

Storage Design Options for VMware® Virtual Desktop Infrastructure



Contents

- Overview 1**
- Base Image Considerations..... 1**
 - Operating Systems Settings..... 1
 - Application Sets..... 1
 - User Data 2
 - Configuration Concerns..... 3
- Storage Protocols 3**
- Storage Technology Options..... 5**
 - Standard Storage 5
 - Thin Provisioning 5
 - Virtual Machine White Space 6
 - Thin Provisioning Individual Virtual Machines 6
 - Thin Provisioning Entire Datastores..... 6
 - Thin Provisioning Summary 7
 - Single Instancing (deduplication)..... 7
 - Network-booted Operating Systems..... 8

Overview

Companies planning to deploy VDI face decisions about the use of both local and shared storage, and in the case of shared storage solutions, choosing between differing technologies available in today's market.

Selecting the appropriate storage model is important for both performance and costs reasons. Certain solutions require less overhead than others, as do different implementations of the same technology. Costs can vary greatly depending on which storage options are chosen. Fortunately organizations can leverage a myriad of best practices to help drive these costs down, while improving performance.

This paper provides information on technical concepts related to storage implementations in a VMware® Virtual Desktop Infrastructure (VDI) environment.

Base Image Considerations

Most VDI implementations deploy hosted desktops from a small number of base, or *golden master*, images. Since these base images are meant to be reused, it makes sense to take special care when creating them. Image size can be optimized through adjusting operating system (OS) settings, application sets, and user data. Optimizing OS settings has the additional benefit of improving overall system performance.

Operating Systems Settings

Reducing the Operating System's footprint, regardless of the underlying storage technology, is a VDI architecture best practice. The operating system, once virtualized, no longer needs many of the components that are found in a traditional desktop PC implementation. There are various utilities available which can be used to streamline the base operating system by removing unnecessary features.

While the primary goal is to reduce the image size, increased performance is an important side effect: removing unnecessary components reduces the overhead associated with unneeded processes and services within a virtual desktop.

Application Sets

In a VDI deployment, the manner in which applications are deployed directly affects the size of the final desktop image. Historically, in a traditional desktop environment applications are provided via a number of different mechanisms, including: installation directly to the local hard disk, streamed to the desktop, or deployment centrally through a server-based computing (SBC) model.

While the basic deployment methodologies remain unchanged from a traditional desktop environment, VDI opens up new opportunities for application management. For instance, with VDI multiple users can leverage a single generic desktop image which includes the base operating system as well as the necessary applications. Administrators create pools of desktops based on this single golden master image, When a user logs in, a new desktop based on the base image, with hotfixes, application updates and additions, is assigned from the pool.

Managing a single base template rather than multiple individual desktop images, each with its own application set, reduces the overall complexity of application deployment. This base template can be cloned in different ways (e.g., at the virtual disk level, at the datastore or volume level, etc.) to provide desktops with varying storage requirements for different types of users.

It is important to note that when using storage-based or virtualization-based snapshot technologies, the addition of locally-installed applications across a large number virtual machines may result in higher storage requirements and decreased performance. For this reason it's recommended that only individual desktops based on a customized image have their applications deployed directly to the virtual machine.

While traditional server based computing (SBC) models have met with limited success as a means to deploying centralized desktops, centralized application presentation can be used to reduce the overhead associated with providing applications to virtualized desktops.

SBC not only reduces storage requirements on a per virtual machine basis, but can also increase the number of virtual desktop workloads per VDI server since the application processing overhead is off-loaded to the server hosting the applications. SBC also allows for flexible load management and easier updating of complex front-line business applications as well as global application access. Regardless of VDI design, SBC can be leveraged for the overall application distribution method.

User Data

A best practice related to user-specific information on the virtual desktop is to redirect as much user data as possible to network-based file shares. In order to gain the maximum benefit of shared storage technologies, the individual virtual desktop should be thought of as disposable. While persistent (1-to-1, typically dedicated) virtual desktops are possible, some of the key benefits of VDI come from the ability to update the base image easily and allow the VDI infrastructure to distribute the changes as an entirely new virtual desktop. Updates to a VDI desktop are performed on the golden master image, and new machines are provisioned as needed.

With persistent desktops and pools, it is possible to store user data locally. However, it is still recommended to direct data to centralized file storage. By separating the desktop image from the data, administrators can easily update the virtual desktop image more easily.

