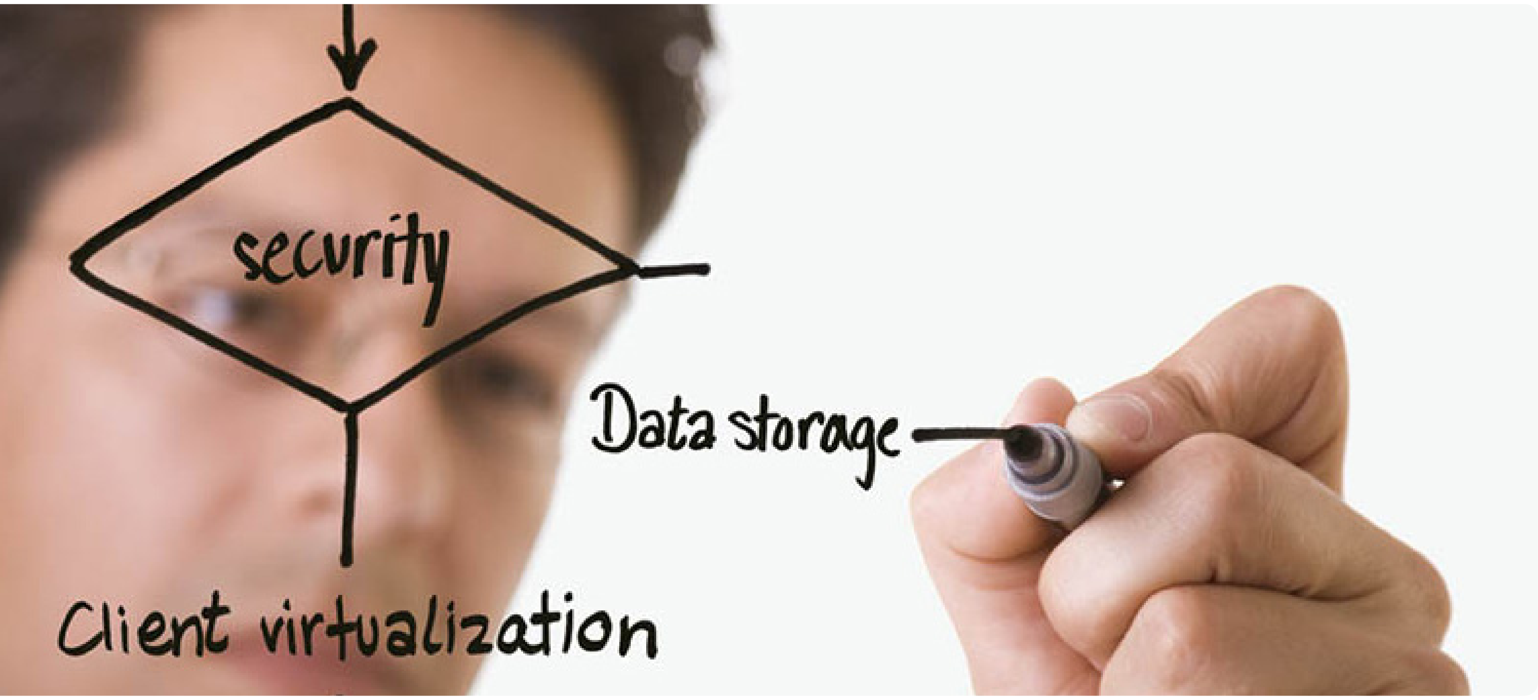
Software

Mythbusting Goes Virtual

Debunking four common vSphere “Truths”

Written by Scott D. Lowe



Introduction

The information being presented in this paper comes courtesy of the great minds of Eric Sloof, a VMware Certified Instructor, vExpert, consultant and active VMware community member; and Mattias Sundling, vExpert and Dell evangelist focused on the virtualization space. The information presented here was discussed in depth during an April 2, 2012 webcast with Mattias Sundling and Eric Sloof.

Regardless of the underlying technology solution, as anything becomes increasingly popular and widespread in use, certain pieces of sometimes inaccurate information about that product become permanent fact, often taking on legend-like status. Moreover, as a product matures, it changes; it evolves by taking on new features, shedding old ones and improving the functionality everywhere else. However, no matter how much a product matures and no matter how much it evolves, many products carry with them myths that follow through the ages. Myths that may or may not have once been true, but are used as truisms nonetheless even as the version count rises ever higher. In this white paper, we will expose four such myths about vSphere.

Myth #1: RDMs have better performance than VMFS

What is RDM?

A raw device mapping (RDM) is created when a vSphere administrator has configured a virtual machine’s virtual disk to point directly to, for example, a LUN (logical unit number) on a storage array. With an RDM in place, a virtual machine can access storage just like it’s any other disk.

RDMs operate as follows: The virtual machine’s initial access to an RDM virtual disk results in the virtual machine being pointed to a small mapping file. This mapping file is a symbolic link containing the raw ID of the intended storage on the storage array. Once it learns that raw ID, the virtual machine points directly to the raw ID on the storage array and no longer needs to make use of the mapping file, as illustrated in Figure 1.

