SQLskills Immersion Event
IE0: Accidental/Junior DBA

Module 1: Hardware Considerations

Jonathan Kehayias
Jonathan@SQLskills.com

Overview

- The importance of processor selection
- Intel or AMD processors?
- How to evaluate and compare processors
- Determining how much physical memory to buy
- Planning SQL Server I/O subsystems
- SAN considerations
- Virtualization considerations

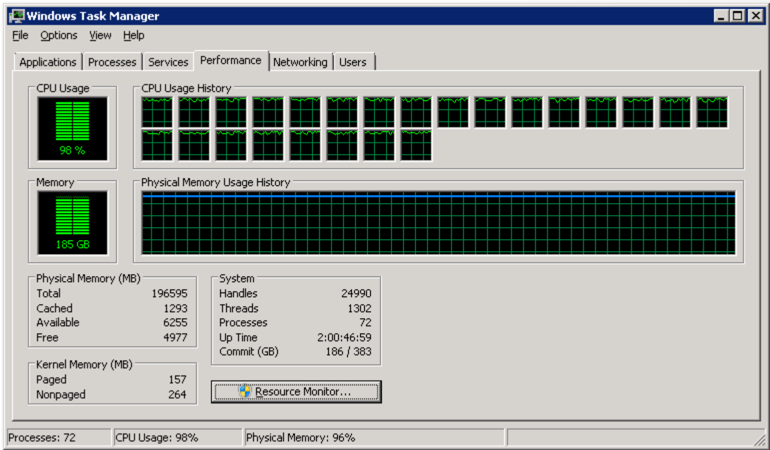
Why Does Database Hardware Matter?

- **Database servers are mission critical assets**
  - Performance and scalability problems are immediately noticeable
  - Multiple applications typically depend on the database server
- **Very difficult to compensate for poor hardware choices**
  - Inadequate I/O performance and capacity can cripple the system
  - Insufficient memory capacity can cause extra I/O pressure
  - Insufficient CPU capacity hurts performance and scalability
- **Wise hardware selection can save money on SQL license costs**
  - Physical core counts are cost driver for SQL Server 2012+ core licensing
  - New two-socket servers can often replace older four-socket servers
  - New four-socket servers can often replace older eight-socket servers
  - It is possible to save so much on license costs that your hardware is free!

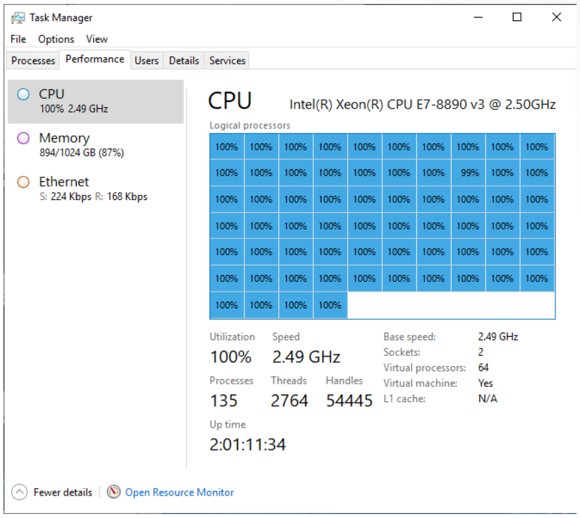
Top Scalability Issues with SQL Server

- **Poor application architecture and design**
- **Poor database architecture and design**
- **Poor indexing strategy and maintenance**
- **Improper instance configuration settings**
- **Improper database configuration settings**
- **Inadequate storage subsystem**
- **Old or inappropriate hardware**
  - Watch Glenn's Pluralsight course: Scaling SQL Server 2012, Part 1
    - http://pluralsight.com/training/Courses/TableOfContents/scaling-sqlserver2012-part1
- **Key Point: Hardware cannot solve all problems!**

