Analysis of VDI Storage Performance During Bootstorm

Introduction

Virtual desktops are gaining popularity as a more cost effective and more easily serviceable solution. The most resource-dependent process for many virtual desktop pools is the bootstorm. Enterprises deploying a virtual desktop infrastructure (VDI) environment have a variety of choices when deciding what sort of storage to deploy for their VDI's. Solid State Storage is known to dramatically decrease boot times compared to hard disk drives (HDD), and may be a good storage choice for a pool of Virtual Machines (VMs).

Demartek evaluated the performance of Internal HDD, Fibre Channel (FC) HDD, and Fibre Channel solid state drives (SSD) under bootstorm conditions and compared VM boot times, server load, and disk throughput performance metrics. In addition, Demartek compared results for different cache configurations and host bus adapter (HBA) queue depths.

This evaluation was conducted in the Demartek lab in Colorado.

Executive Summary and Key Findings

Demartek evaluated the differences SSD and HDD storage had on bootstorm performance, and analyzed the effects of queue depth and cache configuration on that performance.

For Fibre Channel SSDs we found that:

- Average VM boot time ranged from 2.3x to 4.2x faster than any HDD storage configuration tested.
- CPU usage ranged from 16% to 105% higher than any HDD storage configuration tested.
- Lower drive latency during the bootstorm enabled VM boots to be initiated 1.6x to 6.9x faster than any HDD storage configuration tested.

For Internal HDDs we found that:

- Doubling the controller cache size from 512 MB to 1024 MB decreased average boot time by 18%.

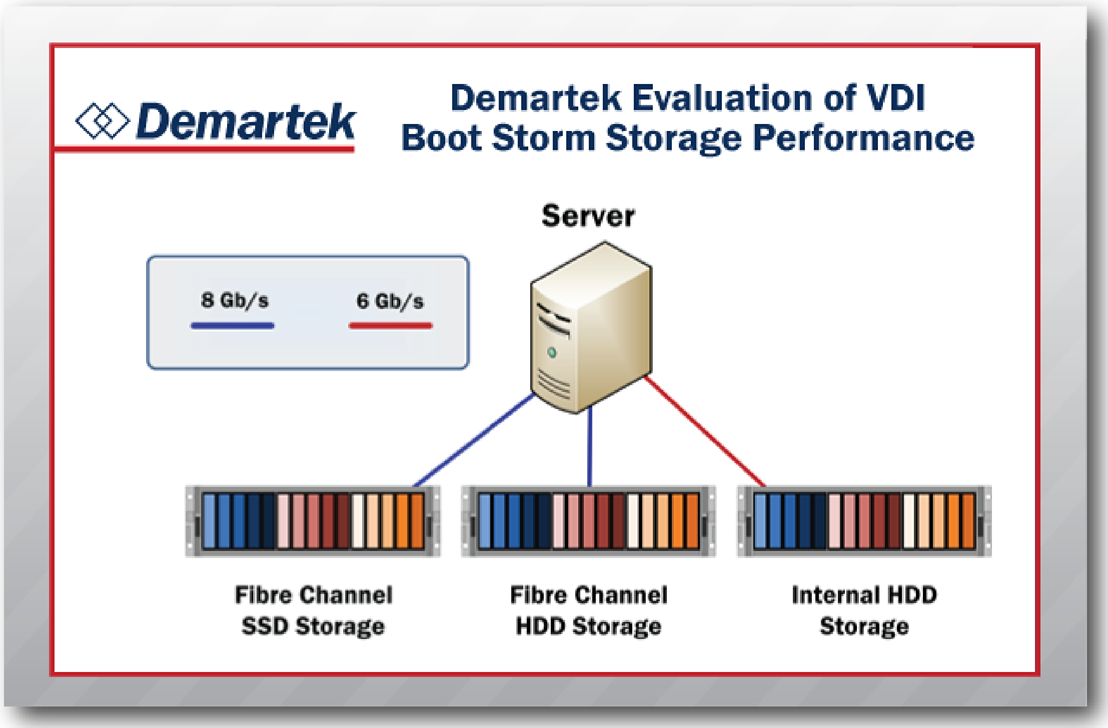
For Fibre Channel host bus adapters (HBAs) we found that:

- Increasing the HBA queue depth from 32 to 254 decreased HDD boot times by 33% and decreased SSD boot times by 18%.
- Increasing the HBA queue depth from 32 to 254 increased HDD CPU usage by 54% and increased SSD CPU usage by 6%.

We recommend that enterprises consider SSDs for VDI storage. If HDDs must be used, a larger cache and increased HBA queue depth will greatly increase performance.

Test Configuration

90 Windows 7 VMs were set up in VMware vSphere 5.1 on the internal HDDs. Each VM had a startup script which recorded the time in a unique file and then transferred the file over to a server. These VMs were updated as of 5/14/13, and then cloned to storage on the FC SSD and FC HDD drives. Scripts accessing the ESXi console recorded the time and then sent commands to power on a VM. ESXTOP was used to record server load and disk I/O data.



Script Delays

When ESXi software sent commands to start all 90 VMs at the same time, the software gave errors when attempting to power on some of the VMs. Most of the errors were due to loading shared libraries. As a result, not all VMs would be issued the power-on command by ESXi software. Adding a slight delay to the VM power-on commands led to all VMs booting properly.

Multiple attempts at delaying power-on commands were tried for each configuration until the minimum time required was found. This time varied for each storage configuration. This fact led to the conclusion that ESXi software libraries wait for a response from storage while powering-on VMs. Lower latency storage configurations made it possible for the ESXi software to handle more boot requests at the same time without denying requests.

Caching

Two different internal HDD configurations were tested, one with a 512MB cache RAID controller and one with a 1024MB cache RAID controller replacing the original RAID controller.

The Fibre Channel HDD configuration included a 1744 MB data cache. During the first test it was configured for read-only caching, and during the second test it was configured for write-back caching as well. Write back caching is not recommended for usage unless there is a backup battery or data can afford to be lost.

Queue Depth

The initial tests using Fibre Channel storage used the default queue depth of 32 for the FC HBA. During the second round of tests, the queue depth was increased to 254.

Data Collection

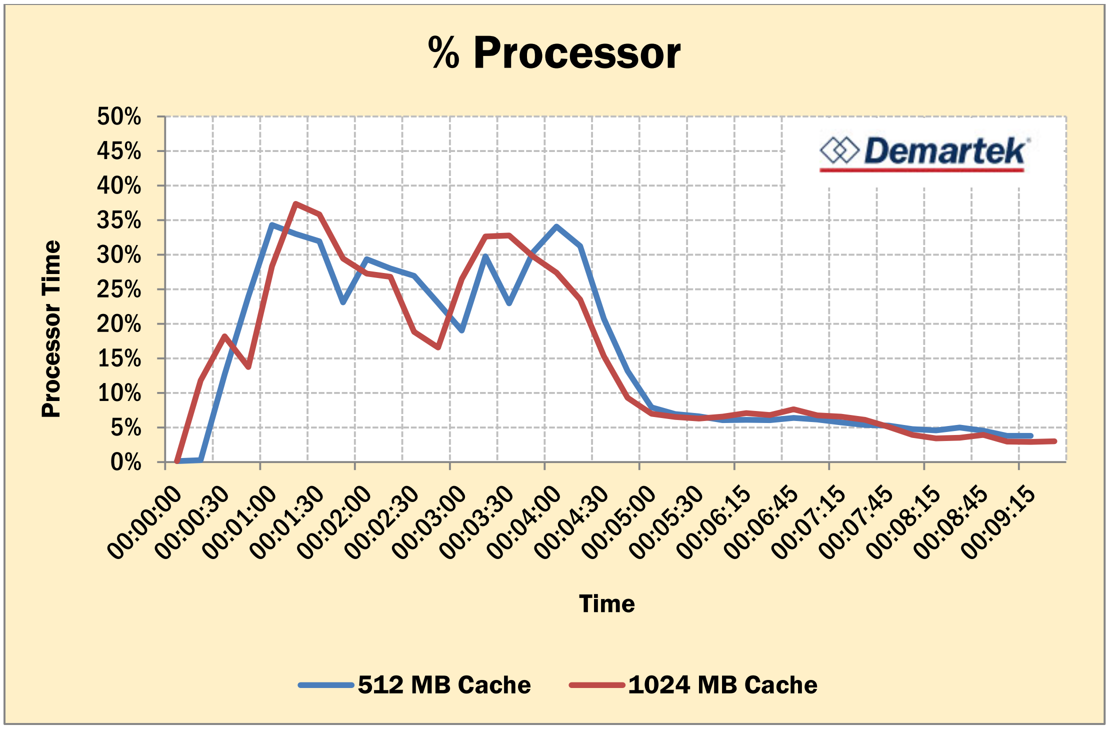
Server load and disk I/O data was collected in intervals of 15 seconds using ESXTOP. Boot-related timestamps gathered by scripts were rounded to the same 15 second intervals for graph production.

