

SQL Server 2012: Evaluating and Sizing Hardware

Module 5: Choosing an Appropriate Storage Type for SQL Server 2012

Glenn Berry

Glenn@SQLskills.com



Introduction

- **Considering your workload for storage evaluation and sizing**
- **Common storage types for SQL Server 2012**
- **RAID levels and SQL Server 2012 workloads**
- **The effects of RAID levels on storage sizing and performance**
- **Different methods of evaluating storage performance**
- **Traditional magnetic storage vs. flash-based storage**
- **Storage sizing techniques for performance**

Considering Your Workload for Storage

- **SQL Server can have several different workload types**
- **Three most common types:**
 - Online Transaction Processing (OLTP)
 - Relational Data Warehouse (DW)
 - Online Analytical Processing (OLAP)
- **These workload types have different I/O access patterns**
 - OLTP workload has frequent writes to data files and log file
 - Also has random reads from data files if database does not fit in memory
 - Random I/O performance is very important
 - DW workload has large sequential reads from data files
 - Sequential I/O performance is very important
 - OLAP workload has lots of random reads from cube files
 - Random I/O performance is very important

Additional Workload Considerations

- **You may have a mixed I/O workload for several reasons**
 - If you have multiple databases on the same instance
 - This complicates and randomizes the I/O workload
 - If you have multiple databases with log files on the same LUN
 - This will make the I/O workload on that LUN random instead of sequential
 - If you will be using HA/DR features that read from the transaction log
 - This will cause reads from the LUN where the log files are located
 - Index creation and maintenance will cause sequential I/O pressure
 - Reads and writes to data files, and writes to log file
 - Database backups will cause sequential I/O pressure
 - Reads from data files and log files, and writes to the backup file(s)
 - Database restores will cause sequential I/O pressure
 - Reads from backup file(s) and writes to data files and log file

Choosing the Storage Type for SQL Server

- **Depends on server usage, performance requirements, budget**
 - Existing infrastructure, employee skillset, and politics also matter
- **Five main storage types**
 - Internal drives - traditional magnetic drives or solid state drives (SSDs)
 - PCI-E flash-based storage cards
 - Direct-attached storage (DAS) - traditional magnetic drives or SSDs
 - Storage area networks (SAN) - traditional magnetic drives or SSDs
 - Server message block (SMB) file shares (Windows Server 2012)
- **Internal, DAS and SAN can use hybrid or tiered-storage**
 - Mixture of magnetic storage and SSD storage
 - Good compromise between space, performance, and cost
- **Storage details can make a huge difference for I/O performance**
 - 10K drives versus 15K drives, 3Gbps SAS versus 6Gbps SAS
 - Number of drive spindles, RAID levels, and amount of free space
 - Bandwidth of RAID controller, HBA, or iSCSI NIC is very important

RAID Basics

- **Redundant array of inexpensive disks (RAID)**
 - Standardized method of managing multiple drives with a controller
 - Provides redundancy and higher performance than a single drive
 - Allows higher capacity logical drives than is possible with one drive
- **Hardware RAID controllers manage multiple drives**
 - Server RAID controllers have dedicated cache memory
 - RAID controller cache can be used for reads or writes or both
 - For SQL Server, it is usually better to use RAID cache for writes
- **Several different RAID levels are commonly used**
 - RAID 1
 - RAID 5
 - RAID 50
 - RAID 10

RAID 1

- **RAID 1 is called mirroring**
 - Requires two physical drives
 - Data is copied to both drives
 - Requires 50% storage space overhead
 - Drive array can survive the loss of one drive
 - You need to replace the failed drive and allow the RAID controller to automatically rebuild the mirror as soon as possible
 - No performance impact after the loss of one drive
- **Very common to install the OS to a RAID 1 volume on a server**
 - Usually done with two internal drives in the server
 - This allows the server to operate normally after losing one drive

RAID 5

- **RAID 5 is called striping with parity**
 - Requires at least three physical drives
 - Data is striped between all drives
 - After data is written to all drives, parity information is calculated and then striped to all of the drives
 - This causes a write performance penalty
 - This allows the array to survive the loss of one drive in the array
 - Performance is severely affected after the loss of one drive
 - Failed drive must be replaced as soon as possible
 - Requires $1/(\text{the number of drives})$ as storage overhead
- **RAID 5 is very popular with I.T. departments**
 - It is quite economical because of low storage overhead
 - Risk of failure goes up as you add more drives to the array

