



Technical Report

VMware vSphere and ESX 3.5 Multiprotocol Performance Comparison Using FC, iSCSI, and NFS

Jack McLeod, Saad Jafri, and Karun Nijaguna, NetApp
January 2010 | TR-3808-0110

ABSTRACT

This document compares the performance of 4GbE FC, 1GbE and 10GbE iSCSI, and 1GbE and 10GbE NFS protocols using both VMware® vSphere™ 4.0 and ESX 3.5 on NetApp® storage systems. This document seeks to compare the individual protocol performance and CPU utilization at varying workloads.

TABLE OF CONTENTS

1	INTRODUCTION	4
1.1	EXECUTIVE SUMMARY	4
1.2	PURPOSE	4
1.3	METHODOLOGY	4
2	PERFORMANCE SUMMARY	6
2.1	COMPARING VSPHERE AND ESX 3.5 PROTOCOL PERFORMANCE AND EFFICIENCY	6
2.2	COMPARING PROTOCOL PERFORMANCE IN VSPHERE	6
3	RELATIVE PERFORMANCE COMPARISON OF VSPHERE AND ESX 3.5 USING A 4K WORKLOAD 8	
3.1	RELATIVE THROUGHPUT AND CPU UTILIZATION COMPARISON USING FC	8
3.2	RELATIVE THROUGHPUT AND CPU UTILIZATION COMPARISON USING NFS	9
3.3	RELATIVE THROUGHPUT AND CPU UTILIZATION COMPARISON USING ISCSI	10
4	RELATIVE PERFORMANCE COMPARISON OF VSPHERE AND ESX 3.5 USING AN 8K WORKLOAD 12	
4.1	RELATIVE THROUGHPUT AND CPU UTILIZATION COMPARISON USING FC	12
4.2	RELATIVE THROUGHPUT AND CPU UTILIZATION COMPARISON USING NFS	13
4.3	RELATIVE THROUGHPUT AND CPU UTILIZATION COMPARISON USING ISCSI	14
5	PERFORMANCE COMPARISON USING MULTIPLE PROTOCOLS IN VSPHERE WITH A 4K WORKLOAD, 4GB FC, AND GIGABIT ETHERNET	15
5.1	RELATIVE THROUGHPUT COMPARISON	15
5.2	RELATIVE CPU UTILIZATION COMPARISON	15
5.3	RELATIVE LATENCY COMPARISON	16
6	PERFORMANCE COMPARISON USING MULTIPLE PROTOCOLS IN VSPHERE WITH AN 8K WORKLOAD, 4GB FC, AND GIGABIT ETHERNET	18
6.1	RELATIVE THROUGHPUT COMPARISON	18
6.2	RELATIVE ESX SERVER CPU UTILIZATION COMPARISON	18
6.3	RELATIVE LATENCY COMPARISON	19
7	PERFORMANCE COMPARISON USING MULTIPLE PROTOCOLS IN VSPHERE WITH A 4K WORKLOAD, 4GB FC, AND 10 GIGABIT ETHERNET	21
7.1	RELATIVE THROUGHPUT COMPARISON	21
7.2	RELATIVE ESX SERVER CPU UTILIZATION COMPARISON	21
7.3	RELATIVE LATENCY COMPARISON	22
8	PERFORMANCE COMPARISON USING MULTIPLE PROTOCOLS IN VSPHERE WITH AN 8K WORKLOAD, 4GB FC, AND 10 GIGABIT ETHERNET	24
8.1	RELATIVE THROUGHPUT COMPARISON	24
8.2	RELATIVE ESX SERVER CPU UTILIZATION COMPARISON	24
8.3	RELATIVE LATENCY COMPARISON	25

9	TEST DESIGN AND CONFIGURATION	27
9.1	HARDWARE AND SOFTWARE ENVIRONMENT	27
9.2	CONNECTING NETAPP STORAGE TO THE VMWARE DATA CENTER	28
9.3	PROVISIONING NETAPP STORAGE TO THE VMWARE DATA CENTER	31
9.4	CONFIGURING VMWARE ESX 3.5 AND VSPHERE DATA CENTER	35
9.5	WORKLOAD DEFINITION AND IOMETER ENVIRONMENT	36
10	REFERENCES	38
11	ACKNOWLEDGEMENTS	38
12	FEEDBACK	38

1 INTRODUCTION

NetApp storage arrays and VMware vSphere natively support data access using FC, iSCSI, and NFS protocols. Because of the deployment and management differences in each protocol, determining which of these three protocols to use is one of the key steps in designing a virtualized infrastructure. With this in mind, knowing how each protocol performs in terms of throughput and CPU utilization can assist in making this important design consideration.

1.1 EXECUTIVE SUMMARY

The protocol performance analysis shown in this technical report demonstrates that vSphere on NetApp is capable of providing the performance required for any type of production environment with any of the protocols tested. Listed below are several key items found during the tests that illustrate this point.

- In the tests it was shown that with vSphere VMware improved both iSCSI and NFS in terms of performance with all protocols when measured against FC fixed path, performing within 5% of each other throughout the tests.
- VMware significantly improved CPU efficiency with both iSCSI and NFS in vSphere, with iSCSI showing the highest single improvement of all the protocols.
- When FC fixed path was compared to FC with round-robin (a feature supported natively by NetApp FAS controllers) load balancing performance was shown to be very similar throughout all tests.

Also, please note that these tests were not designed to demonstrate the potential maximum bandwidth available for each protocol, but to simulate real-world environments using low, medium, and high levels of throughput.

1.2 PURPOSE

This technical report, completed jointly by VMware and NetApp, shares the results of testing conducted to compare the performance of FC, software-initiated iSCSI, and NFS in an ESX 3.5 and vSphere environment using NetApp storage. The results compare the performance of the three protocols with a goal of aiding customer decisions as they build out their virtual infrastructures while also demonstrating the protocol enhancements made from ESX 3.5 to vSphere.

The performance tests sought to simulate a “real-world” environment. The test and validation environment is composed of components and architectures commonly found in a typical VMware implementation that include using the FC, iSCSI, and NFS protocols in a multiple virtual machine (VM), multiple ESX 3.5 and/or vSphere host environment accessing multiple data stores. The performance tests used realistic I/O patterns, I/O block sizes, read/write mixes, and I/O loads common to various operating systems and business applications, such as Windows® Server 2008, Windows Vista, Microsoft® SharePoint® 2007, and Microsoft Exchange 2007.

1.3 METHODOLOGY

During the tests we measured the total throughput generated using each protocol at a variety of points simulating low, typical, and heavy workloads as experienced by ESX and vSphere environments. While a typical ESX or vSphere environment might not be driven to these levels, it is valuable to know how the protocols behave at extremely high levels of activity.

We configured a VMware data center consisting of eight ESX host systems on Rackable Solutions S44 servers. We installed either ESX 3.5 or vSphere on the host systems, depending on the test configurations. In all cases, each ESX host was configured with a total of 20 VMs running 32-bit Windows Server 2003 Enterprise Edition and SP2. Additionally, we used two NetApp FAS3170 controllers in a cluster failover (CFO) configuration to provide storage for the data accessed by the VMs during the performance testing. We used the identical test infrastructure and loads for all three protocols under test. Please consult Figures 25, 26, and 27 for a diagram of the environment.

Once the environment was set up we used the industry standard Iometer benchmark to measure the performance of each protocol using workloads ranging from light to heavy amounts of I/O from the ESX and

vSphere hosts to the NetApp FAS3170 storage used for the tests. The tests were conducted using a workload consisting of a 4K or 8K request size, 75% read, 25% writes, and 100% random access with an increasing number of virtual machines ranging from 32, 96, and 160 executing 128, 384, and 640 total outstanding I/Os, respectively. Each VM ran the lometer dynamo application and was configured to generate a constant four outstanding I/Os spread evenly across either eight VMware ESX 3.5 servers or eight VMware vSphere host servers.

Following are the primary sets of test data presented in this document:

- **vSphere performance relative to ESX 3.5:** Using both ESX 3.5 and vSphere, we compared the performance and host CPU utilization of vSphere relative to ESX 3.5 for FC using fixed data paths, iSCSI, and NFS with Gigabit Ethernet. Because jumbo frames were not officially supported on ESX 3.5 there are no iSCSI or NFS results presented using jumbo frames with these results.

When comparing relative throughput between ESX 3.5 and vSphere, a baseline value of 100 is used to represent the throughput generated when using a specific protocol in ESX 3.5. The test results using vSphere are represented as a percentage relative to the observed throughput of the baseline generated using ESX 3.5. vSphere values greater than 100 indicate vSphere throughput exceeded that of ESX 3.5. vSphere values less than 100 indicate vSphere throughput lower than ESX 3.5.

When comparing relative CPU utilization between ESX 3.5 and vSphere, a baseline value of 100 is used to represent the average ESX 3.5 CPU utilization observed when using a specific protocol in ESX 3.5. The test results using vSphere are represented as a percentage relative to the observed ESX CPU utilization using ESX 3.5. vSphere values greater than 100 indicate vSphere consumed more ESX CPU resources compared to ESX 3.5. vSphere values less than 100 indicate vSphere consumed less ESX CPU resources compared to ESX 3.5.

- **vSphere performance comparison for all protocols with FC, Gigabit Ethernet (GbE) and 10 Gigabit Ethernet (10GbE):** Using only vSphere, we compared the performance, host CPU utilization, and latency of FC using round-robin load balancing, iSCSI, and NFS with 1GbE and 10GbE with and without jumbo frames relative to FC using fixed paths.

For our tests with FC, all eight of the vSphere hosts contained two primary paths to the LUNs servicing the VM's. As a result we conducted tests using both the fixed and round robin path selection policies supported by vSphere. When using the fixed path selection policy we configured one of the paths as the preferred path and the other as an alternative path. In this case, the vSphere host always uses the preferred path to the disk when that path is available. If the vSphere host cannot access the disk through the preferred path, it tries the alternative path(s). This is the default policy for active-active storage devices.

When testing with the round-robin path selection policy enabled, the vSphere host uses an automatic path selection algorithm and rotates through the available paths. In our tests, this option implements load balancing across both of the available physical paths. Load balancing is the process of spreading server I/O requests across all available host paths with the goal of optimizing performance.

When comparing relative throughput using vSphere, a baseline value of 100 is used to represent the throughput generated using FC with fixed data paths. The throughput generated using other protocol configurations are represented as a percentage relative to the observed throughput of the baseline. Results generated using protocols other than FC with fixed data paths are represented as a percentage relative to the baseline. Values greater than 100 indicate throughput exceeding that of FC with fixed paths. Values less than 100 indicate throughput lower than FC with fixed paths.

When comparing relative CPU utilization using vSphere, a baseline value of 100 is used to represent the average ESX CPU utilization observed using FC with fixed data paths. The average ESX CPU utilization observed using other protocol configurations are represented as a percentage relative to the average ESX CPU utilization observed using FC with fixed data paths. Values greater than 100 indicate average ESX CPU utilization exceeding that observed using FC with fixed paths. Values less than 100 indicate average ESX CPU utilization lower than that observed using FC with fixed paths.

When comparing relative latencies using vSphere, a baseline value of 100 is used to represent the average latencies reported by lometer using FC with fixed data paths. The average latencies observed using other protocol configurations are represented as a percentage relative to the average latencies observed using FC with fixed data paths. Values greater than 100 indicate average latencies exceeding that observed using FC with fixed paths. Values less than 100 indicate average latencies lower than that observed using FC with fixed paths.

2 PERFORMANCE SUMMARY

This section presents a high-level summary of the results of our testing as described above. For complete test results for all test cases please refer to the remainder of this report.

As stated above, these test cases were not designed to highlight the differences between the protocols from a raw bandwidth perspective. The smaller 4K and 8K requests generated large numbers of requests (that is, IOPS) but don't necessarily move amounts of data large enough to saturate the GigE and 10GigE networks or FC connections. For example, with a given load we would not expect to see FC deliver twice the numbers of IOPS compared to iSCSI or NFS.

2.1 COMPARING VSPHERE AND ESX 3.5 PROTOCOL PERFORMANCE AND EFFICIENCY

This section provides a high level summary of our tests comparing throughput and protocol efficiency between ESX 3.5 and vSphere. For these tests we consider a protocol to be more efficient compared to others if it consumes less ESX CPU resources when generating a specific level of throughput. We found the following items to be of interest:

- With respect to overall performance as measured in IOPS, we found that vSphere delivered comparable performance to ESX 3.5 for all the configurations we tested. With respect to CPU utilization, we found that vSphere consumed approximately 6% to 23% less ESX host CPU resources when using either FC or NFS, depending on the load.
- With regard to the iSCSI protocol, we found that vSphere consumed approximately 35% to 43% less ESX host CPU resources compared to ESX 3.5, depending on the load.
- When using ESX 3.5, iSCSI consumed approximately 26% to 42% more ESX host CPU resources compared to NFS, depending on the load. With vSphere, ESX host CPU resources consumed using iSCSI were approximately 5% to 6% lower compared to NFS.

Overall, our tests showed vSphere delivered comparable performance and greater protocol efficiency compared to ESX 3.5 for all test configurations. With the significant gains in protocol efficiency for iSCSI, both NetApp and VMware believe this will provide our joint customers with enhanced flexibility when designing their vSphere environments.

2.2 COMPARING PROTOCOL PERFORMANCE IN VSPHERE

This section provides a high level summary of our tests comparing throughput, ESX CPU utilization and latencies in vSphere only. We found the following items to be of interest:

- We observed performance with all protocols in vSphere to be within approximately 9% of the performance generated using FC with fixed data paths.
- Comparing CPU utilization on vSphere only, we found that NFS and iSCSI consumed 10% to 45% more ESX CPU resources compared to FC using fixed data paths. This was true whether using Gigabit Ethernet or 10 Gigabit Ethernet with NFS and iSCSI.
- We observed that performance generated using FC with round-robin load balancing was comparable to that observed using FC configured with fixed data paths for all workloads. Additionally, we found that using FC with round-robin load balancing consumed slightly more vSphere CPU resources compared to using FC configured with fixed data paths.
- Overall average latencies for all test cases closely tracked the performance differences as measured in IOPS between the protocols. This is expected as all test cases were run for the same time duration and, in general, higher numbers of IOPS map directly to lower overall average latencies.

vSphere now officially supports jumbo frames with NFS and iSCSI. As a result a series of additional tests were conducted using the same workloads described in section 1.2 above using both NFS and iSCSI with jumbo frames enabled for both Gigabit and 10GbE to determine the effect on performance and protocol efficiency of using jumbo frames. Due to the smaller request sizes used in the workloads, it was not

expected that enabling jumbo frames would improve overall performance. Our observations after running these tests are as follows:

- Using NFS with jumbo frames enabled using both Gigabit and 10GbE generated overall performance that was comparable to that observed using NFS without jumbo frames and required approximately 6% to 20% fewer ESX CPU resources compared to using NFS without jumbo frames, depending on the test configuration.
- Using iSCSI with jumbo frames enabled using both Gigabit and 10GbE generated overall performance that was comparable to slightly lower than that observed using iSCSI without jumbo and required approximately 12% to 20% fewer ESX CPU resources compared to using iSCSI without jumbo frames depending on the test configuration.