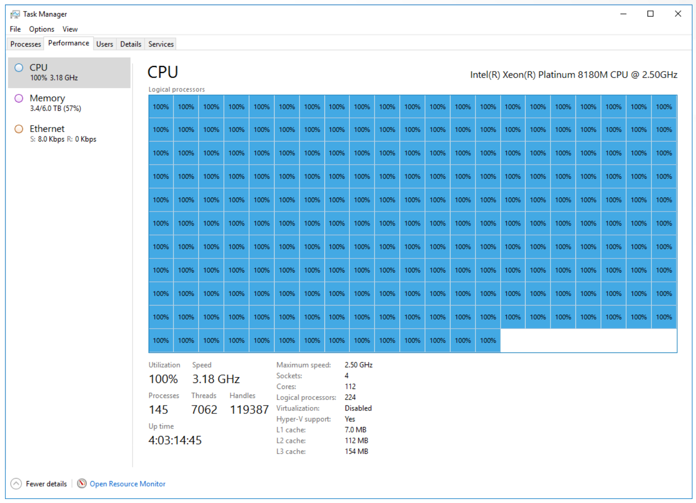
Is This Your Nightmare…



Or Is This Your Nightmare…

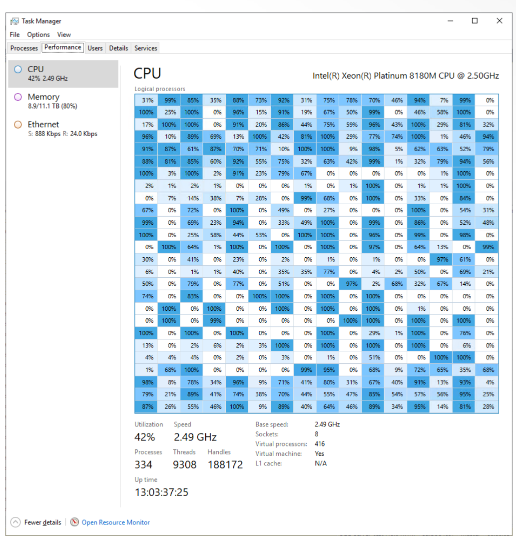


Or Is This Your Nightmare…



SQLskills immerse yourself in sql server

This Is Our Nightmare…



SQLskills immerse yourself in sql server

SQLskills immerse yourself in sql server

The Costs of Scaling Up

- **Scaling up is easy from an engineering perspective**
  - Very little development or testing effort is required
- **Scaling up is expensive from a capital cost perspective**
  - Hardware costs increase exponentially as you add sockets
  - SQL Server 2012+ core-license costs can add up very quickly
- **Higher license limits with Windows 2019 and SQL Server 2019**
  - Unlimited logical processors and 24TB of RAM (1024 cores on Windows 2016)
  - You can currently get 6TB of RAM in an four-socket machine with 64GB DIMMs
    - Windows Server 2012 R2 is still limited to 4TB of RAM
    - Windows Server 2016+ supports 24TB of RAM
- **Even NUMA architecture hardware does not scale linearly**
  - Typically a 70-80% scaling factor as you double the number of sockets

SQL Server and Hardware Selection

- **Nobody will ever complain that a database server is too fast!**
  - Don't be needlessly frugal with your database server hardware
    - You will be blamed for performance whether you selected the hardware or not
  - Server hardware is very affordable compared to SQL Server licenses
  - Don't ever reuse "old" database hardware for a new version of SQL Server
- **Be aware of SQL Server hardware licensing limits**
  - There are differences between SQL Server 2008 R2, 2012, 2014, and 2016+
    - 64GB RAM limit for SQL Server 2008 R2 and 2012 Standard Edition
    - 128GB RAM limit for SQL Server 2014 Standard Edition
    - Four-socket or 16 core limit for SQL Server 2008 R2 and 2012/2014 Standard Edition
    - Four-socket or 24 core limit for SQL Server 2016 Standard Edition with SP1+
    - Eight-socket limit for SQL Server 2008 R2 Enterprise Edition
    - SQL Server Enterprise Edition can use OS limit for RAM and processors

Sockets and Cores Explained

- **A processor socket holds a single physical processor**
  - One-socket, two-socket, and four-socket servers are the most common
  - SQL Server 2008 R2 licensing was based on processor sockets
- **A physical processor can have multiple physical cores**
  - Modern processors typically have four to eighteen physical cores
  - SQL Server 2012+ Enterprise licensing is based on physical cores
- **A physical processor core can be divided into two logical cores**
  - This is an Intel-only feature, called hyper-threading (HT)
  - Logical cores do count for virtualized SQL Server 2012+ instances
    - Unless you have enough physical core licenses for the entire host machine
    - And you have Microsoft Software Assurance (SA)

Database Server Specific Hardware Factors

- **Non-Uniform Memory Access (NUMA)**
  - Both AMD and current Intel systems support NUMA
    - Intel has supported NUMA since the Nehalem microarchitecture
  - Eliminates the old front-side bus contention issue of SMP architecture
  - This is more important as the number of sockets increase
- **Larger L2 and L3 processor cache sizes**
  - Cache size and performance is very important for database performance
  - Finding data in L2/L3 cache is much faster than finding it in main memory
  - Modern Intel processors have very large shared L3 caches
- **High single-threaded performance is very important for OLTP**
  - Most OLTP queries are short-duration, run on a single thread

© SQLskills, All rights reserved.
https://www.sqlskills.com

Intel Specific Considerations

- **Hyper-threading (HT)**
  - Physical cores divided into logical cores
    - Latest processors work with HT much better than the old NetBurst Pentium 4
    - A cache miss is not as expensive as it used to be
  - Works especially well with single-threaded OLTP workloads
    - Can give 20-30% additional capacity for concurrent queries
    - All Intel-based TPC-E submissions have had HT enabled
    - Test with your workload if you are unsure
- **Turbo Boost 2.0**
  - Boosts the speed of individual cores when other cores are idle
  - Very helpful for OLTP performance (up to 12% improvement)
  - Modern servers have ample cooling capacity
  - Always enable Turbo Boost