RAID 50

- **RAID 50 is called striping across multiple RAID 5 data sets**
 - Requires at least six physical drives
 - Minimum of two, three-drive RAID 5 arrays
 - Requires $1/(\text{number of drives})$ in each RAID 5 array for storage overhead
 - Can survive the loss of one drive in each RAID 5 array
 - Performs better than RAID 5 after the loss of one drive
 - Can be a good compromise between RAID 5 and RAID 10
 - Less expensive than RAID 10
 - More expensive than RAID 5, but provides better redundancy
 - Not all RAID controllers support RAID 50

RAID 10

- **RAID 10 is called a striped set of mirrors**
 - Data is mirrored and then striped
 - Possible to survive the loss of more than one drive
 - Requires a minimum of four physical drives
 - Must be an even number of physical drives
 - No write performance penalty
 - Very well-suited to write intensive workloads
 - Ideal for SQL Server log files
 - Requires a 50% storage space overhead
 - More expensive than RAID 5
- **RAID 10 is very popular with database administrators**
 - Provides better write performance and better redundancy than RAID 5
 - It is more expensive than RAID 5

Raid Level and SQL Server Workloads

- **The number of spindles in an array is extremely important**
 - A larger number of smaller drives will perform much better than a small number of larger drives
- **RAID 5 has a write performance handicap**
 - RAID 5 cannot survive the loss of more than one disk in an array
 - RAID 5 arrays with larger numbers of disks are more likely to lose a disk
 - Try to put infrequently accessed data on RAID 5 to save money
- **RAID 10 and RAID 1 have very good write performance**
 - RAID 10 also has more redundancy than RAID 5
 - Always try to use RAID 1 or RAID 10 for log files
- **As a DBA, don't negotiate with yourself on storage**
 - Ask for RAID 10, and then negotiate down if necessary
 - Use RAID 5 for data files and backup files if necessary

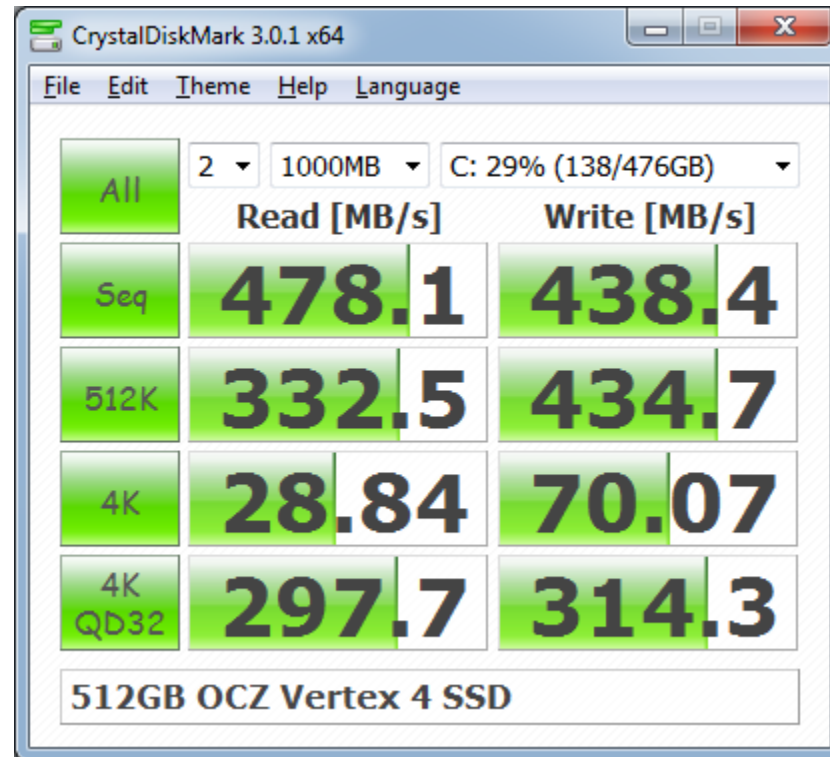
Testing Your Logical Drive Performance

- **CrystalDiskMark is a quick way to test drive performance**
 - Enables you to test each logical drive in a few minutes
 - Tests sequential and random I/O performance
 - Use for first round of testing, before you use SQLIO
- **SQLIO allows you to do much more detailed logical drive testing**
 - Does not require SQL Server to be installed
 - Does not generate a database specific workload
 - Can be much more time consuming to run comprehensive tests