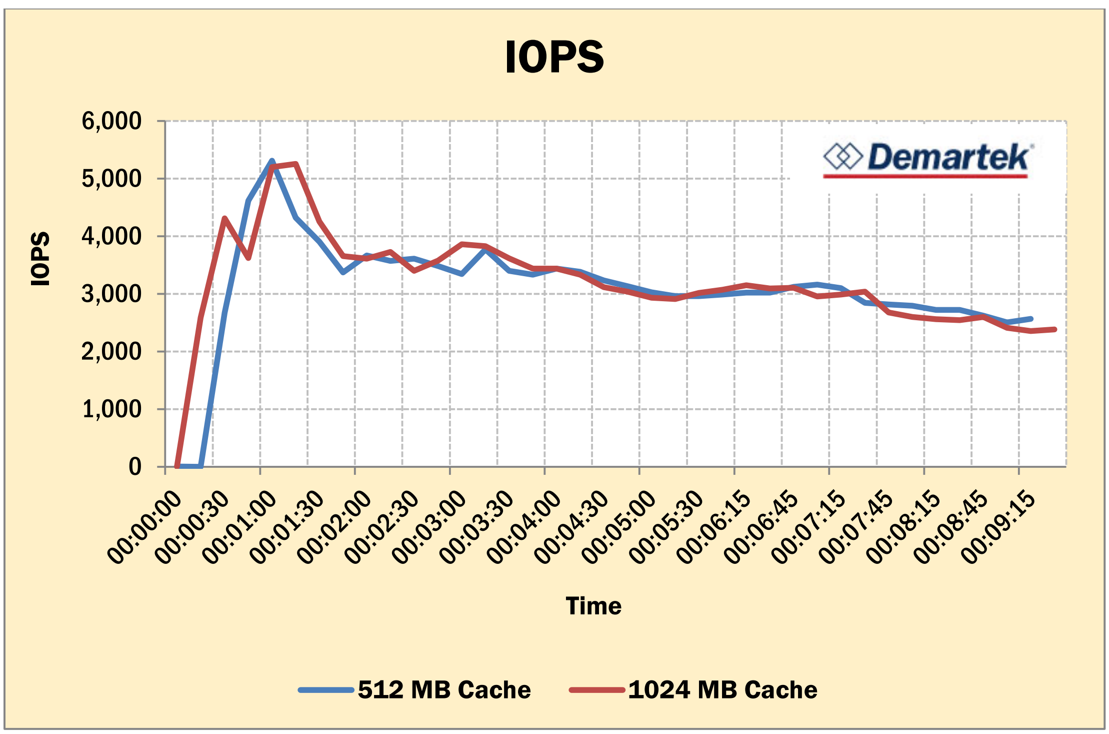
Internal HDD

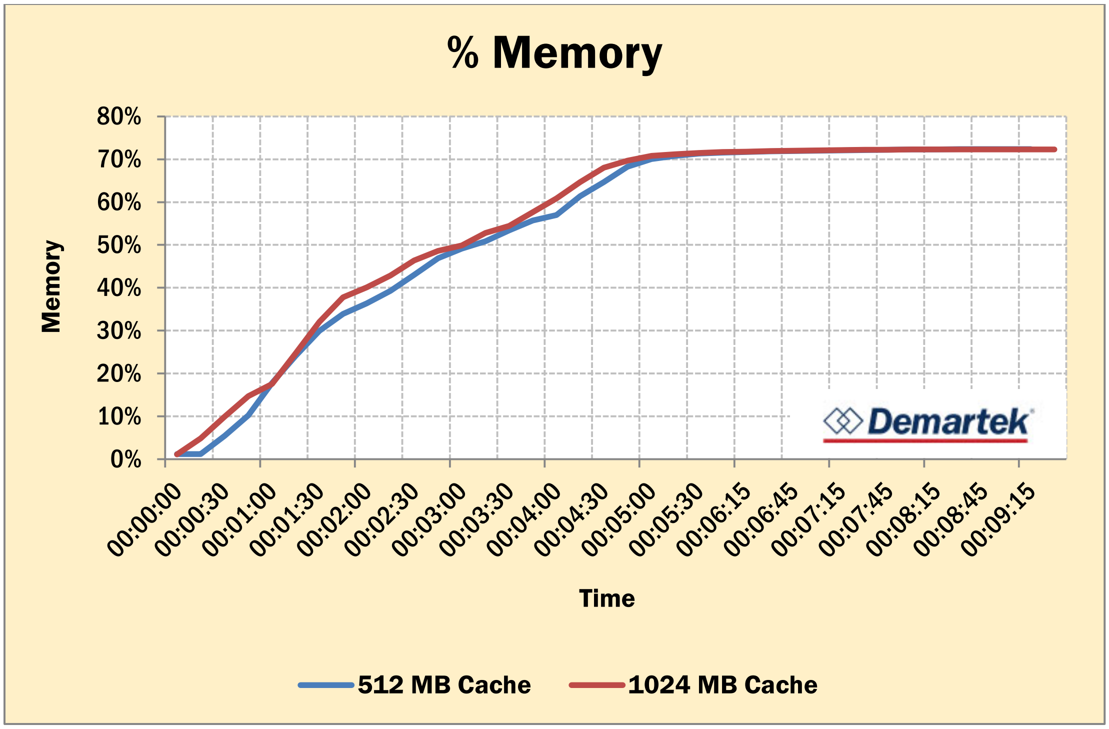
We found that the Internal HDDs had a difficult time keeping up with demand. While doubling the read/write cache size from 512MB to 1024MB offered noticeable improvement, some users would have to wait over 8 minutes before their desktop booted. The server CPU was not over-taxed during the bootstorm, topping out at 41% CPU utilization while using the 1024MB cache.

Demartek	Time Taken to Issue Start Commands (mm:ss)	Time Taken for a VM to Boot			Maximum CPU Utilized	Maximum IOPS
		Minimum	Maximum	Average		
512MB Cache	1:59	00:53	09:40	04:30	37%	5,382
1024MB Cache	1:47	00:34	08:20	03:42	41%	5,842

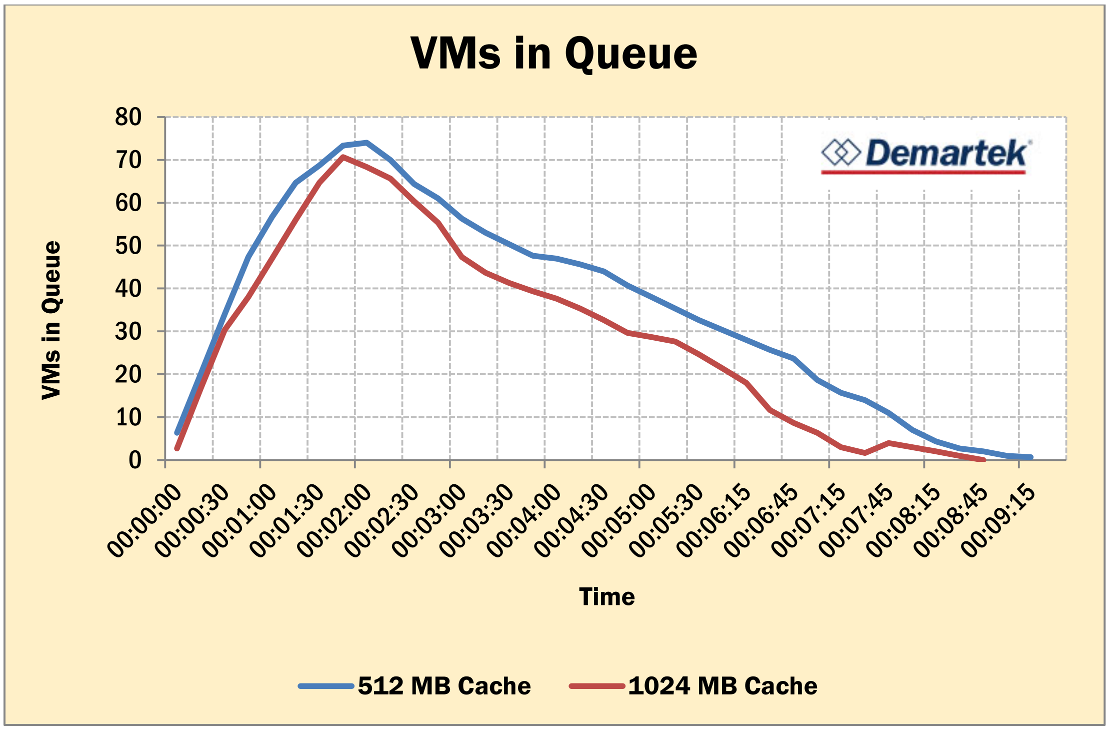
The graphs below show the results averaged over three tests for each configuration. Processor Time peaked slightly higher for the larger cache, while IOPs and memory usage only varied slightly.