General Processor Considerations

- **Purposely over-provision processors (if you have the budget)**
  - Better single-threaded performance is very important for OLTP workloads
  - Higher core counts increase overall capacity and scalability
  - Excess CPU capacity is very useful for reducing I/O requirements
    - SQL Server data compression (Enterprise Edition only)
    - Backup compression (native or 3rd party)
    - Log stream compression for database mirroring and AlwaysOn AGs
- **Processors are relatively inexpensive**
  - Adding I/O capacity is usually much more expensive than a good CPU
  - The license costs per core are the same, so pick the right processor
  - Don't pick a lower speed processor from the same family to save money
  - Lower core count processors can save a lot of money on license costs

SQL Server 2019 Licensing Considerations

- **SQL Server 2019 Enterprise Edition is licensed per physical core**
  - You pay four-core minimum per physical socket
    - Retail cost is $6874.00 per physical core
    - This is another reason to retire old hardware with less than four cores
- **If SQL Server 2019 licensing costs are the biggest issue**
  - Consider "frequency-optimized" eight-core model CPU
    - Intel Xeon E5-2667 v3 – eight cores, 3.2GHz Base to 3.6GHz Turbo
  - Otherwise, get higher core count for more scalability
    - Intel Xeon E5-2699 v3 – eighteen cores, 2.3GHz Base to 3.6GHz Turbo
  - Consider processor architecture and cache sizes, not just clock speed
    - Physical vs. logical cores
    - Only physical cores matter for non-virtualized SQL Server core-based licensing

Why Use Two-Socket Servers?

- **Two-socket systems can handle a high percentage of workloads**
  - Newer, faster Intel processors compared to four-socket servers
    - Two-socket space has much higher sales volume than four-socket space
    - Two-socket Intel processor release cycle is 12-18 months ahead of four-socket
- **Lower RAM capacity, but up to 24 memory slots**
  - Two-socket servers support up to 1.5TB of RAM with 64GB RDIMMs
    - May go up to 3TB with LRDIMMs
    - Up to 768GB with 32GB DIMMs
    - Up to 384GB or RAM with more affordable 16GB DIMMs
- **Can have fewer PCI-E expansion slots than four-socket servers**
  - Typically four to six PCI-E slots in a modern two-socket server
  - This can limit maximum I/O capacity somewhat
  - Intel Sandy Bridge-EP and newer have PCI-E 3.0 support
    - This has double the bandwidth of PCI-E 2.0 standard in older servers

Why Use Four-Socket Servers?

- **Tradition has dictated four-socket database servers**
  - Four-socket servers have more physical and logical cores than two-socket
    - Higher total CPU capacity increases total server capacity
    - Lower single-threaded CPU performance because of older processor models
  - Higher total memory capacity
    - More memory slots and higher capacity memory controllers (in the CPU)
  - More PCI-E expansion slots (limited to PCI-E 2.0 prior to Xeon E7 v2 family)
    - Used for FC HBAs, DAS RAID controllers, NICs, PCI-E flash storage cards
    - Number and type of PCI-E slots sets upper limit for total I/O capacity
  - Four-socket servers have better reliability, availability and servicing (RAS) features
    - Memory error recovery in SQL Server 2012+ Enterprise with Xeon E7 family
    - Requires Windows 2012 Server or newer

(1) Four-Socket Server vs. (2) Two-Socket Servers

- **(1) Four-socket server - Dell PowerEdge R920 (4U)**
  - (4) Xeon E7-8891 v3 2.8GHz processors (80 logical cores with HT)
  - 3TB RAM (with 96 * 32GB DIMMs), 10 PCIe 3.0 slots
  - 4290 tpsE TPC-E score -- 53.62/logical core (estimated)
  - $84,823 hardware cost (no OS license or internal storage)
  - $274,960 SQL Server 2019 Enterprise Edition license cost (40 core licenses)
  - **$359,783 Total cost**
- **(2) Two-socket servers – Dell PowerEdge R730 (2U each)**
  - (4) Xeon E5-2667 v3 3.2GHz processors (64 total logical cores with HT)
  - 1.5TB total RAM (with 48 * 32GB DIMMs), 14 PCIe 3.0 slots (total)
  - 4664 total tpsE TPC-E score -- 72.87/logical core (estimated)
  - $36,394 hardware cost (no OS license or internal storage)
  - $219,968 SQL Server 2019 Enterprise Edition license cost (32 core licenses)
  - **$256,362 Total cost**

Intel or AMD Processors?

- **Intel is completely dominant in single-threaded performance**
  - Two-socket space since December 2008 (Xeon 5500 series)
  - Four-socket space since April 2010 (Xeon 7500 series)
- **AMD processors are more expensive to license**
  - Modern AMD processors have high physical core counts
    - SQL Server 2012 Core Factor table gives 25% license discount
    - Even with this discount, AMD licensing costs are much higher than Intel
- **AMD processors are less expensive (hardware cost)**
  - Processor cost is a very small component of total cost of the server
    - Not a compelling reason to pick an AMD processor
  - Opteron 6100/6200/6300 series works better for DW/DSS workloads
    - Opteron 6200/6300 series (Bulldozer/Piledriver) not good for OLTP

