VMware vSphere VMFS

VMware Storage

Table of contents

VMware vSphere VMFS

Introduction

This section covers the introduction of VMware vSphere VMFS. It provides insights on the functionalities and capabilities of VMFS and how it works benefiting the organization.

VMware vSphere® VMFS is a high-performance cluster file system (CFS) that enables virtualization to scale beyond the boundaries of a single system. Designed, constructed and optimized for vSphere virtual infrastructures, VMFS increases resource utilization by providing multiple virtual machines with shared access to a consolidated pool of clustered storage. It offers the foundation for virtualization spanning multiple servers, enabling services such as virtual machine snapshots, VMware vSphere Thin Provisioning, VMware vSphere vMotion®, VMware vSphere Distributed Resource Scheduler (vSphere DRS), VMware vSphere High Availability (vSphere HA), VMware vSphere Storage DRS and VMware vSphere Storage vMotion®.

VMFS reduces management overhead by providing a highly effective virtualization management layer that is especially suitable for large-scale enterprise datacenters. Administrators employing VMFS find it easy and straightforward to use, and they benefit from the greater efficiency and increased storage utilization offered by the use of shared resources.

This paper provides a technical overview of VMFS, including a discussion of features and their benefits. It highlights how VMFS capabilities enable greater scalability and decreased management overhead and it offers best practices and architectural considerations for deployment of VMFS. The paper concludes with an overview of recent enhancements to VMFS6.

Background

This section provides a brief about VMware vSphere VMFS and it's benefits in the virtual environment.

In today’s IT environment, systems administrators must balance competing goals: finding ways to scale and consolidate their environment while decreasing the management overhead required to provision and monitor resources. Virtualization provides the answer to this challenge. VMware vSphere enables administrators to run more workloads on a single server, and it facilitates virtual machine mobility without downtime.

A key feature of vSphere is the ability for all machines to dynamically access shared resources such as a pool of storage. VMware® vCenter provides a management interface that can easily provision, monitor and leverage the shared disk resources. Without such an intelligent interface, the operational costs of scaling virtual machine workloads and their storage resources might affect the benefits of virtualization.

VMware has addressed these needs by developing VMFS to increase the benefits gained from sharing storage resources in a virtual environment. VMFS plays a key role in making the virtual environment easy to provision and manage. It provides the foundation for storage access to virtual machines by making available an automated CFS along with cluster volume management capabilities for the virtual environment.

VMFS Technical Overview

This section provides a technical overview of the VMFS storage management interface.

VMFS is a high-performance CFS that provides storage virtualization that is optimized for virtual machines. Each virtual machine is encapsulated in a small set of files; VMFS is the default storage management interface for these files on physical disks and partitions.

VMFS empowers IT organizations to greatly simplify virtual machine provisioning by efficiently storing the entire machine state in a central location. It enables multiple instances of vSphere hosts to access shared virtual machine storage concurrently. It also enables virtualization-based distributed infrastructure services such as vSphere DRS, vSphere HA, vMotion, vSphere Storage DRS and Storage vMotion to operate across a cluster of vSphere hosts. In short, VMFS provides the foundation that enables the scaling of virtualization beyond the boundaries of a single system.

Figure 1 shows how multiple vSphere hosts with several virtual machines running on them can use VMFS to share a common clustered pool of storage.

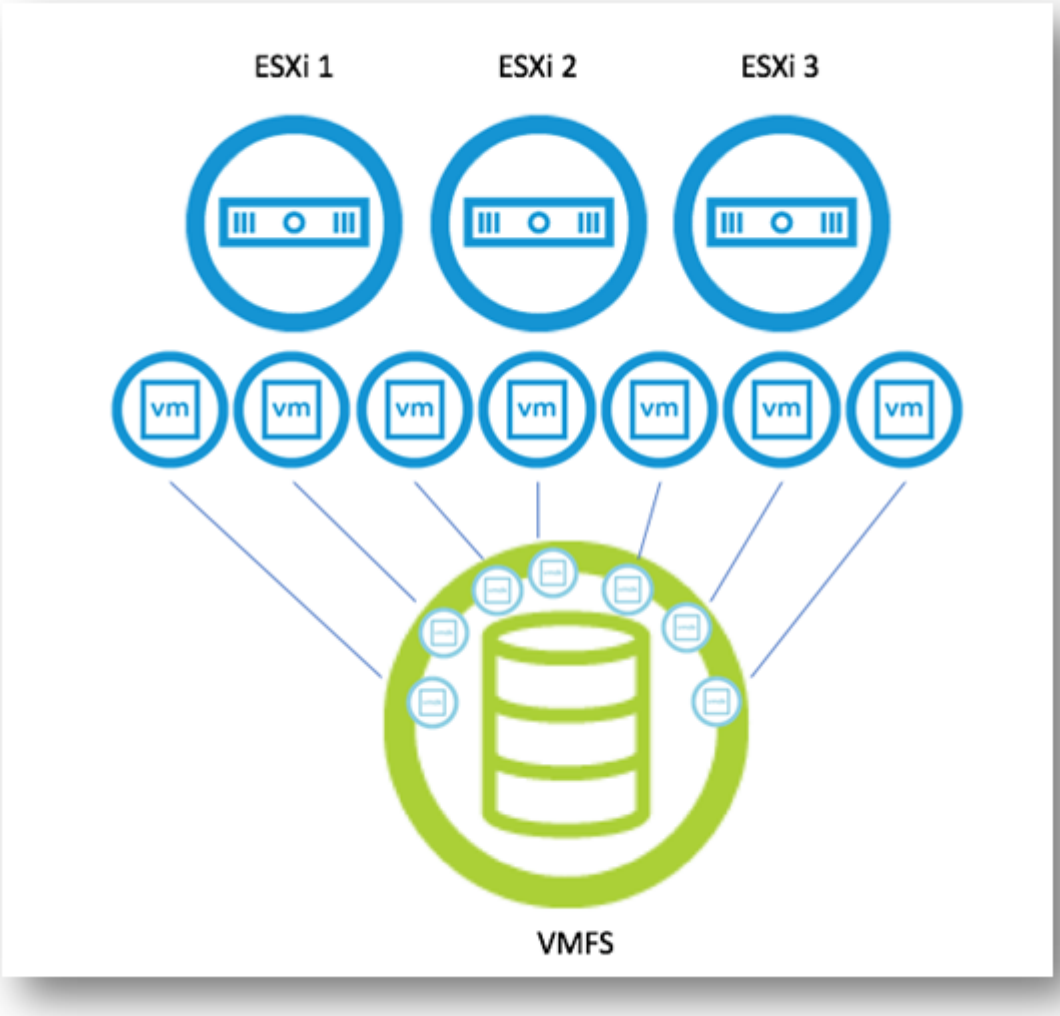

Figure 1. VMFS as a Common Pool of Storage

Each of the three vSphere hosts has a number of virtual machines running on it. The lines connecting them to the disk icons for the virtual machine disks (VMDKs) are logical representations of the association between and allocation of the larger VMFS volume, which is made up of one large logical unit number (LUN). A virtual machine detects the VMDK as a local SCSI target. The virtual disks are really just files on the VMFS volume.

