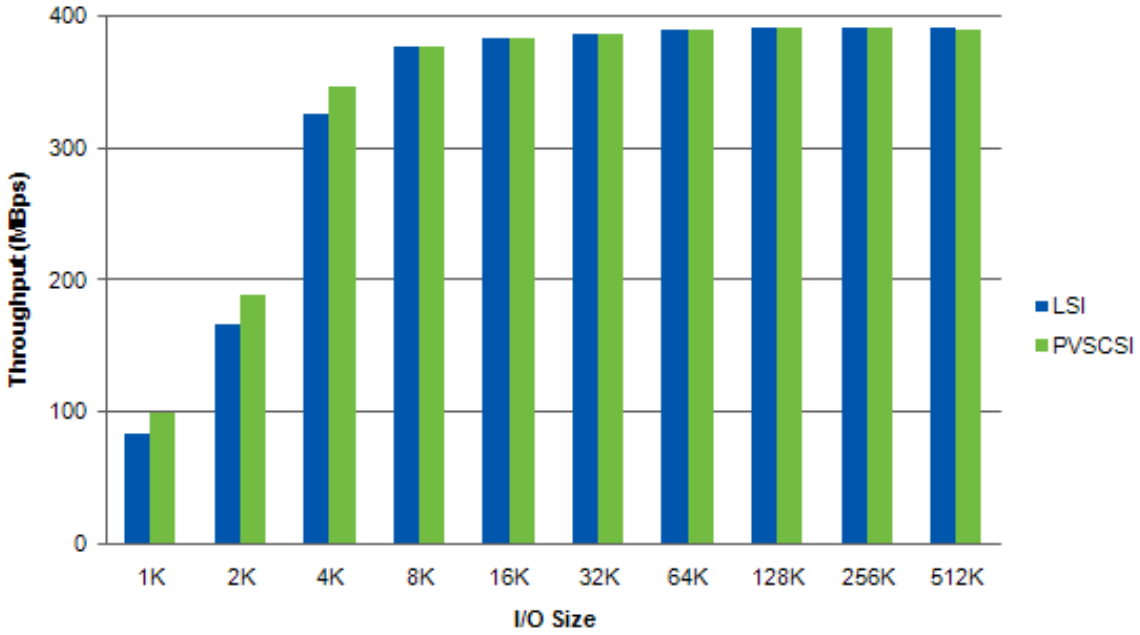
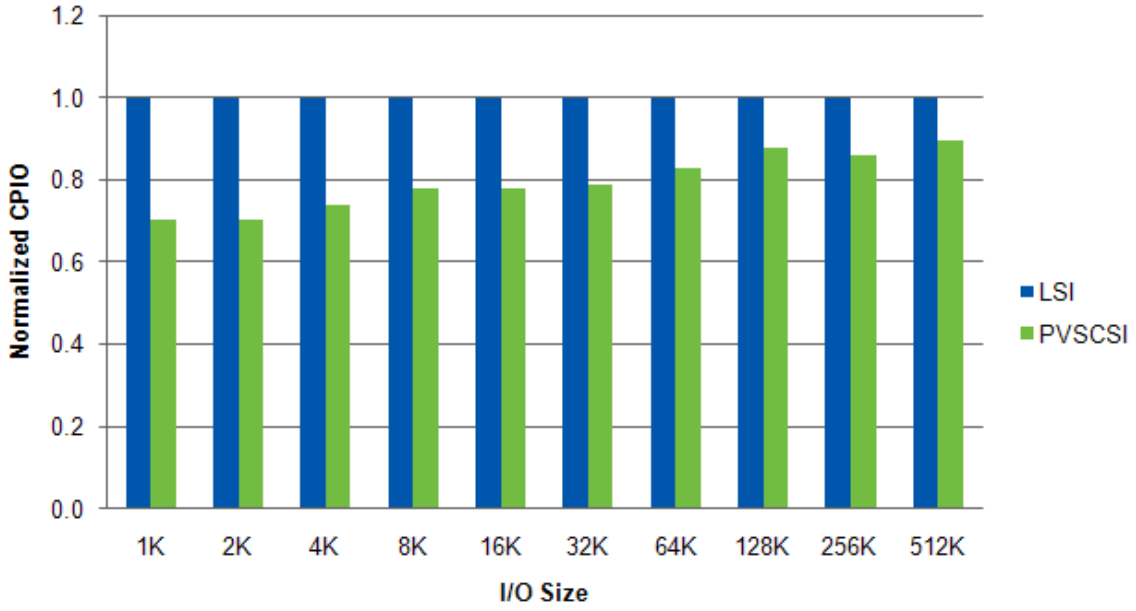


Figure 4 illustrates the CPIO of PVSCSI normalized to LSI Logic. Figure 2 shows that PVSCSI reduces the CPIO by 10%~30% compared to LSI Logic. It is observed that PVSCSI provides much better CPIO efficiency than LSI Logic with all the block sizes from 1KB to 512KB.