NetApp and VMware believe these tests validate that FCP, iSCSI, and NFS storage protocols using Gigabit and 10 Gigabit Ethernet are production ready, even for mission-critical applications.

3 RELATIVE PERFORMANCE COMPARISON OF VSPHERE AND ESX 3.5 USING A 4K WORKLOAD

3.1 RELATIVE THROUGHPUT AND CPU UTILIZATION COMPARISON USING FC

Figure 1 below compares the performance of vSphere using FC with fixed data paths relative to the performance of ESX 3.5 using FC with fixed data paths as the baseline. We found that performance generated with vSphere using FC was as good to slightly better than that observed with ESX 3.5 using FC.

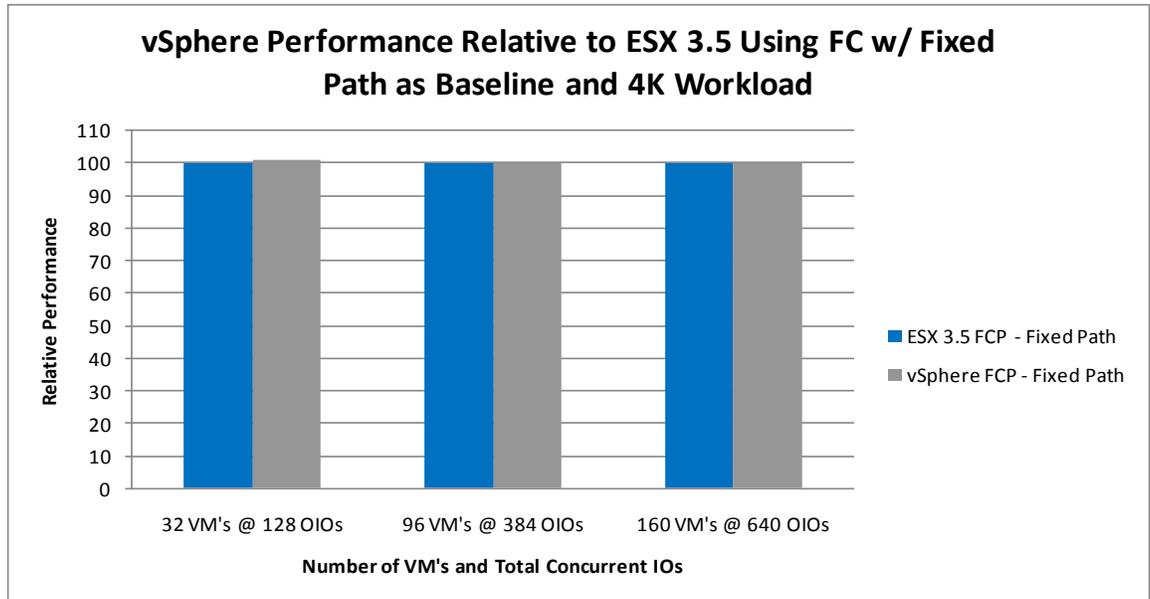


Figure 1) vSphere performance relative to ESX 3.5 using 4Gb FC and 4K request size.

Figure 2 below compares the average ESX host CPU utilization observed in vSphere using FC with fixed data paths relative to the average ESX host CPU utilization observed in ESX 3.5 using FC with fixed data paths as the baseline. We found that average CPU utilization in vSphere was approximately 9% to 11% lower compared to ESX 3.5 using FC.

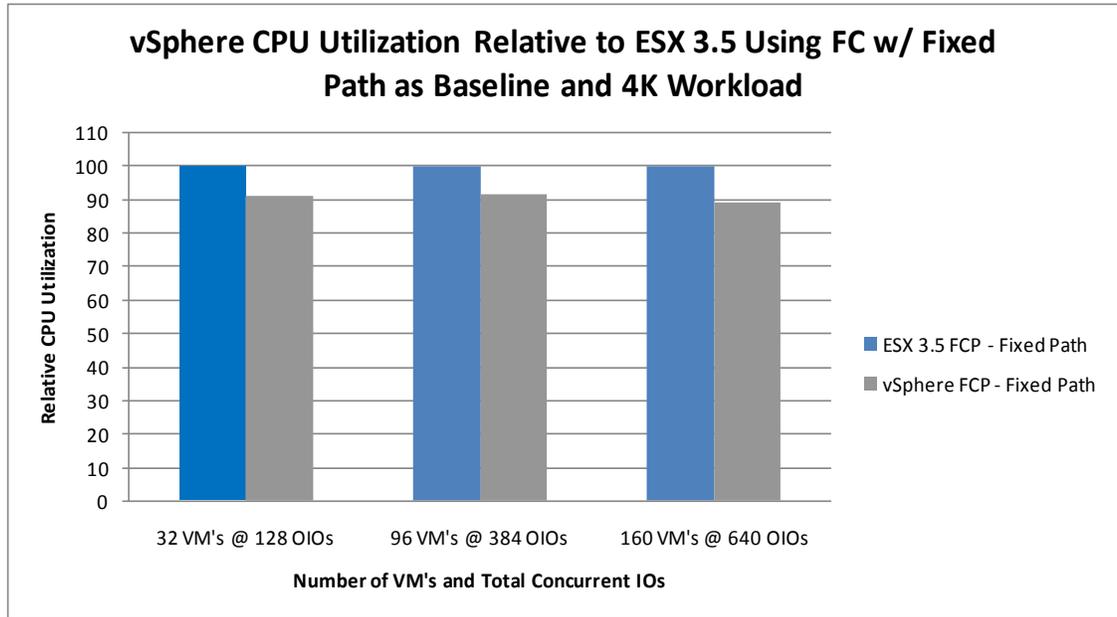


Figure 2) vSphere CPU utilization relative to ESX 3.5 using 4Gb FC and 4K request size.

3.2 RELATIVE THROUGHPUT AND CPU UTILIZATION COMPARISON USING NFS

Figure 3 below compares the performance of vSphere using NFS and gigabit ethernet relative to the performance of ESX 3.5 using NFS and gigabit ethernet as the baseline. We found that performance generated with vSphere using NFS was comparable to that observed with ESX 3.5 using NFS.

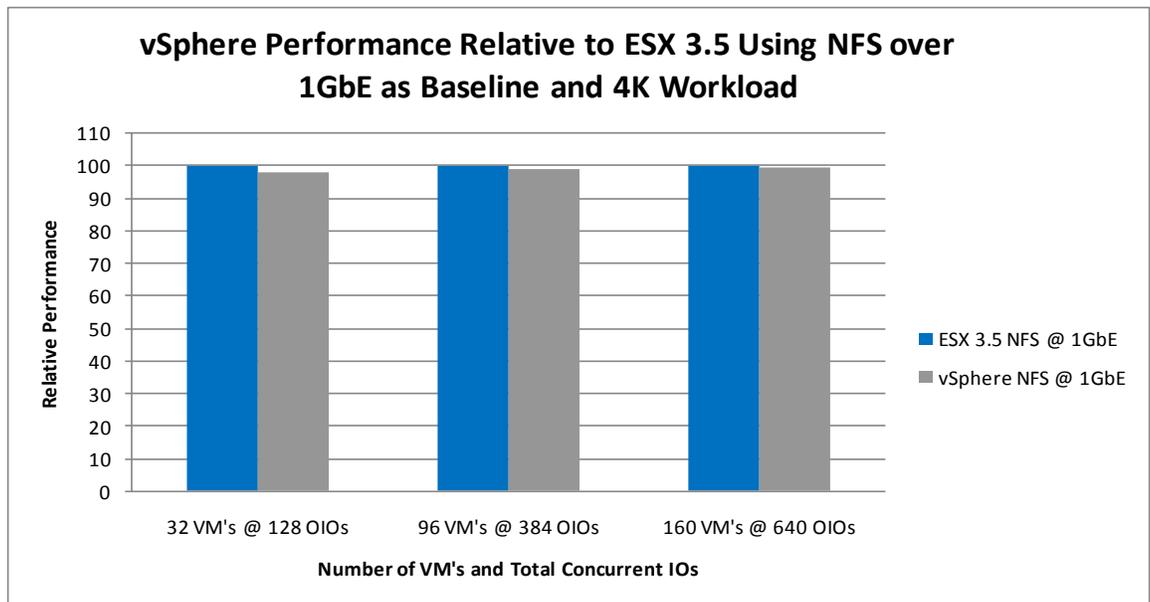


Figure 3) vSphere performance relative to ESX 3.5 using NFS over 1GbE and 4K request size.

Figure 4 below compares the average ESX host CPU utilization observed in vSphere using NFS and gigabit ethernet relative to the average ESX host CPU utilization observed in ESX 3.5 using NFS and gigabit ethernet as the baseline. We found that average CPU utilization in vSphere was approximately 6% to 14% lower compared to ESX 3.5 using NFS.

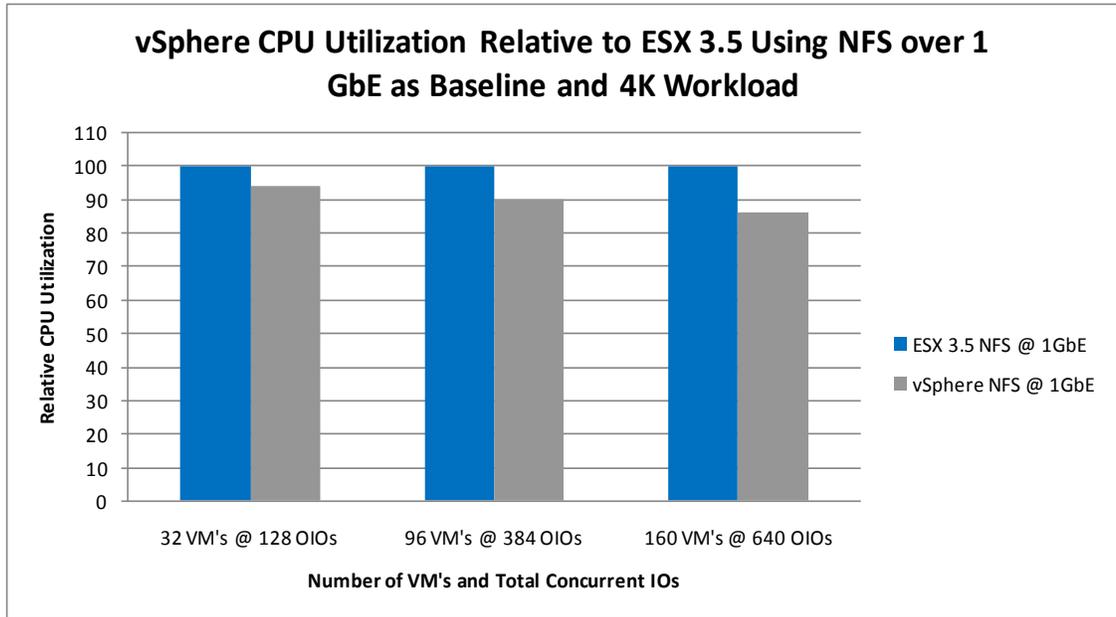


Figure 4) vSphere CPU utilization relative to ESX 3.5 using NFS over 1GbE and 4K request size.

3.3 RELATIVE THROUGHPUT AND CPU UTILIZATION COMPARISON USING ISCSI

Figure 5 below compares the performance of vSphere using iSCSI and gigabit ethernet relative to the performance of ESX 3.5 using iSCSI and gigabit ethernet as the baseline. We found that performance generated with vSphere using iSCSI was comparable to that observed with ESX 3.5 using iSCSI.

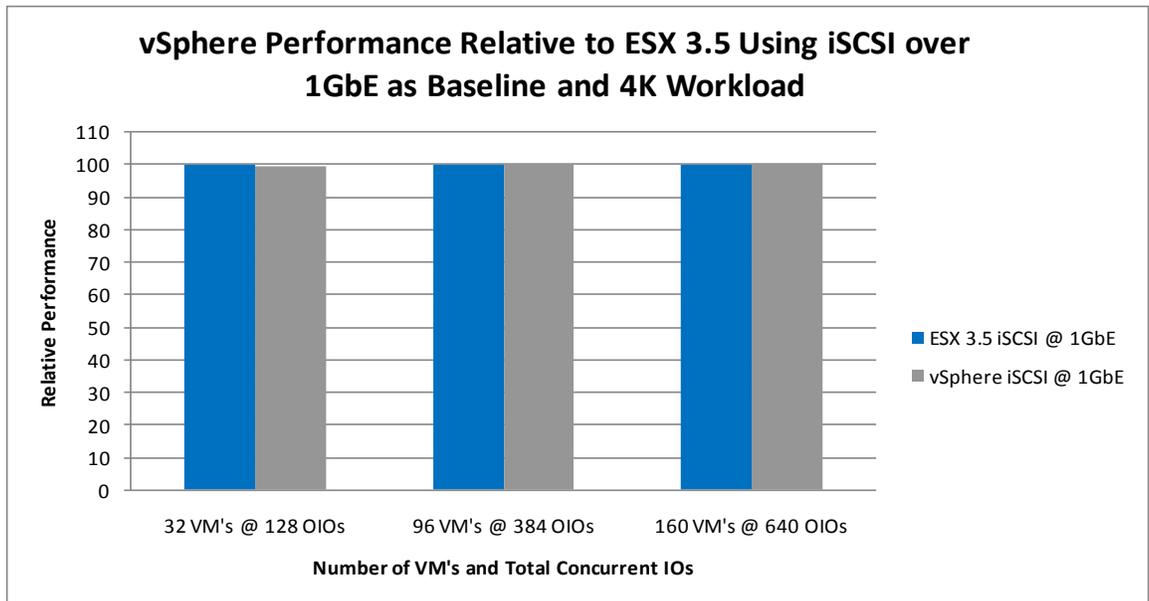


Figure 5) vSphere performance relative to ESX 3.5 using iSCSI over 1GbE and 4K request size.

Figure 6 below compares the average ESX host CPU utilization observed in vSphere using iSCSI and gigabit ethernet relative to the average ESX host CPU utilization observed in ESX 3.5 using iSCSI and gigabit ethernet as the baseline. We found that average CPU utilization in vSphere was approximately 35% lower compared to ESX 3.5 using iSCSI.

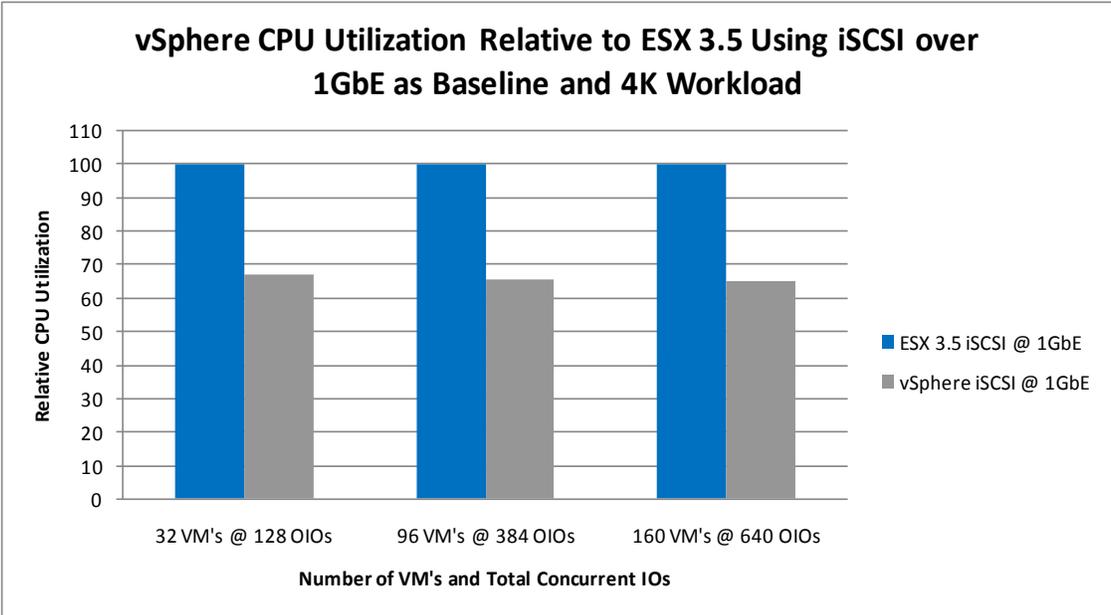


Figure 6) vSphere CPU utilization relative to ESX 3.5 using iSCSI over 1GbE and 4K request size.