19

Comparing Performance with TPC-E

- **TPC-E OLTP benchmark available since 2007**
  - http://tpc.org/tpce/results/tpce_perf_results.asp
  - Much more realistic than old TPC-C OLTP benchmark
    - Less dependency on I/O subsystem performance, requires fault-tolerance
  - Only SQL Server systems have been submitted so far
    - 68 official submissions as of April 14, 2014
  - TPC-E benchmark is CPU-bound with adequate I/O capacity
    - Benchmark is a good indicator of CPU performance
    - Look at Executive Summary and Full Disclosure Report for details
  - TPC-E terminology translation
    - Processors = sockets
    - Cores = physical cores
    - Threads = logical cores

20

TPC-E Score Analysis (SQL Server 2014)

- **Divide actual tpsE score by number of processors**
  - Indicator of scalability
    - Intel Xeon E7-8890 v3: 11,058.99 tpsE divided by 8 processors = 1382.37/processor
    - Intel Xeon E7-8890 v3: 6964.75 tpsE divided by 4 processors = 1741.19/processor
    - Intel Xeon E5-2699 v3: 3772.08 tpsE divided by 2 processors = 1886.04/processor
- **Divide actual tpsE score by number of threads**
  - Indicator of single-threaded performance
    - Intel Xeon E7-8890 v3: 11,058.99 tpsE divided by 288 threads = 38.40/thread
    - Intel Xeon E7-8890 v3: 6964.75 tpsE divided by 144 threads = 48.37/thread
    - Intel Xeon E5-2699 v3: 3772.08 tpsE divided by 72 threads = 52.39/thread
- **This shows that lower socket count servers perform better**
  - Scalability still decreases as you add sockets, even with NUMA

TPC-E Score Analysis (SQL Server 2012)

- **Divide actual tpsE score by number of processors**
  - Indicator of scalability
    - Intel Xeon E7-8870: 5457.20 tpsE divided by 8 processors = 682.15/processor
    - Intel Xeon E7-4870: 3218.46 tpsE divided by 4 processors = 804.61/processor
    - Intel Xeon E5-2690: 1881.76 tpsE divided by 2 processors = 940.88/processor
    - Intel Xeon E5-2697 v2: 2590.93 tpsE divided by 2 processors = 1295.47/processor
- **Divide actual tpsE score by number of threads**
  - Indicator of single-threaded performance
    - Intel Xeon E7-8870: 5457.20 tpsE divided by 160 threads = 34.11/thread
    - Intel Xeon E7-4870: 3218.46 tpsE divided by 80 threads = 40.23/thread
    - Intel Xeon E5-2690: 1881.76 tpsE divided by 32 threads = 58.81/thread
    - Intel Xeon E5-2697 v2: 2590.93 tpsE divided by 48 threads = 53.98/thread

Geekbench CPU/Memory Benchmark

- **Geekbench is a CPU/memory benchmark**
  - Quick assessment of CPU/memory performance
  - No configuration required, takes a couple of minutes to run
- **Correlates reasonably well with TPC-E scores**
  - Can be used to "adjust" TPC-E score for slightly different CPU
- **Online database of Geekbench scores**
  - http://browse.geekbench.ca/
  - Search for similar system by server model number
  - Make sure to compare 32-bit scores to 32-bit scores!
  - Make sure to compare 64-bit scores to 64-bit scores!

General Memory Considerations

- **Maximize your physical RAM (within SQL Server license limits)**
  - Larger buffer pool cache reduces physical reads from disk subsystem
    - More data in the buffer pool (logical vs. physical reads)
  - RAM is faster and much less expensive than any disk subsystem
    - RAM is less expensive than enterprise-class storage
    - Orders of magnitude difference in latency
  - Can reduce the frequency of lazy writes and checkpoints
    - Helps even out the write workload to your data files
  - Physical memory is very important for In-Memory OLTP (SQL Server 2014+)
    - 256GB of physical memory in SQL Server 2014
    - OS maximum in SQL Server 2016+ Enterprise
    - 32GB of physical memory in SQL Server 2016 Standard with SP1+
  - Consider using Buffer Pool Extension (BPE) feature in Standard Edition
    - After you have the license limit for RAM in Standard Edition
    - BPE file can be 4X the max server memory size in Standard Edition

DDR4 PC4-21300 ECC Memory Prices

64GB module	$900.00	$14.06/GB
32GB module	$352.00	$11.00/GB
16GB module	$214.00	$13.38/GB
8GB module	$105.00	$13.13/GB

- Retail prices from Crucial.com (3/28/2017)

- **Just 3.5 years ago capacity/price sweet spot was 32GB modules!**

DDR4 PC4-23400 ECC Memory Prices

128GB LRDIMM	$1340.00	$10.47/GB
64GB LRDIMM	$397.00	$6.20/GB
64GB RDIMM	$369.00	$5.76/GB
32GB RDIMM	$189.00	$5.90/GB
16GB RDIMM	$108.00	$6.75/GB
8GB RDIMM	$84.00	$10.50/GB

- Retail prices from CDW.com (9/8/2020)

