SearchBusinessAnalytics

TechTarget

E-Guide

Optimizing SQL Server Hardware and performance

SQL Server and virtualization are made for each *other, and the situation is getting better all the time. Not just for workload management and consolidation either, but also for high availability. However, to make it all happen you're going to need hardware, and buying hardware for a virtualization host that will be running SQL Server is a bit different than selecting hardware for SQL Server itself. In this E-Guide, readers will learn tips on choosing the right hardware and using it to optimize performance.*

Contents

SQL Server virtualization is inevitable: Get the right hardware

By: Don Jones, Contributor

Let's face it: If you haven't already virtualized SQL Server instances in your environment, you're going to do so eventually, if not "real soon now."

SQL Server and virtualization are made for each other, and the situation is getting better all the time. Not just for workload management and consolidation either, but also for high availability. A new breed of technologies is out there now that can provide multiprocessor power to SQL Server on multiple hosts, keeping virtual machine instances in lockstep with one another and enabling zero-nanosecond failover in the event that one instance goes down.

But to make it all happen you're going to need hardware, and buying hardware for a virtualization host that will be running SQL Server is a bit different than selecting hardware for SQL Server itself. You also have to plan your SQL Server instances. Busy instances that handle big databases might go onto virtual machines (VMs) all their own, while smaller instances might be teamed up within a single VM.

Remember that a VM becomes your basic unit of management: You can move VMs to different hosts, fail them over and so forth -- but every instance

Contents

within each VM goes along for the ride. Focus on creating VMs that need as few virtual processors as possible to do their job; that will make each VM more granular in terms of the workload it handles, and it will make it easier for those VMs to co-exist with other VMs on the same host.

When outfitting that host, there are three things to consider: disk throughput, memory and processors. Your money is best spent, initially, on processors. Ignore blade servers and compact 1U servers for SQL Server hosts: You'll squeeze more processor sockets and cores into a 4U chassis, and that chassis will often run with lower cooling and power requirements than a similar 1U or 2U chassis.

Find the "sweet spot" for processor speed -- where you're getting the best performance for your dollar -- don't just buy the fastest. A few extra megahertz aren't going to deliver a vast performance improvement. Do focus on server-class processors, though. If you're the kind of person who believes he can build a server from off-the-shelf Centrino-based motherboards, abandon that theory when it comes to SQL Server virtualization hosts, please.

Memory is the next expense. The more, the merrier. Modern hypervisors typically let you *overcommit* memory, meaning you can configure your VMs to use more memory, in total, than the host actually has. Many environments do well with a 50% overcommit, but SQL Server is a real memory hog. Analyze SQL Server instances to see how much memory they're typically consuming, plan your overcommit accordingly, and don't put VMs on the same host if they're all running SQL Server instances, which tend to max out their memory allocation.

Bear in mind that SQL Server, more than many other server applications, will try to use whatever memory the operating system is willing to give it—so if Windows *thinks* it has 12 GB of memory, SQL Server will often make its best effort to utilize that. That behavior can make overcommit tricky, so proceed with caution. In fact, most experienced database administrators don't like to use memory overcommit at all when they're virtualizing SQL Server.

Contents

That said, the amount of memory is the one thing you can skimp on when buying a server. That's because you can add more later -- provided you put the largest memory modules possible in your server, leaving free slots for future expansion. Don't cheap out on the memory you *do* buy, however. Get error-correcting memory that's speed-matched to the server's motherboard. In other words, buy whatever your chosen server vendor recommends for your server, and ideally buy the memory from that same vendor. After all, that vendor is most likely to offer you support if you have problems with it.

Disk is last, and in most SQL Server cases you'll be building a storage area network (SAN) rather than relying heavily on internal storage within the servers. (You might build a mirrored set of internal hard drives to run SQL Server and Windows themselves, not to store data.) In order of priority, build your SAN for fault tolerance, speed and size; if you think you need a 10-terabyte SAN, size is the last thing you price out.