In a Windows infrastructure, there are several key guidelines for user data that should be considered. While, historically, roaming profiles often had issues, with proper design they can in fact be stable and be successfully leveraged in a VDI environment. The key to the successful use of roaming profiles is to keep the profile as small as possible. Using folder redirection, especially beyond the defaults that are found in standard group policy objects, roaming profiles can be stripped down to the bare minimum.

Key folder redirections that can be used to reduce profile size include:

- Application Data
- My Documents
- My Pictures
- My Music
- Desktop
- Favorites
- Cookies
- Templates

Regardless of whether persistent or non-persistent pools are used, local data should not exist on the VM image. If an organization requires local data storage, it should be limited.

Locking down the desktop, including measures to prevent users from creating folders on the virtual machine's root drive, is also important. Many of the policy settings and guidelines that exist for traditional server-based computing are leveraged in the VDI design process for security lock-down.

A final benefit of redirecting all data is that only the user data and base template needs to be archived or backed up; backups of the VDI desktop are unnecessary.

Configuration Concerns

When building a virtual desktop image, make sure the virtual machine does not consume unnecessary computing resources.

The following items can safely be removed or modified to increase performance and scalability:

- Turn off graphical screensavers, use a basic blank windows logon screensaver only
- Disable offline files and folders
- Disable all GUI enhancements, such as themes, etc., except for font smoothing
- Disable any COM ports
- Delete locally cached roaming profile at logoff

Additional information for preparing the base image can be found in the Windows XP deployment guide for VDI: http://www.vmware.com/files/pdf/XP_guide_vdi.pdf.

It is important to consider how storage is being accessed by the applications executing on the hosted desktop in a VDI implementation. For example, if multiple virtual machines are sharing a base image, and they all run virus scan at the same time, performance can degrade dramatically because the virtual machines would all be attempting to use the same I/O path at once, excessive simultaneous access to storage resources will result in a performance penalty.

Depending on the plan for storage reduction, the swap file of the virtual machine should be located separately from the snapshot file. In certain cases, leveraging snapshot technology for storage savings can cause the swap file usage to increase snapshot size to the point where it can degrade performance. An example of this could be the use of a virtual desktop in an environment where there is intensive disk activity and extremely active use of memory (particularly where all memory is used and thus page file activity is increased) Again, this is only applicable when array-based snapshots are being used to save on shared storage usage.

Storage Protocols

Historically, many virtualization implementations used either iSCSI or fibre channel (FC) protocols in their architecture. While both iSCSI and FC are certainly capable of providing excellent levels of service to the virtualization layer, VDI brings out key flexibility advantages around the NFS protocol.

It is important to note that most of the storage technologies in this paper are neither protocol-specific nor vendor specific. NFS is mentioned in detail here because of the typical stigma it has had in the past, and because of the potential advantages it has for VDI.

Storage, not bandwidth, is typically the bottleneck in a virtual environment. Response time and I/O are what matter most, and the reduced latency of NFS can often be more suitable than comparable fibre channel or iSCSI solutions, given the same disk subsystem.

It is also important to consider the queue depth of the storage architecture, something that NFS can avoid. In larger environments with many hosts, excessive queue depth can cause serious

performance issues. This degradation in performance occurs because the host's physical adapter queue depth is typically oversubscribed in today's standard-sized ESX server. As hosts and clusters scale out, this oversubscription may become pronounced and performance suffers.

With a basic configuration, fibre channel will outperform NFS for I/O with a single VM in a single datastore. However, as ESX servers are scaled out, NFS can out-perform FC when the I/O of multiple virtual machines hit a shared datastore.

It is important to keep in mind that, as virtualization hosts in the datacenter grow exponentially, so do LUNs, Zones, HBAs, VMFS volumes, etc. With VDI, the added flexibility of NFS is very important.

NFS can often scale better because of the simplicity it provides. Since with VDI, many more virtual machines are incorporated in the design, the flexibility advantages of NFS over other storage technologies become clear.

In the past, the key perceived deficiency with NFS was assumed poor performance. In reality, NFS performance virtualized environments can be quite impressive, provided NFS is being used from an enterprise grade array and disk subsystems.