Each vSphere host stores its virtual machine files in a specific subdirectory on VMFS. When a virtual machine is operating, VMFS has a lock on those files so other vSphere hosts cannot update them. VMFS ensures that the virtual machine cannot be opened by

more than one vSphere host in the cluster unless explicitly instructed to do so, as in the case of clustering applications running in the virtual machine, or fault tolerance or linked clones.

Each of the three vSphere hosts detects the entire LUN. The LUN is a clustered volume, and VMFS provides the distributed lock management that arbitrates access, enabling vSphere hosts to share the clustered pool of storage. The point of control moves from the SAN to the VMkernel, with no loss of security.

Features of VMFS

The following technical features of VMFS are among those that make it suitable for use in a virtual environment:

- Automated file system with hierarchical directory structure
- Optimization for virtual machines in a clustered environment
- Lock management and distributed logical volume management
- Dynamic datastore expansion by spanning multiple storage extents
- CFS with journal logging for fast recovery
- Thin-provisioned virtual disk format for space optimization
- Virtual machine–level point-in-time snapshot copy management
- Encapsulation of the entire virtual machine state in a single directory
- Support for VMware vSphere Storage APIs for Array Integration (VAAI)

Benefits of VMFS

This section provides the benefits of VMFS.

As an intelligent and automated storage interface for virtual machine environments, VMFS provides both an automated CFS capability and intelligent cluster volume management functions. It has a number of benefits that make it particularly well suited as a CFS for the virtual environment. It is included with vSphere at no additional cost and is tailored to virtual machine performance patterns.

Enables Automated CFS Capability

VMFS is automated and optimized for virtual machines. It enables multiple vSphere hosts to access the same virtual machine storage. Virtual machines can be dynamically and automatically migrated between vSphere hosts.

Optimizes Virtual Machine Access

VMFS provides the SCSI access layer for virtual machines to efficiently read and write data on the underlying disk. VMFS6, the most recent release, uses a unified 1MB file block allocation for large files, and sub-block allocation for small files and directories. VMFS is rigorously tested and certified for a wide range of Fibre Channel, Fibre Channel over Ethernet (FCoE) and iSCSI storage systems, and it is optimized to support large files while also performing many small concurrent writes.

Encapsulates the Entire Virtual Machine State in a Single Directory

VMFS stores all of the files that make up the virtual machine in a single directory and automatically creates a new subdirectory for each new virtual machine. This location is often referred to as the “virtual machine home”.

Simplifies Provisioning and Administration of Virtual Machines

VMFS reduces the number of steps required to provision storage for a virtual machine. It also reduces the number of interactions required between virtualization administration (vSphere administrators) and the storage administration team to allocate storage to a new virtual machine. vSphere administrators appreciate the automated le naming and directory creation as well as the user-friendly hierarchical le system structure that eases navigation through the les that form the virtual machine environment.

Provides Distributed Infrastructure Services for Multiple vSphere Hosts

VMFS provides on-disk locking that enables concurrent sharing of virtual machine disk files across many vSphere hosts. In fact, VMFS enables virtual disk files to be shared by as many as 32 vSphere hosts (depending on the use-case). Furthermore, it manages storage access for multiple vSphere hosts and enables them to read and write to the same storage pool at the same time.

It also provides the means by which vSphere DRS and vMotion can dynamically move an entire virtual machine from one vSphere host to another in the same cluster without having to restart the virtual machine. vSphere Storage DRS and Storage vMotion offer the capability to move a virtual machine home from one datastore to another without downtime, which enables migration of virtual machines off of an overcrowded or saturated datastore or to a datastore with less usage or a different performance capacity.

Facilitates Dynamic Growth

Through the use of a volume management layer, VMFS enables an interface to storage resources so that several types of storage (Fibre Channel, iSCSI and FCoE) can be presented as datastores on which virtual machines can reside. Enabling dynamic growth of those datastores through aggregation of storage resources provides the ability to increase a shared storage resource pool without incurring downtime. Through the VMFS Volume Grow feature, a datastore on block-based storage now can be expanded on an underlying LUN that has been expanded within the storage array. VMFS also enables dynamic growth of the virtual machine disk.

Provides Intelligent Cluster Volume Management

VMFS simplifies administration with an intelligent interface that makes it easy to manage allocation and access of virtual disk resources, providing the ability to recognize and mount snapshot copies at the datastore or LUN level. VMFS has a volume signature that can be resignatured to manage additional but convergent copies of a given datastore on block-based storage.

Optimizes Storage Utilization

VMFS enables virtual disk thin provisioning as a means to dramatically increase storage utilization. With dynamic allocation and intelligent provisioning of available storage capacity in a datastore, Thin Provisioning reduces the amount of space that is allocated but not used in a datastore.

Enables High Availability with Lower Management Overhead

VMFS enables portability of virtual machines across vSphere hosts to provide high availability while lowering management overhead. As a CFS and cluster volume manager (CVM), VMFS enables unique virtualization services that leverage live migration of running virtual machines from one vSphere host to another. VMFS also facilitates automatic restart of a failed virtual machine on a separate vSphere host, and it supports clustering virtual machines across different vSphere hosts. File-level lock management provides the foundation needed for the multi-server virtualization that enables vSphere HA, vSphere DRS, vMotion, vSphere Storage DRS, Storage vMotion and VMware vSphere Fault Tolerance (FT), causing less downtime and faster recovery.

Simplifies Disaster Recovery

Because VMFS stores virtual machine files in a single subdirectory, disaster recovery, testing and cloning are greatly simplified. The entire state of the virtual machine can be remotely mirrored and easily recovered in the event of a disaster.