The Fibre Channel results indicate that PVSCSI provides much better CPU efficiency than LSI for all the I/O block sizes. Note that the advantage of PVSCSI in throughput can be observed only when the workload is driving very high I/O rates. The difference in throughput may not be observed when the workload is driving low I/O rates or when the storage and wire bandwidth limits are reached.

Figure 4. Normalized CPIO for Fibre Channel (lower is better)



SW iSCSI

This section compares the performance of PVSCSI and LSI Logic with 10Gb SW iSCSI. On the 10Gb TCP/IP protocol, the standard MTU of 1500 bytes was used because it is the most widely used size of protocol data unit.

Figure 5. Random read throughput for 10Gb SW iSCSI (higher is better)

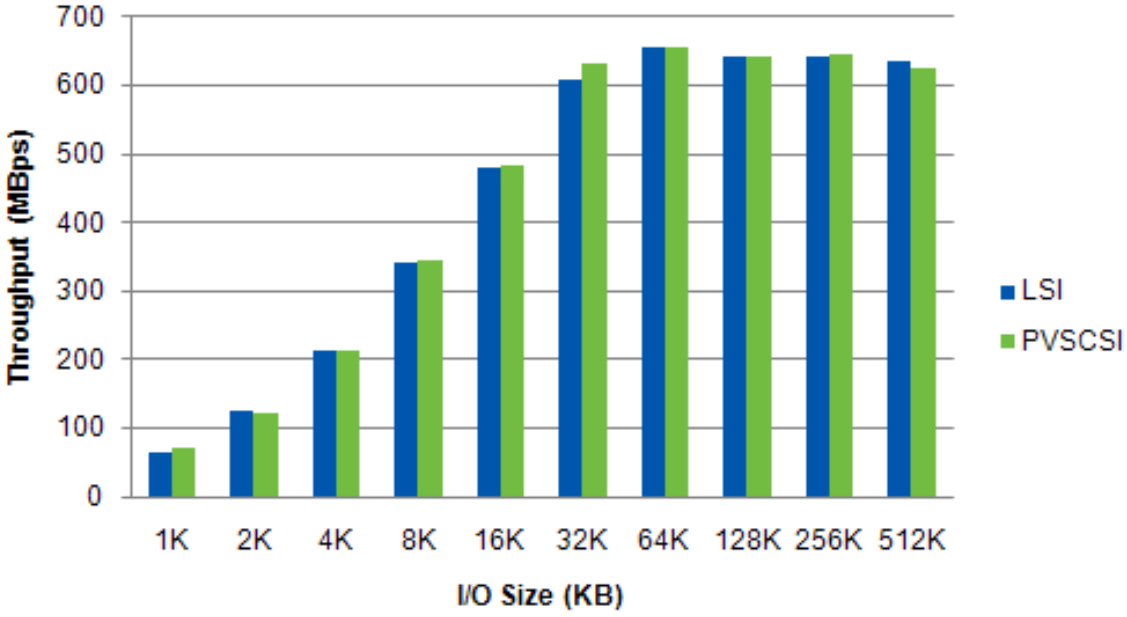
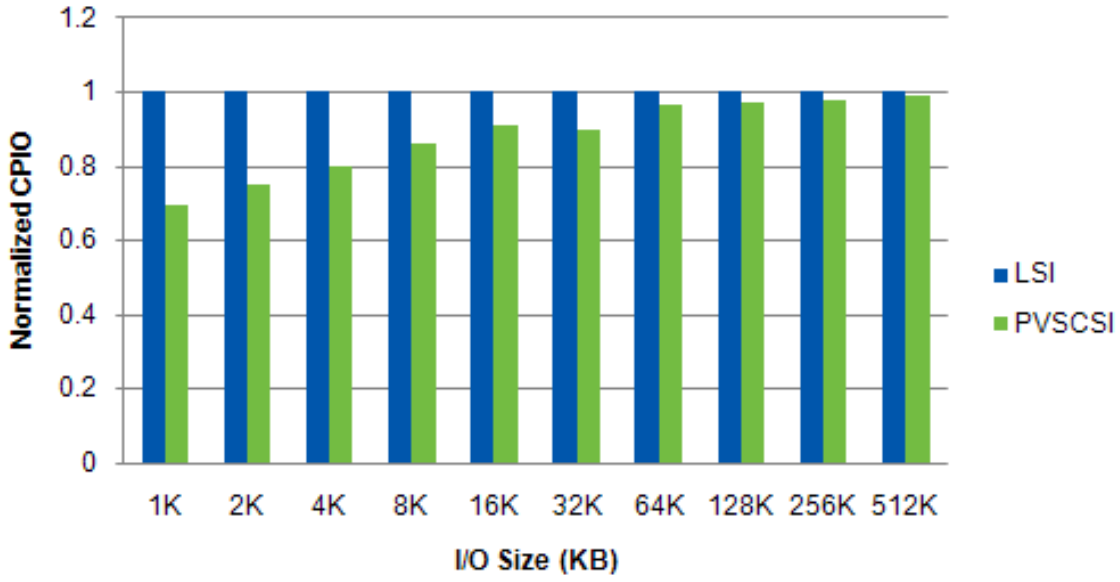