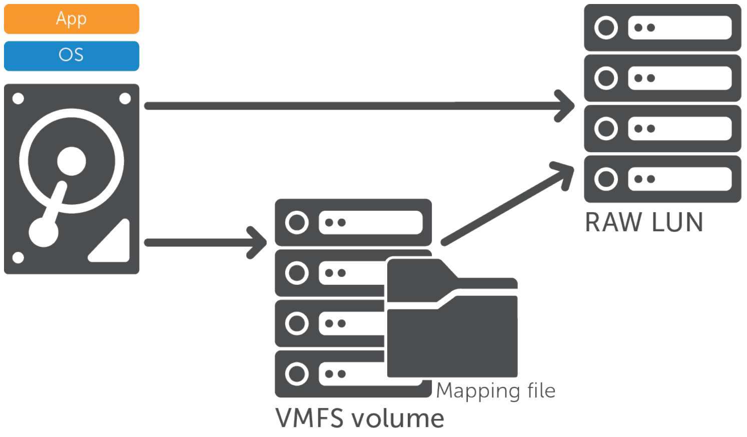


Figure 1. A VM initially accesses an RDM virtual disk using a mapping file, but subsequently uses the raw ID.

The source of the myth

Because the virtual machine is accessing storage directly and not going through some of the abstraction that takes place when the hypervisor is placed in the middle, there is a myth that RDMs have superior performance over virtual storage devices that make use of vSphere Virtual Machine File System (VMFS) datastores. Evidence of this myth abounds in forum articles and other resources outlining administrators' attempts to use RDMs to eke out as much performance as possible for storage-intensive workloads, such as those supporting databases.

RDMs have two modes: virtual and physical

When considering the use of RDMs, bear in mind that they come in two different flavors:

- **Virtual compatibility mode**—When an RDM is configured in virtual mode, it appears to the guest operating system just like a virtual disk does when it's housed inside a VMFS volume. With this mode, administrators are still able to enjoy the benefits that come with the use of VMFS, including advanced file locking and snapshots. Further, because virtual mode continues to provide a level of hardware abstraction, it is more portable across storage hardware than physical mode.
- **Physical compatibility mode**—When an RDM is in physical mode, the volume the characteristics of the mapped device, which provides the greatest flexibility in managing the volume using native SAN tools. However, physical RDMs lose some of the features found with virtual volumes, including the ability to be snapshotted, cloned, made into a template, or migrated if the migration involves copying the disk.

It's not unreasonable to make the assumption that "raw" would translate into increased performance of the virtual machine, but this myth has been well and truly busted and, in fact, RDMs operate with performance characteristics on par with VMFS storage. This is demonstrated as one starts to peer under the covers at what's happening with the host system and, in particular, how these storage calls are interacting with the hypervisor kernel. By monitoring the kernel, the entire story of how storage operates becomes clear. Through this monitoring, an administrator can watch the "hidden" story of storage activities and what impact these activities have on overall performance.

Testing the myth

To evaluate this myth, Eric performed tests using three distinct scenarios, two involving RDMs and one using VMFS as primary storage. The tests use a single virtual machine configured with a SCSI adapter, but with four different volumes, each configured like this:

- Virtual RDM

- Physical RDM
- VMDK file on VMFS
- A disk that connects to an iSCSI target through the use of the Microsoft iSCSI initiator that ships with all current editions of Windows

Otherwise, the environment was configured as follows:

- vSphere 5.0, virtual machine hardware version 8
- The virtual machine was running Windows Server 2008
- It was configured with 4 gigabytes of memory
- A single virtual CPU was added to the virtual machine
- The virtual machine was connected to the local area network's Cisco 2960 switch
- The storage being used is an Iomega PX6

In measuring directly the latency as storage commands make their way through the kernel, Eric discovered that there isn't much of a difference in any of the storage configurations since they all have to go through the kernel, except the iSCSI option, which just goes out over the network and connects to an iSCSI target directly. However, at 1 Gbps, iSCSI had a top speed throughput rate of 112.6 MBps.

The results?
Busted!

In testing, Eric discovered that there was very little difference between either of the RDM configurations and the VMFS configuration. In other words, while there may be other reasons to choose a RDM-based volume over a VMFS-based volume, doing so for performance reasons alone isn't necessary.

VMware's test results

Even VMware has busted this myth in a pretty big way, as shown in Figure 2. The PDF file from which the chart was sourced includes a wide variety of test cases that fully debunk the RDM vs. VMFS myth.

Reasons to choose VMFS over RDMs

Now, understanding that performance isn't a reason to choose RDMs, what are some better reasons to choose VMFS? VMware has spent years improving VMFS and, with vSphere 5, had made tremendous improvements to this robust, cluster-aware file system with features such as:

- Storage I/O control
- Storage vMotion
- Storage DRS

> While there may be other reasons to choose a RDM-based volume over a VMFS-based volume, doing so for performance reasons alone isn't necessary.