And with automated handling of virtual machine files, VMFS provides encapsulation of the entire virtual machine so that it easily can become part of a disaster recovery solution. The following VMFS features are among those that are especially useful in disaster recovery:

- Hardware independence between primary and secondary sites
- Encapsulation—all files for a virtual machine in one place
- Robust journal le system capability for CFS metadata
- Integration of raw disk maps (RDMs) in the VMFS structure
- Resignature option to handle storage array–based snapshots

VMware vCenter Site Recovery Manager Server (SRM Server) and VMware vSphere Replication leverage many of these features in the replication and disaster recovery of virtual environments.

Comparing VMFS to Conventional File Systems

Conventional file systems allow only one server to have read/write access to a specific file at a given time. In contrast, VMFS is a CFS that leverages shared storage to enable multiple vSphere hosts to have concurrent read and write access to the same storage resources. VMFS also has distributed journaling of changes to the VMFS metadata to enable fast and resilient recovery across multiple vSphere clusters.

On-disk locking in VMFS ensures that a virtual machine is not powered on by multiple installations of a vSphere host at the same time. With vSphere HA enabled, if a server fails, the on-disk lock for each virtual machine is released, enabling the virtual machine to be restarted on other vSphere hosts. Moreover, VMFS provides virtual machine–level snapshot capability, enabling fast point-in-time recovery. VM Backup products from many VMware partners leverage this feature to provide consistent backup of virtual environments.

VMFS does not have every feature found today in other CFS and CVM systems. However, there is no other CFS or CVM that provides the capabilities of VMFS. Its distributed locking methods forge the link between the virtual machine and the underlying storage resources in a manner that no other CFS or CVM can equal.The unique capabilities of VMFS enable virtual machines to join a VMware cluster seamlessly, with no management overhead.

Best Practices for Deployment and Use of VMFS

This section provides the best practices for the deployment and use of VMFS.

This section offers some best practices, insight, and experience in addressing several questions that often arise when deploying and using VMFS volumes. It is not intended to provide the definitive answer for every question, because often there is no single right answer. The intent here is to discuss what the trade-offs and considerations are, as well as to offer some insights into choosing the answer that best fits a specific configuration.

The following topics are addressed:

- How large should LUNs be created for a given VMFS volume?
- Should we isolate storage for virtual machines or share a consolidated pool of storage?
- Should we use RDMs or VMFS volumes?
- Should we use disk spanning? If so, are there any concerns or suggestions?
- Is the array using spinning media or is it all-flash?

How Large a LUN?

The best way to configure a LUN for a given VMFS volume is to size for throughput first and capacity second. That is, you should aggregate the total I/O throughput for all applications or virtual machines that might run on a given shared pool of storage; then make sure you have provisioned enough back-end devices and appropriate storage service to meet the requirements.

This is actually no different from what most system administrators do in a physical environment. It just requires an extra step, to consider when to consolidate a number of workloads onto a single vSphere host or onto a collection of vSphere hosts that are addressing a shared pool of storage.

Each storage vendor likely has its own recommendation for the size of a provisioned LUN, so it is best to check with the vendor. However, especially for arrays based on spinning disk, if the vendor's stated optimal LUN capacity is backed with a single disk that has little or no storage array write cache, the configuration might result in low performance in a virtual environment. In this case, a better solution might be a smaller LUN striped within the storage array across many physical disks, with some write cache in the array. The RAID protection level also factors into the I/O throughput performance. For All-Flash storage arrays, many of these considerations become moot due to the high performance levels that All-Flash storage arrays can deliver.

Because there is no single correct answer to the question of how large your LUNs should be for a VMFS volume, the more important question to ask is, "How long would it take one to restore the virtual machines on this datastore if it were to fail?" The recovery time objective (RTO) is now the major consideration when deciding how large to make a VMFS datastore. This equates to how long it would take an administrator to restore all of the virtual machines residing on a single VMFS volume if there were a failure that caused data loss. With the advent of very powerful storage arrays, including All-Flash storage arrays, the storage performance has become less of a concern. The main concern now is how long it would take to recover from a catastrophic storage failure. Another important question to ask is, "How does one determine whether a certain datastore is over-provisioned or under-provisioned?" There are many performance screens and metrics that can be investigated within vCenter or VMware vRealize Operations to monitor datastore I/O rates and latency. Monitoring these metrics is the best way to determine whether a LUN is properly sized and loaded. Because workload can vary over time, periodic tracking is an important consideration. vSphere Storage DRS can also be a useful feature to leverage for load balancing virtual machines across multiple datastores, from both a capacity and a performance perspective.

Isolation or Consolidation?

The decision of whether to "isolate" or "consolidate" storage resources for virtual environments is a topic of some debate. The basic answer depends on the nature of the I/O access patterns of that virtual machine. If you have a very heavy I/O-generating application, in many cases VMware vSphere Storage I/O Control can assist in managing fairness of I/O resources among virtual machines. Another consideration in addressing the "noisy neighbor" problem is that it might be worth the potentially inefficient use of resources to allocate a single LUN to a single virtual machine. This can be accomplished using either an RDM or a VMFS volume that is dedicated to a single virtual machine. These two types of volumes perform similarly with varying read and write sizes and I/O access patterns. Figure 2 illustrates the differences between isolation and consolidation.