- **Current capacity/price sweet spot is 64GB modules!**

SQL Server I/O Planning

- **Poor I/O performance is one of the most common bottlenecks we see during our server health check consulting engagements**
- **Don't just plan for capacity requirements - sizing storage for SQL Server should be done in terms of I/O operations per second (IOPS), average transfer size, maximum latency expectations, and high availability/redundancy requirements**
- **Good performance using traditional rotating spindles generally means:**
  - Using the fastest rotational speed possible (15K RPM disks)
  - Using smaller capacity drives in higher numbers
  - Overprovisioning physical capacity and only using a portion of the total storage
- **SSDs may meet performance requirements in a single drive, but won't have redundancy**

SQLskills immerse yourself in sql server

Physical Layout Considerations

- **Degree of separation depends on I/O workload and I/O subsystem**
  - E.g. it may be fine to have several databases on one volume
  - E.g. SAN can do a better job of spreading I/O costs than a single drive
- **Consider separating:**
  - Log files from data files
  - Different databases on different volumes
  - SQL Server files separate from other uses (e.g. OS files)
- **Decide on file/filegroup layout**
  - tempdb
  - Multiple data files for increased performance?
  - Multiple filegroups for partitioned maintenance?
  - Multiple filegroups to allow partial database availability?
  - Usually don't put all logs on one LUN and all data on another LUN
- **WP: Physical Database Storage Design**
  - http://www.microsoft.com/technet/prodtechnol/sql/2005/physdbstor.mspx

SQLskills immerse yourself in sql server

Data Reading

- **Reads can be:**
  - Single/multiple pages from a data file
  - Single/multiple extents from a data file
  - Variable size chunks of FILESTREAM files
  - Usually random, except for large scans and backups
- **Misconception that SQL Server always reads extents**
  - But it will do sometimes to 'ramp up' the buffer pool (http://bit.ly/KV4Dbv)
- **Reads can come from the buffer pool or backups**
- **Read performance can be dramatically affected by:**
  - Number of files and file placement
  - Incorrect I/O subsystem configuration
  - Buffer pool memory and memory pressure
  - Ability to perform efficient read-ahead on indexes

Data Writing

- **Data file writes can be:**
  - Single/multiple pages
  - Single/multiple extents (for bulk operations)
- **Data file pages are written when:**
  - A checkpoint occurs (for whatever reason)
  - The lazywriter forces a dirty page from the buffer pool
  - A bulk operation flush occurs (a.k.a 'eager writes')
  - A database mirror is processing log records
    - Dirty pages are continuously flushed out, leading to heavy I/O load
    - Does not happen for Availability Groups
- **Write performance can be dramatically affected by:**
  - Number of files and file placement
  - Incorrect I/O sub-system configuration

Transaction Log Writes

- **Writes are always sequential**
- **No performance gain from having multiple log files**
  - SQL Server ALWAYS perform sequential writes of log records
- **There are specific limits on transaction log I/Os**
- **Limit of outstanding I/Os**
  - 64-bit: 32 outstanding I/Os (up to 2008 R2), 112 (2012 onwards)
  - 32-bit: 8 outstanding I/Os
- **Amount of outstanding I/O**
  - SQL Server 2008 onwards: limit of 3,840KB at any given time
  - Prior to SQL Server 2008: limit of 480KB at any given time

Transaction Log Reads

- **The following can cause sequential transaction log reads:**
  - Backups
  - Transactional replication or Change Data Capture
  - Database Mirroring or Availability Groups
  - A checkpoint in the SIMPLE recovery mode
- **The following can cause random transaction log reads:**
  - Transaction rollbacks
  - Crash recovery
  - Creating a database snapshot
  - Running DBCC CHECKDB
  - Manually looking in the transaction log with fn_dblog

Latency Matters!

- **How long it takes an operation to finish measured in milliseconds**
- **Monitoring**
  - Physical Disk: Avg. Disk Sec/Read
  - Physical Disk: Avg. Disk Sec/Write
- **General SQL Server guidelines**
  - < 8ms: excellent
  - < 12ms: good
  - < 20ms: fair
  - > 20ms: poor
- **Reality is "It Depends"**
  - Should be SLA-based to meet business requirements
  - Log writes should be as fast as possible (i.e. minimal latency)

Traditional HDD
Access Time

- **The response time or amount of time it takes for the drive to begin to transfer data**
- **Response time = seek time + rotational latency**
  - Seek time: the time it takes the read/write head to move from the current track to another track on the hard drive.
  - Rotational latency: the time it takes to rotate the disk platter to the correct position for access by the read/write head

HDD Spindle [RPM]	Average rotational latency [ms]
7,200	4.17
10,000	3.00
15,000	2.00

SSDs
Access Time

- SSD access time does not depend on mechanical moving parts, but electrical connections to solid state memory, so the access time is very fast and consistent across cells
- General average seek time ranges between 0.08ms and 0.16ms

SSDs NAND Types
SLC vs. MLC vs. TLC

- **SLC: Single-Level Cell flash stores a single bit value with two states**
  - Lower voltages results in longer endurance of 100,000 write cycles
- **MLC: Multi-Level Cell flash stores a double bit value with four states**
  - Higher data density per cell at the trade off of higher voltages with reduce cell endurance of 10,000 write cycles
- **TLC: Triple-Level Cell flash stores a triple bit value with eight states**
  - Higher data density per cell with reduced cell endurance of 5,000 write cycles