NFS Advantages

- Thinly provisioned disks
- The ability to expand/decrease the NFS volume in the storage subsystem with the change being immediately visible in the VDI environment
- No RDMs to manage
- No storage subsystem configuration - for example fiber switches, HBAs, zoning, LUNs, etc.
- The ability to clone a single VM or multiple VMs within the datastore - from the storage subsystem - without the full use of storage.
- Backing up the files that constitute a virtual machine becomes a very simple matter because the storage subsystem can see the virtual disk file directly.
- No single disk I/O queue exists with NFS, as it does with a VMFS volume on iSCSI or Fibre. Performance is strictly dependent on the size of the pipe and the disk array hardware.
- Disaster recovery can be simplified - failover to mirrored volumes can be done simply and in minutes. iSCSI and Fibre Channel VMFS volume require additional configuration steps.

NFS Disadvantages

- No support for Microsoft Clustering Services (while not necessarily important to desktops, this can be important to overall shared storage choice, assuming desktops will share the same storage with servers)
- Risk of poor performance if not implemented correctly.

NFS Configuration Tips

- ESX's link aggregation is supported only on a single switch (or stacked) - not across separate trunked switches.
- Allocate a maximum of 40-50 .vmdk's per storage volume.
- Set the vSwitch routing to IP Hash routing.

- Set the physical network switch to use SRC-DEST-IP balancing.
- Create a multi-mode ether channel group on the array and use IP aliasing with multiple IP addresses. (Switch must support 802.3ad)
- Create several datastores by using the same mount point, but use the various IP addresses to mount from (IP aliasing).
- Leverage multiple IP address with several datastores from a single ESX server to spread the NFS traffic across multiple uplinks from a single vSwitch.

Storage Technology Options

There are five core options today for storage under VDI:

- Standard shared storage, similar to basic virtual server deployments
- Storage-level thin provisioning, either at the virtual machine or volume level
- Single-instancing, also known as data deduplication
- Boot-from-network of the Operating System
- Local Storage

Proper use of one or more of these technologies should reduce the overall storage requirements of most VDI environments. Which technology to use in a given environment will vary depending on organizational, availability, and performance requirements.

Standard Storage

VDI with *standard shared storage* resembles the storage usage of typical virtual servers. A logical unit number (LUN) is created from a volume on shared storage and presented to the virtualization host, which uses the LUN to store fully provisioned individual virtual machines.

VDI workloads using *standard full storage* volumes are deployed to larger LUN sizes than those deployed for virtual servers. Desktop workloads typically have much lower I/O needs than server workloads, but desktops often have large footprints relative to their I/O needs, especially when application sets are installed within the template. A server LUN might be 300GB, while a desktop LUN might be up to 500GB. However, smaller LUN sizes may be necessary for snapshots with VDI because the use of snapshots can affect storage subsystem performance.

Standard shared storage for VDI should only be used for certain types of desktops with unique needs, such as a one-off application set for a unique user, information technology staff, or perhaps in small VDI implementations that require only a limited number of desktops.

Thin Provisioning

Thin provisioning is a term used to describe several reduction methods:

- Reduce the white space in the virtual machine disk file.
- Reduce the space used by identical virtual machines by cloning the virtual disk file within the same volume so that writes go to a small snapshot, with a number of virtual machines sharing a base image.

- Reduce the space used by a set of cloned virtual machines by thin cloning the entire volume in the shared storage device. Each volume itself would have a virtual clone, or snapshot, of the base volume.

Thin provisioning starts with some type of standard shared storage, then either the individual virtual machines or the entire volume containing the virtual desktops can be cloned. The virtual clone is made at the storage layer, so that a virtual machine's disk writes go into some type of snapshot file, or else block-level writes are tracked by the shared storage for the individual machine/volume clones.

Thin provisioning methods can be utilized in many areas, and reduction technologies can be layered, or stacked, but their impact on performance, when used separately or in combination, should be evaluated. The layered approach looks great on paper, but because of the look-up tables used in the storage subsystem, the impact may be undesirable. Storage savings requirements must be balanced with the overhead a given solution brings to the design.

Virtual Machine White Space

Reduction of white space on the virtual disk file refers to the removal of unused space in the virtual disk file. The storage used by a virtual machine is based on the actual amount of data in the disk file. For example, using ESX, a .vmdk on an NFS mount is provisioned as thin by default, so, a 40GB .vmdk with 24GB of data would use 24GB of storage when provisioned as thin.

Thin Provisioning Individual Virtual Machines

Base .vmdk virtual disk sharing at the hypervisor level is an option that has been technically possible for some time; This approach provides the ability to leverage a base .vmdk file with multiple snapshots, and does not require manual configuration and/or customization of .vmx file. Leveraging snapshots an result in an excellent reduction in storage without the aid of any other storage reduction techniques.