CrystalDiskMark 3.0.1 x64

- **Very easy to use, no complicated configuration required**
 - You can choose the file size for the test runs
 - 50MB, 100MB, 500MB, 1000MB, 2000MB, 4000MB
 - You can choose the file type
 - Random data or non-random data
 - Some SSD controllers use compression for performance
 - Random data is not very compressible
 - You can choose the number of test runs (1-9)
- **Quickly measures sequential and random I/O performance**
 - Sequential reads and writes in MB/second
 - Large and small random reads and writes at different queue depths
 - Measured in MB/sec and IOPS
 - Higher queue depth is more similar to a SQL Server workload
 - Free download
 - <http://bit.ly/TDoGOi>

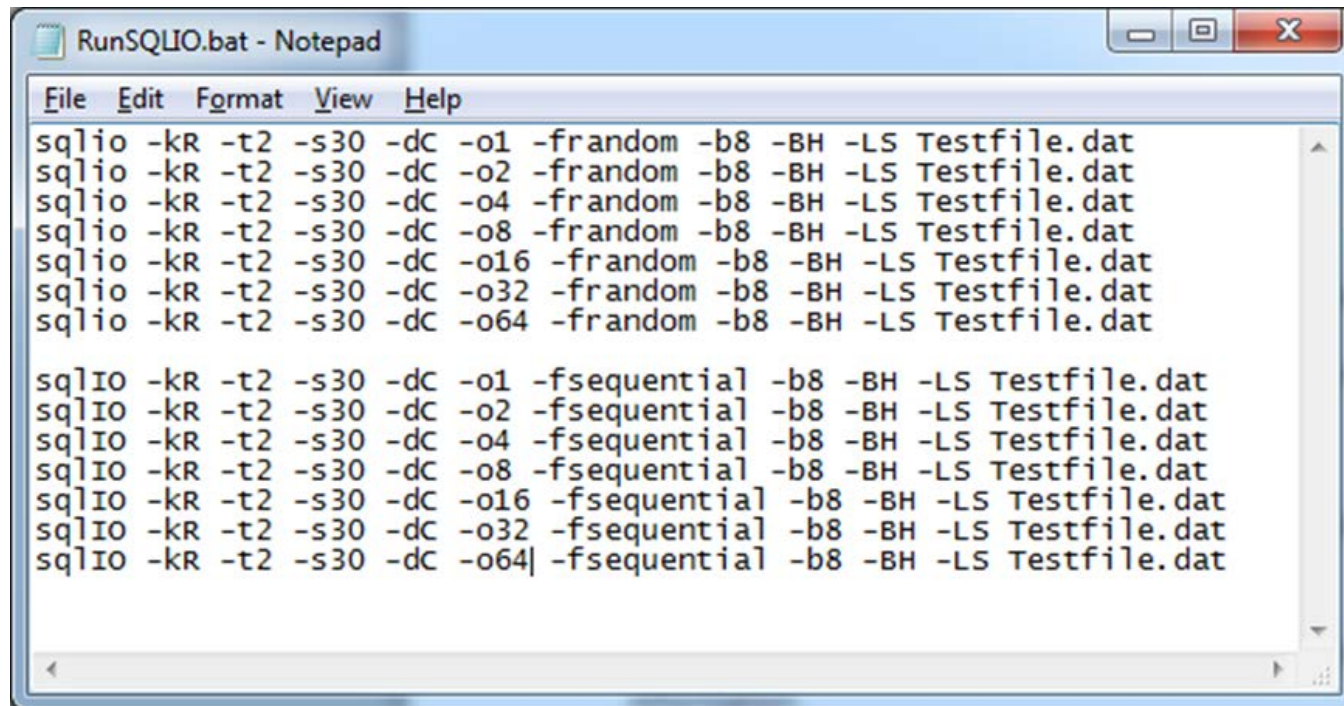
CrystalDiskMark Example Output



SQLIO Disk Benchmark

- **Despite the name, it has nothing to do with SQL Server**
 - Free tool developed by Microsoft to evaluate I/O performance
 - You can use it on any server running a recent version of Windows
- **Command-line utility**
 - Requires some expertise to properly configure and run
 - Can take a long time to run a comprehensive set of tests
- **Allows you to test the limits of your I/O subsystem**
 - Measures IOPS
 - Sequential throughput in MB/second
 - Latency in milliseconds
- **Download location**
 - <http://bit.ly/QxwUV8>