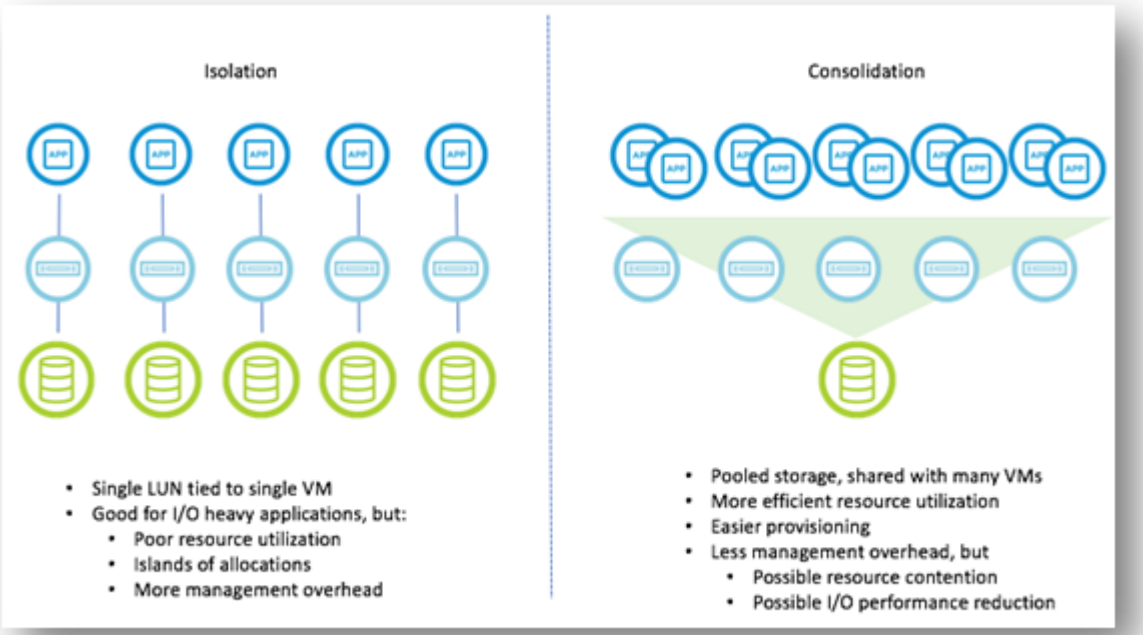

Figure 2. Differences Between Consolidation and Isolation

The following are detailed arguments regarding isolation and consolidation:

Isolated Storage Resources

One school of thought suggests limiting the access of a single LUN to a single virtual machine. In the physical world, this is quite common. When using RDMs, such isolation is implicit, because each RDM volume is mapped to a single virtual machine.

The downside to this approach is that as you scale the virtual environment, you may quickly reach the upper limit of 512 LUNs per host (this new maximum was introduced in vSphere 6.5). You also must provision an additional disk/LUN each time you want to increase storage capacity for the virtual machine. This can lead to significant management overhead; in some environments, it might take days for a request to provision a disk/LUN to be serviced by the storage administration team.

Another consideration is that every time you must expand the capacity of a virtual machine, your minimum commit size is the allocation of a LUN. Although many arrays allow LUNs of any size, the storage administration team might balk at carving up lots of small LUNs because this configuration makes it more difficult for them to manage the array. Most storage teams prefer to allocate LUNs that are fairly large; they like the system administration or applications teams to divide those LUNs into smaller chunks higher up in the stack. VMFS suits this allocation scheme perfectly and is one of the reasons VMFS is so effective in the virtualization storage management layer.

Consolidated Pools of Storage

The consolidation school wants to gain additional management productivity and resource utilization by pooling the storage resource and sharing it, with many virtual machines running on several vSphere hosts. Dividing this shared resource among many virtual machines enables better flexibility as well as easier provisioning and ongoing management of the storage resources for the virtual environment.

Compared to strict isolation, consolidation normally offers better utilization of storage resources. The cost is additional resource contention, which under some circumstances can lead to reduction in virtual machine I/O performance. However, vSphere offers Storage I/O Control and vSphere Storage DRS to mitigate these risks.

At this time, there are no clear rules of thumb regarding the limits of scalability. For most environments, the ease of storage resource management and incremental provisioning offers gains that outweigh any performance impact. As you will see later in this paper, however, there are limits to the extent of consolidation.

Best Practice: Mix Consolidation with Some Isolation

In general, vSphere Storage DRS may be used to detect and mitigate storage latency and capacity bottlenecks by load balancing virtual machines across multiple VMFS volumes. Additionally, vSphere Storage I/O Control can be leveraged to ensure fairness of

I/O resource distribution among many virtual machines sharing the same VMFS datastore. However, the vSphere Storage DRS and vSphere Storage I/O Control features might not always be available. Another option is to separate heavy I/O workloads from the shared pool of storage. This optimizes the performance of those high-transactional throughput applications—an approach best characterized as “consolidation with some level of isolation.”

Because workloads can vary significantly, there is no exact formula that determines the limits of performance and scalability regarding the number of virtual machines per LUN. These limits also depend on the number of vSphere hosts sharing concurrent access to a given VMFS volume. The key is to remember the upper limit of 512 LUNs per vSphere host and consider that this number can diminish the consolidation ratio if you take the concept of “one LUN per virtual machine” too far.

Many different applications can easily and effectively share a clustered pool of storage. And the increase in disk utilization and improvements in management efficiency clearly can compensate for the minor performance reductions caused by the additional contention.

Use of RDMs or VMFS?

Another question is when to use VMFS and when to use RDMs. This section explains the trade-offs.

About RDMs

Even with all the advantages of VMFS, there still are some cases where it makes more sense to use RDM storage access. The following scenarios call for raw disk mapping:

- Migrating an existing application from a physical environment to virtualization
- Using Windows Server Failover Cluster (WSFC) for failover clustering in a virtual environment
- Implementing N-Port ID Virtualization (NPIV)
- Separating heavy I/O workloads from the shared pool of storage

At this point, it should be pointed out that VMware recommends avoiding RDMs if at all possible. Many newer versions of applications now allow for high availability without the need for failover clustering. Microsoft SQL Server AlwaysOn Availability Groups and Microsoft Exchange Database Availability Groups (DAG) are two such examples. However, VMware understands that, in certain cases, failover clustering is still desired, so we continue to support RDMs for this use-case.