Sharing the base image within the storage subsystem is another exciting idea. As with sharing the .vmdk within a VMFS volume at the hypervisor, there are storage devices are capable of sharing a base .vmdk image file *at the storage layer*. Sharing at the storage layer instead of using a snapshot within a VMFS volume, offers much greater the scalability. A ratio of 1:20 or higher would not be out of the question.

File Layer Thin Provisioning Tips

- The guest OS must be in a state to be duplicated by using Microsoft's Sysprep utility.
- Pay close attention to how individual virtual machines write data. For example, do not install large applications in the individual virtual machine clones, go back to the base image and update it instead.
- Understand Windows pagefile usage. Some virtual machine usage scenarios can cause abnormal snapshots and performance issues.

Thin Provisioning Entire Datastores

Leveraging thin provisioning at the datastore level opens up additional possibilities for storage reduction. The entire datastore, rather than the individual virtual machine, is cloned and represented to the ESX cluster as a different datastore. The clone, however, does not actually use twice the storage, it is *virtually cloned* in the storage subsystem of the shared storage device.

The idea behind this is to virtually clone an original base golden datastore so a datastore with multiple (20 or more) virtual machines can be virtually cloned several times over. Each virtual machine could then be powered on and used individually, but still use a common storage footprint across a 100+ virtual machine delta.

This type of thin provisioning is heavily dependent on the storage manufacturer's snapshot technology; some storage vendors provide better performance than others.

Datastore Layer Thin Provisioning Tips

- For VMFS datastores, enable the *enableresignature* option in VirtualCenter to use a cloned datastore.
- As with file layer provisioning, guest operating systems in the virtual machines in the base datastore must be in a state to be duplicated. Leverage sysprep, and have the VM ready to come with and make modifications on Power On, including a new hostname, SID, domain membership, networking information.
- Performance will not be linear. Multiple base image datastores may be necessary. Scalability of clones per base datastore will greatly depend on the storage vendor's snapshot method and efficiency.
- Be cautious of the LUN/volume limits of VMFS datastores and NFS mounts.

Thin Provisioning Summary

Expect to see several methods of thin provisioning working together to compress VDI storage. In the future, it will be possible to leverage image-level thin provisioning with datastore virtual cloning. Scalability of these solutions is very promising especially with NFS, however, there is little accurate field data available for real-world scenarios under VDI.

Single Instancing (deduplication)

The concept of single instancing, or deduplication, of shared storage data is quite simple: The system searches the shared storage device for duplicate data, and reduce the actual amount of physical disk necessary by matching up data that is identical. Deduplication was born in the database world where the administrators created the term to search for duplicate records in merged databases. For the discussion of shared storage, deduplication is the algorithm used to find and delete duplicate data objects: files, chunks, or blocks. The original pointers in the storage system are modified so that the object can still be found, but the physical location on disk is shared with other pointers. If the data object is written to, the write goes to a new physical location and no longer shares a pointer.

There are varying approaches to deduplication, as well as quite a bit of misinformation about what deduplication offers front-line storage.

Regardless of method, the deduplication is based on two elements: *hashes* and *indexing*.

A *hash* is the unique digital fingerprint that each object is given. The hash value is generated by a formula under which it is unlikely that the same hash value will be used twice. However, it is important to note that it is *possible* to have the same hash value between two objects. Some in-line systems use only the basic hash value, which can cause data corruption. Any system lacking a secondary check for duplicate hash value could be risky to the VDI deployment.

Indexing contains either hash catalogs or lookup tables.

Hash catalogs are used by the storage subsystem to identify duplicates without having to be in the middle of the actual reads and writes to disk. This allows indexing to function out of the way of native disk usage.

A *lookup table* is used to extend a hash catalog to file systems that do not support multiple block references. The lookup table can be used in the middle of the system I/O and the native file system. This has a disadvantage, however, as the lookup table can then become a point of failure.

In the backup storage market, deduplication is an obvious strategy to adopt: the potential for duplicate data is huge. Ratios of 20:1 and higher are common. Identical data in backups is even more pronounced, and the speed at which the deduplication must take place is often much slower than the needs of front-line virtual machine workloads.