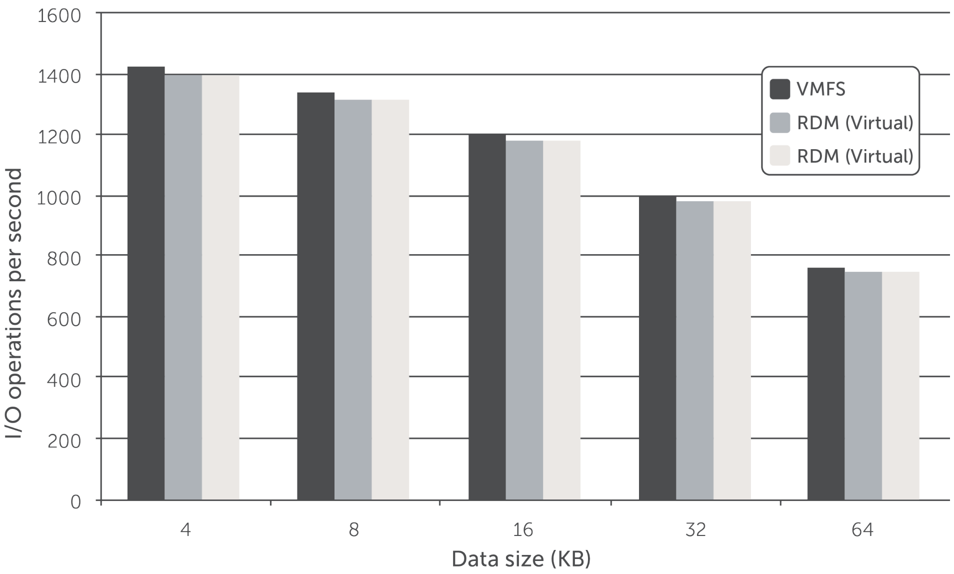


Figure 2. Random mixed I/O per second (higher is better)

- Large volume size: 64 TB
- Large VMDK file size: 2 TB
- Changed block tracking (CBT) support (CBT tracks all of the storage blocks in a virtual machine that have changed since a point in time.)

When to choose RDMs over VMFS
Even tough RDMs don't offer better performance, there are times when an RDM should be considered. When a virtual machine needs access to a particularly large single volume—one that is greater than 2 TB in size—an administrator might consider using a physical RDM, which provides direct access to a volume of up to 64 TB in size and is not subject to VMDK file size limits, which remain at 2 TB. Note that this 64 TB capability is valid only for physical RDMs; virtual RDMs are still limited to a size of 2 TB.

Another time when RDMs may reasonably come into play is when there is a need to perform SAN snapshotting, which results in snapshots not supported by vSphere. Before a SAN can take a snapshot, the virtual machine must be quiesced, which means that the virtual machine needs to flush buffered data to disk and prepare for the snapshot. If you are using SAN snapshots, which are not communicating with the vSphere layer, then you need to use RDM with native file systems, such as like NTFS or EXT3. Another scenario that requires the use of RDMs comes when there is a need to cluster virtual machines with Microsoft Clustering Services. Microsoft's clusters are not aware of what happens with VMFS and expect to see fully configurable and raw storage in order to operate. In that case, use an RDM and present to the virtual machine the storage that it's expecting.

Using CBT can drastically shrink backup windows, and the disk subsystem will also get less utilized during backups.

Myth #2: Changed block tracking causes significant overhead on your virtual machines

What is changed block tracking?

First introduced in vSphere 4, changed block tracking (CBT) is a VMKernel-based driver that tracks all of the storage blocks in a virtual machine that have changed since a point in time. This feature is incredibly powerful because backup and replication technologies can rely on vSphere's own vStorage advanced programming interfaces (APIs), rather than either on drivers and software developed from scratch or on traditional full and incremental backup methodologies for data protection.

Requirements for using CBT

A number of requirements must be met for CBT to operate:

- Since CBT was introduced in vSphere 4, the host must be running at least that version of vSphere.
- CBT must actually be enabled for the virtual machine. This will be discussed below.
- The virtual machine being tracked must be running virtual hardware version 7 or above.
- The virtual machine must be using a storage mechanism that runs through the vSphere storage stack. Such mechanisms include VMFS, NFS and RDMs in virtual compatibility mode. However, an RDM in physical compatibility mode is not supported. iSCSi initiators installed inside a virtual machine do not work with CBT, either.

Benefits of CBT

By using CBT, administrators can drastically shrink their organization's backup windows since the backup application doesn´t need to scan the VMDK files for block changes when doing incremental or differential backups. Even when a full backup is performed, CBT can be useful in that the CBT API will remove all the unallocated blocks when a full VMDK file is read, so only blocks that have actually been allocated to the virtual machine will be processed. On top of reduced backup window the disk subsystem will also get less utilized during backups.