% Processor
Processor Time
50%
45%
40%
35%
30%
25%
20%
15%
10%
5%
0%
00:00:00
00:00:30
00:01:00
00:01:30
00:02:00
00:02:30
00:03:00
00:03:30
00:04:00
00:04:30
00:05:00
00:05:30
00:06:15
00:06:45
00:07:15
00:07:45
00:08:15
00:08:45
00:09:15
Time
Demartek
512 MB Cache
1024 MB Cache

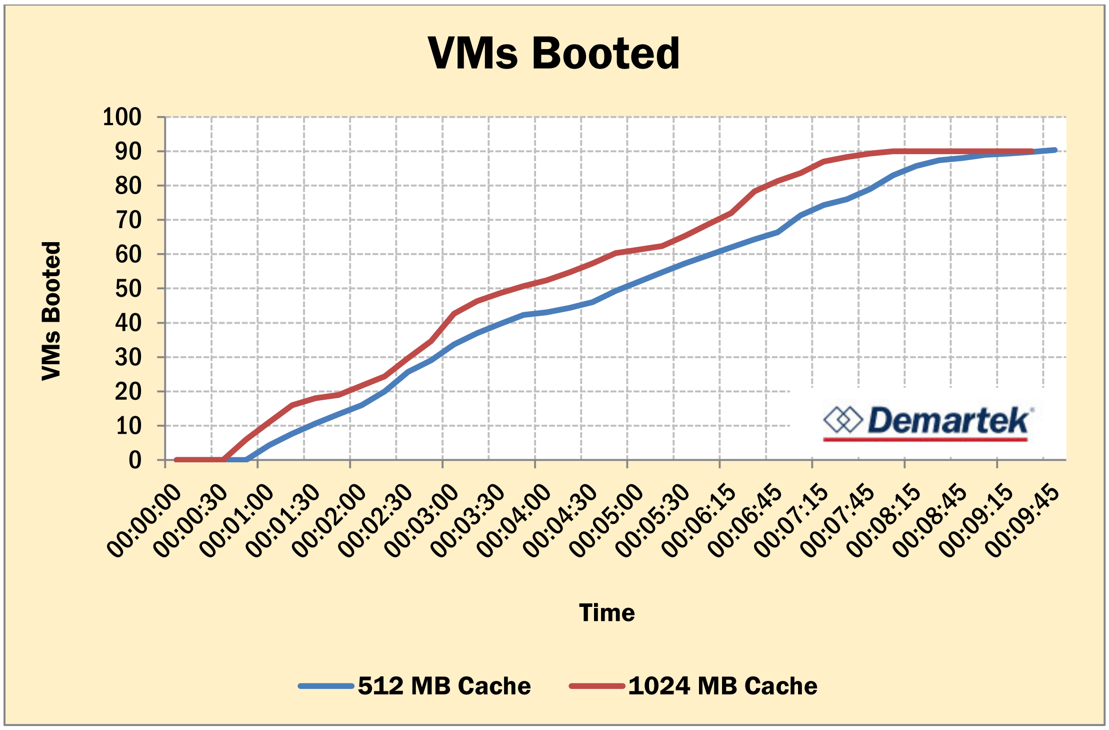

IOPS
IOPS
6,000
5,000
4,000
3,000
2,000
1,000
0
00:00:00
00:00:30
00:01:00
00:01:30
00:02:00
00:02:30
00:03:00
00:03:30
00:04:00
00:04:30
00:05:00
00:05:30
00:06:15
00:06:45
00:07:15
00:07:45
00:08:15
00:08:45
00:09:15
Time
Demartek
512 MB Cache
1024 MB Cache


% Memory
Memory
80%
70%
60%
50%
40%
30%
20%
10%
0%
00:00:00
00:00:30
00:01:00
00:01:30
00:02:00
00:02:30
00:03:00
00:03:30
00:04:00
00:04:30
00:05:00
00:05:30
00:06:15
00:06:45
00:07:15
00:07:45
00:08:15
00:08:45
00:09:15
Time
512 MB Cache
1024 MB Cache
Demartek

However, we found that bootstorms are a cache-friendly workload. Increasing cache size to 1024MB enabled the ESXi software to handle VM start requests faster, which lowered the number of VMs in queue and caused the queue to be completed faster. Boot times were reduced for all VMs by an average of 18%.



Increasing cache size to 1024MB enabled the first VMs to complete boot sooner, giving this configuration a head start. In addition, there was a peak of 8 VMs completing boot in 15 seconds with the 1024MB cache, while the 512MB cache typically peaked at 6 VMs completing boot in 15 seconds. Boots were completed faster with the larger cache, cutting down on boot-time for all VMs.



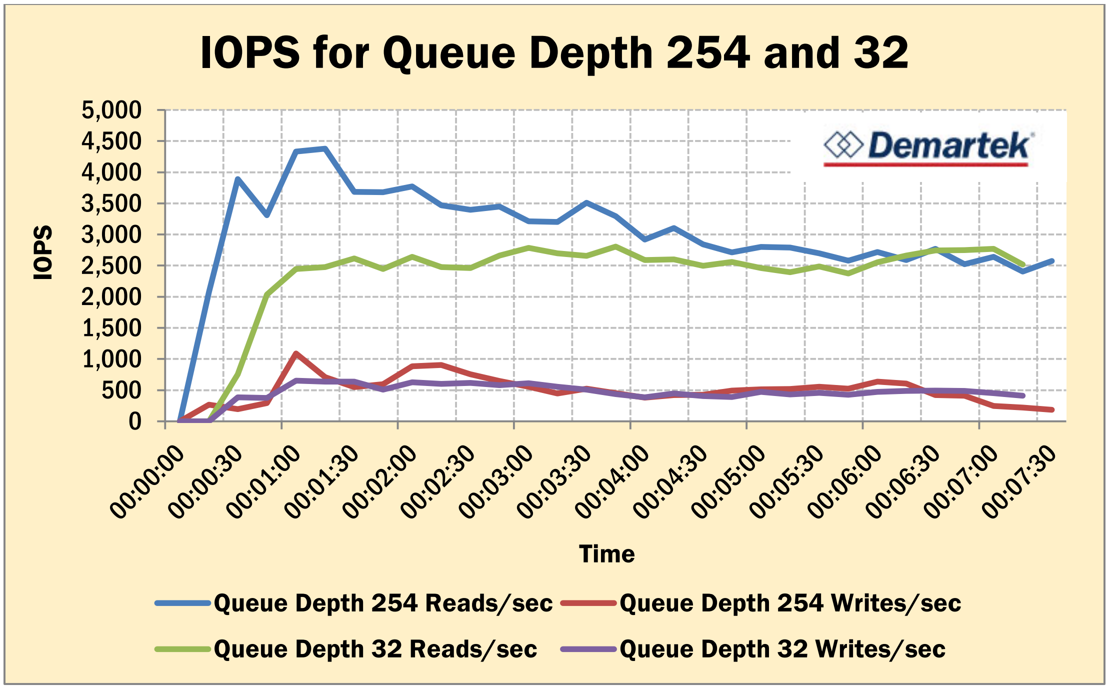
Fibre Channel HDD

The initial Fibre Channel HDD test with queue depth 32 and write through did not have near the same performance as the internal HDD configuration. However, enabling write-back caching and increasing queue-depth to 254 for the second set of tests improved performance dramatically, resulting in a 33% faster boot time on average than the queue depth 32 configuration. This more than brought the performance to levels on par with internal HDDs. Average boot times for the final tests were 9% faster than for the internal HDDs.

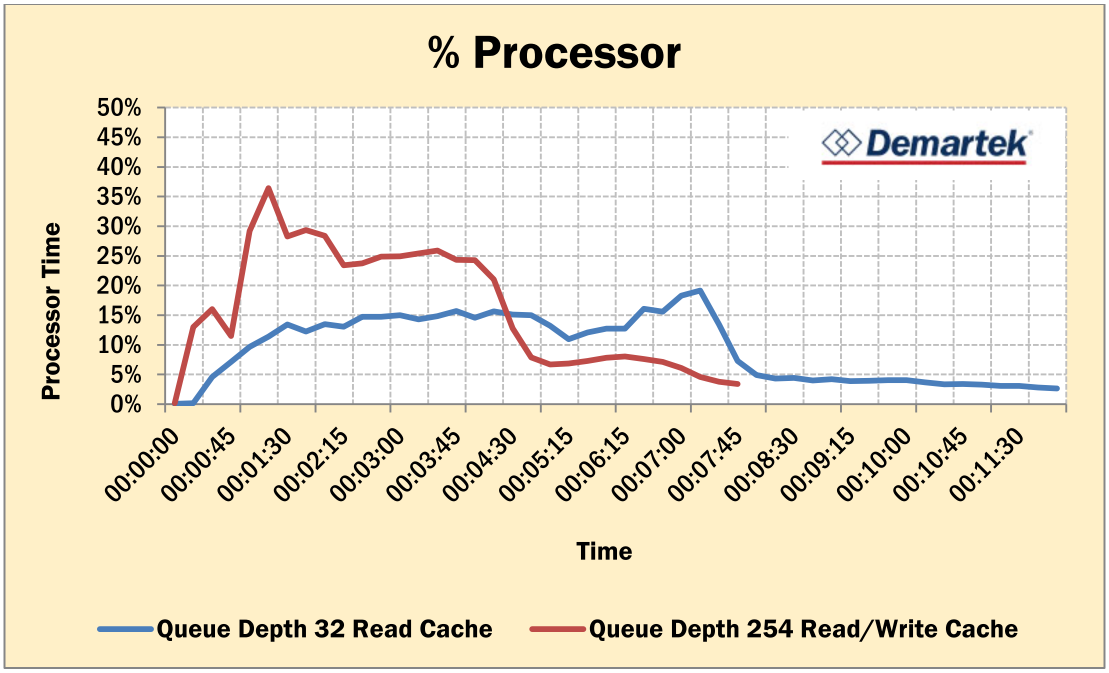
Demartek	Time Taken to Issue Start Commands (mm:ss)	Time Taken for a VM to Boot			Maximum CPU Utilized	Maximum IOPS
		Minimum	Maximum	Average		
Queue Depth 32, 1744MB Read Cache	04:36	00:53	11:20	05:02	25%	3,278
Queue Depth 254, 1744MB Read/Write Cache	01:38	00:41	06:42	03:22	38%	5,669