4 RELATIVE PERFORMANCE COMPARISON OF VSPHERE AND ESX 3.5 USING AN 8K WORKLOAD

4.1 RELATIVE THROUGHPUT AND CPU UTILIZATION COMPARISON USING FC

Figure 7 below compares the performance of vSphere using FC with fixed data paths relative to the performance of ESX 3.5 using FC with fixed data paths as the baseline. We found that performance generated with vSphere using FC with fixed data paths was comparable to that observed with ESX 3.5 using FCP with fixed data paths.

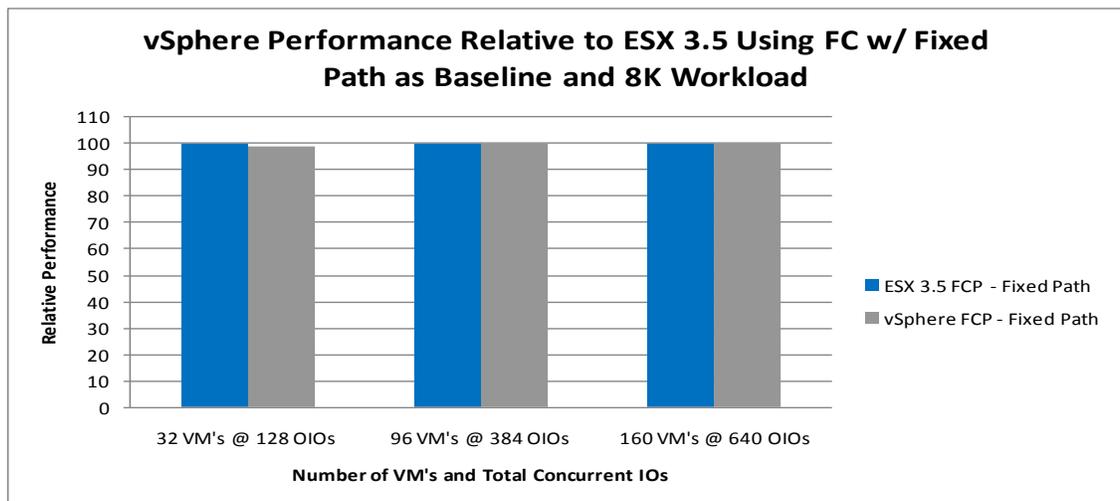


Figure 7) vSphere performance relative to ESX 3.5 using 4Gb FC and 8K request size.

Figure 8 below compares the average ESX host CPU utilization observed in vSphere using FC fixed data paths relative to the average ESX host CPU utilization observed in ESX 3.5 using FC fixed data paths as the baseline. We found that average CPU utilization in vSphere was approximately 3% to 8% lower compared to ESX 3.5 using FC.

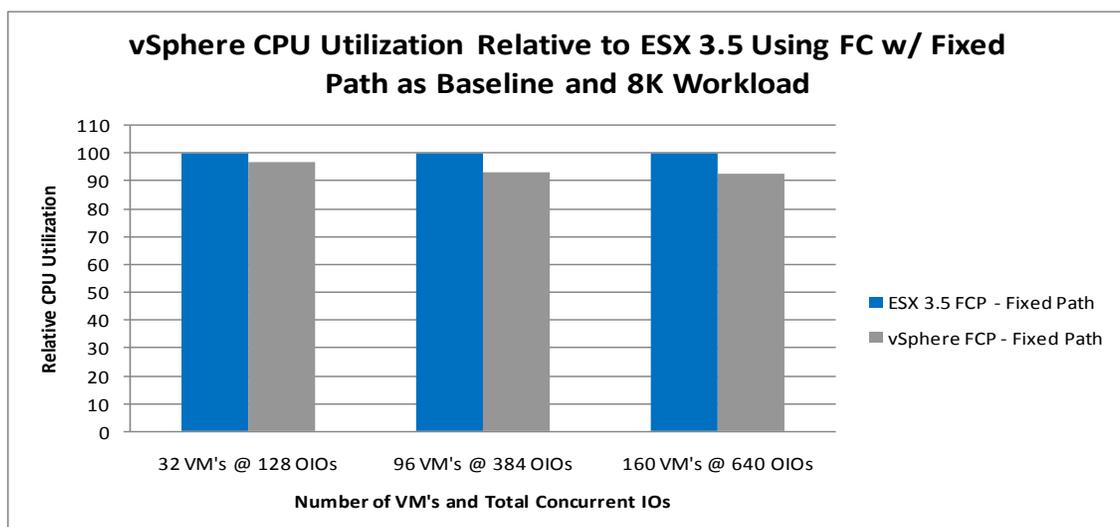


Figure 8) vSphere CPU utilization relative to ESX 3.5 using 4Gb FC and 8K request size.

4.2 RELATIVE THROUGHPUT AND CPU UTILIZATION COMPARISON USING NFS

Figure 9 below compares the performance of vSphere using NFS and gigabit ethernet relative to the performance of ESX 3.5 using NFS and gigabit ethernet as the baseline. We found that performance generated with vSphere using NFS was comparable to that observed with ESX 3.5 using NFS.

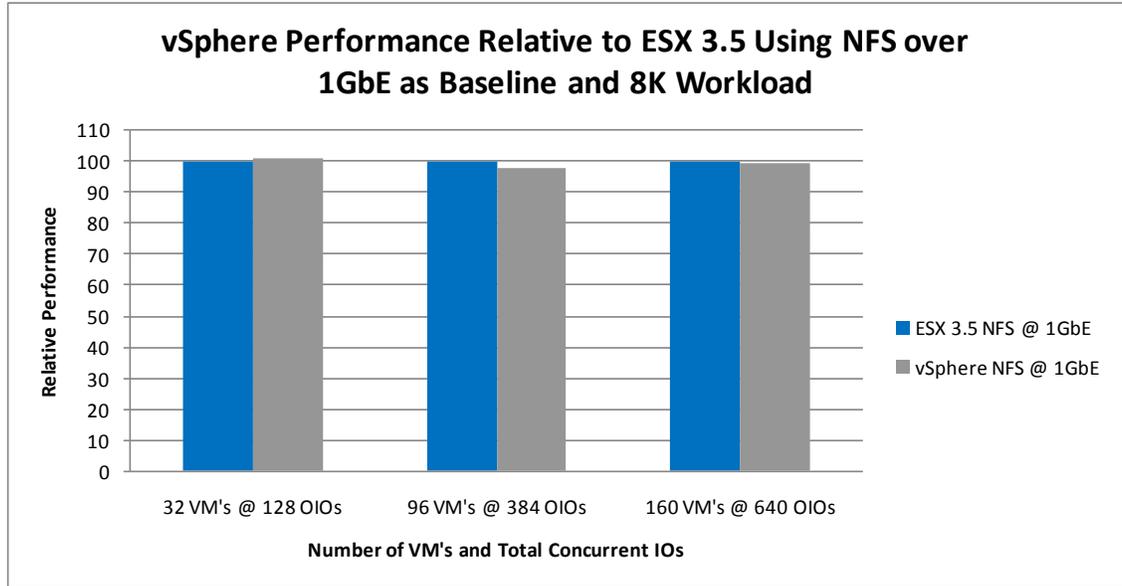


Figure 9) vSphere performance relative to ESX 3.5 using NFS over 1GbE and 8K request size.

Figure 10 below compares the average ESX host CPU utilization observed in vSphere using NFS and gigabit ethernet relative to the average ESX host CPU utilization observed in ESX 3.5 using NFS and gigabit ethernet as the baseline. We found that average CPU utilization in vSphere was approximately 12% to 23% lower compared to ESX 3.5 using NFS.

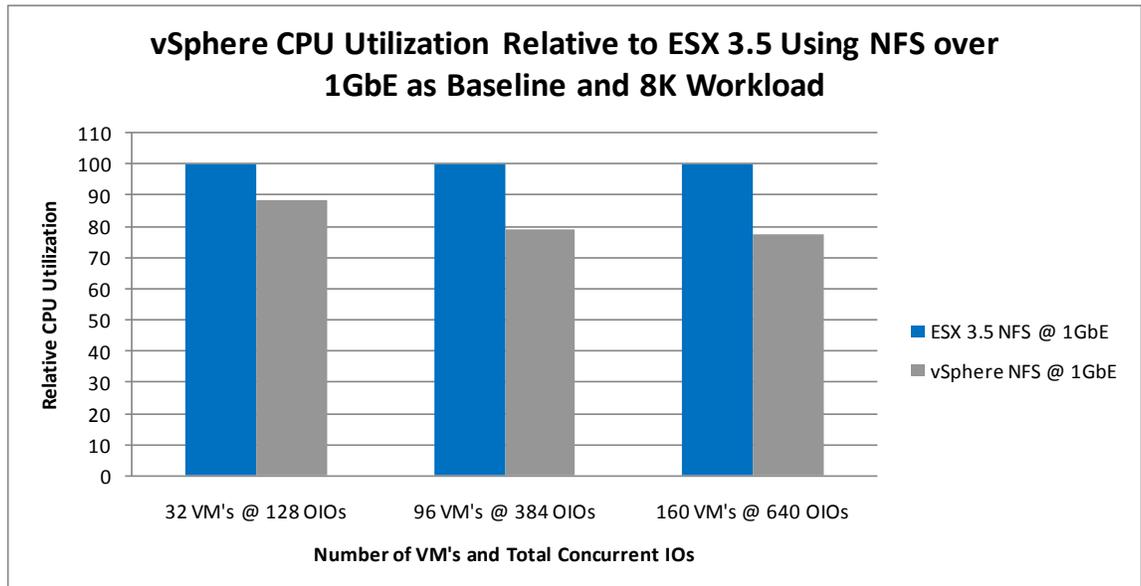


Figure 10) vSphere CPU utilization relative to ESX 3.5 using NFS over 1GbE and 8K request size.

4.3 RELATIVE THROUGHPUT AND CPU UTILIZATION COMPARISON USING ISCSI

Figure 11 below compares the performance of vSphere using iSCSI and gigabit ethernet relative to the performance of ESX 3.5 using iSCSI and gigabit ethernet as the baseline. We found that performance generated with vSphere using iSCSI was comparable to that observed with ESX 3.5 using iSCSI.

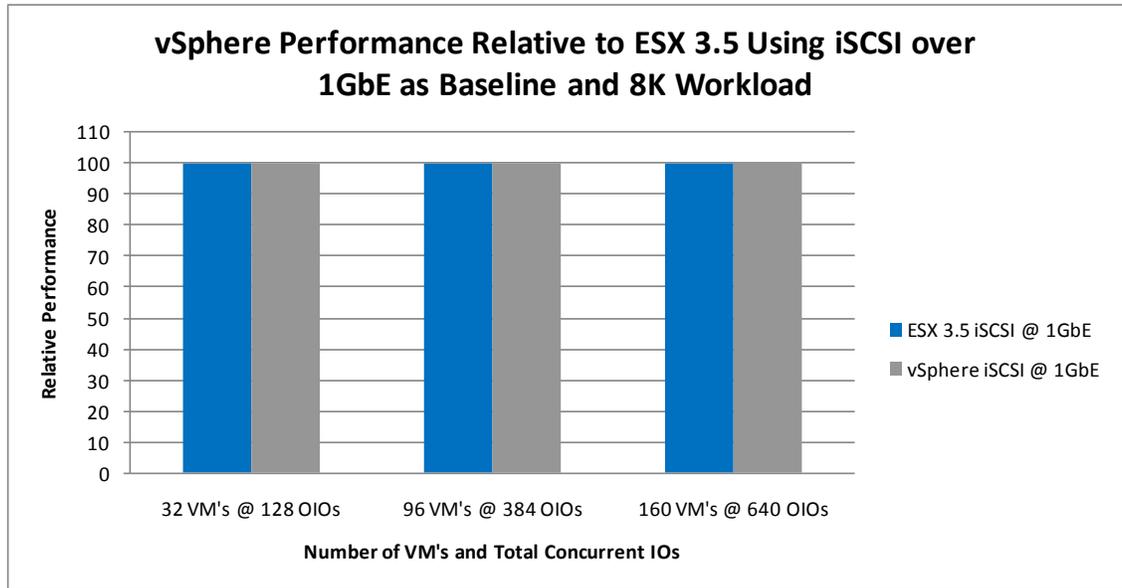


Figure 11) vSphere performance relative to ESX 3.5 using iSCSI over 1GbE and 8K request size.

Figure 12 below compares the average ESX host CPU utilization observed in vSphere using iSCSI and gigabit ethernet relative to the average ESX host CPU utilization observed in ESX 3.5 using iSCSI and gigabit ethernet as the baseline. We found that average CPU utilization in vSphere was approximately 40% to 43% lower compared to ESX 3.5 using iSCSI.

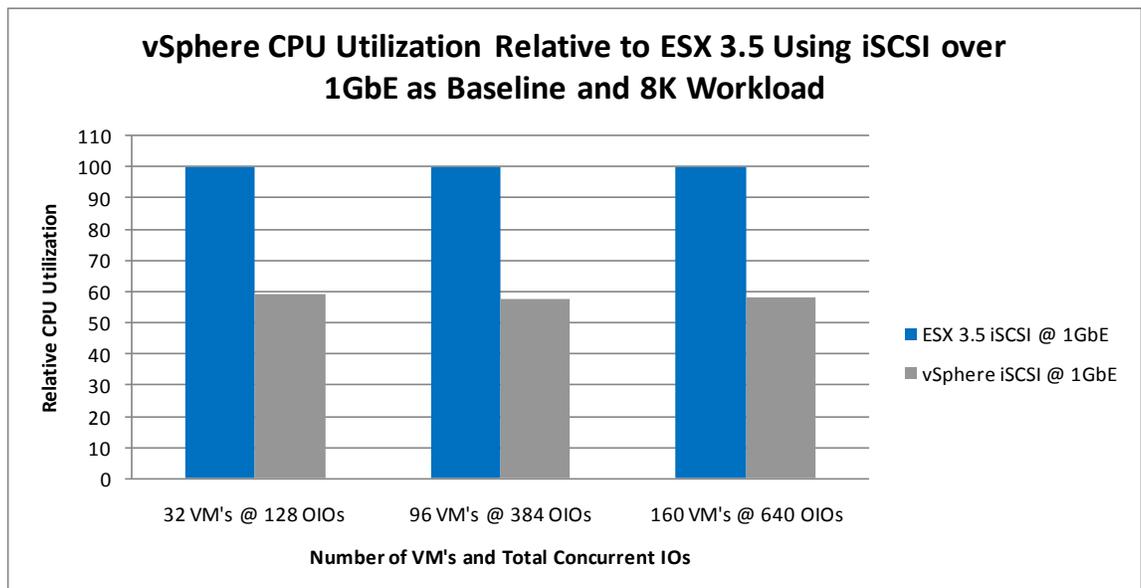


Figure 12) vSphere CPU utilization relative to ESX 3.5 using iSCSI over 1GbE and 8K request size.