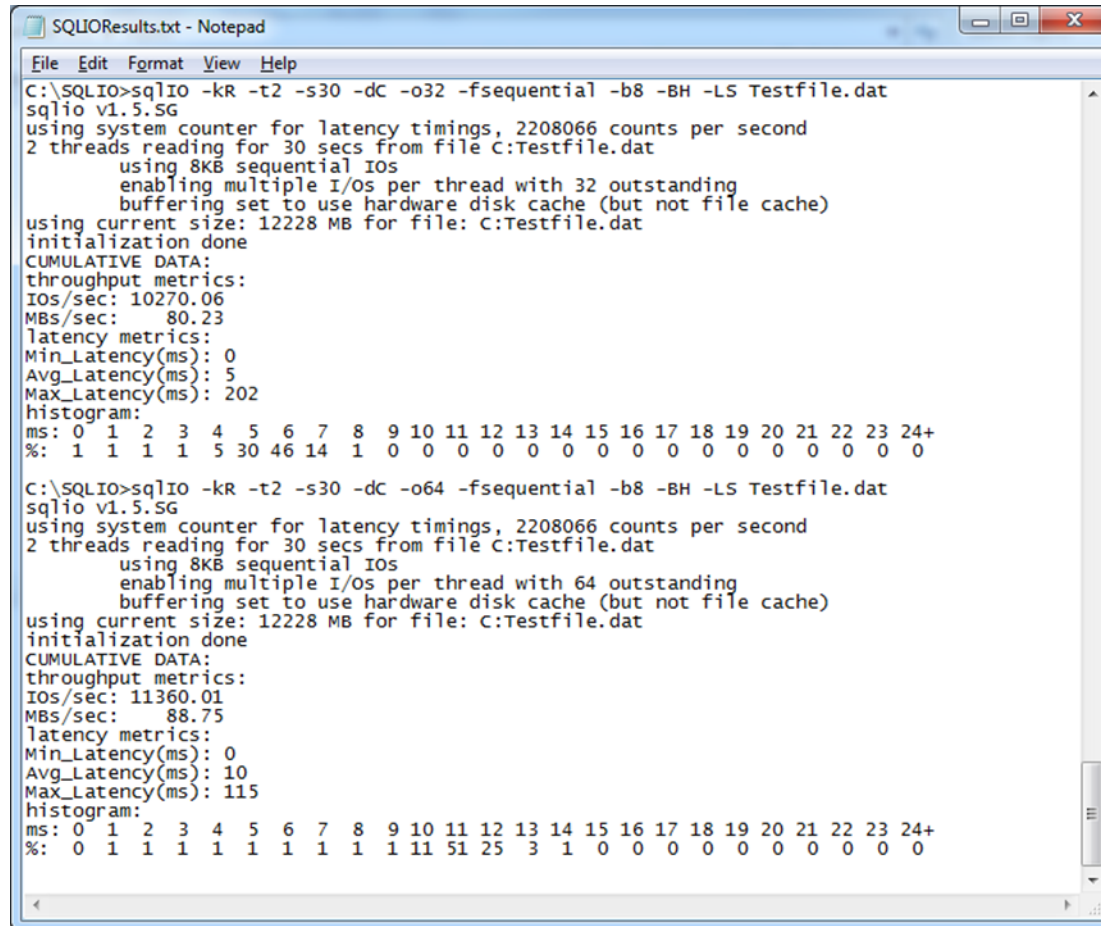
SQLIO Example Batch File



```
RunSQLIO.bat - Notepad
File Edit Format View Help
sqlio -kR -t2 -s30 -dC -o1 -frandom -b8 -BH -LS Testfile.dat
sqlio -kR -t2 -s30 -dC -o2 -frandom -b8 -BH -LS Testfile.dat
sqlio -kR -t2 -s30 -dC -o4 -frandom -b8 -BH -LS Testfile.dat
sqlio -kR -t2 -s30 -dC -o8 -frandom -b8 -BH -LS Testfile.dat
sqlio -kR -t2 -s30 -dC -o16 -frandom -b8 -BH -LS Testfile.dat
sqlio -kR -t2 -s30 -dC -o32 -frandom -b8 -BH -LS Testfile.dat
sqlio -kR -t2 -s30 -dC -o64 -frandom -b8 -BH -LS Testfile.dat

sqlIO -kR -t2 -s30 -dC -o1 -fsequential -b8 -BH -LS Testfile.dat
sqlIO -kR -t2 -s30 -dC -o2 -fsequential -b8 -BH -LS Testfile.dat
sqlIO -kR -t2 -s30 -dC -o4 -fsequential -b8 -BH -LS Testfile.dat
sqlIO -kR -t2 -s30 -dC -o8 -fsequential -b8 -BH -LS Testfile.dat
sqlIO -kR -t2 -s30 -dC -o16 -fsequential -b8 -BH -LS Testfile.dat
sqlIO -kR -t2 -s30 -dC -o32 -fsequential -b8 -BH -LS Testfile.dat
sqlIO -kR -t2 -s30 -dC -o64 -fsequential -b8 -BH -LS Testfile.dat
```