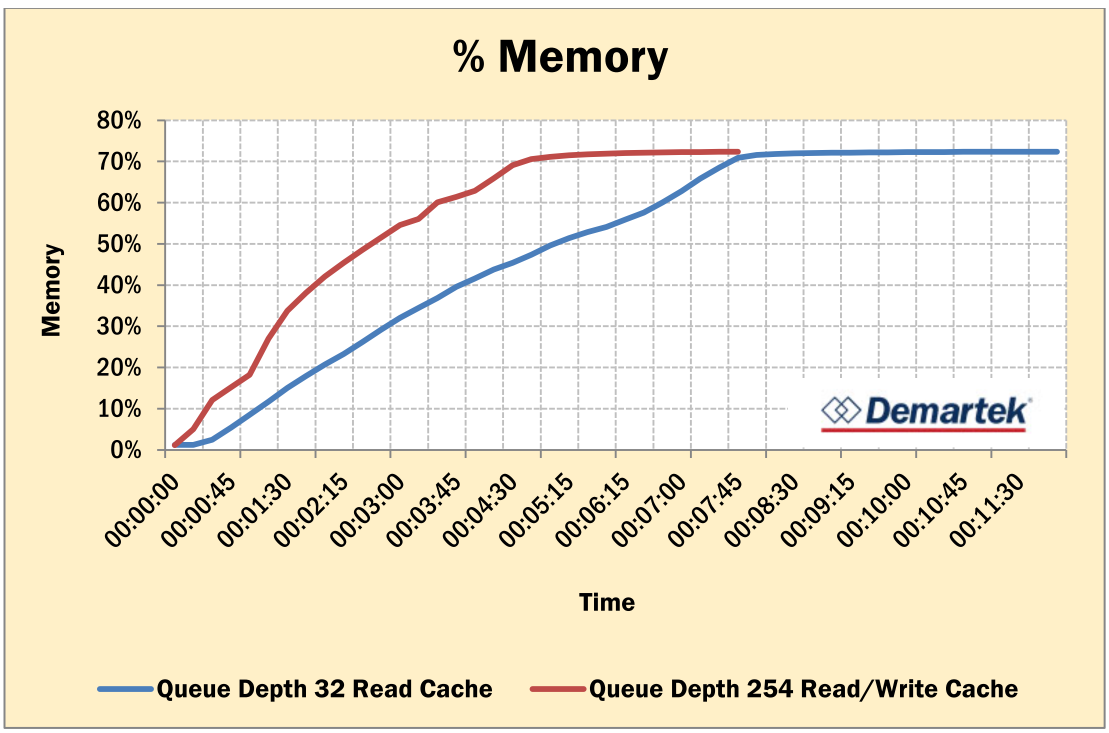
An analysis of read and write operations can help us determine the source of the bulk of the performance improvements. Writes per second increased from 354 to 497 when the caching was enabled and queue depth was increased to 254, a 40% improvement. Reads/sec increased from an average of 2246 to 3010 when queue depth increased to 254, a 34% improvement. Read caching was enabled for both tests, so the improvement in reads/sec is entirely due to the increase in queue depth. We can see that writes/sec increased by 6% more than reads/sec increased. We can conclude that the majority of the improvement in writes/sec was due to the increase in queue depth, with the additional increase in writes/sec being due to the write cache. As most of the performance improvement appears to be due to the increase in queue depth, the initial bottleneck appears to have been due to the HBA settings.

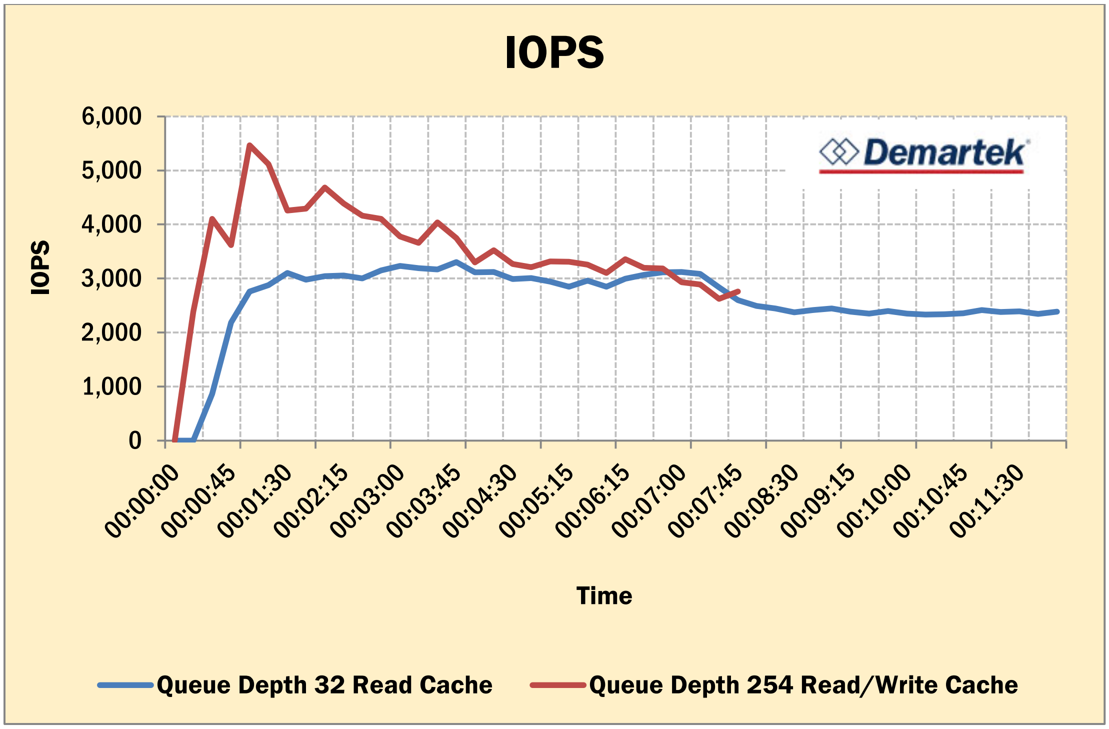
The graphs below show results averaged over three tests for each configuration.



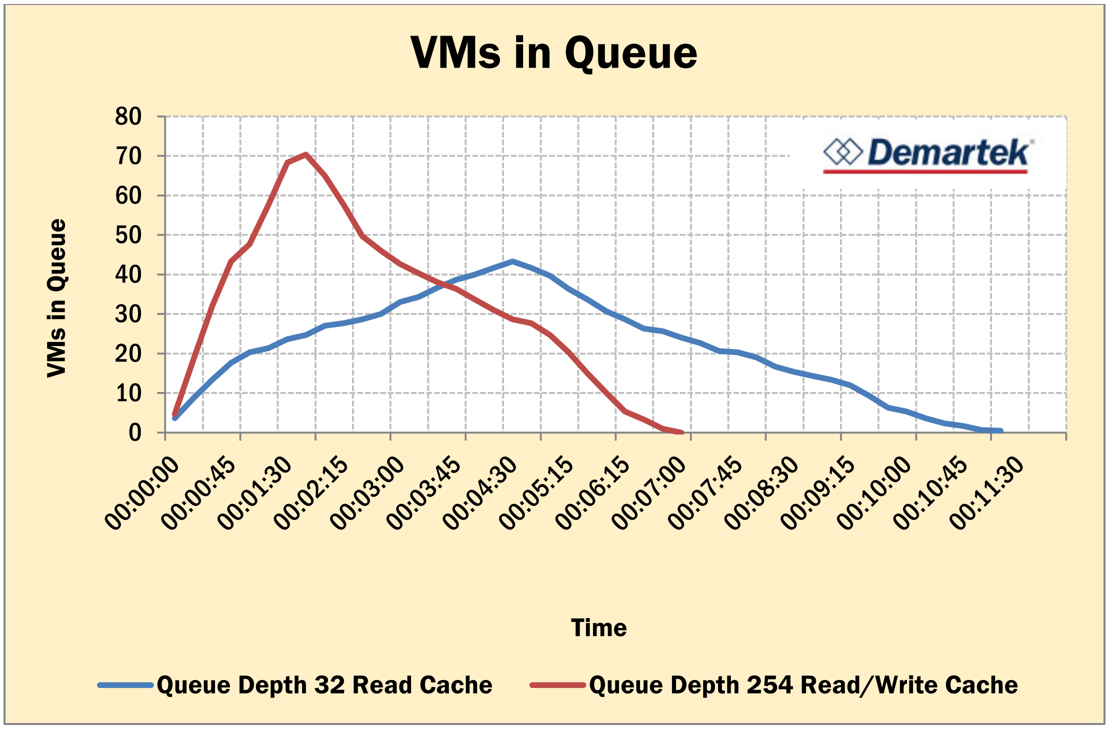
The graphs below indicate that faster storage drove the processor 54% harder and resulted in VMs completing boot sooner.