5 PERFORMANCE COMPARISON USING MULTIPLE PROTOCOLS IN VSPHERE WITH A 4K WORKLOAD, 4GB FC, AND GIGABIT ETHERNET

5.1 RELATIVE THROUGHPUT COMPARISON

Figure 13 below compares the performance of vSphere using FC with round-robin load balancing, NFS and iSCSI using gigabit ethernet relative to the performance of FC with fixed data paths as the baseline. The following general observations were noted during the performance tests:

- All configurations tested generated throughput within 9% of FC using fixed data paths.
- FC throughput generated using round robin was comparable to FC throughput using fixed data paths.
- Performance using NFS with jumbo frames was comparable to NFS performance without jumbo frames configured.
- Performance using iSCSI with jumbo frames was 4% to 6% lower compared to iSCSI performance without jumbo frames configured.
- As the load increased, throughput using iSCSI and jumbo frames improved slightly relative to iSCSI without jumbo frames.

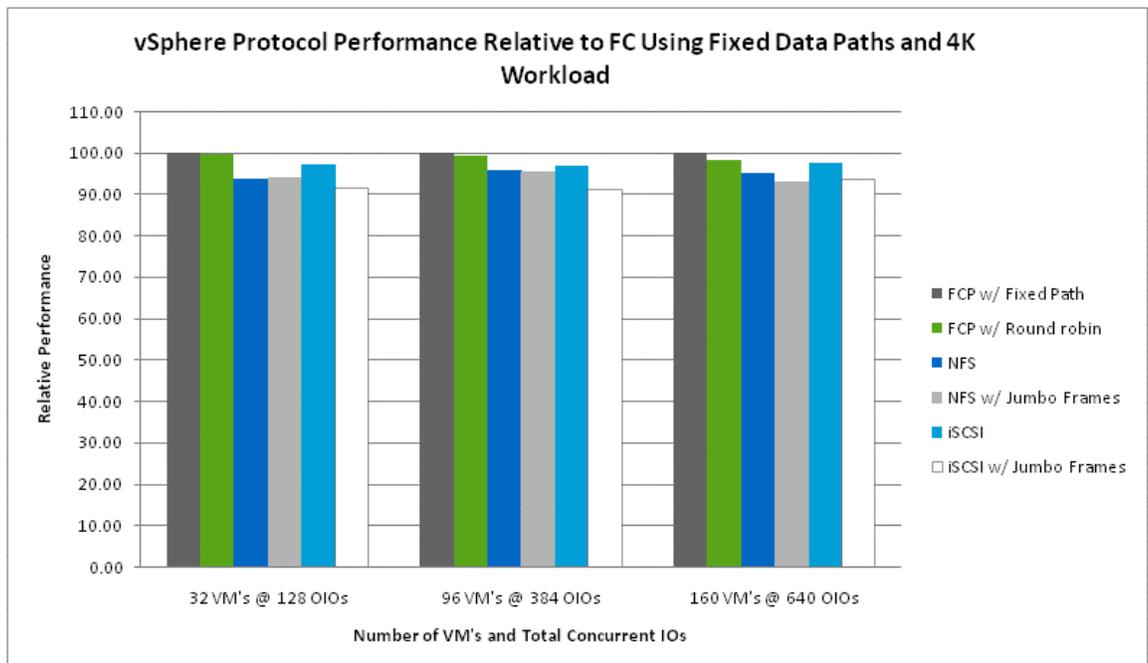


Figure 13) vSphere performance for 4Gb FCP, 1Gb iSCSI, and 1Gb NFS with 4K request size.

5.2 RELATIVE CPU UTILIZATION COMPARISON

Figure 14 below compares the average ESX host CPU utilization observed in vSphere using FC with round-robin load balancing, NFS and iSCSI using gigabit ethernet relative to the performance of FC with fixed data paths as the baseline. The following general observations were noted during the performance tests:

- We found that NFS and iSCSI using gigabit ethernet consumed 10% to 43% more ESX CPU resources compared to FC using fixed data paths.

- We found that using FC with round-robin load balancing consumed comparable amounts of vSphere CPU resources compared to using FC configured with fixed data paths.
- iSCSI with and without jumbo frames enabled consumed less vSphere CPU resources compared to NFS with and without jumbo frames enabled.
- Enabling jumbo frames on NFS or iSCSI resulted in lower vSphere host CPU utilization compared to NFS and iSCSI without jumbo frames enabled.

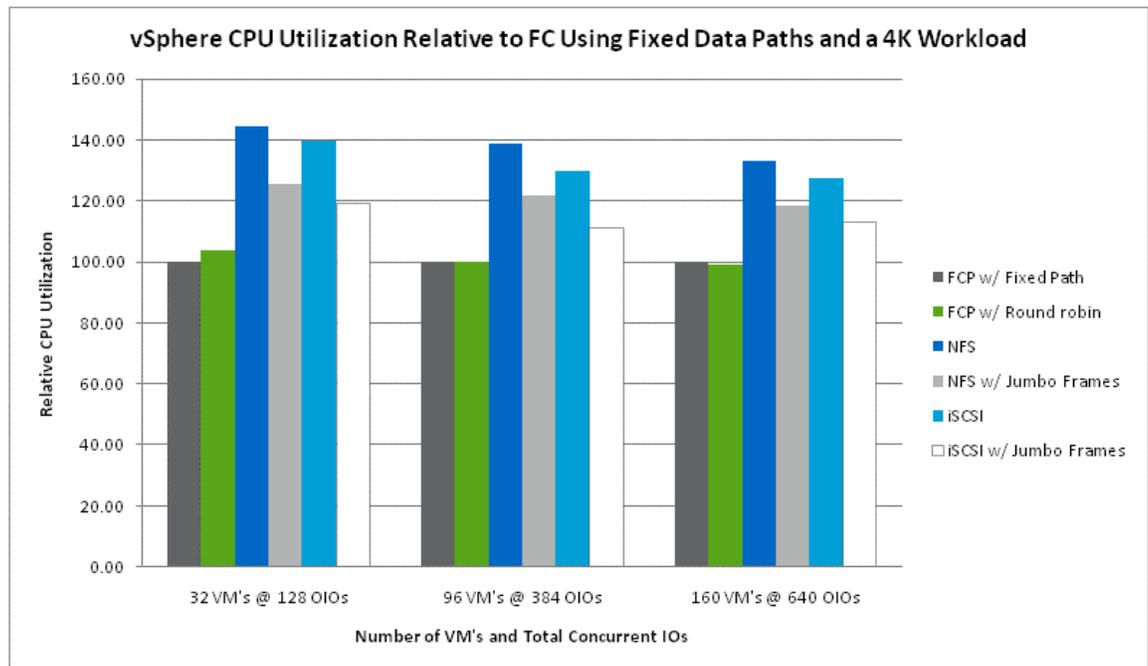


Figure 14) vSphere CPU utilization for 4Gb FCP, 1Gb iSCSI, and 1Gb NFS with 4K request size.

5.3 RELATIVE LATENCY COMPARISON

Figure 15 below compares the average latencies observed in vSphere using FC with round-robin load balancing, NFS and iSCSI using gigabit ethernet relative to the performance of FC with fixed data paths as the baseline. The following general observations were noted during the performance tests:

- For all configurations tested we observed overall average latency within 9% of FC using fixed data paths.
- FC latency using fixed paths was slightly lower compared to FC using round robin.
- In general, latencies observed for NFS using jumbo frames were comparable to NFS without jumbo frames.
- In general, latencies observed for iSCSI using jumbo frames were 4% to 6% higher compared to iSCSI without jumbo frames.

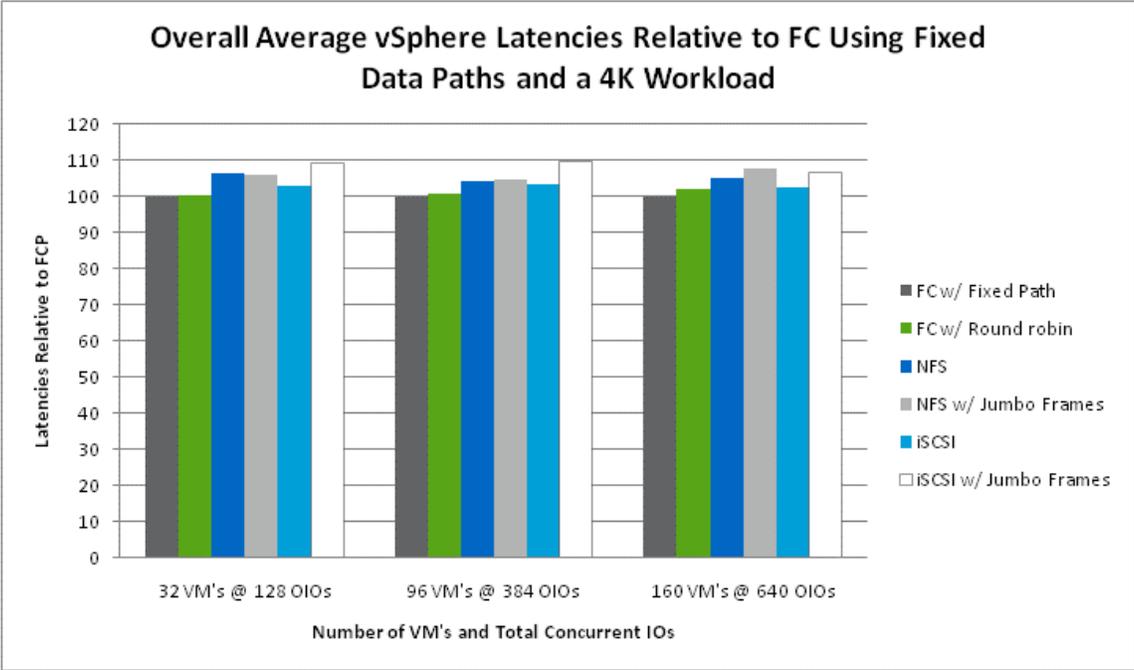


Figure 15) vSphere average latencies for 4Gb FCP, 1Gb iSCSI, and 1Gb NFS with 4K request size.

6 PERFORMANCE COMPARISON USING MULTIPLE PROTOCOLS IN VSPHERE WITH AN 8K WORKLOAD, 4GB FC, AND GIGABIT ETHERNET

6.1 RELATIVE THROUGHPUT COMPARISON

Figure 16 below compares the performance of vSphere using FC with round-robin load balancing, NFS and iSCSI using gigabit ethernet relative to the performance of FC with fixed data paths as the baseline. The following general observations were noted during the performance tests:

- With the exception of iSCSI using jumbo frames at a load of 128 outstanding I/Os, all configurations tested generated throughput within 9% of FC using fixed data paths.
- FC throughput generated using round-robin load balancing was comparable to FC throughput using fixed paths.
- Performance using NFS with jumbo frames was comparable to NFS performance without jumbo frames configured.
- Performance using iSCSI with jumbo frames was 3% to 9% lower compared to iSCSI performance without jumbo frames configured.
- As the load increased, throughput using iSCSI and jumbo frames improved relative to iSCSI without jumbo frames.

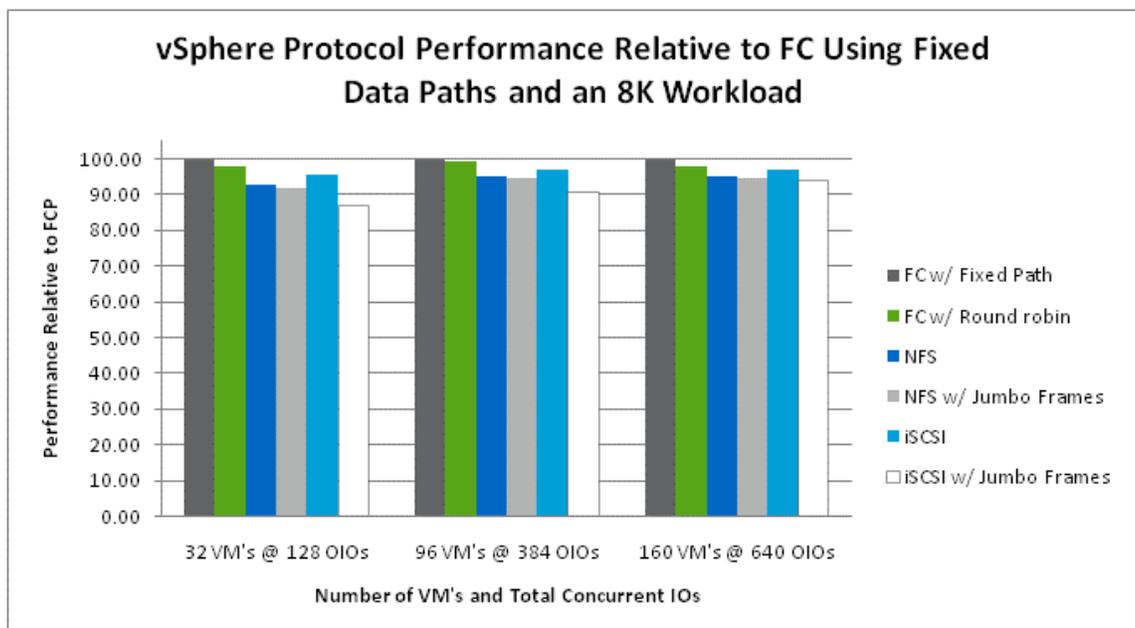


Figure 16) vSphere performance for 4Gb FCP, 1Gb iSCSI, and 1Gb NFS with 8K request size.

6.2 RELATIVE ESX SERVER CPU UTILIZATION COMPARISON

Figure 17 below compares the average ESX host CPU utilization observed in vSphere using FC with round-robin load balancing, NFS and iSCSI using gigabit ethernet relative to the performance of FC with fixed data paths as the baseline. The following general observations were noted during the performance tests:

- We found that NFS and iSCSI using gigabit ethernet consumed 10% to 45% more ESX CPU resources compared to FC using fixed data paths.

- We found that using FC with round-robin load balancing consumed comparable amounts of vSphere CPU resources compared to using FC configured with fixed data paths.
- iSCSI with and without jumbo frames enabled consumed less vSphere CPU resources compared to NFS with and without jumbo frames enabled.
- Enabling jumbo frames on NFS or iSCSI resulted in lower vSphere host CPU utilization compared to NFS and iSCSI without jumbo frames enabled.