SSD Concepts

- **Write Amplification**
  - Garbage Collection – data is written in pages but only erased in blocks
    - Used pages within a block are read and rewritten to another block to allow reclamation of the stale pages by erasing the current block
- **Overprovisioning**
  - Reserves a permanent or temporary portion of the SSD capacity as working space for the controller
    - Improves performance and endurance of the NAND cells
- **Wear Leveling**
  - Distributes writes as evenly as possible across the cells to increase endurance and reduce a single cell from wearing out under heavy writes
  - May increase write amplification as static data is moved from a underwritten cell to a heavily written cells

RAID

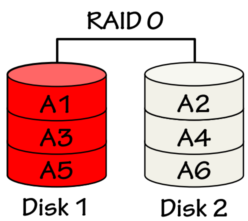
Redundant Array of Independent/Inexpensive Disks

- **Combines multiple disk drive components into a single logical unit, distributing data across the disks for redundancy and improved performance**
- **Configurations or levels are named by the word RAID and a numeric value specific to the configuration for standardization**
- **Common RAID levels**
  - RAID 0 - Striping
  - RAID 1 – Mirroring
  - RAID 5 – Striping with parity
  - RAID 6 – Striping with double-parity

RAID Levels: RAID 0

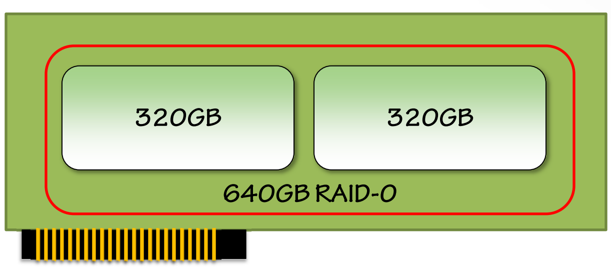
Striping

- **Usage scenarios:**
  - Data you don't care about
  - Performance matters more than data loss risk
- **Pros:**
  - Fast read and write performance
- **Cons:**
  - Total loss of data from single disk failure



PCI-Express SSDs

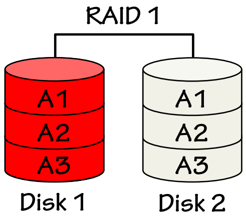
- **Some PCI-Express SSDs like the FusionIO ioDrive Duo 640 present two physical disks to Windows**



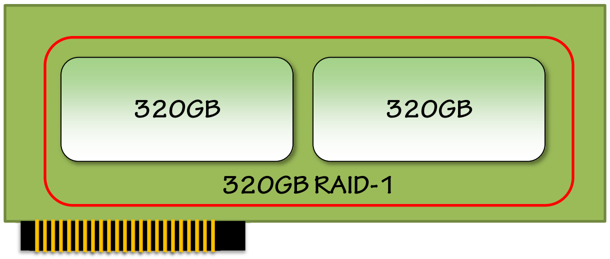
- **To get full capacity from the card requires Windows RAID 0**

RAID Levels: RAID 1
Mirroring

- **Usage scenarios:**
  - Important data that requires redundancy to protect from disk failures
- **Pros:**
  - Mirrored storage protects from single disk loss
- **Cons:**
  - Only provides half the storage
  - Write performance equal to one disk

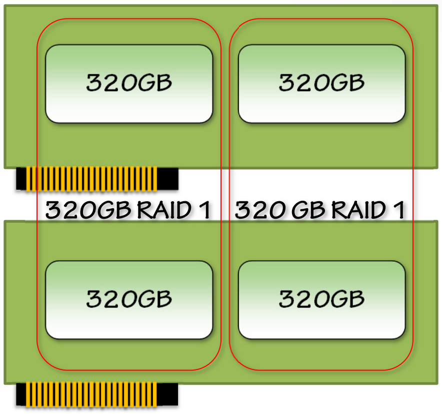


PCI-Express SSDs and RAID 1



- **RAID 1 using a single card still does not provide redundancy against a controller failure on the card**

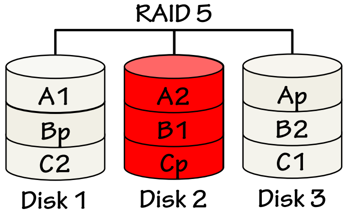
PCI-Express SSDs



- **640GB capacity with full redundancy requires two cards and RAID 1 across the cards**

RAID Levels: RAID 5
Striping with Parity

- **Usage scenarios:**
  - Where reads exceed writes, data files
- **Pros:**
  - Maximize available capacity from minimal number of disks with redundancy
- **Cons:**
  - Protection from single disk failure only as loss of two disks results in total data loss
  - Write penalty for parity calculation
  - Significant performance impact occurs when degraded



Nested RAID Levels

- **Implement one level of RAID on top of another level of RAID**
- **Can be used to create better redundancy as well as higher levels of performance**
- **Common nested RAID levels are**
  - 10 or 1+0: striping over mirrored pairs
  - 01 or 0+1: mirroring over striping
  - 50 or 5+0: striping over single-parity striping
  - 60 or 6+0: striping of double-parity striping
  - 100 or 10+0 or 1+0+0: striping over a striped set of mirrored pairs