% Processor
Processor Time
50%
45%
40%
35%
30%
25%
20%
15%
10%
5%
0%
00:00:00
00:00:45
00:01:30
00:02:15
00:03:00
00:03:45
00:04:30
00:05:15
00:06:15
00:07:00
00:07:45
00:08:30
00:09:15
00:10:00
00:10:45
00:11:30
Time
Queue Depth 32 Read Cache
Queue Depth 254 Read/Write Cache

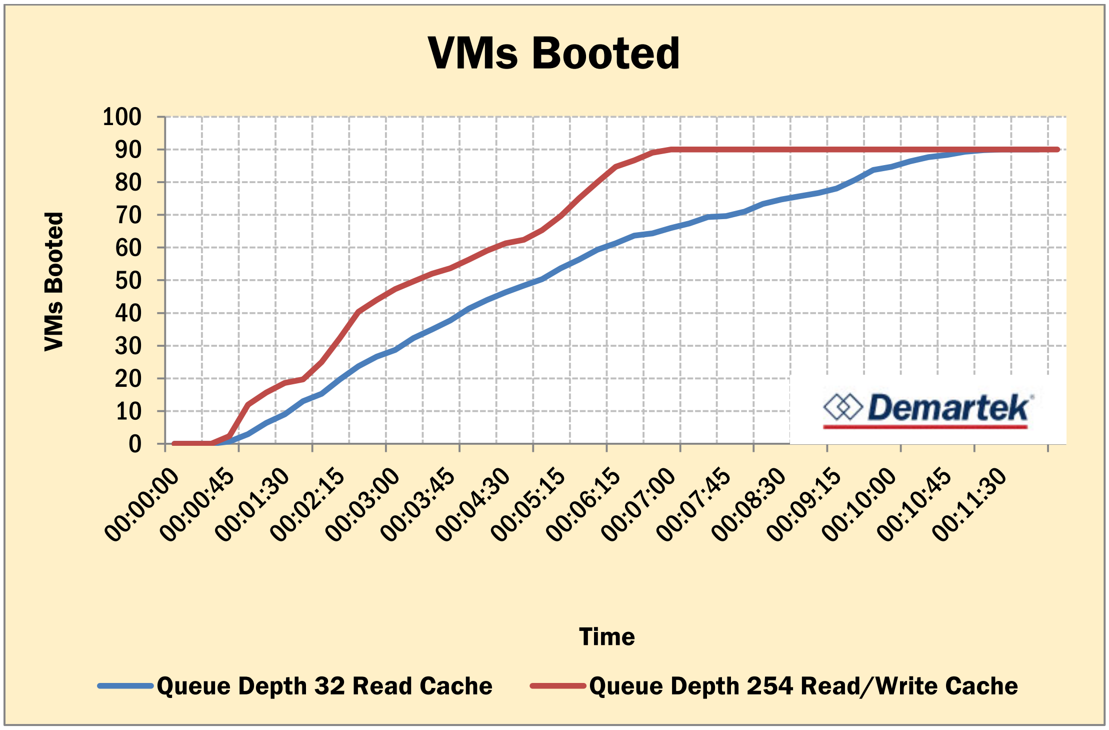

% Memory
Memory
80%
70%
60%
50%
40%
30%
20%
10%
0%
00:00:00
00:00:45
00:01:30
00:02:15
00:03:00
00:03:45
00:04:30
00:05:15
00:06:15
00:07:00
00:07:45
00:08:30
00:09:15
00:10:00
00:10:45
00:11:30
Time
Queue Depth 32 Read Cache
Queue Depth 254 Read/Write Cache



As mentioned previously, ESXi shared libraries wait for a response from storage while powering on VMs, requiring software power-on requests to be delayed to avoid ESXi errors. Below we can see that the reduced latency with queue depth 254 enabled the VM queue to be populated 64% faster, which enabled more VMs to boot at the same time and all VMs to boot faster.


VMs in Queue
80
70
60
50
40
30
20
10
0
VMs in Queue
00:00:00
00:00:45
00:01:30
00:02:15
00:03:00
00:03:45
00:04:30
00:05:15
00:06:15
00:07:00
00:07:45
00:08:30
00:09:15
00:10:00
00:10:45
00:11:30
Time
Queue Depth 32 Read Cache
Queue Depth 254 Read/Write Cache

The rate at which VMs were able to complete boot was faster for queue depth 254 than it was for queue depth 32, as evidenced by the slope of these lines. Again this shows that the bottleneck was mostly due to the HBA settings.



Fibre Channel SSD

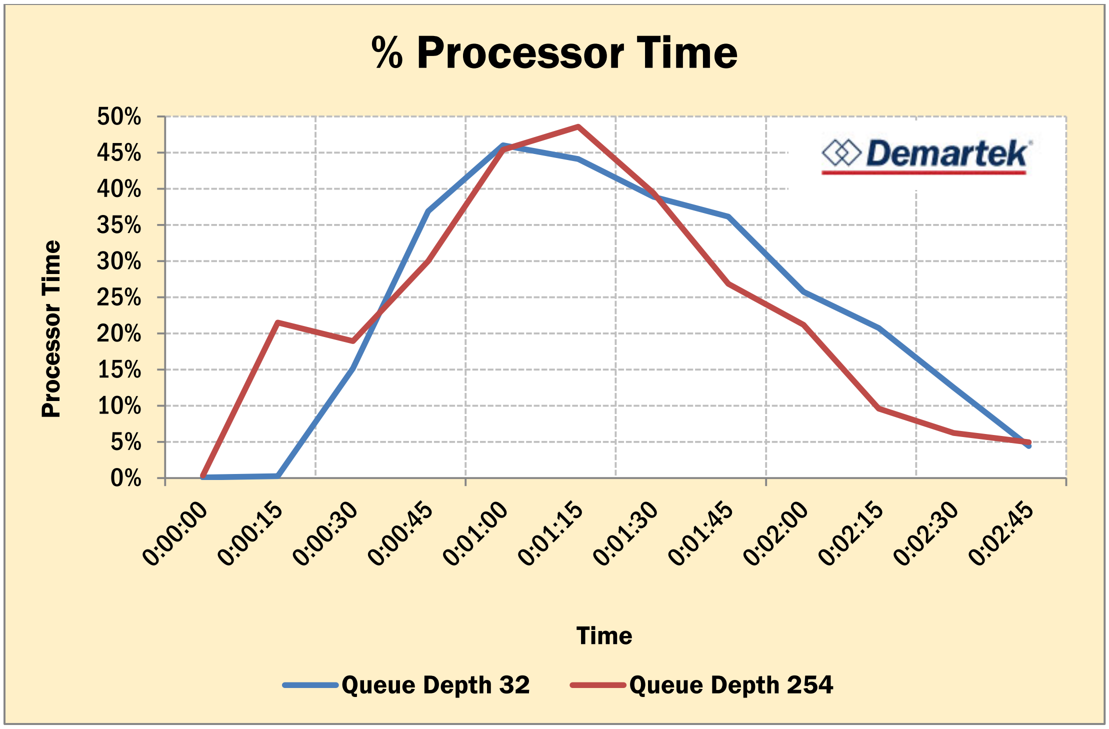
Fibre Channel SSDs were tested with queue depths of 32 and 254. Increasing the queue depth enhanced performance, resulting in an 18% improvement in average boot time.