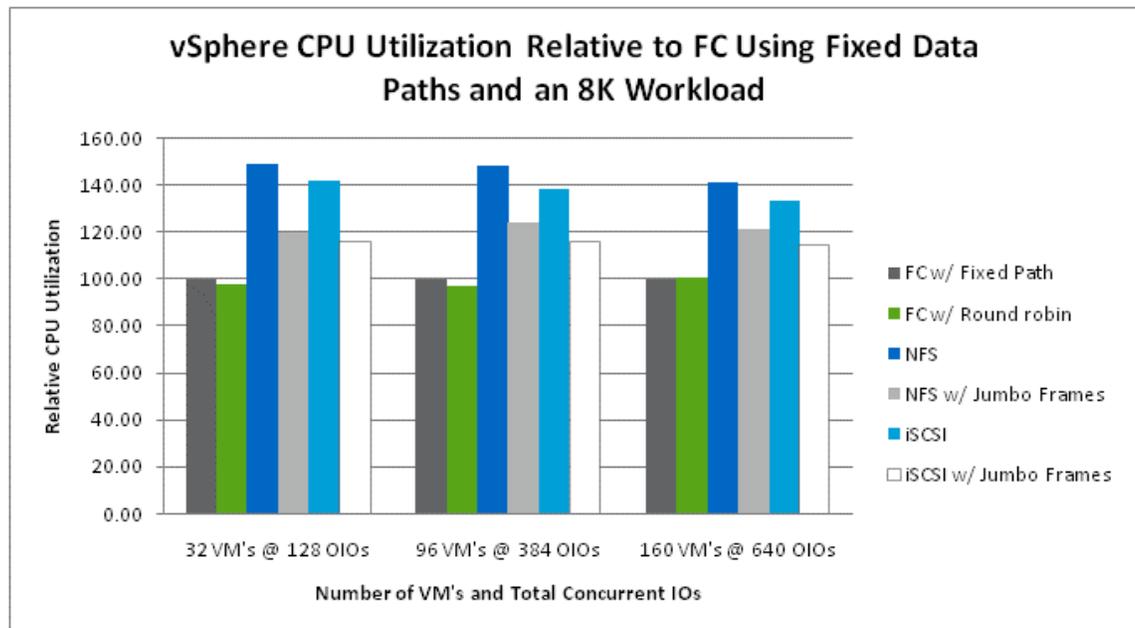


Figure 17) vSphere CPU utilization for 4Gb FCP, 1Gb iSCSI, and 1Gb NFS with 8K request size.

6.3 RELATIVE LATENCY COMPARISON

Figure 18 below compares the average latencies observed in vSphere using FC with round-robin load balancing, NFS and iSCSI using gigabit ethernet relative to the performance of FC with fixed data paths as the baseline. The following general observations were noted during the performance tests:

- With the exception of iSCSI using jumbo frames, we observed overall average latency within 8% of FC using fixed data paths.
- FC latency using fixed paths was slightly lower compared to FC using round-robin load balancing.
- Latencies observed using NFS with jumbo frames were comparable to NFS latencies without jumbo frames.
- Latencies observed using iSCSI with jumbo frames were approximately 3% to 9% higher compared to iSCSI without jumbo frames.
- As the load increased, latencies using iSCSI and jumbo frames improved relative to iSCSI without jumbo frames.

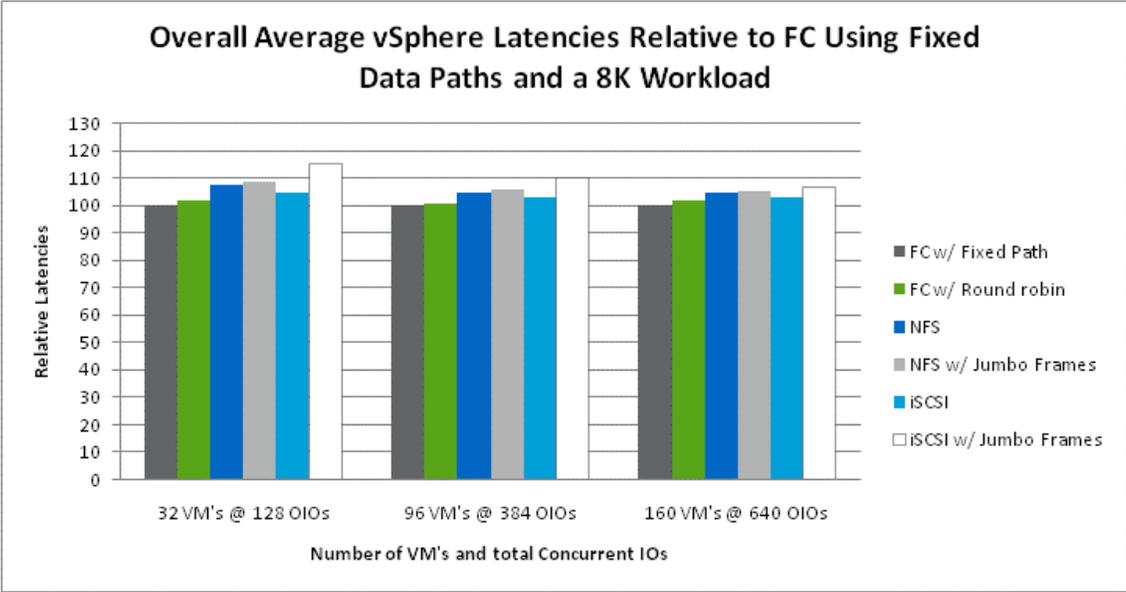


Figure 18) vSphere average latencies for 4Gb FCP, 1Gb iSCSI, and 1Gb NFS with 8K request size.

7 PERFORMANCE COMPARISON USING MULTIPLE PROTOCOLS IN VSPHERE WITH A 4K WORKLOAD, 4GB FC, AND 10 GIGABIT ETHERNET

7.1 RELATIVE THROUGHPUT COMPARISON

Figure 19 below compares the performance of vSphere with NFS and iSCSI using 10 gigabit ethernet relative to the performance of FC with fixed data paths as the baseline. The following general observations were noted during the performance tests:

- All configurations tested generated throughput within 9% of FC using fixed data paths.
- Performance using NFS with jumbo frames was comparable to NFS performance without jumbo frames configured.
- Performance using iSCSI with jumbo frames was 5% to 7% lower compared to iSCSI performance without jumbo frames configured.

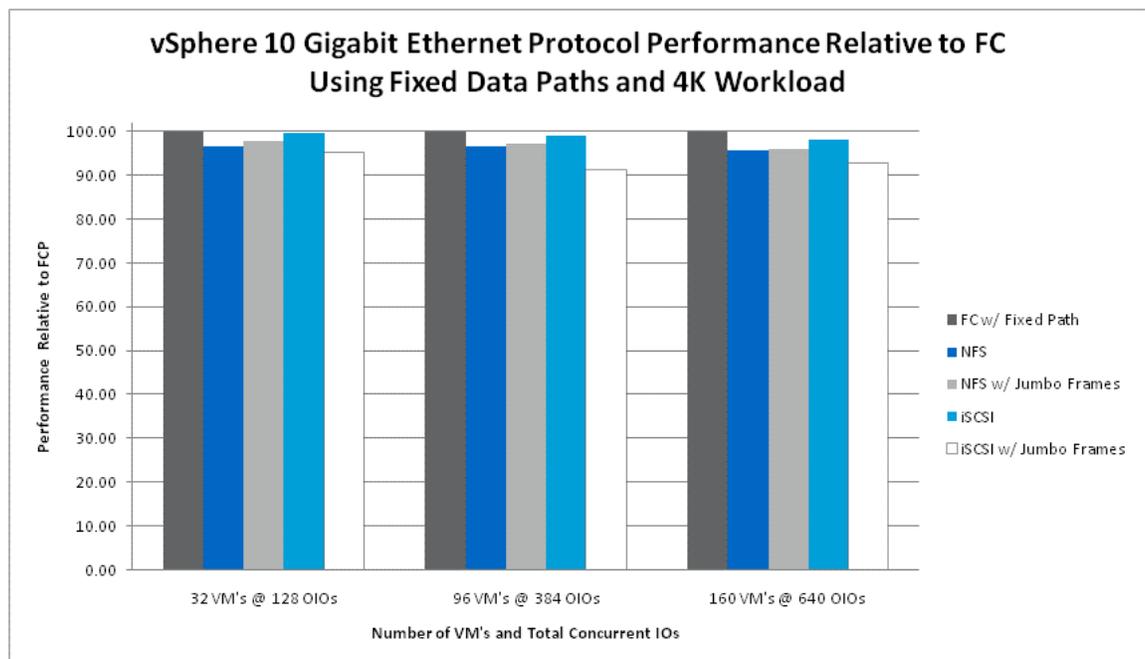


Figure 19) vSphere performance for 4Gb FCP, 10Gb iSCSI, and 10Gb NFS with 8K request size.

7.2 RELATIVE ESX SERVER CPU UTILIZATION COMPARISON

Figure 20 below compares the average ESX host CPU utilization observed in vSphere with NFS and iSCSI using 10 gigabit ethernet relative to the performance of FC with fixed data paths as the baseline. The following general observations were noted during the performance tests:

- We found that NFS and iSCSI using 10 gigabit ethernet consumed approximately 12% to 40% more ESX CPU resources compared to FC using fixed data paths.
- iSCSI with and without jumbo frames enabled consumed less vSphere CPU resources compared to NFS with and without jumbo frames enabled.
- Enabling jumbo frames on NFS or iSCSI resulted in lower vSphere host CPU utilization compared to NFS and iSCSI without jumbo frames enabled.

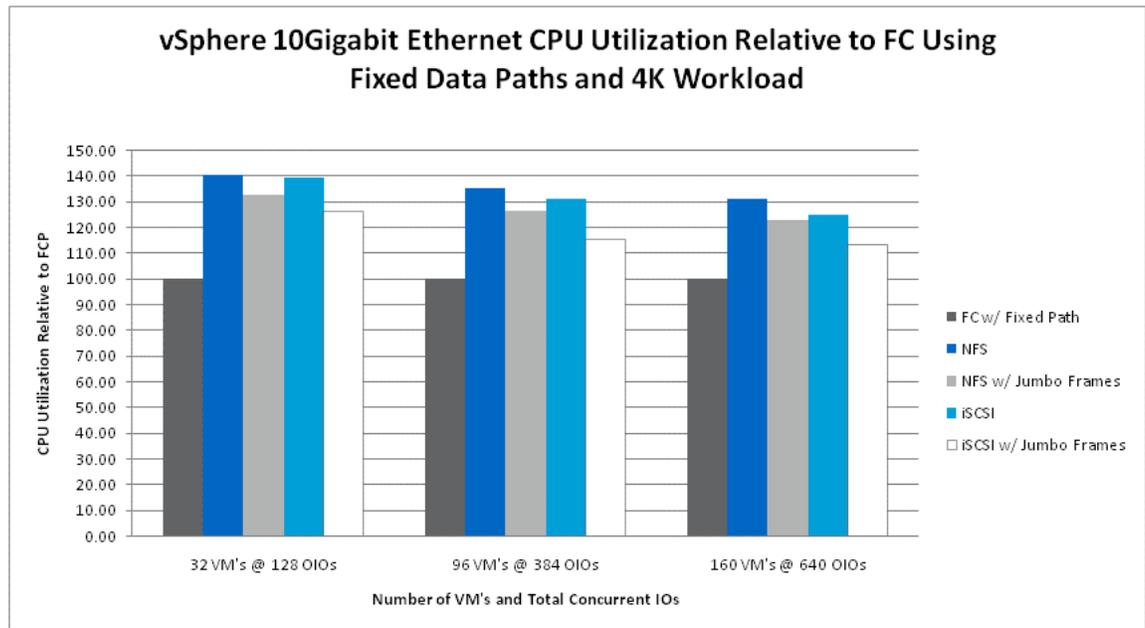


Figure 20) vSphere CPU utilization for 4Gb FCP, 10Gb iSCSI, and 10Gb NFS with 4K request size.

7.3 RELATIVE LATENCY COMPARISON

Figure 21 below compares the average latencies observed in vSphere using NFS and iSCSI using 10 gigabit ethernet relative to the performance of FC with fixed data paths as the baseline. The following general observations were noted during the performance tests:

- For all configurations tested we observed overall average latency within 10% of FC using fixed data paths.
- Latencies observed using NFS with jumbo frames were comparable to NFS latencies without jumbo frames.
- Latencies observed using iSCSI with jumbo frames were 5% to 8% higher compared to iSCSI performance without jumbo frames configured, depending on the load.

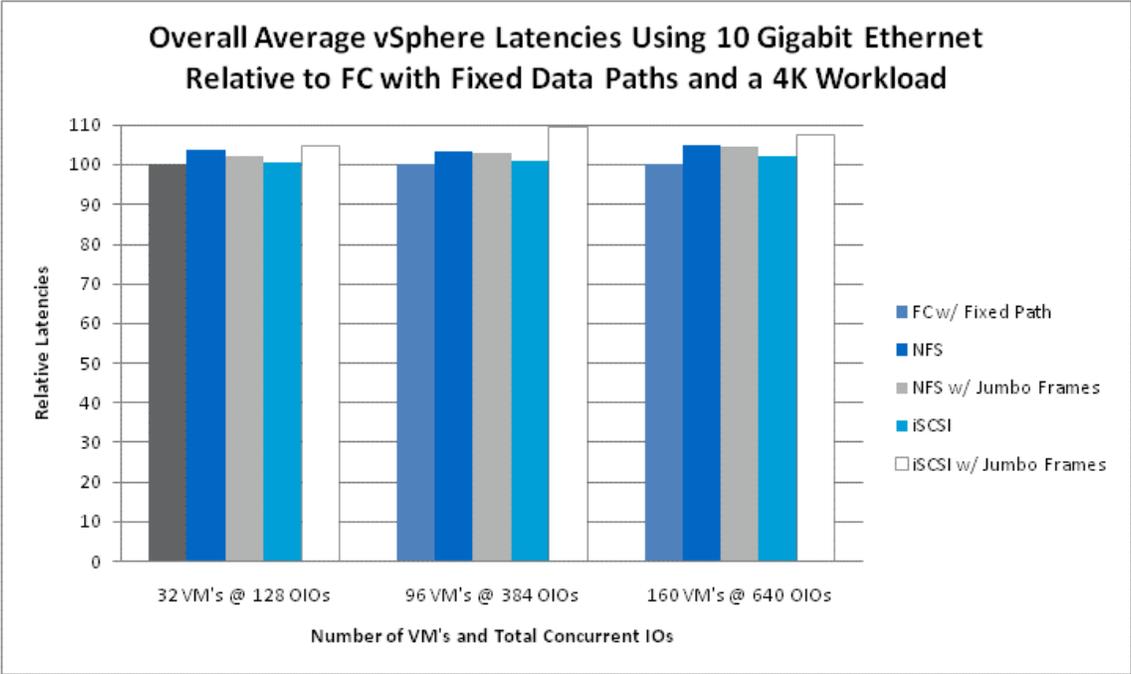


Figure 21) vSphere average latencies for 4Gb FCP, 10Gb iSCSI, and 10Gb NFS with 4K request size.