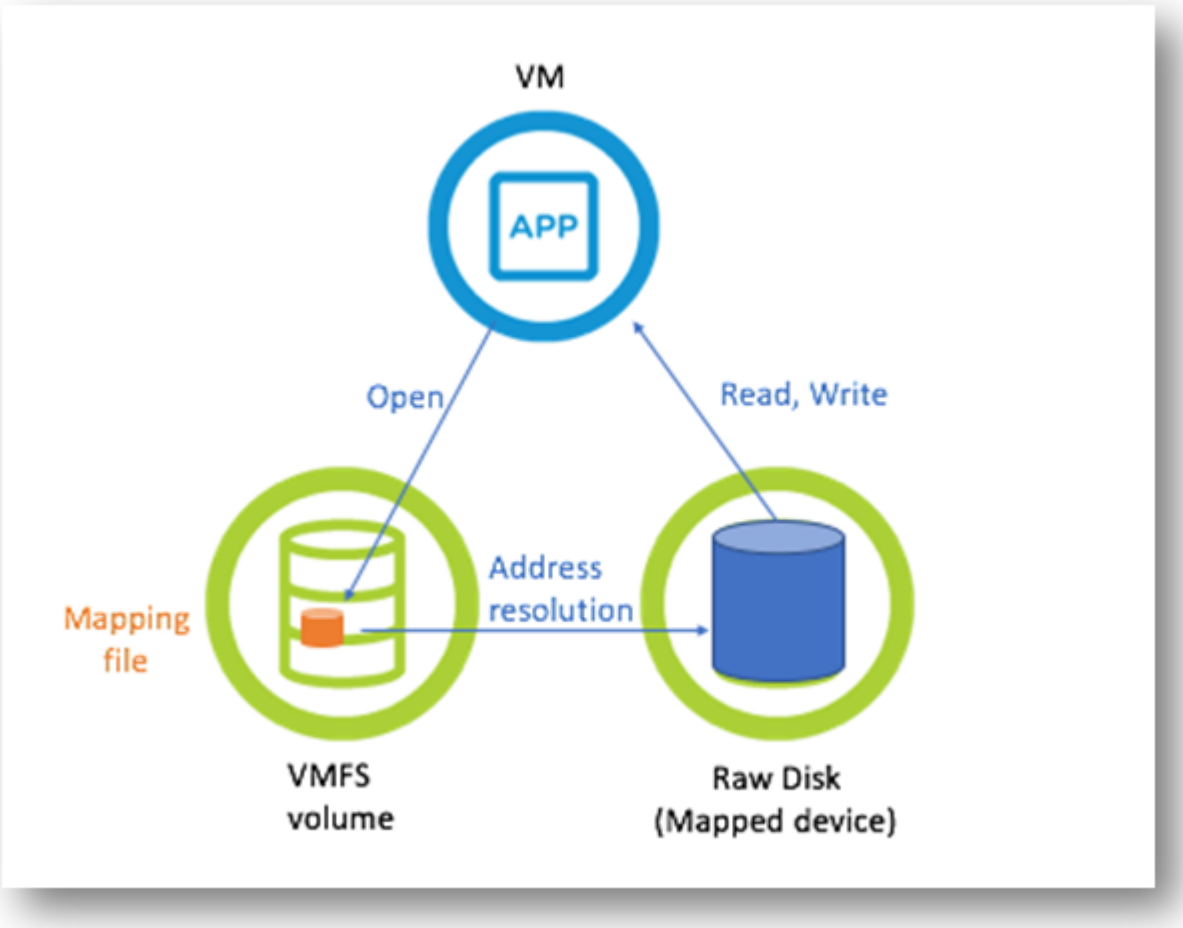

Figure 3 Raw Disk Mapping

An RDM is a symbolic link from a VMFS volume to a raw volume. The mapping makes volumes appear as files in a VMFS volume. The mapping file—not the raw volume—is referenced in the virtual machine configuration. The mapping file, in turn, contains a reference to the raw volume.

Using RDMs, you can do the following:

• Add them to virtual machines using the vSphere client.

• Use vMotion to migrate virtual machines.

• Use file system features such as distributed file locking, permissions and naming.

RDMs have the following two compatibility modes:

• **Virtual compatibility mode** enables a mapping to act exactly like a virtual disk file, including the use of array-based snapshots.

• **Physical compatibility mode** enables direct access of the SCSI device, for those applications needing lower level control.

vMotion, vSphere DRS and vSphere HA are all supported for RDMs that are in both physical and virtual compatibility modes. Storage vMotion is supported for virtual compatibility mode RDMs, but a side-effect is that the RDM is converted to a VMDK during the migration. Storage vMotion is not allowed on physical compatibility mode RDMs.

Why Use VMFS?

For most applications, VMFS is the clear choice. It provides the automated file system capabilities that make it easy to provision and manage storage for virtual machines running on a cluster of vSphere hosts. VMFS has an automated hierarchical file system structure with user-friendly file-naming access. It automates the subdirectory naming process to make administration more efficient in managing RDMs. It enables a higher disk utilization rate by facilitating the process of provisioning the virtual disks from a shared pool of clustered storage.

As you scale the number of vSphere hosts and the total capacity of shared storage, VMFS greatly simplifies the process. It also enables a larger pool of storage than might be addressed via RDMs. Because the number of LUNs that a given cluster of vSphere hosts can discover is currently capped at 512 in vSphere 6.5, you can reach this number rather quickly if mapping a set of LUNs to every virtual machine running on the vSphere host cluster. Using RDMs usually requires more frequent and varied dependence on the storage administration team, because each LUN must be sized for the needs of each specific virtual machine to which it is mapped.

With VMFS, however, you can carve out many smaller VMDKs for virtual machines from a single VMFS volume. This enables the partitioning of a larger VMFS volume—or a single LUN—into several smaller virtual disks, which facilitates a centralized management utility (vCenter) to be used as a control point. The control point resides at the vSphere host level, between the storage array and the virtual machine.

With RDMs, there is no way to break up the LUN and address it as anything more than a single disk for a given virtual machine. One example of this limitation is a case where a user provisioned several 500GB LUNs and wanted to test relative performance on a few virtual machines. The plan called for testing with 100GB virtual disks. With an RDM, the only choice was to address the entire 500GB RDM to the virtual machine and use only the first 100GB. This wasted the other 400GB of that LUN. Using VMFS with a 500GB volume, on the other hand, enabled the creation of five directly addressable virtual disks of 100GB each on the shared VMFS volume.

Even if performance is the main consideration, you can employ a single VMFS volume for each virtual machine, in much the same way as an RDM volume is isolated to a single virtual machine. (Used this way, the VMFS and the RDM volumes provide similar performance.) The bigger question is whether to isolate or consolidate, and that is not limited to use of RDMs for isolation and VMFS for consolidation.

Why Use RDMs?

This section specifies the use cases where RDM storage access is preferred over VMFS.

Even with all the advantages of VMFS, there still are some cases where it makes more sense to use RDM storage access. The following scenarios call for raw disk mapping:

- Migrating an existing application from a physical environment to virtualization
- Using Windows Server Failover Cluster (WSFC) for failover clustering in a virtual environment
- Implementing N-Port ID Virtualization (NPIV)
- Separating heavy I/O workloads from the shared pool of storage

At this point, it should be pointed out that VMware recommends avoiding RDMs if at all possible. Many newer versions of applications now allow for high availability without the need for failover clustering. Microsoft SQL Server AlwaysOn Availability Groups and Microsoft Exchange Database Availability Groups (DAG) are two such examples. However, VMware understands that, in certain cases, failover clustering is still desired, so we continue to support RDMs for this use-case.