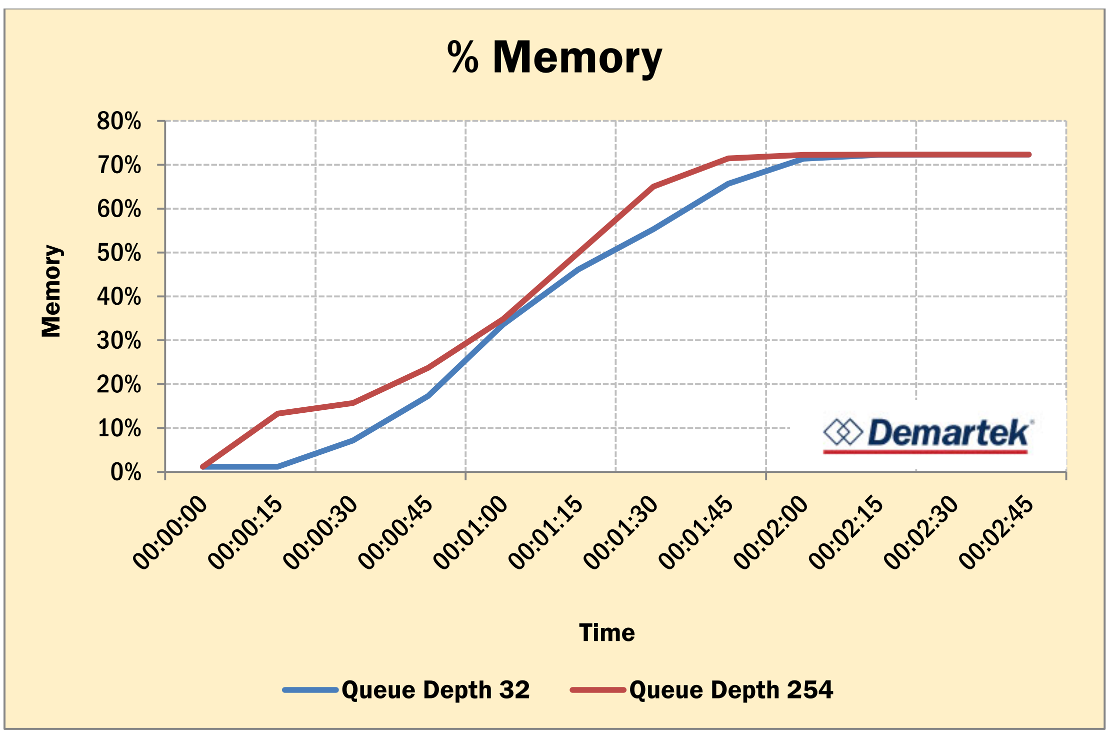
The SSDs were the only drives to produce boot times across the board that would be acceptable to an end user, with the maximum boot time topping out at 2 minutes 17 seconds for the slower queue depth 32 configuration, and less than two minutes for the faster queue depth 254 configuration.

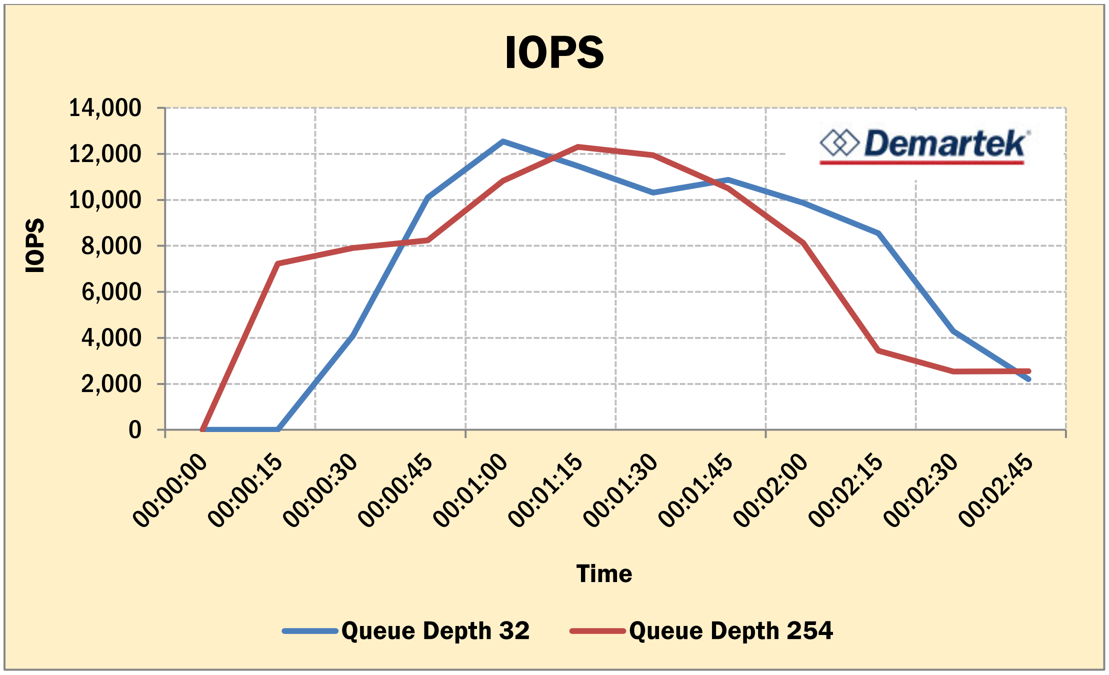
Demartek	Time Taken to Issue Start Commands (mm:ss)	Time Taken for a VM to Boot			Maximum CPU Utilized	Maximum IOPS
		Minimum	Maximum	Average		
Queue Depth 32	1:01	00:35	02:17	01:28	48%	12,981
Queue Depth 254	00:40	00:16	01:57	01:12	50%	12,489

The graphs below show the results averaged over three tests for each configuration.

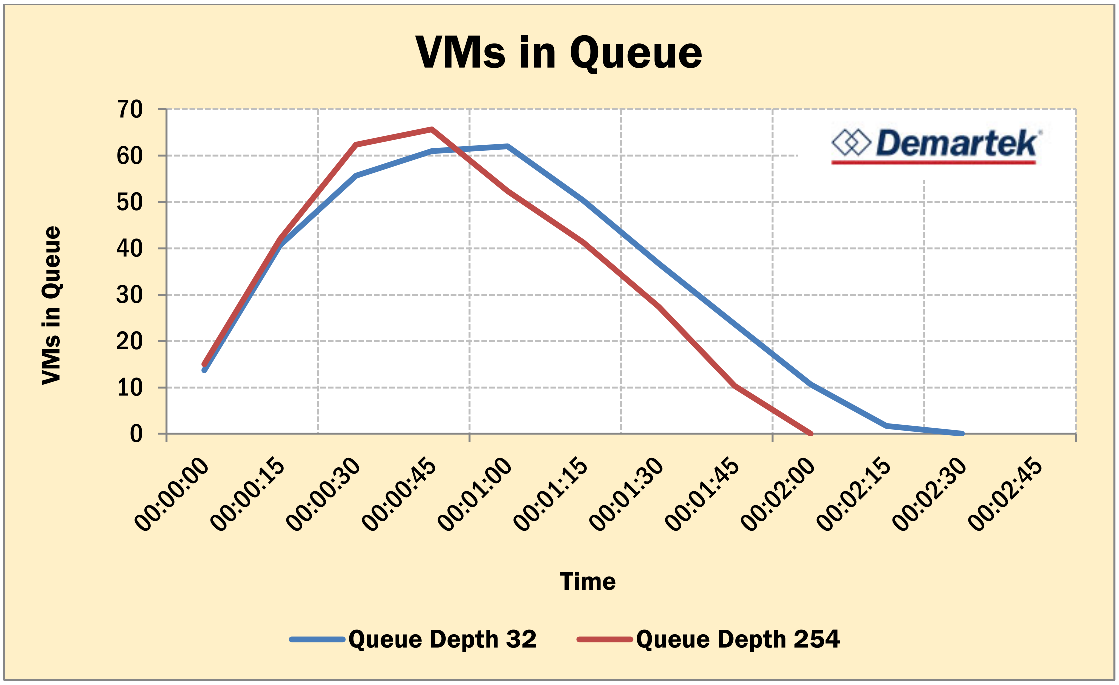
IOPs and CPU vary only slightly between the two tests. CPU was driven only 6% harder on average and IOPS actually decreased by 3% when queue depth was increased to 254. A possible explanation for the decrease in IOPs could be the HBA taking some of the load away from the SSDs.