First, make sure you can afford to make that storage redundant enough to survive the failure of a handful of actual disks, and you have to build it to be fast enough to support SQL Server. SQL Server's most common bottleneck is storage speed, so it's almost impossible to build a SAN that's "too fast." That's especially true with virtualization, which brings its own I/O overhead as data is written to virtual disk images.

While SQL Server is perfectly capable of being run in a virtual machine, buying hardware for SQL Server virtualization hosts and configuring the virtual machines requires a specialized approach. Simply moving your existing SQL Server instances into poorly configured virtual machines, or to poorly provisioned hosts, can significantly degrade performance. There's no need to take that risk: Keep these tips in mind and you'll have an efficient virtualization infrastructure that's SQL Server-ready.

ABOUT THE AUTHOR

Don Jones *is a senior partner and principal technologist at strategic consulting firm Concentrated Technology. Contact him through his website, www.ConcentratedTech.com.*

Sponsored by DELL | Microsoft

Optimize SQL Server hardware performance

By: Jeremy Kadlec, Contributor

Contents

Tuning SQL Server for performance is in many ways like constructing a house. You must find a suitable location with rich soil, build a solid foundation to support two or three levels, install the electrical and plumbing, finish the walls for painting and decorating, and finally conduct ongoing maintenance.

Comparable items to address when tuning SQL Server performance include:

- Hardware
- Windows Server
- SQL Server
- Database design
- Indexes design
- T-SQL development
- Network infrastructure
- Front-end code
- Platform maintenance

Although no single component is more important than any other in achieving high performance, it is imperative to start correctly. If you do not build a solid foundation, the remainder of the application will always be plagued by deficiencies in earlier components. Furthermore, although it is possible to rebuild the hardware once it has been deployed, or simply move to another hardware platform, with proper planning the overall platform can support the application in the long term avoiding a time-consuming upgrade.

Hardware planning

Proper hardware planning is the first step in achieving high performance. It is imperative to determine what hardware to use based on requirements that can be calculated by capacity planning for CPUs, memory, drives, network interface cards (NICs), etc. At times this decision is rather easy; you can use corporate standards to simplify the hardware support. If standards do not

Sponsored by DELL | Microsoft

exist, seize the opportunity and begin building standards to simplify the overall management.

Contents

While standardization is critical to supporting a large number of servers, it should be complimented by the latest technologies so you may continuously achieve higher performance at lower costs. Two current hardware technologies that should be considered in the short and long-term are 64-bit technologies and database accelerators.

64-bit architectures

As hardware vendors deliver 64-bit servers at reasonable costs, Windows Server 2003 and SQL Server 2000 are able to leverage additional resources to improve overall performance from a scale-up perspective. In the 64-bit world, 64 CPUs and 1 TB of RAM can be addressed directly, far exceeding the typical 32-bit world of 4 CPUs and 3 GB of memory – much more than double the resources.

Currently the 64-bit platform has some limitations in terms of tools and full support from all applications, but this too will change over time. It is important now to keep an eye on the 64-bit architectures as they mature and gain greater support from software vendors to improve scale-up options on some rather large servers.

Hardware-based accelerators

Database accelerators have been getting a fair amount of press lately for SQL Server. At a conceptual level, database accelerators are hardware devices with CPUs and memory. They store the entire database or a subset of tables in memory to process user transactions that write back to the SQL Server, which ultimately stores the data. The advantage is that these devices can sometimes support the entire database or core tables in dedicated memory, which outperforms accessing the data from disk. This can help from a scale up or scale out perspective because a single database accelerator can alleviate hardware bottlenecks without a change to the existing SQL Server.

One company that offers this type of solution is XPrime. Consider this option if no further software optimizations can be determined and uptime can't be jeopardized until an upgrade can be tested and completed.

Ideal hard drive layout

Database accelerators have been getting a fair amount of press lately for SQL Server. At a conceptual level, database accelerators are hardware devices with CPUs and memory. They store the entire database or a subset of tables in memory to process user transactions that write back to the SQL Server, which ultimately stores the data. The advantage is that these devices can sometimes support the entire database or core tables in dedicated memory, which outperforms accessing the data from disk. This can help from a scale up or scale out perspective because a single database accelerator can alleviate hardware bottlenecks without a change to the existing SQL Server.