CBT is enabled or disabled on a per-VM basis, so you can choose to use it for some virtual machine and not use it for others. By default, this feature is disabled in a virtual machine and must be proactively enabled by the administrator. Why is this the default? First, common thinking holds that CBT introduces a small amount of performance overhead in the virtual machine, which does make sense since there is additional tracking taking place. However, the myth goes that CBT introduces significant performance overhead. Further, vSphere doesn't know whether or not the backup application in use supports CBT, so it's possible that the feature remains disabled so that the administrator can make a decision that makes the most sense for the organization. That said, a significant amount of today's modern vSphere-aware backup applications do, in fact, support CBT.

Testing the myth

Mattias managed to locate one of the creators of the CBT API at VMware and asked him about some of the particulars related to CBT in order to gain a better understanding of the performance implications that come with CBT's use. From this resource, Mattias was able to gather the following tidbits:

- Memory impact
  - CBT uses a maximum of 256 KB of RAM for a 2 TB virtual disk.
  - About 1.25 KB of RAM is required for every 10 GB of virtual disk space.
- CPU impact
  - Flipping a bit when an I/O request is complete.
- Storage
  - Space: Change tracking file requires 512 KB per 10 GB virtual disk.
  - I/O: When disk is closed, tracking information is written to disk.

Based on these values, it's safe to say that the actual overhead incurred with CBT is just about zero. The memory impact is negligible. The primary impact is on storage, where the CBT file is created. This is a static file that requires about 0.5 megabyte for each 10 GB of a VMDK file. This impact is felt as soon as CBT is enabled, since the CBT needs to allocate upfront all the space that's needed for tracking the information of the VMDK file. Further impact is experienced when a disk is closed. This may happen when, for example, a snapshot is removed from a virtual machine. All of the changes are written from memory to the CBT driver, which then writes it to the CBT tracking file. Also note that the size of the CBT file doesn't change over time; all of the space it needs is allocated upfront, so this is a one-time hit.

Not one to simply state something as fact, Mattias took to his lab and ran side-by-side tests. One test was with CBT enabled and the other was with the feature disabled. He came to the conclusion that he was simply unable to detect any measurable overhead caused by CBT except when it came to the CBT file itself. That file he could actually see in the file system, so there was obvious evidence that it was there, but the size was negligible.

The results?
Busted!

The conclusion is that the overhead introduced by changed block tracking is so small as to be, for all practical purposes, zero. However, the benefits of CBT are significant for an organization that is using a CBT-aware backup and replication tool.

The overhead introduced by CBT is so small as to be, for all practical purposes, zero. However, the benefits of CBT are significant for organizations using CBT-aware backup and replication tools.

Myth #3: Resource pools should always be used to categorize and allocate resources to virtual machines

Ways of managing resources in vSphere

There are many different ways that resources can be managed in vSphere.

- Resource pools—With the release of VMware ESX 3 came the introduction of resource pools, which can be used to categorize virtual machines based on functionality or designation (such as production or test), and which can also be used to allocate resources based on allocated shares. A resource pool can be used to limit a group of virtual machines for CPU and memory and to place reservations on these resources. The use of shares is a way to give priority to more important virtual machines in case of vSphere host is running out of resources.

- **vApps**—There is another kind of resource pool called a vApp. A vApp is a miniature resource pool used to combine multi-tier web applications—such as a web server, database server and a middleware server—into a single vApp, which allows the related resources to be managed in a central way. With a vApp, you can configure the startup order for the virtual machines that are a part of the vApp and can distribute IP address to member machines and configure shared resources, including CPU and memory. In the sense that vApps share resources, they are a form of a resource pool, sometimes considered a luxury resource pool because there are usually just a few virtual machines participating in the vApp.
- **Virtual machines**—Another component that must be considered when considering the use of resource pools is the individual virtual machine. Actually, it is necessary to think about all of the individual virtual machines since they will participate in resource contests with vApps and resource pools.

So, there are three kinds of objects that are contending for resources at the vCenter level: resource pools, vApps and virtual machines.

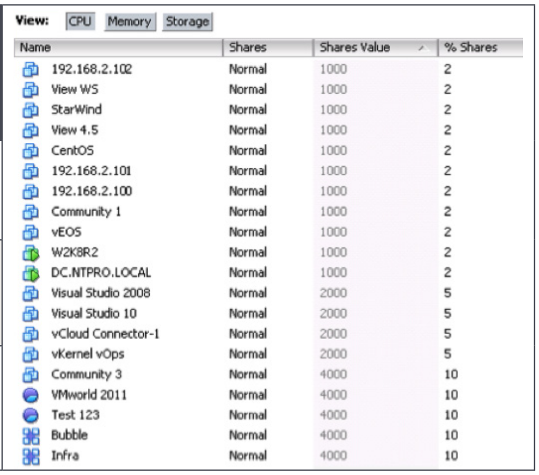
Every new resource pool has the same share value as any other resource pool

Unfortunately, many people don't realize what's happening behind the scenes when resource pools are being used to manage resources for groups of virtual machines. When a resource pool is created, it has the same default settings as every other brand new resource pool. Most critically, regardless of how many virtual machines reside in the resource pool—five or five hundred—every resource pool still has the same share value. Again, that's interesting because not every resource pool will have the same number of virtual machines inside.