% Processor Time
Processor Time
50%
45%
40%
35%
30%
25%
20%
15%
10%
5%
0%
0:00:00
0:00:15
0:00:30
0:00:45
0:01:00
0:01:15
0:01:30
0:01:45
0:02:00
0:02:15
0:02:30
0:02:45
Time
Queue Depth 32
Queue Depth 254
Demartek

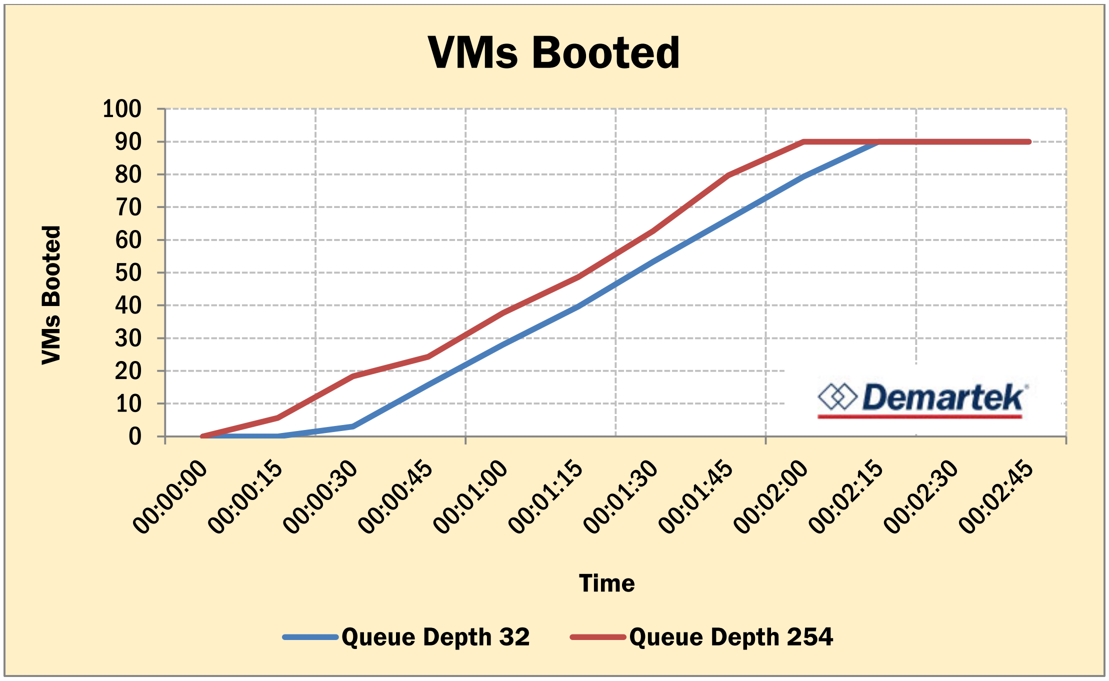

% Memory
Memory
80%
70%
60%
50%
40%
30%
20%
10%
0%
00:00:00
00:00:15
00:00:30
00:00:45
00:01:00
00:01:15
00:01:30
00:01:45
00:02:00
00:02:15
00:02:30
00:02:45
Time
Queue Depth 32
Queue Depth 254
Demartek



Most of the performance increase may be attributed to the VM Queue populating slightly faster, with the time taken to put all VMs into the queue reducing by 24%.We can see the results of this in the graph below where there are more VMs in queue at the beginning of the test, which gives the queue depth 254 configuration a head start.

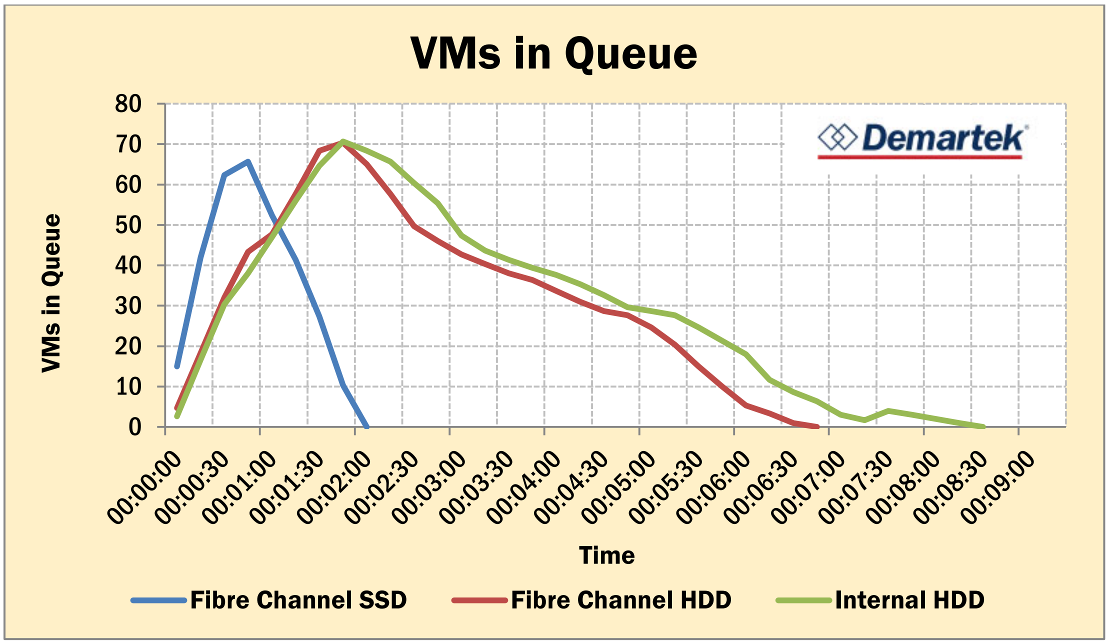


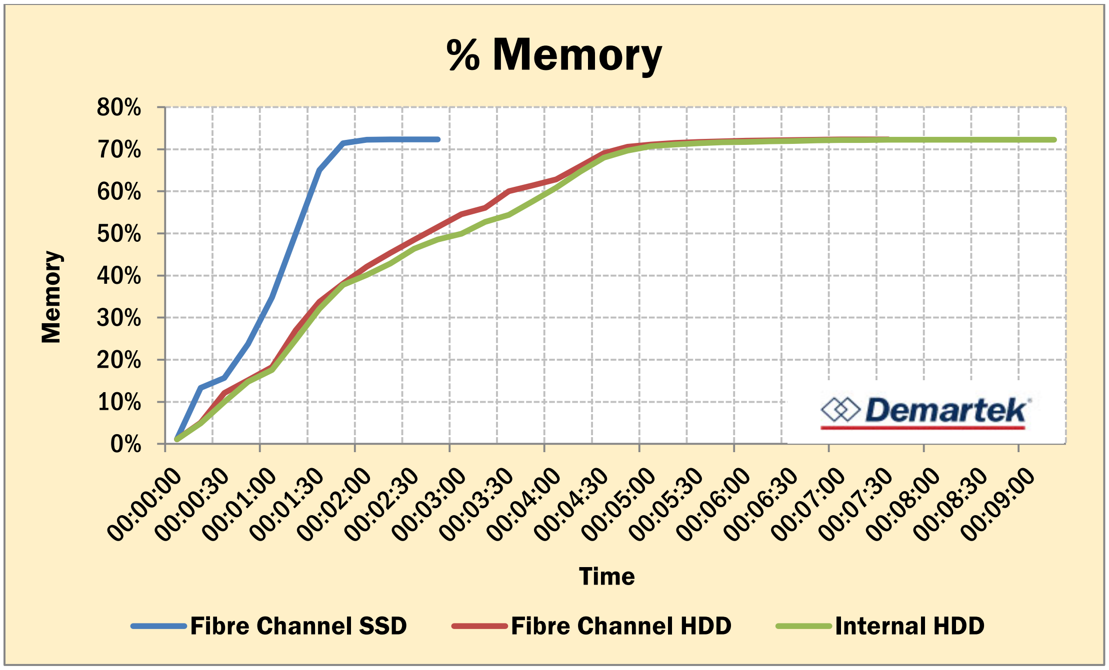
A graph of VMs completing boot shows again how the queue depth 254 configuration gets a head start. The time before the first VM completes boot is shorter when the queue depth is increased to 254, but the slope of both lines for queue depth 254 and queue depth 32 are the same once the boots are started, suggesting that the advantage was again that the first boots were completed faster, giving a head start when the queue depth was increased.


VMs Booted
100
90
80
70
60
50
40
30
20
10
0
VMs Booted
00:00:00
00:00:15
00:00:30
00:00:45
00:01:00
00:01:15
00:01:30
00:01:45
00:02:00
00:02:15
00:02:30
00:02:45
Time
Queue Depth 32
Queue Depth 254
Demartek