RDM Scenario 1: Migrating an Existing Application to a Virtual Server

Figure 4 shows a typical migration from a physical server to a virtual one. Before migration, the application running on the physical server has two disks (LUNs) associated with it. One disk is for the OS and application files; a second disk is for the application data.

To begin, use the VMware vCenter Converter to build the virtual machine and to load the OS and application data into the new virtual machine and associated VMDK.

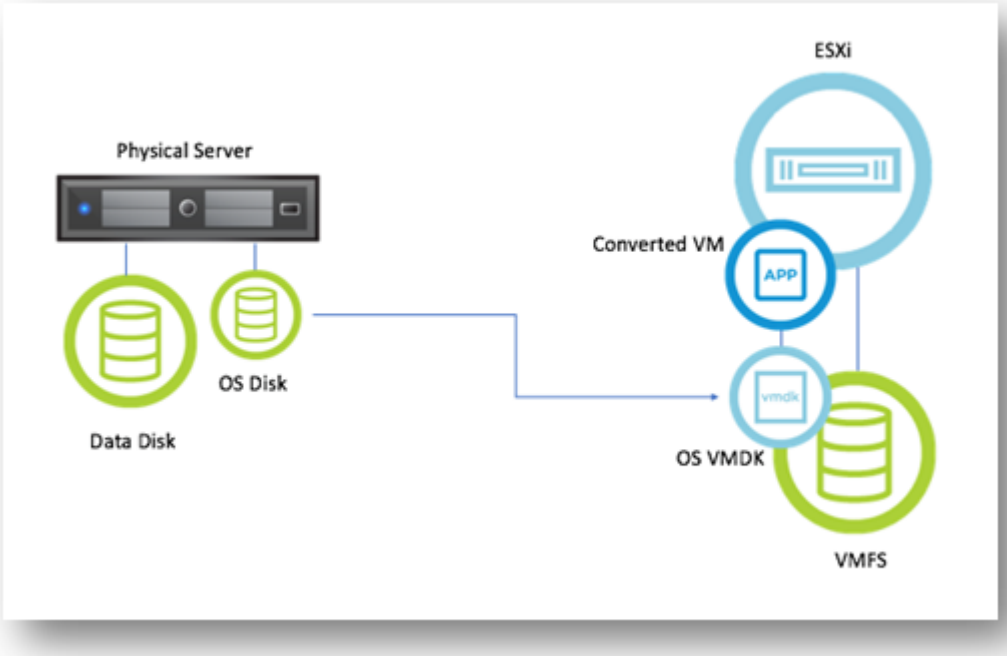

Figure 4. VMware vCenter Converter

Next, remove access to the data disk from the physical machine and make sure the disk is properly zoned and accessible from the vSphere host. Then create an RDM for the new virtual machine pointing to the data disk. This enables the contents of the existing data disk to be accessed just as they are, without the need to copy them to a new location.

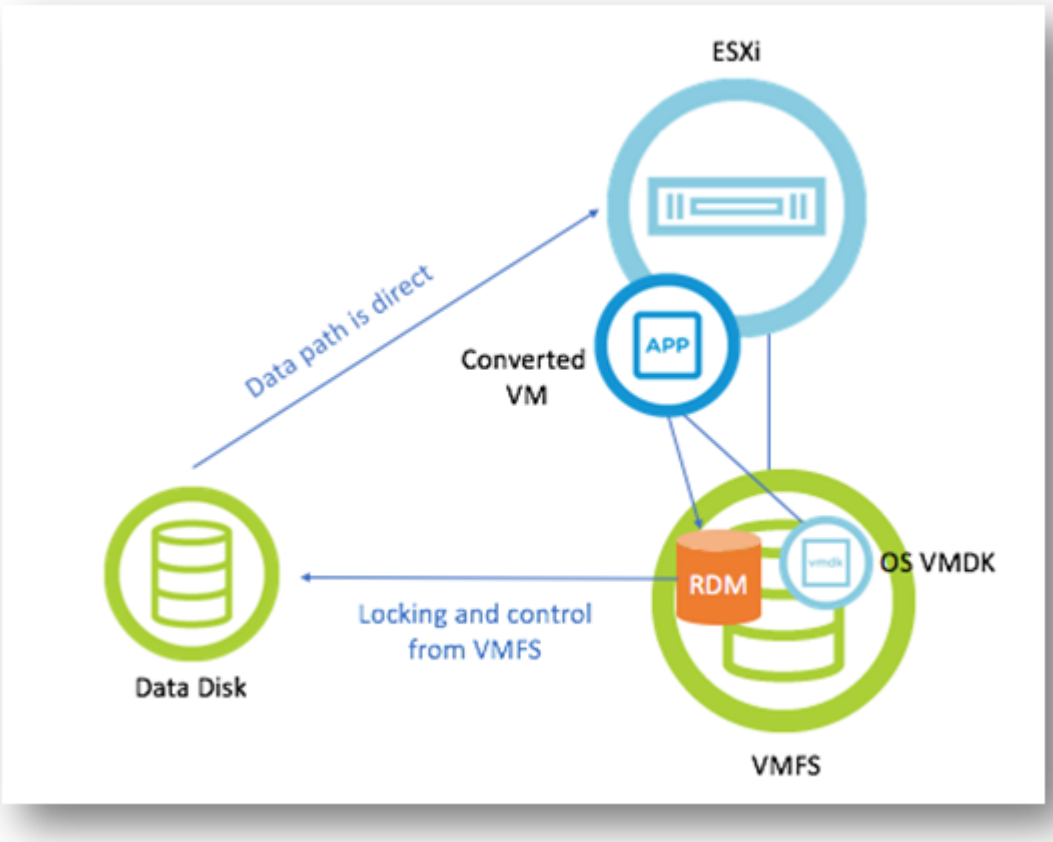

Figure 5. Raw Disk Map Creation

The path to the data disk located on the RDM is stored in the VMFS. Although VMFS provides security access control and locking, the data path to the RDM is direct access. As with virtual disks, VMFS controls access to make sure there is no simultaneous access to the data disk from other virtual machines. Because RDMs enable vMotion, the VMFS can transfer control of the RDM to the destination vSphere host when a virtual machine migrates, assuming that the device is also visible on the destination host.