All of these objects are living in the vCenter inventory. As vApps and resource pools are created, they are created as siblings to the existing virtual machines and each of these resource pool types carries a share value of 4,000. For each vCPU that is included in a virtual machine, that virtual machine gets 1,000 shares. So, a virtual machine with four vCPUs carries a share value of 4,000 as well.

So, this means that each of these objects—a vApp with, say, three virtual machines; a resource pool with 20 virtual machines; and a single virtual machine with four vCPUs—all receive the same number of shares, as shown in Figure 3.

Sibling CPU Shares Value
vApp = 4000
Resource Pool = 4000
Virtual Machine = number of vCPUs * 1000



Name	Shares	Shares Value	% Shares
192.168.2.102	Normal	1000	2
View WS	Normal	1000	2
StarWind	Normal	1000	2
View 4.5	Normal	1000	2
CentOS	Normal	1000	2
192.168.2.101	Normal	1000	2
192.168.2.100	Normal	1000	2
Community 1	Normal	1000	2
vEOS	Normal	1000	2
W2K8R2	Normal	1000	2
DC.NTPRO.LOCAL	Normal	1000	2
Visual Studio 2008	Normal	2000	5
Visual Studio 10	Normal	2000	5
vCloud Connector-1	Normal	2000	5
vKernel vOps	Normal	2000	5
Community 3	Normal	4000	10
VMworld 2011	Normal	4000	10
Test 123	Normal	4000	10
Bubble	Normal	4000	10
Infra	Normal	4000	10

Figure 3. A vApp with three VMs, a resource pool with 20 VMs, and a single VM with 4 vCPUs all receive 4000 shares.

Now, let's suppose that an administrator creates what is often referred to as a "monster virtual machine" with 32 vCPUs. This virtual machine will consume 32,000 shares. If placed at the same level as the aforementioned objects, this means that this single virtual machine will get eight times as many shares as the vApp and the resource pool that holds 20 individual virtual machines. Therefore, administrators need to act with caution when placing resources at the same level in vCenter.

Example: how a pool of 24 Ghz is divided among various resource pools

Figure 4 illustrates what happens from a per-VM perspective in the situations that have been discussed thus far. Note that all of the objects shown reside at the same level in vCenter and therefore each object divides the available resources—24 GHz worth—equally into 6 GHz chunks. In each box, you can see what that division means for each virtual machine that might be running inside a particular object. For the resource pool with eight virtual machines, each virtual machine gets 0.75 GHz of resource capacity, while the virtual machines with four vCPUs get a full 6 GHz of resourcing.

By not paying adequate attention to these kinds of details, a vSphere administrator risks creating a situation that is not immediately obvious from a troubleshooting perspective and that can have a major negative impact on the entire environment.

Bear in mind that this impact happens only when the default resource pool settings are kept in place. However, from a myth perspective, it means that resource pools really don't make it much easier to manage virtual machine resources in any comprehensive way.

The results?
Busted!

Resource pools really don't make it much easier to manage virtual machine resources in any comprehensive way.

How to use resource pools effectively

Obviously, if it's really necessary, one can use vApps and resource pools, but it's imperative that the administrator refuse to simply accept the default options that come with these containers and ensure that the environment remains balanced even while it's constantly changing with the addition and deletion of new virtual machines. This means that the shares must be manually adjusted for each object in the environment to ensure that resources are allocated in a way that

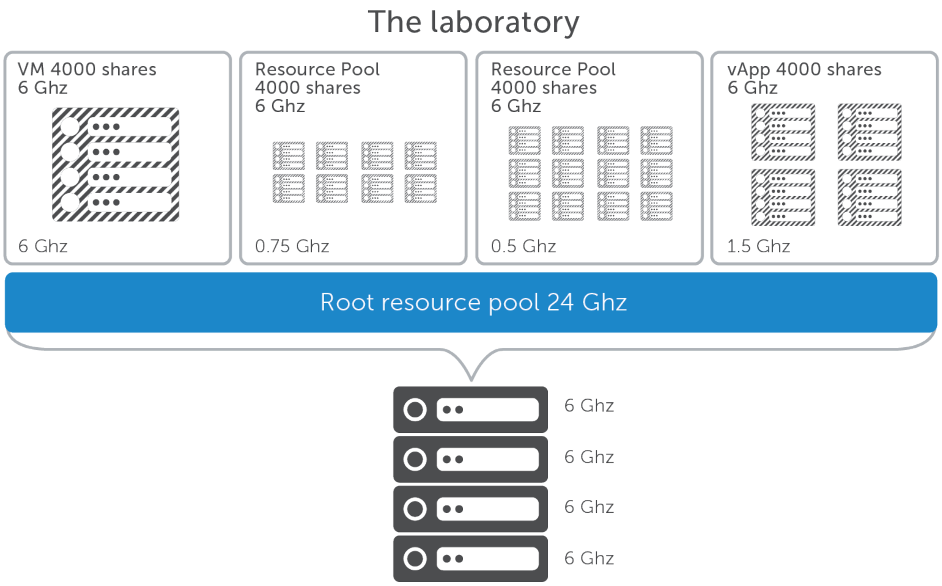


Figure 4. How a resource pool of 24 Ghz is divided by default among various resource pools