SQLskills immerse yourself in sql server

45

Nested RAID Levels: RAID 10 or 1+0
Striping Over Mirrored Pairs

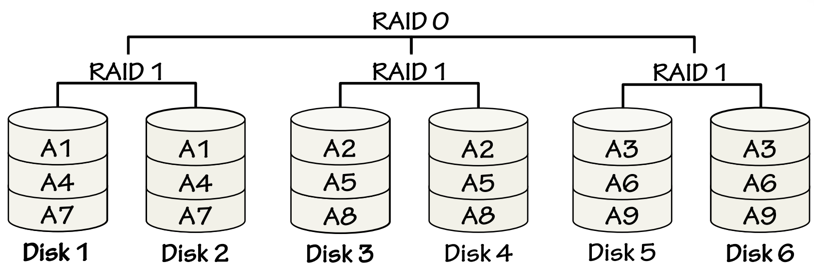
- **Usage scenarios:**
  - Transaction log files, heavy write data files
  - Redundancy is more important than cost
- **Pros:**
  - Fast read and write performance
  - Supports multiple disk failures as long as two of the failed disks aren't in the same RAID 1 pair
- **Cons:**
  - Doubles the cost of storage

SQLskills immerse yourself in sql server

46

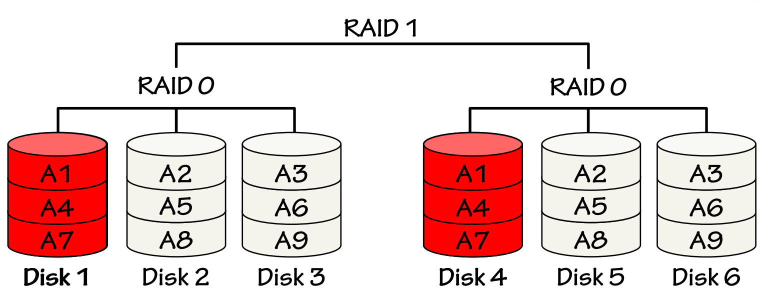
Nested RAID Levels: RAID 10 or 1+0
Striping Over Mirrored Pairs

- Single disk loss keeps array available
- Multiple disk losses in different RAID 1 keeps array available
- Double disk loss in same RAID 1 results in array failure



Nested RAID Levels: RAID 01 or 0+1
Mirroring Over Striping

- IS NOT THE SAME AS RAID 10/1+0
- Single disk loss keeps array available
- Double disk loss in different RAID 0 results in array failure



Other RAID Configurations

- **RAID 6**
  - Similar to RAID 5 but provides double parity disks allowing for up to two disks to fail and the array remains available
  - Common recommendation for some SAN vendors
- **RAID 50, RAID 100, etc**
  - Specialized RAID configurations that stripe other RAID levels to increase performance

Cloud Storage (Azure and AWS)

- **Cloud infrastructure performance is based entirely on the configuration appropriately being sized to requirements**
- **Maximum IOPS limits per VM and per disk volume are independent of each other**
  - AWS EC2 instance type maximum I/O throughput limits (https://sqlskills.com/help/aws-ebs)
  - Azure VM size and type IOPS limits (https://sqlskills.com/help/azvmio)
- **Achieving appropriate storage throughput usually requires striping multiple volumes together using Windows Storage Spaces (not S2D)**
  - Must use PowerShell for configuration to set column count (# disks to stripe data across) – THIS IS INCREDIBLY IMPORTANT!!! Column count cannot be changed after VDISK creation.

Using SAN Storage

- **Storage Area Networks (SANs) virtualize and consolidate storage in the data center**
  - Network multiple storage devices together to provide block level access to servers, which share the underlying configuration
- **There are many SAN vendors and models on the market, each with different configuration options and limitations**
  - Hybrid arrays may mix HDDs and SDDs together into storage pools for tiered performance optimizations internally
  - Storage pools may only allow RAID 5/6 and prevent isolating workloads
- **Front access ports connect servers to the storage**
  - Port speed matters – 1Gb/sec iSCSI is usually a bottleneck for SQL Server
  - Ports are shared across all connected servers and must be monitored for total workload throughput across the enterprise, not just SQL Server in shared configurations

51

SAN Advantages

- **Shared storage**
  - Increases disk utilization
  - Reduces management by making it easier to create new volumes and dynamically allocate storage
  - Create diskless servers that boot from SAN only
- **Advanced features**
  - Mirroring, snapshots, continuous data protection, clustering and geo-clustering only offered by SANs
- **Performance**
  - Almost unlimited number of spindles, controllers, and cache can be put together to meet the requirements depending on the SAN model and vendor

52

SAN Disadvantages

- **Unpredictable performance**
  - When you share your disks, controllers, and fiber switches between a dozen of servers is very difficult to have predictable performance
- **Higher latency**
  - Distance the I/Os have to travel; added layers of switches, cabling and ports
  - PCI Bus → HBA → FC switches → FC ports → array processors → disks
- **Limited bandwidth**
  - FC 16Gb/s max,
  - iSCSI 10Gb/s max
  - Note: 1Gb/s = 128MB/s
- **Cost**