One company that offers this type of solution is XPrime. Consider this option if no further software optimizations can be determined and uptime can't be jeopardized until an upgrade can be tested and completed.

Disk	Disks	Drive Size	Controller	RAID	Size	Volume	Purpose
0	2	72 GB	1 - 1	1	72	C:\	Windows, SQL Server Program Files and System Databases (Master, Model, MSDB) (Random)
1	2	72 GB	1 - 1	1	72	D:\	Tempdb (Random)
2	2	72 GB	1 - 2	1	72	E:\	Database Transaction Logs (Sequential)
3	5	72 GB	2 - 1	5	288	F:\	Databases (Random)
4	3	72 GB	2 - 2	5	144	G:\	Backups and Batch Processing (Sequential)
-	14	-	-	-	648	-	-

Sponsored by DELL | Microsoft

Contents

Local storage or a storage area network (SAN)

The big question for many organizations is how to support a high number of drives on a single server. On top of that, they must consider how to manage databases exceeding the designated amount of storage, or having to separate specific databases from one another on a single server.

There are three basic options: The first is to leverage servers with many internal disks to support storage. Second is a server with a few internal disks and a locally attached drive array. Third is a server with a few internal disks and a connection to a storage area network (SAN).

Management can be more difficult with locally attached storage because it is distributed, unlike a SAN, which has centralized storage where adding new storage to a server can be as simple as a point-and-click process. However, performance may be better for local storage because only a single server leverages the disk drive as compared to a SAN, where five to 10 servers may access the same drives and cause contention. The SAN's answer to this performance issue is a large amount of cache rather than access the data from disk.

Then there is the egg-and-basket metaphor: A single server with dedicated storage having an IO issue can only affect one server, whereas that same problem on a SAN can potentially affect many servers. This can be taken a step further when an update is needed to the SAN; all of the servers leveraging storage from the SAN need to be shutdown. When updating a single server with local storage, the downtime only affects the server requiring the update, not all servers. Manageability versus performance must be balanced.

Conclusion

It is no mystery that hardware is the foundation for SQL Server high performance, but the platform still requires the proper subsequent design and development to achieve the needed performance in the long term. While many hardware innovations are available, keep in mind that slacking off on the development phases is not an option. You must set up SQL Server

hardware properly and keep optimization best practices at hand throughout the application's lifecycle to achieve high performance. Good luck!

Contents

ABOUT THE AUTHOR

Jeremy Kadlec is the Principal Database Engineer at Edgewood Solutions, a technology services company delivering professional services and product solutions for Microsoft SQL Server. He has authored numerous articles and delivers frequent presentations at regional SQL Server Users Groups and nationally at SQL PASS. Jeremy is also the SearchSQLServer.com Performance Tuning expert.

Contents

Contents

Sponsored by DELL | Microsoft

Contents

Free resources for technology professionals

TechTarget publishes targeted technology media that address your need for information and resources for researching products, developing strategy and making cost-effective purchase decisions. Our network of technology-specific Web sites gives you access to industry experts, independent content and analysis and the Web's largest library of vendor-provided white papers, webcasts, podcasts, videos, virtual trade shows, research reports and more —drawing on the rich R&D resources of technology providers to address market trends, challenges and solutions. Our live events and virtual seminars give you access to vendor neutral, expert commentary and advice on the issues and challenges you face daily. Our social community IT Knowledge Exchange allows you to share real world information in real time with peers and experts.

What makes TechTarget unique?

TechTarget is squarely focused on the enterprise IT space. Our team of editors and network of industry experts provide the richest, most relevant content to IT professionals and management. We leverage the immediacy of the Web, the networking and face-to-face opportunities of events and virtual events, and the ability to interact with peers—all to create compelling and actionable information for enterprise IT professionals across all industries and markets.

Related TechTarget Websites

SearchCRM

SearchDataManagement

SearchContentManagement

SearchOracle

SearchSAP

SearchSQLServer