SQLIO Example Output



```
SQLIOResults.txt - Notepad
File Edit Format View Help
C:\SQLIO>sqlio -kr -t2 -s30 -dc -o32 -fsequential -b8 -BH -LS Testfile.dat
sqlio v1.5.SG
using system counter for latency timings, 2208066 counts per second
2 threads reading for 30 secs from file C:\Testfile.dat
    using 8KB sequential IOS
    enabling multiple I/Os per thread with 32 outstanding
    buffering set to use hardware disk cache (but not file cache)
using current size: 12228 MB for file: C:\Testfile.dat
initialization done
CUMULATIVE DATA:
throughput metrics:
IOS/sec: 10270.06
MBs/sec: 80.23
latency metrics:
Min_Latency(ms): 0
Avg_Latency(ms): 5
Max_Latency(ms): 202
histogram:
ms: 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24+
%: 1 1 1 1 5 30 46 14 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

C:\SQLIO>sqlio -kr -t2 -s30 -dc -o64 -fsequential -b8 -BH -LS Testfile.dat
sqlio v1.5.SG
using system counter for latency timings, 2208066 counts per second
2 threads reading for 30 secs from file C:\Testfile.dat
    using 8KB sequential IOS
    enabling multiple I/Os per thread with 64 outstanding
    buffering set to use hardware disk cache (but not file cache)
using current size: 12228 MB for file: C:\Testfile.dat
initialization done
CUMULATIVE DATA:
throughput metrics:
IOS/sec: 11360.01
MBs/sec: 88.75
latency metrics:
Min_Latency(ms): 0
Avg_Latency(ms): 10
Max_Latency(ms): 115
histogram:
ms: 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24+
%: 0 1 1 1 1 1 1 1 1 1 11 51 25 3 1 0 0 0 0 0 0 0 0 0 0
```