Virtualization Considerations

- **Changes in hypervisor limitations in recent years have changed the landscape for virtualizing SQL Server**
- **Limitations on vCPUs expanded**
  - 64 vCPUs on Hyper-V 2012 and ESX 5.1, 128 vCPUs on ESX 6.0
- **Memory limitations expanded**
  - 1TB on Hyper-V 2012 and ESX 5.1, 4TB on ESX 6.0
- **Introduce the ability to pass vNUMA configuration through to VM hardware configuration**
  - Affects SQLOS configuration for memory nodes, lazywriters, and connection distribution for higher scalability
- **Hyper-V 2012 Virtual Fiber Channel allows N_Port ID Virtualization (NPIV) to share a single physical interface with multiple virtual HBAs for direct SAN access inside a VM**

Virtualization Hardware Considerations

- **The underlying hardware still matters when virtualizing SQL Server**
  - Common VM host configurations are 4 socket systems that are planned around higher core counts and not single threaded performance
    - AMD Magny-Cours, Bulldozer, and Piledriver processors are commonly selected
  - Higher core counts allow for higher consolidation densities for VMs on a single host
  - Common for older hardware platforms to be used longer as VM hosts
- **Getting the best performance from SQL Server virtualized may require dedicated hardware for SQL VMs**
  - Higher vCPU count reduces the consolidation density due to co-scheduling
  - Mixing wide VMs with smaller VMs is generally problematic
  - Overcommitting resources should generally be avoided for SQL Server workloads

SQL VM CPU Allocation
Considerations

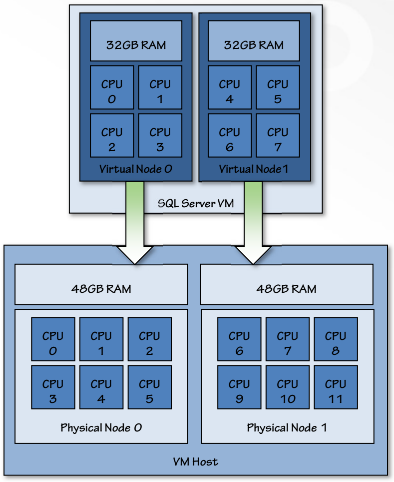
- **How many virtual cores do we actually need?**
  - Consider co-scheduling issues under the hypervisor?
- **How many idle cores are available on the host VM?**
  - Co-scheduling on idle cores will be at full speed until another VM begins to execute and begins to take slices on the CPUs making those SQL threads slower to execute
- **How fast is each physical CPU core?**
- **Is hyper-threading being used on the physical server?**
- **How much of each physical core does the VM get % wise for execution?**
- **Did the last execution run on the same physical cores?**

Virtual NUMA (vNUMA) Configuration

- Prior to ESX 5, if the number of vCPUs exceeded the physical cores in a single NUMA node, the VM would not be NUMA optimized by the hypervisor
- vNUMA in ESX 5+ and Windows Server 2012 Hyper-V allows larger VMs to be partitioned into virtual NUMA nodes to allow NUMA optimization of the VM in the hypervisor
- ESX 5+ only enables vNUMA when a VM has more than 8vCPUs or when the VM is wider than the number of cores in a physical node
  - Can be enabled manually by adding the *numa.vcpu.min* configuration parameter to VM configuration:
    - VM Settings->Options tab->Advanced General->Configuration Parameters
- vNUMA configuration in Windows Server 2012 Hyper-V is accomplished on the NUMA page for the VM settings

vNUMA Optimized Configuration

- By configuring vNUMA for the wide VM, the vCPUs and memory allocations can be localized to the physical node configuration providing NUMA locality for scheduling and memory management to the VM
- For SQL Server this also creates a separate LazyWriter thread for buffer pool management under SQLOS



Key Takeaways

- Don't run newer versions of SQL Server on older hardware, the cost of the hardware can easily be offset by the reduction in core-based licensing costs when upgrading to SQL Server 2012+
- Newer two-socket configurations provide faster single-threaded performance and can handle most workloads with better performance than four-socket configurations
- Select Intel processors optimized for single-threaded performance over higher core counts for OLTP workloads
- Over-provision memory for SQL Server Standard Edition to maximize buffer pool usage and account for memory overheads in Windows
- Consider high-availability requirements for storage when using PCI-Express SSDs, using multiple cards with RAID 1

Review

- Don't reuse old hardware for a new database server
- Consider your workload type as you select hardware
- Size your hardware to minimize SQL Server license costs
- Choose the most appropriate Intel processors
- Get as much RAM as possible
- Don't neglect the I/O subsystem
- Plan virtualization based on SQL Server requirements

Hardware References

- Geekbench
  - http://bit.ly/UGrGbu
- TPC-E OLTP Benchmark
  - http://bit.ly/UGs2Pm
- CPU-Z Tool
  - http://bit.ly/korH23
- Intel Ark Database
  - http://ark.intel.com/
- AnandTech IT
  - http://bit.ly/UGwiyg
- StorageReview
  - http://bit.ly/kOHUL7

61

Questions?