8 PERFORMANCE COMPARISON USING MULTIPLE PROTOCOLS IN VSPHERE WITH AN 8K WORKLOAD, 4GB FC, AND 10 GIGABIT ETHERNET

8.1 RELATIVE THROUGHPUT COMPARISON

Figure 22 below compares the performance of vSphere with NFS and iSCSI using 10 gigabit ethernet relative to the performance of FC with fixed data paths as the baseline. The following general observations were noted during the performance tests:

- All configurations tested generated throughput within 11% of FC using fixed data paths.
- Performance using NFS with jumbo frames was comparable to NFS performance without jumbo frames configured.
- Performance using iSCSI with jumbo frames was 6% to 9% lower compared to iSCSI performance without jumbo frames configured.

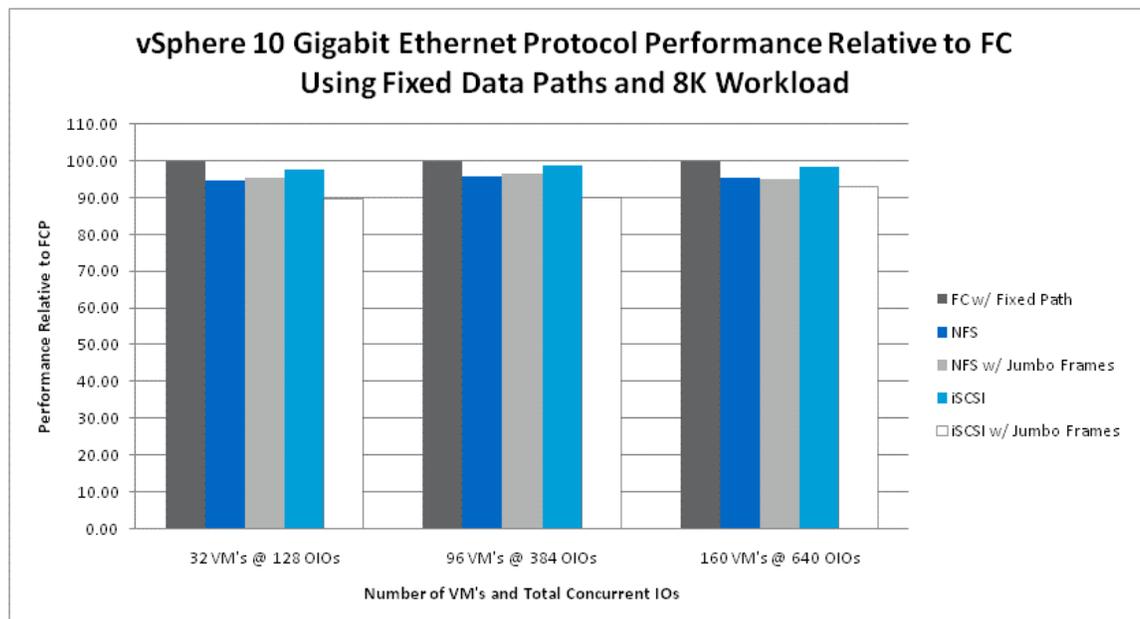


Figure 22) vSphere performance for 4Gb FCP, 10Gb iSCSI, and 10Gb NFS with 8K request size.

8.2 RELATIVE ESX SERVER CPU UTILIZATION COMPARISON

Figure 23 below compares the average ESX host CPU utilization observed in vSphere with NFS and iSCSI using 10 gigabit ethernet relative to the performance of FC with fixed data paths as the baseline. The following general observations were noted during the performance tests:

- We found that NFS and iSCSI using 10 gigabit ethernet consumed approximately 12% to 47% more ESX CPU resources compared to FC using fixed data paths.
- iSCSI with and without jumbo frames enabled consumed less vSphere CPU resources compared to NFS with and without jumbo frames enabled.
- Enabling jumbo frames on NFS or iSCSI resulted in lower vSphere host CPU utilization compared to NFS and iSCSI without jumbo frames enabled.

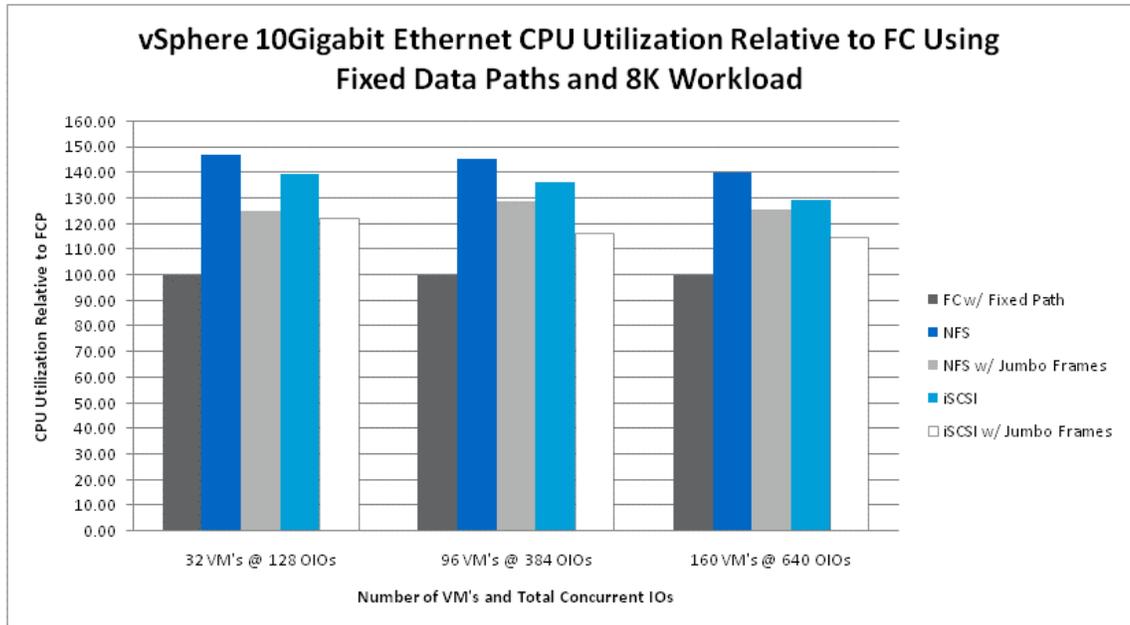


Figure 23) vSphere CPU utilization for 4Gb FCP, 10Gb iSCSI, and 10Gb NFS with 8K request size.

8.3 RELATIVE LATENCY COMPARISON

Figure 24 below compares the average latencies observed in vSphere using NFS and iSCSI using 10 gigabit ethernet relative to the performance of FC with fixed data paths as the baseline. The following general observations were noted during the performance tests:

- With the exception of iSCSI using jumbo frames, we observed overall average latency within 5% of FC using fixed data paths.
- Latencies observed using NFS with jumbo frames was comparable to NFS latencies without jumbo frames.
- Latencies observed using iSCSI with jumbo frames was 6% to 9% higher compared to iSCSI performance without jumbo frames, depending on the load.

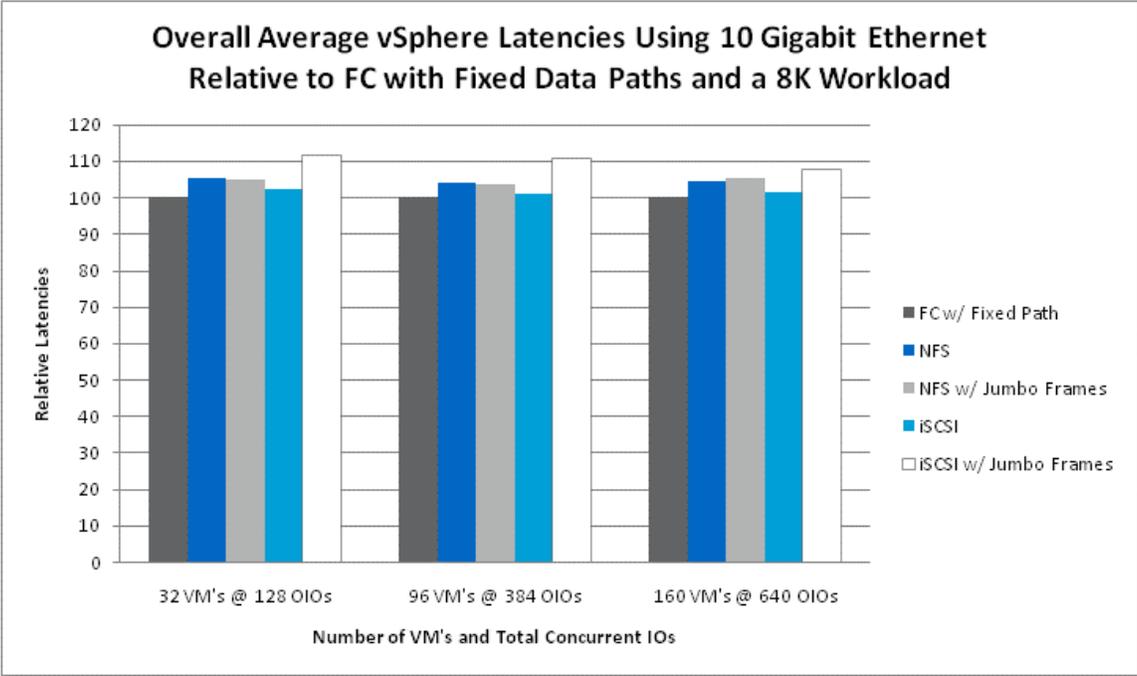


Figure 24) vSphere average latencies for 4Gb FCP, 10Gb iSCSI, and 10Gb NFS with 8K request size.

9 TEST DESIGN AND CONFIGURATION

This section provides the details of the hardware used for the testing. While there is only one physical testbed, there are differences in each setup required for testing the various protocols. These include, but are not limited to, how the NetApp storage is provisioned and presented to the ESX hosts as well as the type of switching infrastructure required to provide connectivity between the ESX hosts and NetApp storage.

In the first phase of our testing, we configured the VMware infrastructure with ESX Server 3.5 U4 and vCenter Server 2.5 U2 and the NetApp storage platform to support 4 Gb/s FCP along with iSCSI and NFS using multiple Gigabit Ethernet connections. After completing tests in this environment, we upgraded the existing Gigabit Ethernet infrastructure to use 10 Gigabit Ethernet and executed the same set of test cases with iSCSI and NFS using ESX 3.5 U4. After completing all tests using ESX 3.5 U4, we upgraded the VMware infrastructure to vSphere 4 while keeping the NetApp storage platform unchanged. We then performed the identical set of tests using vSphere4.

9.1 HARDWARE AND SOFTWARE ENVIRONMENT

Tables 1 and 2 below provide the details of the hardware and software components used to create the ESX data center used for the testing with both ESX 3.5 and vSphere. The data center contained eight ESX host servers connected using FC, Gigabit Ethernet, and 10 Gigabit Ethernet connections to the NetApp storage controllers.

When conducting testing using ESX 3.5, we managed the eight ESX 3.5 servers using vCenter Server 2.5 and the VI Client. When conducting testing using vSphere, we managed the eight vSphere servers using vCenter.

Table 1) Hardware and software components used for ESX and vSphere data center.

Component	Details
Virtual Infrastructure	VMware ESX 3.5/vCenter Server 2.5 and VMware vSphere/vCenter 4.0
Server	Rackable Systems S44
Processors	2 Quad Core Intel® Xeon L5420 2.65GHz
Memory	16GB
Fibre Channel Network	4Gbps FC
Fibre Channel HBA	Emulex Dual Port PCIe FC HBA – Lpe11000
4Gbps Fibre Channel Switch	Brocade 200E
IP network for NFS and iSCSI	1Gbps and 10Gbps Ethernet with a dedicated switch and VLAN
1GB NICs for NFS and software iSCSI	2 Intel 82575EB Ethernet Controllers
Gigabit Ethernet Switch	Cisco Catalyst 3750
10GB NICs for NFS and software iSCSI	1 Chelsio T310 Single port NIC
10Gb Ethernet Switch	Fujitsu XG1200

Table 2) Virtual center details.

Component	Details
Server	Fujitsu Primergy RX300
Processors	2 Dual Core Intel Xeon 3.60 Ghz
Memory	3.25 GB
Operating System	Microsoft Windows Server 2003 Enterprise Edition Service Pack 1

Table 3 below describes the NetApp products used to provide the shared storage for the test configuration.

Table 3) NetApp storage hardware and software components.

Component	Details
Storage System	FAS3170A
Data ONTAP® Version	7.3.1
Number of Drives	140
Size of Drives	144GB
Speed of Drives	15K
Type of Drives	Fibre Channel

***Note:** There is approximately 7.5TB of usable storage on each FAS3170 storage controller.

9.2 CONNECTING NETAPP STORAGE TO THE VMWARE DATA CENTER

This section provides the details relating to how we connected the NetApp FAS3170A storage controllers to the VMware data center created for the testing. This information applies to testing with both ESX 3.5 and vSphere.

9.2.1 Fibre Channel Configuration

The diagram in Figure 25 below shows how the ESX Servers and NetApp FAS3170A storage controllers are connected for all tests using FCP. Brocade 200 FC switches were used to provide the connectivity, and each ESX Server was connected to both FC switches to provide multiple paths to their respective storage.

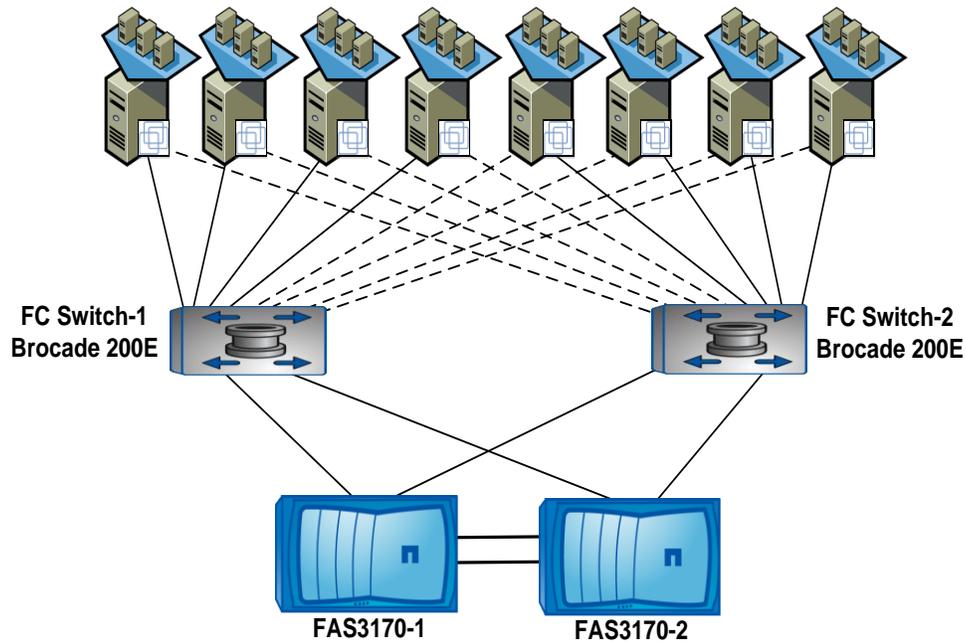


Figure 25) FC connectivity between ESX Servers and NetApp storage.