makes sense for the business. Further, never ignore the impact that larger virtual machines might have on these sibling objects. The monster VM may ultimately consume more resources than desired.

Figure 5 provides another example of how resources are divided. In this figure, there are two virtual machines and a single resource pool. The resource pools gets 4,000 shares, the virtual machine with two vCPUs gets 2,000, and the single-vCPU VM gets 1,000. You can see the corresponding percentages in the figure.

Myth #4: LSI Logic SCSI is always better than paravirtualized SCSI.

What is the paravirtualized SCSI (PVSCSI) adapter?

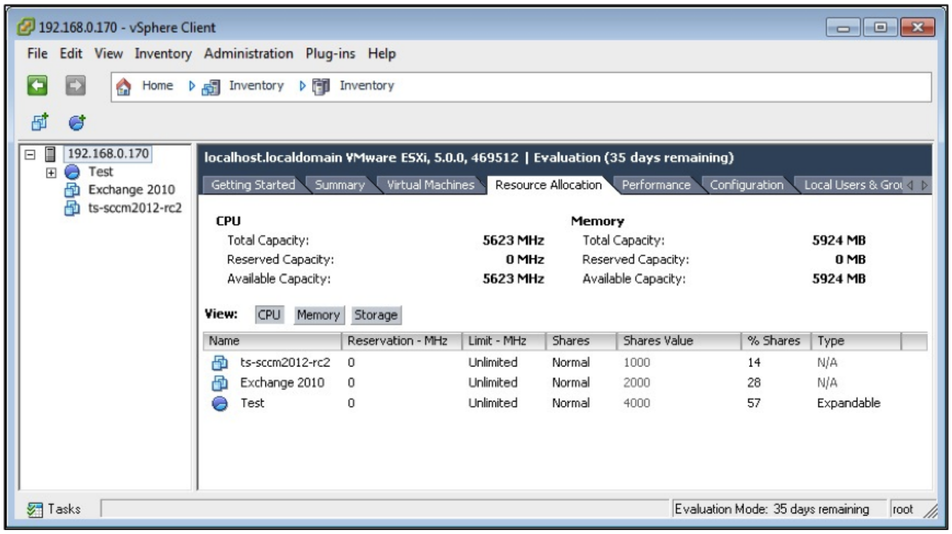
Introduced in vSphere 4, the paravirtualized SCSI (PVSCSI) adapter was intended to provide increased performance to the storage subsystem in a virtual machine (+12%) while decreasing the virtual storage adapter's impact on the vSphere host (-18%). Supported only in hardware version 7 or later, the PVSCSI adapter is supported by a subset of VMware's guest operating systems, including the following:

- Windows Server 2003 or higher
- Red Hat Enterprise Linux 5 or higher
- SUSE Linux Enterprise 11 SP1 or higher
- Ubuntu 10.04 or higher
- Linux 2.6.33 or higher

The origins of the myth

As introduced in vSphere 4, the PVSCSI adapter had quite a number of problems and limitations. For example, upon the initial release of vSphere 4, a PVSCSI adapter could not be used as the adapter for the boot volume of the virtual machine. There was also guidance from VMware indicating that virtual machines with low I/O demand could experience worse I/O performance with the PVSCSI adapter. The guidance indicated that the intent behind PVSCSI was to boost the performance of the monster VM and decrease its impact on the vSphere host. It's no wonder that the myth that LSI should be preferred over PVSCSI continued to spread. However, these issues were resolved in vSphere 4.0 Update 1.

Given that the initial problems with PVSCSI have long been fixed, it's interesting to note that an LSI Logic SCSI adapter is generally the default option recommended by vSphere when a new virtual machine is created. Most administrators have been trained that deviating from default recommendations generally doesn't make sense, so many move forward with this adapter recommendation.