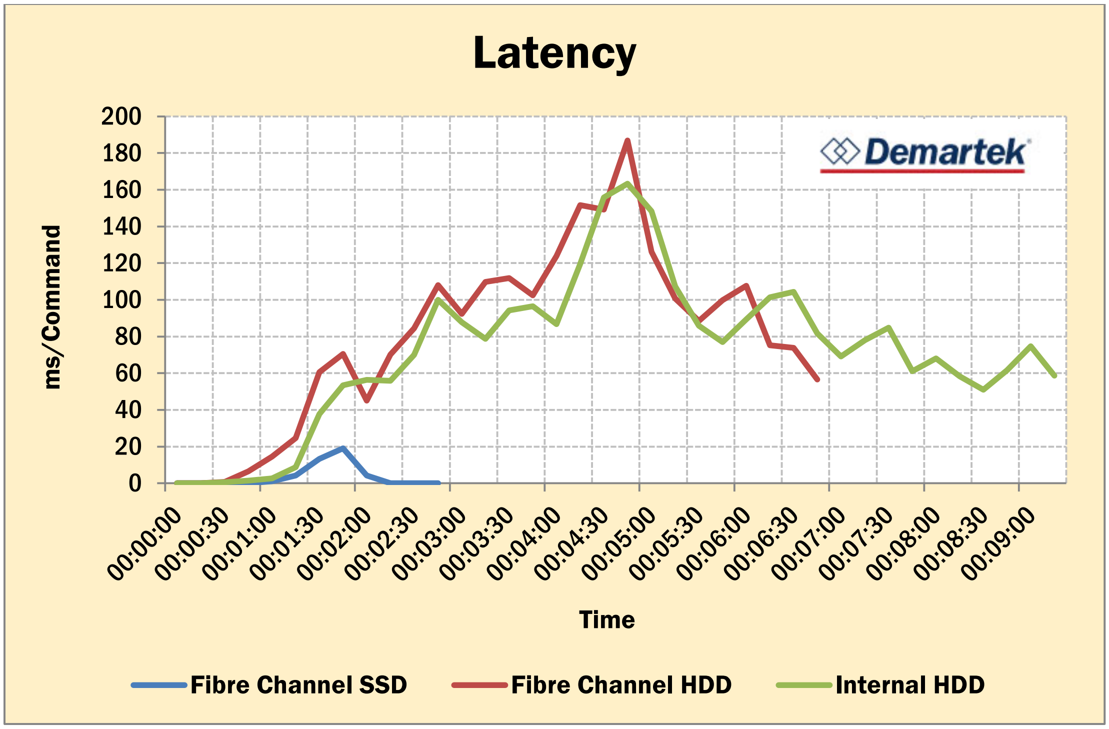
Summary and Conclusion

The data in the graphs below was taken from the queue depth 254 Fibre Channel SSDs, the queue depth 254 Fibre Channel HDDs with write-back, and the 1024MB cache internal HDDs, as these were the optimal configurations for each storage type. The SSDs completed boot much faster than their HDD counterparts.

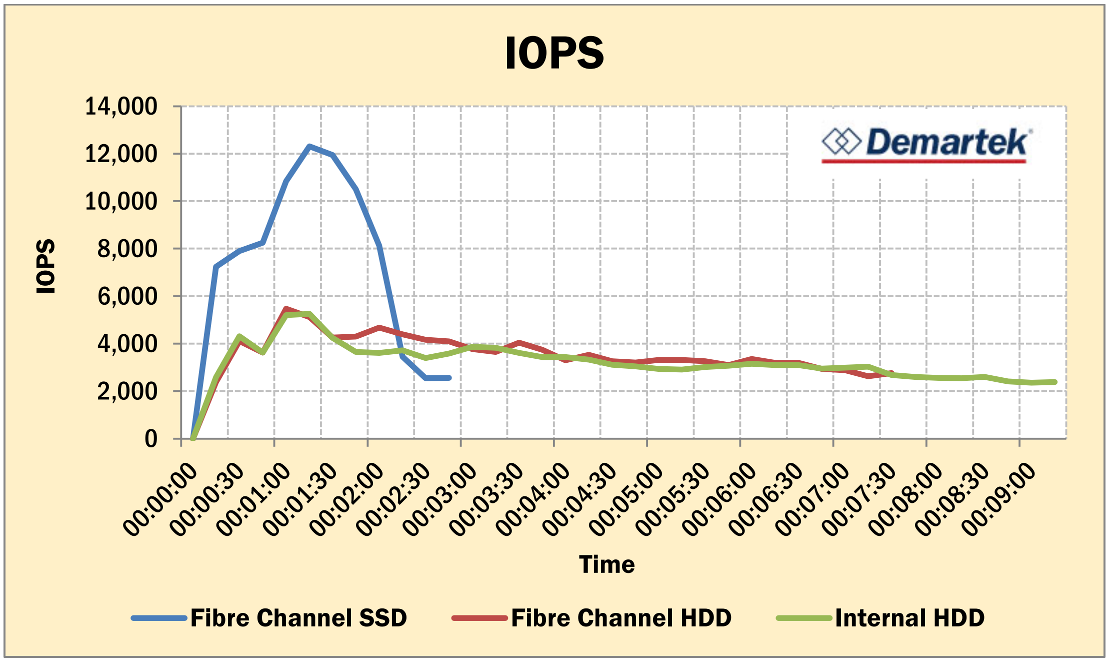




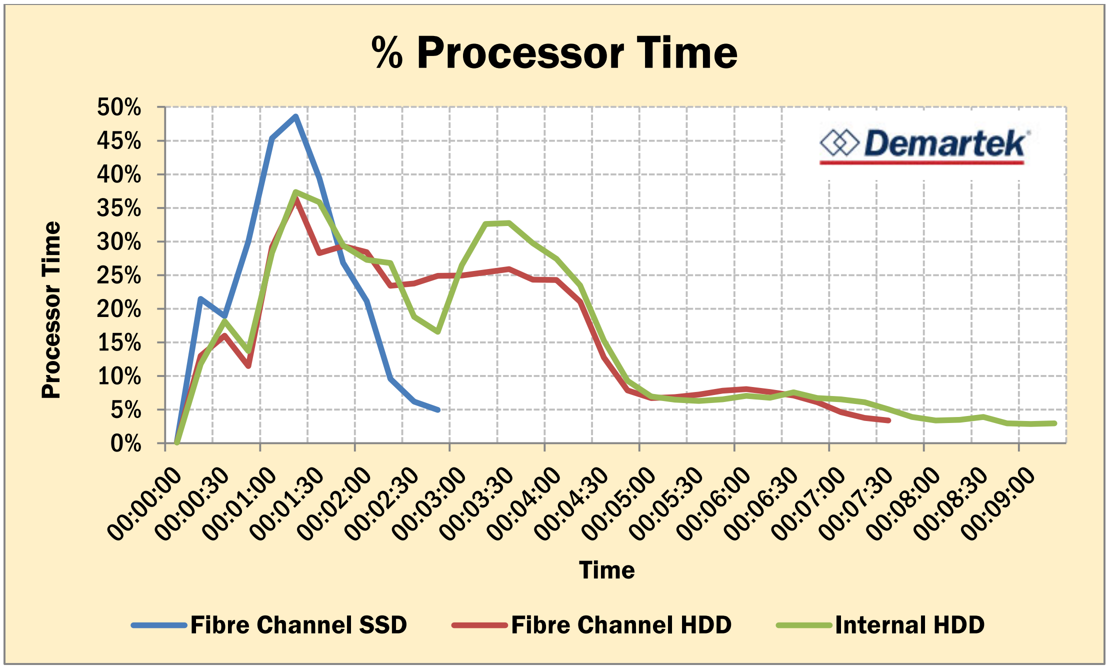
SSDs dramatically outperformed the HDDs in a bootstorm, regardless of configuration. This appears to be due to their low latency. Latency for HDDs increased as more VMs were added to the queue, but latency for solid state drives stayed low the entire time. As the ESXi libraries waited for responses from storage, lower latency SSDs were able to increase ESXi library availability, enabling more VMs to start their boot sequences.


Latency
200
180
160
140
120
100
80
60
40
20
0
ms/Command
00:00:00
00:00:30
00:01:00
00:01:30
00:02:00
00:02:30
00:03:00
00:03:30
00:04:00
00:04:30
00:05:00
00:05:30
00:06:00
00:06:30
00:07:00
00:07:30
00:08:00
00:08:30
00:09:00
Time
Fibre Channel SSD
Fibre Channel HDD
Internal HDD

In addition, solid state drives had the throughput necessary to efficiently handle storage requests from the starting VMs.



However, SSDs drive CPU harder, and may not be an option if processor availability is limited.



HDDs can improve performance by using caching and increasing queue-depth, but do not compete with solid state drive performance.

SSDs are recommended for all bootstorm configurations. Should SSDs not be cost effective, HDDs do have acceptable boot times for the first VMs that are started, before the VM queue gets built up. Data indicates boot times stayed below two minutes for the first 17-19 VMs booted in HDD optimal configurations. HDDs, whether internal or via Fibre Channel, may be acceptable for environments where bootstorms will only consist of 20 VMs or less. It is also possible that more HDD VMs could have booted in under two minutes had the server not been taking new boot requests while attempting to process storage requests from the already booting VMs, possibly delaying their storage requests. Special care must be paid to caching configuration and queue depth when deploying HDDs to achieve optimal performance.

Appendix – Evaluation Environment

The tests were conducted in the Demartek lab in Colorado.

Storage

- Internal HDD
  – 15x 15K 136.13 GB HDD (1.98TB RAID 0 LUN used).
- External HDD (via Fibre Channel)
  – 12x 15K 300GB HDD, 2x 8Gb interfaces (2.5TB RAID 0 LUN used).
- External SSD (via Fibre Channel)
  – 24x 100GB SSD, 2x 8Gb interfaces (2.04TB RAID 0 LUN used).

Server

- Dell PowerEdge R820
  – 4x Intel ® Xeon ® E5-4650, 2.70 GHz, 256 GB RAM
  – ESXi 5.1
  – 1 Dell PERC H710P adapter, 1024MB cache
  – 1 Dell PERC H710 adapter, 512MB cache
  – 1 Brocade 1860 16Gb Fibre Channel HBA (used 1 interface)

Virtual Machines

- – 1 CPU, 2 MB Memory, 20 GB OS drive (thin provisioned)
  – Windows 7 Ultimate SP1, with current updates as of 5/14/13.

The most current version of this report is available at http://www.demartek.com/Demartek_Analysis_of_VDI_Storage_Performance_During_Bootstorm.html on the Demartek website.