ESX and vSphere Servers 1 through 4 are serviced by FAS3170-1, and ESX and vSphere Servers 5 through 8 are serviced by FAS3170-2. Each ESX and vSphere Server is presented with two LUNs shared with another ESX and vSphere Server for VM guest OS and Iometer data file storage. Each ESX host had the following paths to the LUNs on the FAS3170:

- Path 1: HBA_port0 -> Switch1 -> FAS1_port0 -> LUN
- Path 2: HBA_port0 -> Switch1 -> FAS2_port0 -> LUN
- Path 3: HBA_port1 -> Switch2 -> FAS1_port1 -> LUN
- Path 4: HBA_port1 -> Switch2 -> FAS2_port1 -> LUN

9.2.2 Gigabit Ethernet Configuration

The diagram in Figure 26 below shows how the ESX/vSphere servers and NetApp FAS3170A storage controllers were connected for all tests using iSCSI and NFS. A Cisco 4948 Gigabit Ethernet switch was used to provide the connectivity.

Each ESX/vSphere server had two Gigabit Ethernet connections with one of the connections dedicated for use as a service console and internal VM traffic and the other used for iSCSI and NFS traffic.

Each FAS3170 controller contained two Gigabit Ethernet connections on which iSCSI and NFS traffic was serviced. As in the case of FCP, the guest OS and Iometer data file storage for ESX/vSphere Servers 1 through 4 was provided by FAS3170-1, and ESX Servers 5 through 8 were serviced by FAS3170-2. The individual connections for the ESX/vSphere hosts are as follows:

- ESX/vSphere Servers 1 and 2 were serviced by NIC1 of FAS3170-1
- ESX/vSphere Servers 3 and 4 were serviced by NIC2 of FAS3170-1
- ESX/vSphere Servers 5 and 6 were serviced by NIC1 of FAS3170-2
- ESX/vSphere Servers 7 and 8 were serviced by NIC2 of FAS3170-2

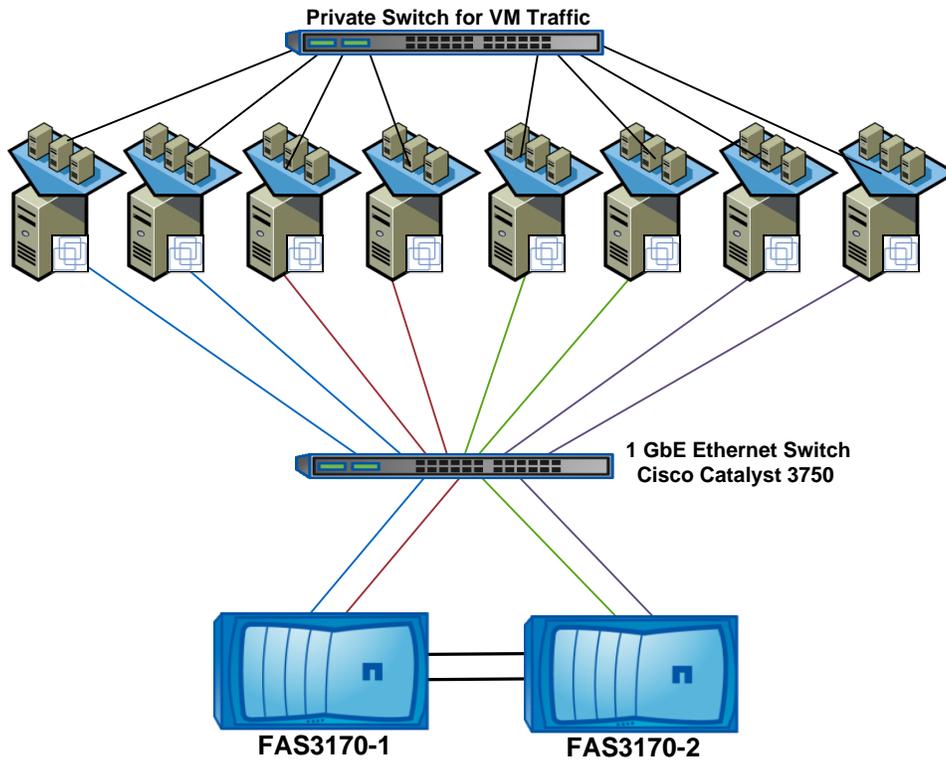


Figure 26) NFS and iSCSI connectivity between ESX Servers and NetApp storage.

9.2.3 10 Gigabit Ethernet Configuration:

The diagram in Figure 27 below shows how the ESX/vSphere servers and NetApp FAS3170A storage controllers were connected for all tests using iSCSI and NFS. A Fujitsu XG1200 10 Gigabit Ethernet switch was used to provide the connectivity.

For the tests using 10 Gigabit Ethernet, we reconfigured the ESX/vSphere servers to use the Chelsio T-310E 10 Gigabit Ethernet NICS and downloaded and installed the VMware certified driver from the VMware site. We reconfigured the networking on each ESX/vSphere server so that the vmkernel traffic used the Chelsio 10 Gigabit Ethernet NIC. The service console and VM network traffic continued to use the same Gigabit Ethernet interface used for the testing using only Gigabit Ethernet.

On each of the FAS3170 storage controllers, we installed and configured two additional Chelsio T310 10 Gigabit Ethernet cards to handle both iSCSI and NFS traffic. For all testing using 10 Gigabit Ethernet, the ESX/vSphere servers continued to use the same LUNs/volumes that were used for the testing with Gigabit Ethernet.

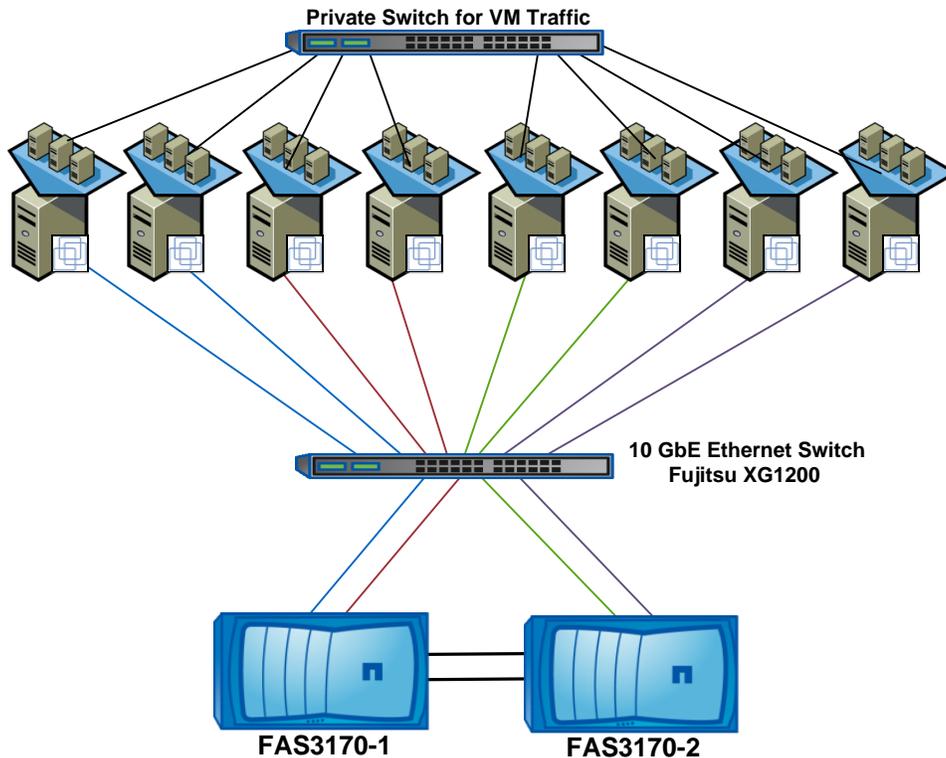


Figure 27) NFS and iSCSI connectivity between ESX Servers and NetApp storage.

9.3 PROVISIONING NETAPP STORAGE TO THE VMWARE DATA CENTER

After configuring the physical infrastructure as described above, the storage was then provisioned on the FAS3170 storage controllers to provide adequate shared storage for creating a set of 20 VMs on each of the eight ESX/vSphere servers.

As mentioned previously, each of the FAS3170 storage controllers is configured with a total of 84 x 144GB/15K RPM FC disk drives. For these tests, separate aggregate was created on each of the FAS3170 storage controllers containing a total of 80 drives configured into 5 x RAID-DP@ groups of 16 disks each while maintaining one spare disk on each storage controller. The result of this configuration is approximately 7.5TB of usable shared storage on each of the FAS3170 storage controllers. After creating the aggregates, a set of FlexVol@ volumes and LUNs was created on the FAS3170 storage controllers to be presented to the ESX Servers as follows:

- On each FAS3170 storage controller, a total of eight FlexVol volumes were created. Four of the FlexVol volumes were 550GB in size, and the remaining four were 650GB in size.
- Configure the four 550GB FlexVol volumes as NFS volumes.
- In each of the other four 650GB FlexVol volumes, a single 550GB LUN was created. These LUNs were used for both the FCP and iSCSI testing.

Figure 28 below shows the layout of the FlexVol volumes on one of the FAS3170 storage controllers that will provide shared storage for ESX Servers 1 through 4. In this case, the FlexVol volumes are named based on the type of protocol that will be used to access the storage. The FlexVol volumes named `esx_blocks1-4` will contain LUNs that will be accessed by the ESX/vSphere hosts using either FCP or iSCSI, depending on the protocol being tested. Additionally, the FlexVol volumes named `esx_nfs1-4` are mapped to the respective ESX/vSphere hosts using NFS. The second FAS3170 storage controller contains the same number of FlexVol volumes created using a similar naming scheme for use with ESX Servers 5 through 8.

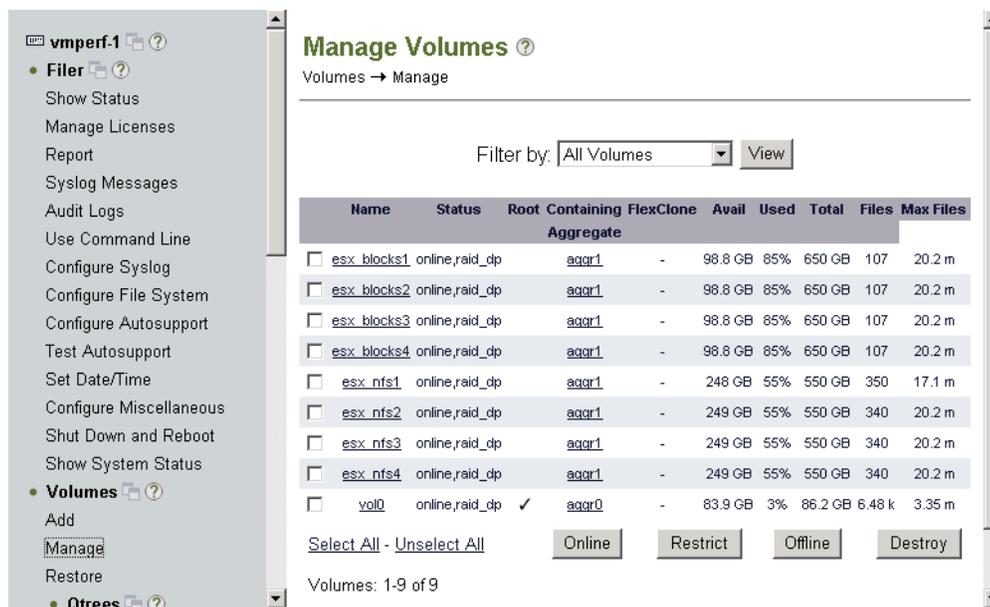


Figure 28) FAS3170 FlexVol volumes configuration.

Each of the eight ESX/vSphere servers was presented with 2 x 550GB LUNs and 2 x 550GB FlexVol volumes respectively on which to create a set of 20 VMs. Each LUN/volume was shared between 2 ESX/vSphere Servers as shown in the following diagram in Figure 29.

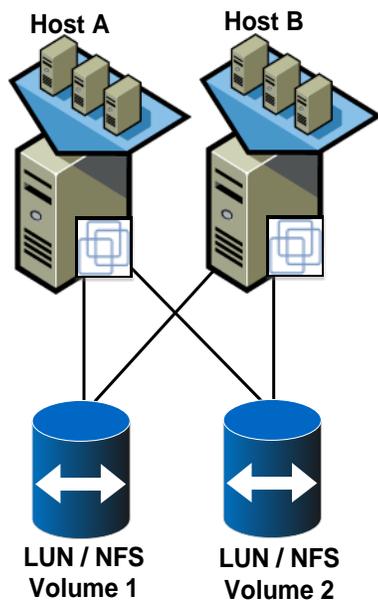


Figure 29) Sharing LUNs/NFS volumes between ESX Servers.

Tests using iSCSI and FCP used the same set of LUNs created on the FAS3170 storage controllers. Additionally, all testing using NFS was conducted using the same set of FlexVol volumes. In order to use the same LUNs for iSCSI and FCP the initiator group feature of Data ONTAP was utilized to allow access to the LUNs based on either the FC WWPNs or iSCSI IQN identifiers defined on the ESX/vSphere hosts.

To change the protocol accessing each of the LUNs, we created two different sets of initiator groups on each of the FAS3170 storage controllers: one for FCP and another for iSCSI. Each initiator group contained the WWPN and IQN information associated with each of the FC and iSCSI initiators on each of the 4 ESX/vSphere servers using the shared storage on the FAS3170.

Each FCP initiator group contained 4 WWPNs associated with the dual ported FC HBA on each of the two ESX/vSphere servers that should see the LUN the specific initiator group maps to. Each iSCSI initiator group contained 2 iSCSI IQNs associated with the iSCSI initiator on each of the two ESX/vSphere servers that should see the LUN this initiator group maps to. Figure 30 below shows the FCP and iSCSI initiator groups defined on one of the FAS3170 storage controllers serving ESX/vSphere servers 1 through 4.



Figure 30) FAS3170 initiator group configurations.

After creating the FCP and iSCSI initiator groups using information from the ESX/vSphere servers, a mapping between each of the ESX Servers and the specific LUNs that will be providing shared storage for the VMs was created. For example, Figure 31 below shows each of the four LUNs mapped to a different FCP initiator group corresponding to a different ESX/vSphere server. In this case, the ESX/vSphere servers will only be able to access the LUNs using FCP.



Figure 31) FAS3170 LUNs configured for tests using FCP.

Once the LUNs are mapped to a specific FC or iSCSI initiator on the ESX/vSphere hosts, the storage is available to the ESX data center. Figure 32 below shows the LUNs defined on one of the FAS3170 storage controllers as seen from one of the ESX/vSphere hosts using FC.