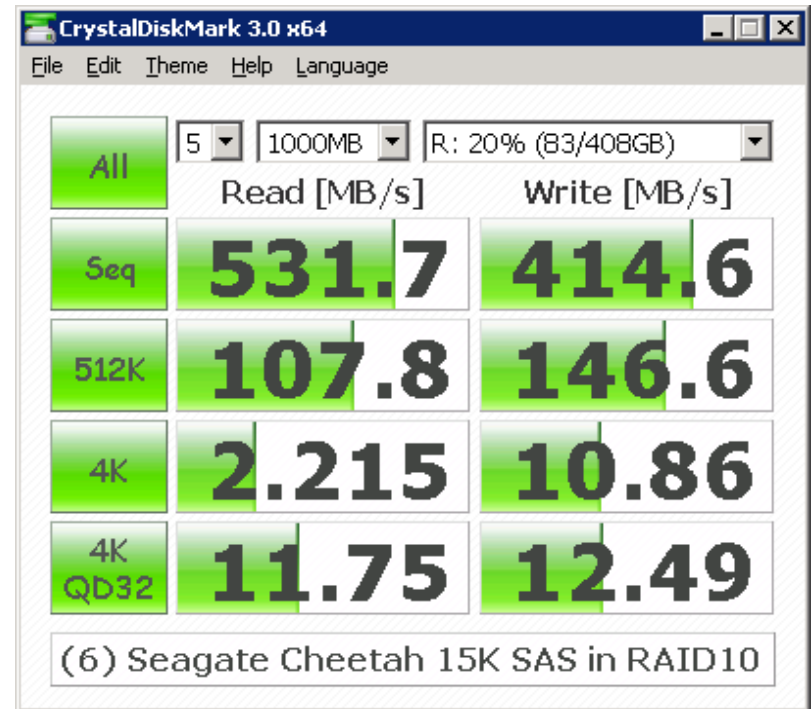
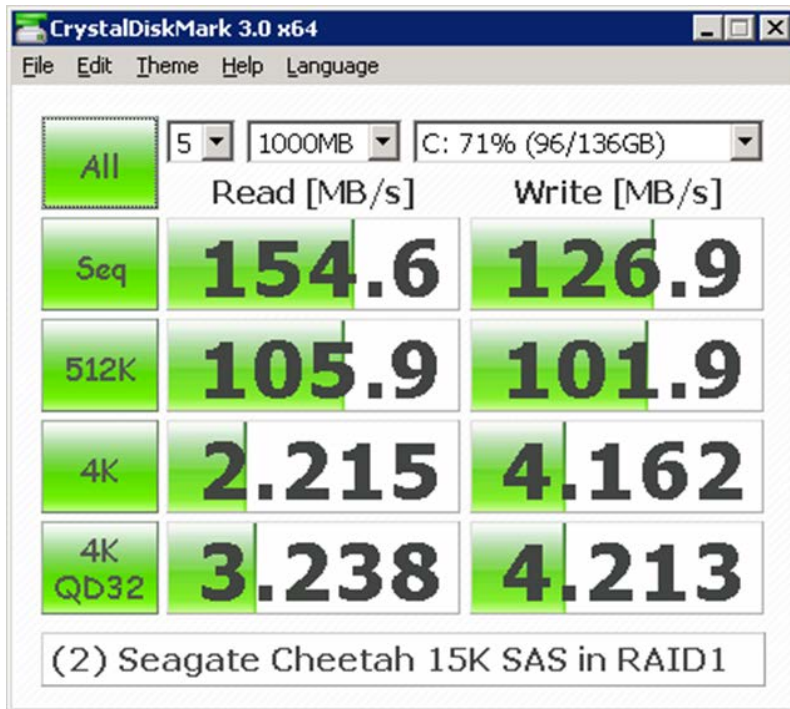
Magnetic Storage vs. Flash-Based Storage

- **Magnetic storage has fair sequential performance**
 - 100-200 MB/sec per disk
- **Magnetic storage has poor random I/O performance**
 - 100-200 IOPs per disk
- **Flash-based storage has good sequential performance**
 - 6Gbps SAS/SATA can do about 550 MB/sec per disk
 - 3Gbps SAS/SATA can do about 275 MB/sec per disk
 - PCI-E storage cards are sometimes only limited by PCI-E slot bandwidth
- **Flash-based storage has excellent random I/O performance**
 - 6Gbps SAS/SATA can do about 100,000 IOPs
 - PCI-E cards are capable of even higher IOPs