There are two methods of deduplication, in-line (on-the-fly) deduplication and post-write deduplication. Which to choose depends greatly on the type of storage utilization. Post-write deduplication is the primary method to be considered as a part of a VDI architecture. In-line deduplication is not yet fast enough to provide appropriate speed for a larger infrastructure, and is designed primarily for backup implementations.

In-line deduplication, deployed today for backups and archiving, deduplication keeps track of object records virtually and writes only the non-duplicate blocks to a (typically proprietary) back-end file system. It requires two file systems, one on the front and one on the back. It must search out duplicate objects before writing the data to disk, and must accomplish this *very* quickly. As processor computing power increases and solid state disk costs decreases, in-line deduplication will become an increasingly viable option for front-line storage. There is quite a bit of effort in this space today, but in-line deduplication still lacks the maturity needed for VDI storage.

With post-write deduplication, the deduplication is performed after the data is written to disk. This allows the deduplication process to happen when system resources are free, and it does not interfere with the front-line storage speeds. However, it does have the disadvantage of needing sufficient space to write all the data first and then consolidate it later, and this temporarily used capacity must be planned into the storage architecture.

Network-booted Operating Systems

Booting the operating system directly onto a virtual or physical device across the network is a relatively new concept. Only in the last few years has the technology been made available as an alternative to a standard OS install base.

This technology allows the OS to be booted over the network from either a 1:1 image or a shared image. The OS's disk drive is an image file on a remote server. The virtual machine typically would use PXE to bootstrap to the imaging server, which would then provide the correct virtual disk to the booting machine across the network.

The protocol used for this technology is very similar to iSCSI (originally local SCSI just wrapped in a TCP payload) but designed for use on the network.

Streaming adds an interesting consideration when the imaging server is used to serve up the virtual disk as a read-only image, with a private write disk for each virtual machine instance. This write cache holds everything that been changed since the computer has booted up. The cache can be stored in various places, including the virtual machine's RAM, the local hard disk, or as a file out on a network file server. When the virtual machine powers off, the changes disappear unless the image is set to keep the write image storage on a file server.

It is also possible to have a private one-to-one mapping; however, the storage needs for the individual disk files still remain, but now the storage would be on the actual imaging server.

Advantages

- SAN Storage space savings – each VM needs only the smallest of .vmdk's to appear to have a local disk to function
- Deployment of new machines is nearly instantaneous.
- Flexibility – very easy to change what the virtual machine is booting and seeing, and very easy to update one base image that deploys lots of individual VM in a pool.

Disadvantages

- No work off-line capability today
- Only supports non-persistent pools by default
- Heavy network traffic
- Increased server hardware requirements
- Scaling issues per imaging server in larger deployments
- Private images reside in one place – if the imaging server fails those desktops would not be accessible thereby potentially failing the desktop VM SLA.

A note of caution: Where private unique images are streamed from a single imaging host, outage could occur with the loss of that host.

This white paper was written by Jason Campagna, VCP, CCIA, MCSE, Chief Technical Architect of Server Centric Consulting.

Server Centric Consulting is a VMware Authorized Consultancy headquartered in St. Louis, MO. This unique 'all technical/no sales company provides architecture, design, and engineering services to the Fortune 1000, Government, Finance, Healthcare, and Education market segments in the areas of Virtualization, Secure Access, and Optimization.

Certified in all major virtualization technologies, Server Centric is an authorized custom engagement and Jumpstart sub-contract partner for VMware, CDW, Dell, and is available for contract through all VMware resellers. Call toll Free: 888 747-4700 or write vdi@servercentric.com.



VMware, Inc. 3401 Hillview Ave Palo Alto CA 94304 USA Tel 877-486-9273 Fax 650-427-5001 www.vmware.com
Copyright © 2008 VMware, Inc. All rights reserved. Protected by one or more of U.S. Patent Nos.
6,961,806, 6,961,941, 6,880,022, 6,397,242, 6,496,847, 6,704,925, 6,496,847, 6,711,672, 6,725,289,
6,735,601, 6,785,886, 6,789,156, 6,795,966, 6,944,699, 7,069,413, 7,082,598, 7,089,377, 7,111,086,
7,111,145, 7,117,481, 7,149,843, 7,155,558, 7,222,221, 7,260,815, 7,260,820, 7,268,683, 7,275,136,
7,277,998, 7,277,999, 7,278,030, 7,281,102, 7,290,253; patents pending.

VMware Part Number: WP-061-SLN-01-01