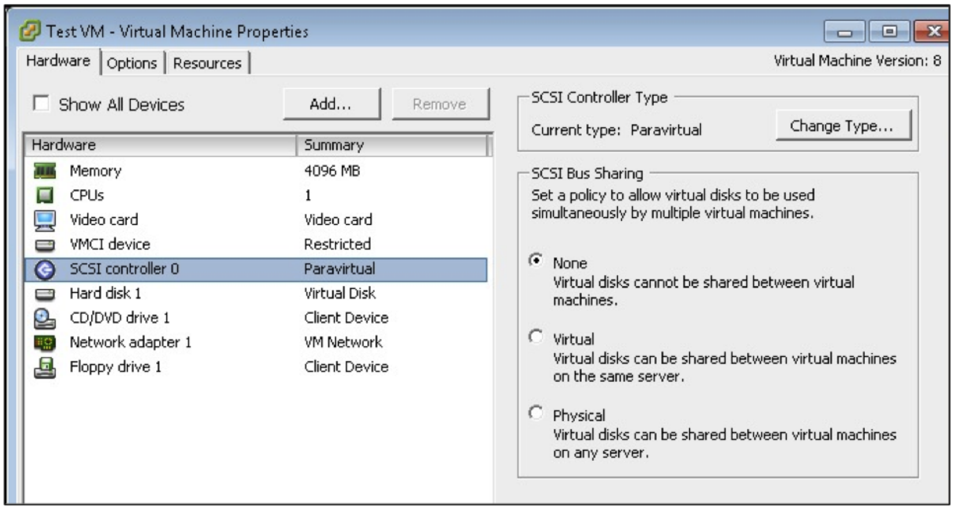


localhost.localdomain VMware ESXi, 5.0.0, 469512 | Evaluation (35 days remaining)

CPU		Memory	
Total Capacity:	5623 MHz	Total Capacity:	5924 MB
Reserved Capacity:	0 MHz	Reserved Capacity:	0 MB
Available Capacity:	5623 MHz	Available Capacity:	5924 MB

Name	Reservation - MHz	Limit - MHz	Shares	Shares Value	% Shares	Type
ts-sccm2012-rc2	0	Unlimited	Normal	1000	14	N/A
Exchange 2010	0	Unlimited	Normal	2000	28	N/A
Test	0	Unlimited	Normal	4000	57	Expandable

Figure 5. Shares can be divided appropriately among two VMs and a resource pool.



Test VM - Virtual Machine Properties

Hardware | Options | Resources

Virtual Machine Version: 8

☐ Show All Devices   Add...   Remove

Hardware	Summary
Memory	4096 MB
CPUs	1
Video card	Video card
VMCI device	Restricted
SCSI controller 0	Paravirtual
Hard disk 1	Virtual Disk
CD/DVD drive 1	Client Device
Network adapter 1	VM Network
Floppy drive 1	Client Device

SCSI Controller Type

Current type: Paravirtual   Change Type...

SCSI Bus Sharing

Set a policy to allow virtual disks to be used simultaneously by multiple virtual machines.

● None
Virtual disks cannot be shared between virtual machines.

○ Virtual
Virtual disks can be shared between virtual machines on the same server.

○ Physical
Virtual disks can be shared between virtual machines on any server.

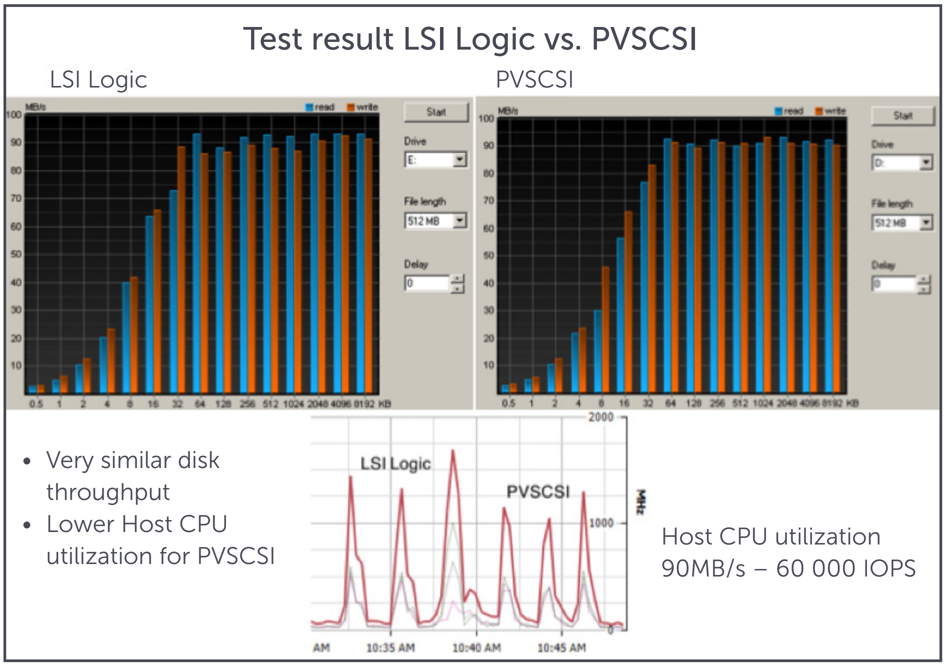
Figure 6. Options for a PVSCSI adapter

It's still important to recognize that the PVSCSI adapter isn't supported by every guest operating system, but the most common ones support it. When the PVSCSI adapter was introduced way back in vSphere 4, it was targeted at those monster VMs that needed particularly fast disk I/O; it was not recommended for the average virtual machine.