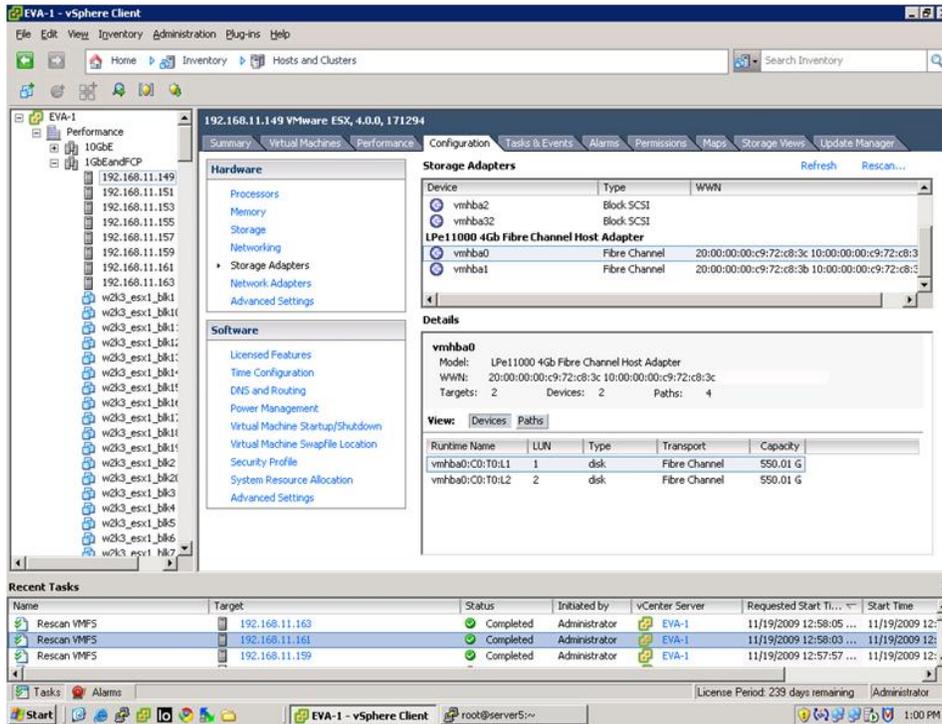


Figure 32) ESX Server view of LUNs using FCP.

To use the same set of LUNs for the iSCSI testing, we removed the LUN mapping using the FC initiator groups and assigned the set of iSCSI initiator groups to each of the same LUNs and simply rescanned the storage on each of the ESX/vSphere hosts such that the LUNs were accessed using iSCSI. In this case, all the VM data remains intact on the ESX/vSphere servers. At this point, all the ESX/vSphere hosts continue to see the same set of LUNs; however they are now accessing them using the iSCSI protocol.

Figure 33 below shows this process after the LUN named esx1_lun has been remapped to be accessed by the iSCSI initiator on ESX/vSphere server 1. Once the LUN remapping process is complete, each of the 4 x 550GB LUNs was mapped to a different iSCSI initiator group.

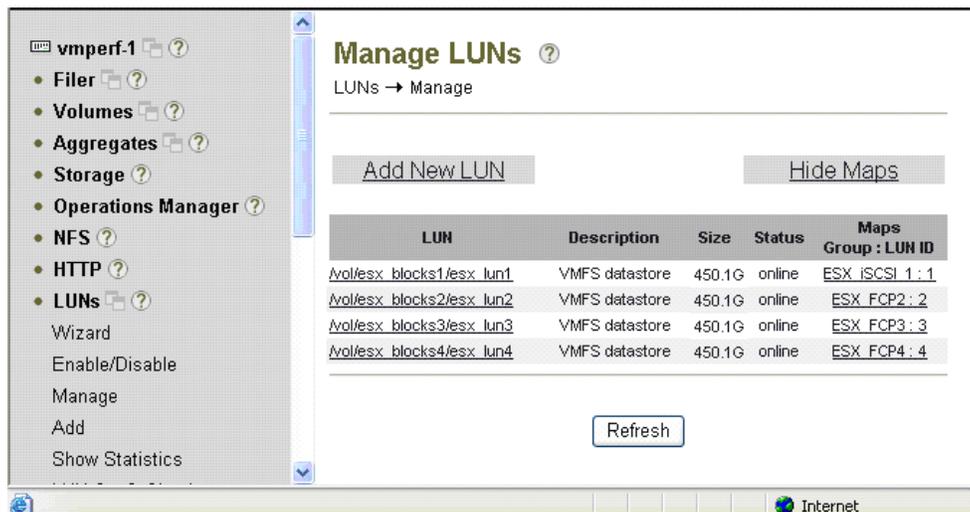


Figure 33) FAS3170 LUN mappings when switching to iSCSI tests.

Figure 34 below shows the same two 550GB LUNs on the same ESX Server now mapped on the ESX Server using iSCSI.

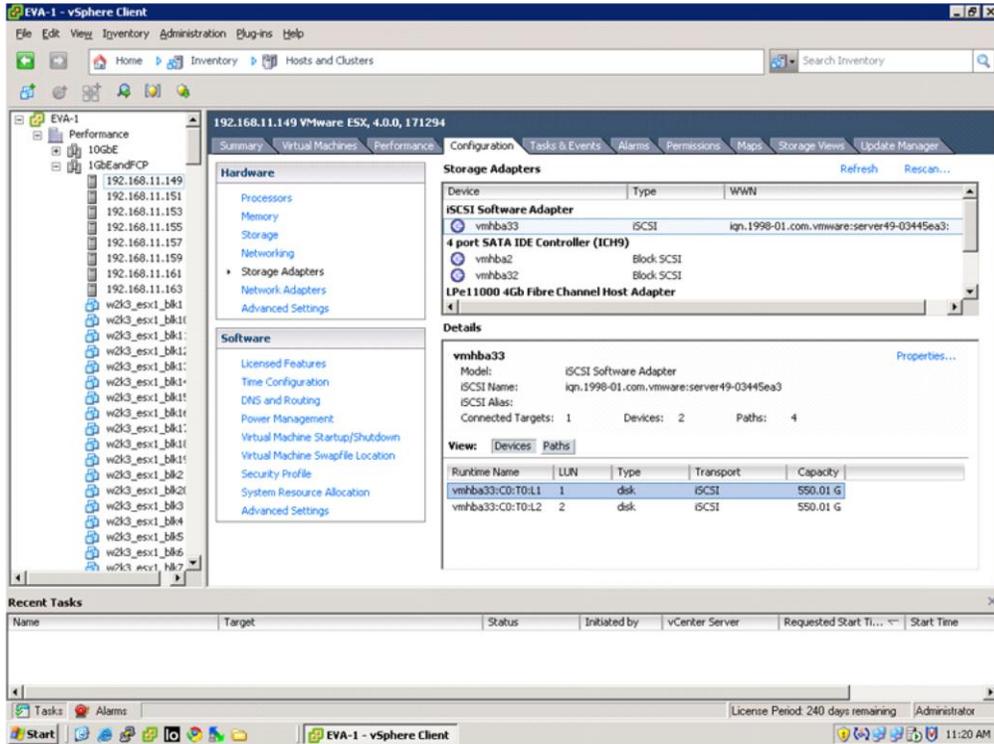


Figure 34) ESX Server view of LUNs using iSCSI.

9.4 CONFIGURING VMWARE ESX 3.5 AND VSPHERE DATA CENTER

After provisioning the storage as described in the previous section, we then created a set of 20 VMs on each of the eight ESX hosts for use during the tests. Table 4 below provides the configuration details for each of the VMs.

Table 4) Virtual machine components.

Component	Details
Operating System	32-bit Windows Server 2003 Enterprise with SP2
Number of Virtual Processors	1
Memory	512MB
Virtual Disk Size for Iometer Data	10GB
Virtual Disk Size for OS	5GB

Each of the ESX and vSphere host servers has access to four data stores provisioned on the FAS3170 storage controllers: two block based LUNs and two NFS volumes. After presenting the storage to the ESX Servers as described previously, a gold master VM was created as follows:

1. Create a VM in the NFS mounted data store of ESX Server 1.
2. Before installing the operating system, create an aligned 5GB partition on the VM according to published NetApp VMware best practices (see NetApp technical report TR-3749 for more information).
3. Install the operating system in the aligned 5GB partition.

4. Configure the network adapter on the VM to use DHCP to retrieve an IP address.
5. Install the lometer dynamo application that will be used later to generate the workload for all test cases.
6. After verifying the gold master VM started correctly, create a new 10GB virtual disk in the VM for use by lometer.
7. Using Windows disk management tools, create an aligned and NTFS formatted 10GB disk within Windows to hold lometer data files. This virtual disk was created in the same ESX data store as the 5GB virtual disk that holds the OS partition.

After creating the gold master VM, standard VMware cloning was used through vCenter Server to create a total of 20 VMs on the two data stores on each ESX Server. Each ESX/vSphere server created a set of 10 VMs on the first LUN/NFS volume and a second set of 10 VMs on the other LUN/NFS volume. Each of the cloned VMs was customized using a vCenter Server customization specification to enable the cloned VMs to have unique computer names and SSIDs. Once powered on, the VMs get unique IP addresses from a standalone DHCP server.

In addition to the configurations discussed above, the following additional changes were made to all of the ESX/vSphere host systems and the FAS3170 controllers in order to increase performance for both the FC and iSCSI protocols.

On each of the ESX 3.5 and vSphere hosts:

- Set FC HBA queue depth to 128.
- Set iSCSI queue depth to 128 (this is now the default setting on vSphere).

On the FAS3170A storage controllers the “no_atime_update” volume option was enabled on the FlexVol volume providing storage accessed using NFS and increased the TCP receive window size volume option for NFS by issuing the following commands:

- `vol options <nfs_flex_vol_name> no_atime_update on`
- `options nfs.tcp.recvwindowsize 64240`

For all tests with iSCSI and NFS that use jumbo frames, we issued the following commands to enable support for jumbo frames on each of the ESX 3.5/vSphere hosts:

- `esxcfg-vmknic -d VMkernel`
- `esxcfg-vmknic -a -i <IP addr> -n <netmask> -m 9000 VMkernel`
- `esxcfg-vswitch -m 9000 <vSwitch name>`

For FC tests that use round-robin load balancing, the following steps were performed on each vSphere host:

- Identified all primary FC paths from each vSphere host to the appropriate NetApp FAS3170 storage controller.
- Used vSphere client to select round-robin load balancing policy and assign the primary paths identified in the first step to be used when load balancing.

9.5 WORKLOAD DEFINITION AND IOMETER ENVIRONMENT

For the testing, the publicly available lometer (found at www.lometer.org/) application was used to generate the load. lometer is a client server application that works both as a workload generator and a measurement tool. The client portion is called “dynamo” and is installed on each of the VMs. The server portion is called the “lometer Controller” and was installed on a standalone server separate from the ESX Servers. The lometer controller is used to manage the dynamo load generators running on the VMs and gather test results from each of the dynamos.

A dedicated server was configured that acted both as a DHCP server to the VMs and as an lometer controller. Before testing, lometer was used to initialize the 10GB data stores on each of the VMs. During the initialization process, lometer generates a 10GB data file on the disk under test called “iobw.tst.” lometer then performs read and write operations to this file during testing. To initialize this file on each VM, the following steps were performed:

1. Power on all VMs on all ESX Servers.

2. Run the lometer dynamo application in all the powered on VMs.
3. Create an lometer access specification to generate a 100% random read load using an 8K request size.
4. Execute the specification on all of the VMs for a total of 5 minutes.

The results of this random read test were not used. It was used only to initialize the virtual disks associated with the VMs.

After initializing the lometer data files as described above, the VMs were powered off and created a Snapshot™ copy of the aggregate containing the eight FlexVol volumes defined above. This Snapshot copy represents the initial state of the system before performing any of our tests. At the end of every test, we powered off all VMs and used this Snapshot copy to restore the storage to the freshly initialized state. This way, it is made sure that all tests started from exactly the same point in time with the lometer data files being in their initialized state.

The workloads used for these tests were a mixture of random reads and writes using both a 4K and 8K request size that represent realistic workloads experienced by ESX in production environments. The specifics of these loads are as follows:

- 4K request size, 75% read, 25% write, 100% random
- 8K request size, 75% read, 25% write, 100% random

For each workload, we measured the following items on both ESX 3.5 and vSphere:

- Throughput in IOPS of FC, iSCSI, NFS protocols relative to each other
- Average ESX Server CPU utilization
- Average ESX Server latency

For these tests, the lometer access specification were configured such that each dynamo instance running on an individual VM generated a constant total of 4 outstanding I/Os. We then increased the load by increasing the total number of VMs participating in a given test. Table 5 below summarizes the three different test configurations we used for each different protocol configuration and includes the number of VMs used, the total number of outstanding I/Os generated by each ESX 3.5/vSphere host, and the total number of outstanding I/Os generated against the NetApp storage from the entire VMware data center.

Table 5) Table illustrating workload scaling keeping four outstanding I/Os per VM.

Number of VMs	Outstanding I/O Per ESX Server	Total Outstanding I/Os
4	16	128
12	48	384
20	80	640

The lometer tests will be run for a total of five minutes with a five minute ramp up time to allow performance to reach steady state before lometer begins to record results. At a minimum, there were 16 VMs per FAS3170 controller having an lometer working data set of 160GB at 10GB per VM. This working set is large enough to make sure that the data is not served from the cache on the FAS3170A, resulting in modest to high levels of disk activity on the FAS3170A, depending on the number of outstanding I/Os used in the test.

Once these tests are run for a protocol in the above described way at five workload levels, the aggregate Snapshot copies are restored that were created after initializing the lometer test files in all VMs. This allows a return to the start and a repeat of the test in the same way for the next protocol. All tests are performed on ESX Server 3.5 and vSphere servers and draw comparison on the data we collect.

10 REFERENCES

[TR-3428: NetApp and VMware Virtual Infrastructure Storage Best Practices](#)

[TR-3749: NetApp and VMware vSphere Storage Best Practices](#)

11 ACKNOWLEDGEMENTS

Special thanks to the following for their contributions:

- Kaushik Banerjee, Senior Manager, Performance Engineering, VMware
- Gil Bryant, Lab Manager, NetApp
- Duaine Fiorrito, Solutions Technologies Lab Manager, NetApp
- Chris Gebhardt, Reference Architect, NetApp
- Keith Griffin, EDCA Lab Support, NetApp
- Abhinav Joshi, Reference Architect, NetApp
- Chethan Kumar, Senior Member of Technical Staff, Performance Engineering, VMware
- Christopher Lemmons, Senior Manager, Workload Engineering, NetApp
- Todd Muirhead, Staff Engineer, Performance Engineering, VMware
- Vaughn Stewart, Technical Marketing Engineer, NetApp
- Ricky Stout, Lab Manager, NetApp
- Wen Yu, Senior Technical Alliance Manager, VMware

12 FEEDBACK

Send an e-mail to xdl-vgibutmevmtr@netapp.com with questions or comments concerning this document.

NetApp provides no representations or warranties regarding the accuracy, reliability or serviceability of any information or recommendations provided in this publication, or with respect to any results that may be obtained by the use of the information or observance of any recommendations provided herein. The information in this document is distributed AS IS, and the use of this information or the implementation of any recommendations or techniques herein is a customer's responsibility and depends on the customer's ability to evaluate and integrate them into the customer's operational environment. This document and the information contained herein may be used solely in connection with the NetApp products discussed in this document.

© Copyright 2009 NetApp, Inc. All rights reserved. No portions of this document may be reproduced without prior written consent of NetApp, Inc. NetApp, the NetApp logo, Go further, faster, Data ONTAP, FlexVol, RAID-DP, and Snapshot are trademarks or registered trademarks of NetApp, Inc. in the United States and/or other countries. Microsoft, Windows, and SharePoint are registered trademarks of Microsoft Corporation. Intel is a registered trademark of Intel Corporation. VMware is a registered trademark and vSphere is a trademark of VMware, Inc. All other brands or products are trademarks or registered trademarks of their respective holders and should be treated as such.