Figure 5 (above) illustrates the read throughput in megabyte per seconds. Both PVSCSI and LSI Logic throughputs are increasing as the I/O block size becomes larger until throughput reaches the wire bandwidth limit. PVSCSI shows similar or better throughput over LSI Logic in most I/O block sizes from 1KB to 512KB. For example, PVSCSI outperforms LSI Logic by up to 9% in the 1KB I/O block size.

Figure 6 illustrates the CPIO of PVSCSI normalized to LSI Logic. In Figure 4 PVSCSI reduces the CPIO by 1%~25% compared to LSI Logic. Note that a greater reduction in CPIO can be observed in small I/O block sizes from 1KB to 8KB, because the benefit of PVSCSI becomes larger as the workload drives higher I/O rates.

Figure 6. Normalized CPIO for 10Gb SW iSCSI (lower is better)



The experiment results in this section show that PVSCSI greatly improves the CPU efficiency and provides better throughput for very heavy I/O workloads. For certain workloads, however, the ESX 4.0 implementation of PVSCSI may have a higher latency than LSI Logic if the workload drives low I/O rates. This is because the ESX 4.0 design of PVSCSI coalesces based on outstanding I/Os and not throughput. So when the virtual machine is requesting a lot of I/O but the storage is not delivering it, the PVSCSI driver is coalescing interrupts.[1]

Conclusion

This document provides a performance comparison of the PVSCSI and LSI Logic storage adapters. The results show that PVSCSI greatly improves the CPU efficiency as was demonstrated in the experiments using Fibre Channel and SW iSCSI storage protocols. PVSCSI also improves the throughput when the workload drives very high I/O rates.

The experiment results also demonstrate that, although PVSCSI provides better CPU efficiency, LSI Logic is able to provide wire-speed throughput, indicating that the storage adapter is not the limiting factor in I/O performance if the system is properly configured.

About the Authors

Moon Chang is a Senior Member of Technical Staff at VMware. As a member of the R&D performance engineering team, he works on the performance analysis and optimization of virtualization and cloud computing software focusing on hypervisor and storage I/O subsystems. He received his PhD degree in Computer Science from Seoul National University, South Korea. He has years of extensive industry and research experience in systems software areas including IBM AIX and Linux operating system kernels, NUMA and scalable server architectures, Cell Broadband Engine and multi-core architectures.

[1] VMware plans to update PVSCSI driver functionality so that it coalesces interrupts based on outstanding I/Os (OIOs) and on throughput (IOPS). Information about the updated driver will be included in http://kb.vmware.com/kb/1017652.

Jinpyo Kim is a Senior Performance Engineer at VMware, where he focuses on optimizing the performance of storage I/Os in a virtualized environment. He received his PhD in Computer Science from University of Minnesota-Minneapolis. Prior to joining VMware, he gained experience in developing I/O subsystems in servers and parallel SCSI, FC, and networked RAID controllers.

Acknowledgements

The PVSCSI project was a collaborative work between the core storage and storage performance teams. The authors would like to thank Maxime Austruy and Glen McCready for providing deep insights and knowledge in core storage development. The authors also would like to thank Scott Drummonds and Chethan Kumar for their comments, and Bing Tsai, Vikram Makhija, and Jennifer Anderson for their support for making this project successful.