RDM Scenario 2: Using Windows Server Failover Clustering in Virtual Environment

Another common use of RDMs is for Windows Server Failover Clustering (WSFC) configurations, although these are becoming less widespread now that Microsoft offers services such as MS Exchange Database Availability Groups and MS SQL Server AlwaysOn Availability Groups, which negate the requirement for shared quorum devices.

WSFC can use clusters in a single vSphere host (cluster in a box); clusters between virtual machines on separate vSphere hosts (cluster across boxes); or clusters across both physical and virtual machines. Each of these scenarios has different requirements for shared storage, which are summarized in the following table:

	CLUSTER IN A BOX	CLUSTER ACROSS BOXES	N+1 CLUSTERING
Virtual disks (VMDKs)	Yes	No	No
Pass-through RDM (physical compatibility mode)	No	Yes	Yes
Non-pass-through RDM (virtual compatibility mode)	Yes	Yes	No

Table 1. Summary of WSFC Shared Storage

Using WSFC in a virtual environment requires a disk for each node in which les specific to that node are stored. Other les require shared access for the quorum disk. Those disks must support native le system access, which requires the use of RDMs in physical compatibility mode. This is another example where RDMs provide a more flexible solution for the use of virtualization technology.

When and How to Use Disk Spanning

It is generally best to begin with a single LUN in a VMFS volume. To increase the size of that resource pool, you can provide additional capacity by either:

1) adding a new VMFS extent to the VMFS volume, or

2) increasing the size of the VMFS volume on an underlying LUN that has been expanded in the array (via a dynamic expansion within the storage array)

Adding a new extent to the existing VMFS volume will result in the existing VMFS volume's spanning across more than one LUN. However, until the initial capacity is filled, that additional allocation of capacity is not yet put to use. The VMFS does not stripe I/O across LUNs when more than one LUN is allocated to a given VMFS volume. Expanding the VMFS volume on an existing, larger LUN will also increase the size of the VMFS volume, but it should not be confused with spanning. However, the VMFS Volume Grow feature can be used to expand a VMFS volume that spans a few VMFS extents as well as one that spans multiple LUNs, provided there is space on those LUNs to expand the VMFS extents onto.

From a management perspective, it is preferable that a single large LUN with a single extent host your VMFS. Using multiple LUNs to back multiple extents of a VMFS volume entails presenting every LUN to each of the vSphere hosts sharing the datastore. Although multiple extents might have been required prior to the release of vSphere 5 and VMFS5 to produce VMFS volumes larger than 2TB, VMFS5 and later versions support single-extent volumes up to 64TB.

Gaining Additional Throughput and Storage Capacity

Additional capacity with disk spanning does not necessarily increase I/O throughput capacity for that VMFS volume. It does, however, result in increased storage capacity. If properly provisioned on the underlying storage array, the additional capacity can be allocated on LUNs other than the first LUN and will result in additional throughput capacity as well. It is very important to be certain you are adding LUNs of similar performance capability (data services and I/O density) when adding to an existing VMFS volume.

The current size limit for a VMFS6 extent is 64TB. In VMFS3, the extent size was 2TB. For large VMFS3 volumes, spanning was required to concatenate multiple 2TB extents. There is a limit of 32 extents in a VMFS volume; the size limit of any VMFS volume is 64TB. Spanning of multiple volumes (LUNs) was required to reach that upper limit and is needed for any VMFS3 volume greater than 2TB in size. This is not necessary for VMFS5 or later, as a single extent VMFS volume can be created with these newer versions.

Suggestions for Rescanning

In prior versions of vSphere, it was recommended that before adding a new VMFS extent to a VMFS volume, you make sure a re-scan of the SAN is executed for all nodes in the cluster that share the common pool of storage. However, in more recent versions of vSphere, there is an automatic rescan that is triggered when the target detects a new LUN, so that each vSphere host updates its shared storage information when a change is made on that shared storage resource. This auto re-scan is the default setting in vSphere and is configured to occur every 300 seconds.

A closer look at VMFS6 enhancements

This section focusses on the enhancements and improvements of VMFS6 in vSphere 6.5.

With the release of vSphere 6.5, a new version of VMFS was released. VMFS6 has a number of new enhancements and improvements, which are now discussed.

Small File Blocks and Large File Blocks

VMFS6 has two new "internal" block sizes, a small file block (SFB) and a large file block (LFB). This is not to be confused with the actual block size used on VMFS6, which continues to be 1MB. The SFB and LFB are used to back files created on VMFS. The SFB is set to 1MB and the LFB is set to 512MB.

When thin files are created on VMFS6, these are backed by SFBs initially. Thick files created on VMFS6 are backed by LFBs where possible. For example, if a VMDK is Lazy Zeroed Thick (LZT) or Eager Zeroed Thick (EZT), this VMDK will always use LFBs as long they are available for allocation. If there is a portion of the thick file which "overflows" an LFB, then the remaining portion is backed by SFBs.

This enhancement to VMFS6 should result in a much faster file creation time, especially for thick files. This is also going to improve the power-on time for VMs. When a VM is powered on, a swap file is created. This swap file is always "thick", so it should be created much faster if the swap file is backed by LFB(s).

Dynamic System Resource Files

System resource files (.fdc.sf, .pbc.sf, .sbc.sf, .jbc.sf) are now extended **dynamically** for VMFS-6. The idea here is that VMFS6 volumes can be created that are small in capacity, so do not consume a huge amount of overhead for metadata. Another reason for this approach is to not place a cap the maximum capacity of a volume with its initial formatting size, but instead allow it to grow dynamically over time. If the filesystem exhausts any resources, the respective system resource file is extended to create additional resources.

VMFS-6 can now support **millions** of files (as long as volume has free space).

New Journal System Resource File

VMFS6 is a distributed, journaling filesystem. Before making changes to any of the files on the volume, a journal entry is created, so that if the operation fails for any reason (e.g. host crash), the consistency of the data can be validated. Prior to VMFS-6, the journal entries used regular file blocks. This has led to some operational issues, for example, in the case of a full filesystem. The easiest way to resolve such a condition would be to remove some files. However, this required the creation of a journal entry on VMFS. Since there are no file block available to create the journal entry, customers were required to take additional manual steps to free some space.