Testing the myth

In order to see for himself how the latest PVSCSI adapter operates, Mattias tested them head-to-head. In the same lab as was used previously, a vSphere 5 environment, Mattias tested the LSI Logic controller and the PVSCSI controller against his iSCSI SAN environment. He used Windows Server 2003 virtual machines and followed all the best practices; for example, he ensured that there were no disk alignment issues.

Paravirtualized SCSI is equal to or faster than LSI SCSI and requires less CPU resources.



Test result LSI Logic vs. PVSCSI

LSI Logic

PVSCSI

- Very similar disk throughput
- Lower Host CPU utilization for PVSCSI

Host CPU utilization
90MB/s – 60 000 IOPS

Figure 7. CPU utilization is significantly less for the PVSCSI controller than for the LSI controller.

From there, he ran a series of disk performance tests, first with the LSI Logic controller and with a paravirtualized SCSI controller. As shown in Figure 7, it's clear that the throughput rates are practically identical but the host utilization for the PVSCSI controller is quite a bit less than it is for the LSI controller. The reason that the graphs don't show much difference in throughout is because the tested disk environment was too small. It had only six spindles and was running older hardware, so it couldn't move much faster than seen in the chart. If the test had used an up-to-date SAN with more disk spindles, there would be clear throughput benefit.

However, as mentioned, there is significantly less CPU utilization with the PVSCSI adapter. As one considers today's modern environments and constrained budgets, it's important to achieve the highest virtual machine density possible in the virtual environment. With the PVSCSI adapter, an organization can fit more virtual machines on a single vSphere host and push a little bit more throughput as well.

With the PVSCSI adapter, an organization can fit more virtual machines on a single vSphere host and push a little bit more throughput as well.

Disk alignment

Disk alignment remains an important consideration for older operating systems. When a virtual disk is not aligned properly, there can be significant performance degradation, sometimes approaching 25–30 percent. In order to improve performance and recover these lost cycles, vSphere administrators should always make sure that disks are aligned for operating systems that require such alignment, such as versions of Windows prior to Windows Server 2008. With Windows 2008, manual alignment adjustment is no longer necessary.

The results?
Busted!

Paravirtualized SCSI is equal to or faster than LSI SCSI and requires less CPU resources. vSphere 5 carries none of the baggage that it did back in the vSphere 4.0 days. Now that PVSCSI supports boot volumes and works well for even smaller, less I/O-intensive volumes, it's an option that should be carefully considered. The worst that will happen is receiving identical performance, but it's more than likely that the environment will operate more efficiently than with the default LSI adapter. Consider using PVSCSI as a part of your virtual machine templates so you automatically get the benefits it provides when deploying new virtual machines. When it comes to boosting the performance of and reducing the host impact from your existing virtual machines, identify your top 5-10 disk intensive virtual machines and consider updating them to PVSCSI.

Conclusion

The myths described in this paper were guidance that made sense at one time. However, as new product versions have been released and administrators have learned more about exactly how certain features operate, these myths no longer represent the truth about vSphere. Let the information in this paper be your definitive answer to these four myths.

Move beyond monitoring

Simplify virtualization and storage management with Foglight for Virtualization.

Try It for 30 Days

For More Information

The information in this document is provided in connection with Dell products. No license, express or implied, by estoppel or otherwise, to any intellectual property right is granted by this document or in connection with the sale of Dell products. EXCEPT AS SET FORTH IN DELL'S TERMS AND CONDITIONS AS SPECIFIED IN THE LICENSE AGREEMENT FOR THIS PRODUCT, DELL ASSUMES NO LIABILITY WHATSOEVER AND DISCLAIMS ANY EXPRESS, IMPLIED OR STATUTORY WARRANTY RELATING TO ITS PRODUCTS INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT. IN NO EVENT SHALL DELL BE LIABLE FOR ANY DIRECT, INDIRECT, CONSEQUENTIAL, PUNITIVE, SPECIAL OR INCIDENTAL DAMAGES (INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF PROFITS, BUSINESS INTERRUPTION OR LOSS OF INFORMATION) ARISING OUT OF THE USE OR INABILITY TO USE THIS DOCUMENT, EVEN IF DELL HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. Dell makes no representations or warranties with respect to the accuracy or completeness of the contents of this document and reserves the right to make changes to specifications and product descriptions at any time without notice. Dell does not make any commitment to update the information contained in this document.

About Dell

Dell Inc. (NASDAQ: DELL) listens to customers and delivers worldwide innovative technology, business solutions and services they trust and value. For more information, visit www.dell.com.

If you have any questions regarding your potential use of this material, contact:

Dell Software
5 Polaris Way
Aliso Viejo, CA 92656
www.dell.com
Refer to our Web site for regional and international office information.