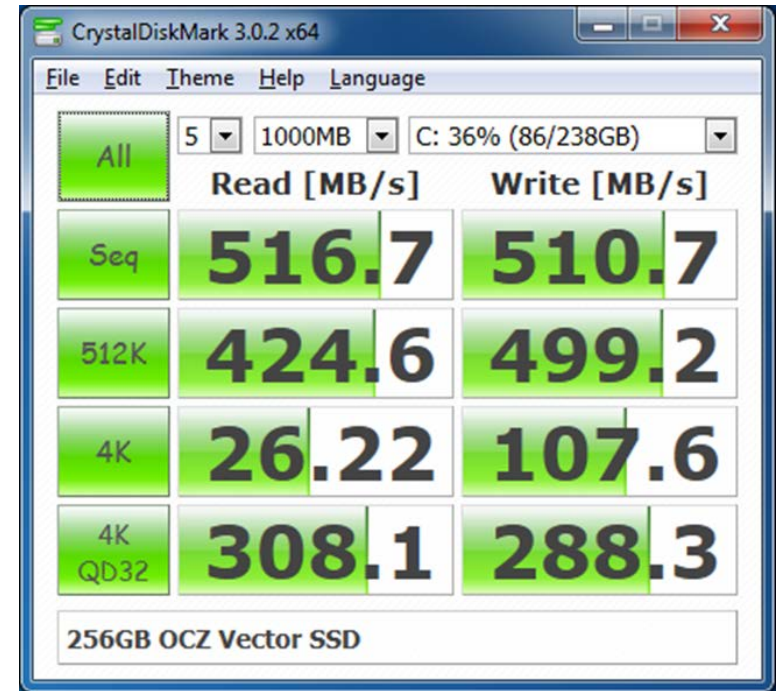
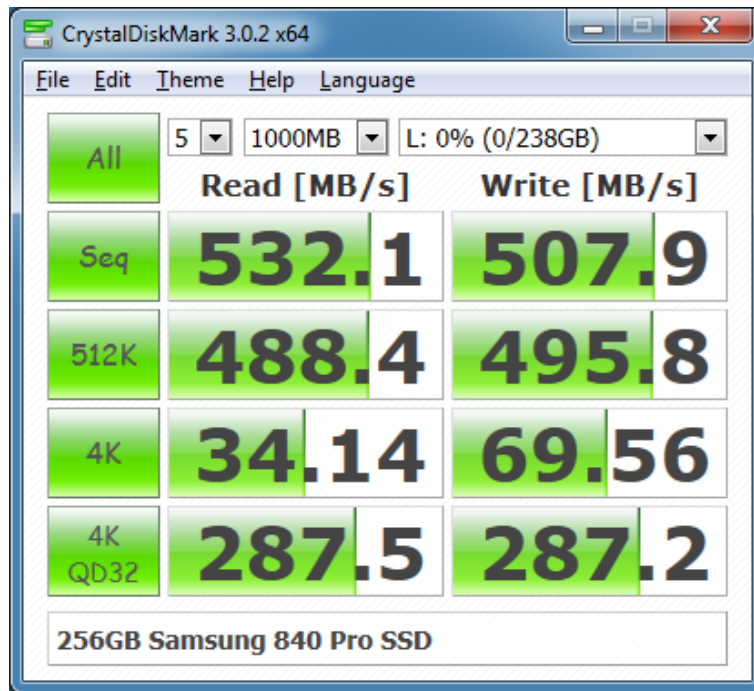
PCI-E Numbers to Know

- **PCI-E 1.0 Bus (one-way)**
 - x4 slot: 750MB/sec
 - x8 slot: 1.5GB/sec
 - x16 slot: 3.0GB/sec
- **PCI-E 2.0 Bus (one-way)**
 - x4 slot: 1.5-1.8GB/sec
 - x8 slot: 3.0-3.6GB/sec
- **PCI-E 3.0 Bus (one-way)**
 - x4 slot: 3.0-3.6GB/sec
 - x8 slot: 6.0-7.0GB/sec
- **So far, only Intel Xeon E5 family has PCI-E 3.0 support**

Traditional Magnetic Drive Performance



Consumer Flash Drive Performance



Storage Sizing Techniques for Performance

- **Do not simply size storage system for disk space requirements**
 - This is a very common mistake that leads to poor performance
- **Use a larger number of smaller disks to meet space requirements**
 - More disks give you more sequential and random I/O performance
 - Limiting factor for sequential performance can be PCI-E slot bandwidth
 - Purposely over-provision disk space if possible (short-stroking)
- **Use a relatively small number of high capacity flash-based disks to replace a large number of traditional magnetic disks**
 - This can save on initial capital costs
 - This can also reduce your electrical power usage

Summary

- **Consider your workload type as you evaluate storage types**
- **Five main types of storage**
 - Internal drives, PCI-E flash-based storage, direct attached storage
 - Storage area network, server message block file shares (WS2012)
- **RAID is commonly used to increase performance and reliability**
- **RAID level has effect on redundancy and disk performance**
- **Use disk benchmark programs to test logical disk performance**
 - CrystalDiskMark and SQLIO
- **Traditional magnetic drives are common and affordable**
- **Flash-based storage has much better random I/O performance**
- **Spindle count and individual drive performance is important**

What is Next?

- **Module 6 will cover hardware and storage sizing techniques**
 - Comparing an existing system to a new system using TPC-E
 - Comparing an existing system to a new system using Geekbench
 - Adjusting benchmark scores for different processors
 - Deciding how much physical memory to buy
 - Comparing storage subsystems using disk benchmarks