With VMFS6, there is a new distinct journal resource file (.jbc.sf). This means that in situations such as the one described previously, having a separate journal resource file will avoid these sorts of issues going forward. Note that the journal resource file can also be dynamically extended, along with the other resources files, on VMFS6.

VM-based Block Allocation Affinity

VMFS uses Resource Clusters to group sets of resources such as file descriptors, sub-blocks, file blocks, etc. These were historically allocated on a per ESXi host basis. Therefore, as an ESXi host created more and more VMs and files, it consumed more and more resources in the resource cluster. This worked well from a contention standpoint since the same ESXi host owned all of the VMs/files in the same resource cluster. However, if a VM was vMotion to another ESXi host, either manually or via DRS load balancing, then there is now two ESXi hosts sharing the same resource cluster. There may now be resource contention if these two ESXi hosts needed to do an operation on their respective VMs simultaneously. The possibility of contention rises as the number of ESXi hosts which shares the VMFS grows.

This situation is alleviated somewhat in VMFS6. VMFS6 introduces a new VM-based block allocation affinity model. When a new file is created on VMFS6, the existing resource clusters are examined. If a resource cluster has any active users, meaning there is already a file using this resource cluster, that resource cluster is not picked for the allocation of resources for this new file. Instead, the block allocation logic now looks for inactive resource clusters that are not in use by any other files.

Essentially, what we have done is moved from a host affinity model for resource clusters to a **VM affinity model for resource clusters**. This should also mean that even in the event of a vMotion of a VM to a different host from where it was created, VMFS-6 should not have multiple hosts contending for resources within the same resource cluster, since we are aiming to have the resource cluster only used by a single VM/file.

Of course, if all of the resource clusters become active/have files consuming resources, then we look for resource clusters with the

least number of users (we are using a simple reference count mechanism to track this). This should also help keep the number of lock contention issues to a minimum, if they arise at all, on VMFS-6.

Parallelism/Concurrent Improvements

Some of the biggest enhancements made to VMFS6 are in the areas of parallelism and concurrent operations. Previous versions of VMFS only allowed single transactions per host on a given filesystem. In VMFS6, support for multiple, concurrent transactions per host are now supported. This is most noticeable in device discovery and filesystem probing. The improvements are significant for fail-over/DR events, where device discovery and resignaturing play a major part of the process. Site Recovery Manager customers should especially benefit from this feature.

Upgrading to VMFS6

It should be noted that there is no direct in-place upgrade from previous versions of VMFS to VMFS6. There are simply too many changes to the underlying filesystem to make upgrading possible. Customers can continue to run earlier versions of VMFS with vSphere 6.5 while they determine how to move to VMFS6. Migration techniques, such as Storage vMotion, can be used to migrate VMs from earlier versions of VMFS to VMFS6.

Conclusion

This section summarizes VMFS, its functionalities and capabilities.

VMware vSphere VMFS provides the foundation for virtualization to span multiple systems. It enables optimized I/O handling of multiple virtual machines sharing a common clustered pool of storage. It also provides more efficient storage management for both the initial provisioning and the ongoing management of the shared storage resources. Leveraging VMware vSphere Thin Provisioning can dramatically increase the utilization rates of your storage resources in a virtual environment. VMFS insight and monitoring are built into VMware vCenter as an intelligent interface that enables efficiency and easy management.

VMFS is the leading cluster le system for virtualization. It is unique in the market today and provides the capabilities that empower virtual machines to scale beyond the limits of a single server, without downtime. VMFS enables VMware vSphere Storage vMotion to migrate virtual machines from one storage resource to another without downtime. It also improves resource utilization and lowers the management overhead for storage administration in a virtual environment. VMFS is a key reason why VMware vSphere is today’s leading virtualization solution, one that is more scalable and reliable than any other offering currently available.

About the Author

This section covers the details about the author of this guide.

Cormac Hogan is a Director and Chief Technologist in the Storage and Availability Business Unit at VMware. Cormac has written a book called “Essential vSAN” available from VMware Press, as well as a number of storage-related white papers. He has given numerous presentations on storage best practices and new features and is a regular presenter at VMworld and VMware User Group meetings.

Glossary

This section provides abbreviations of the short forms used in the guide.

CFS – Cluster File system.

CVM – Cluster Volume Manager.

DAG – Database Availability Group. Highly available framework built into Microsoft products.

Datastore – A formatted file system that a vSphere host mounts and uses as a shared storage pool. It is built upon either a VMFS volume for block-based storage or a mount point for NFS shared storage.

DRS – VMware vSphere Distributed Resource Scheduler.

vSphere host – The hardware on which the vSphere software is loaded and running.

Extent – Part of a larger allocation of a resource, or an appendage to an existing resource.

Fibre Channel (FC) – An ANSI-standard, gigabit-speed network technology used to build storage area networks and transmit data. FC components include HBAs, switches and cabling.

HA – High availability.

iSCSI – Internet small computer serial interface.

LUN – Logical unit number—what the server detects as a single disk resource.

MSCS – Microsoft Cluster Service.

NFS – Network File system, provided by network attached storage (NAS).

RAID – Redundant array of independent disks.

RDM – Raw disk map.

SAN – Storage area network.

SCSI – Small computer serial interface.

Storage vMotion – A live migration of a virtual machine on disk files from one datastore to another. The virtual machine home changes location, but the virtual machine remains running on the same vSphere host.

VCB – VMware consolidated backup.

vCenter – Centralized management utility.

VMware vSphere Client – Virtual infrastructure client, a management access point.

VMDK – Virtual machine disk, a file in the VMFS that backs the storage object for the disk for a virtual machine.

VMFS3 – Version 3 of the Virtual Machine File System.

VMFS5 – Version 5 of the Virtual Machine File System.

VMFS6- Version 6 of the Virtual Machine File System.

VMFS extent – The portion of a VMFS volume that resides on a single LUN.

VMFS volume – The aggregation of block-based storage that backs a datastore. Normally a single LUN, it can be up to 32 LUNs when a VMFS volume spans across multiple VMFS extents.

VMkernel – Core management component of the vSphere host system.

vMotion - A means by which the virtual machine can be moved from one vSphere host to another without any downtime (the virtual machine remains on the same datastore but runs on a new vSphere host).

WSFC – Windows Server Failover Clusters from Microsoft.