# VMware Horizon 6 Storage Considerations

TECHNICAL WHITE PAPER

## Table of Contents

# Overview

This document addresses the challenges associated with end-user computing workloads in a virtualized environment and suggests design considerations for managing them. It focuses on performance, capacity, and operational considerations of the storage subsystem because storage is the foundation of any virtual desktop infrastructure (VDI) implementation. Where possible, it offers multiple solutions to common design choices faced by IT architects tasked with designing and implementing a VMware Horizon storage strategy.

# Typical Storage Considerations

Over the years, many end-user computing environments were designed, engineered, and built without proper consideration for specific storage requirements. Some were built on existing shared storage platform offerings. Others simply had their storage capacity increased without an examination of throughput and performance. These oversights prevented some VDI projects from delivering on the promises of virtualization.

For success in design, operation, and scale, IT must be at least as diligent in the initial discovery and design phases as in deployment and testing. It is essential to have a strong methodology and a plan to adapt or refine certain elements when technology changes. This document aims to provide guidance for the nuances of storage.

Operating systems are designed without consideration for virtualization technologies or their storage subsystems. This applies to all versions of the Windows operating system, both desktop and server, which are designed to interact with a locally connected magnetic disk resource.

The operating system expects at least one local hard disk to be dedicated to each single instance, giving the OS complete control from the device driver upward with respect to the reading, writing, caching, arrangement, and optimization of the file system components on the disk. When installing the operating system into a virtual machine running on a hypervisor, particularly when running several virtual machines simultaneously on that hypervisor, the IT architect needs to be aware of factors that affect how the operating system works.

# VMware Horizon Architecture

Figure 1 presents a logical overview of a validated VMware Horizon® 6 design. The design includes VMware Horizon with View, VMware Workspace™ Portal, and VMware Mirage™, along with the recommended supporting infrastructure. These components work in concert to aggregate identity, access, virtual desktops, applications, and image management in a complete architecture.

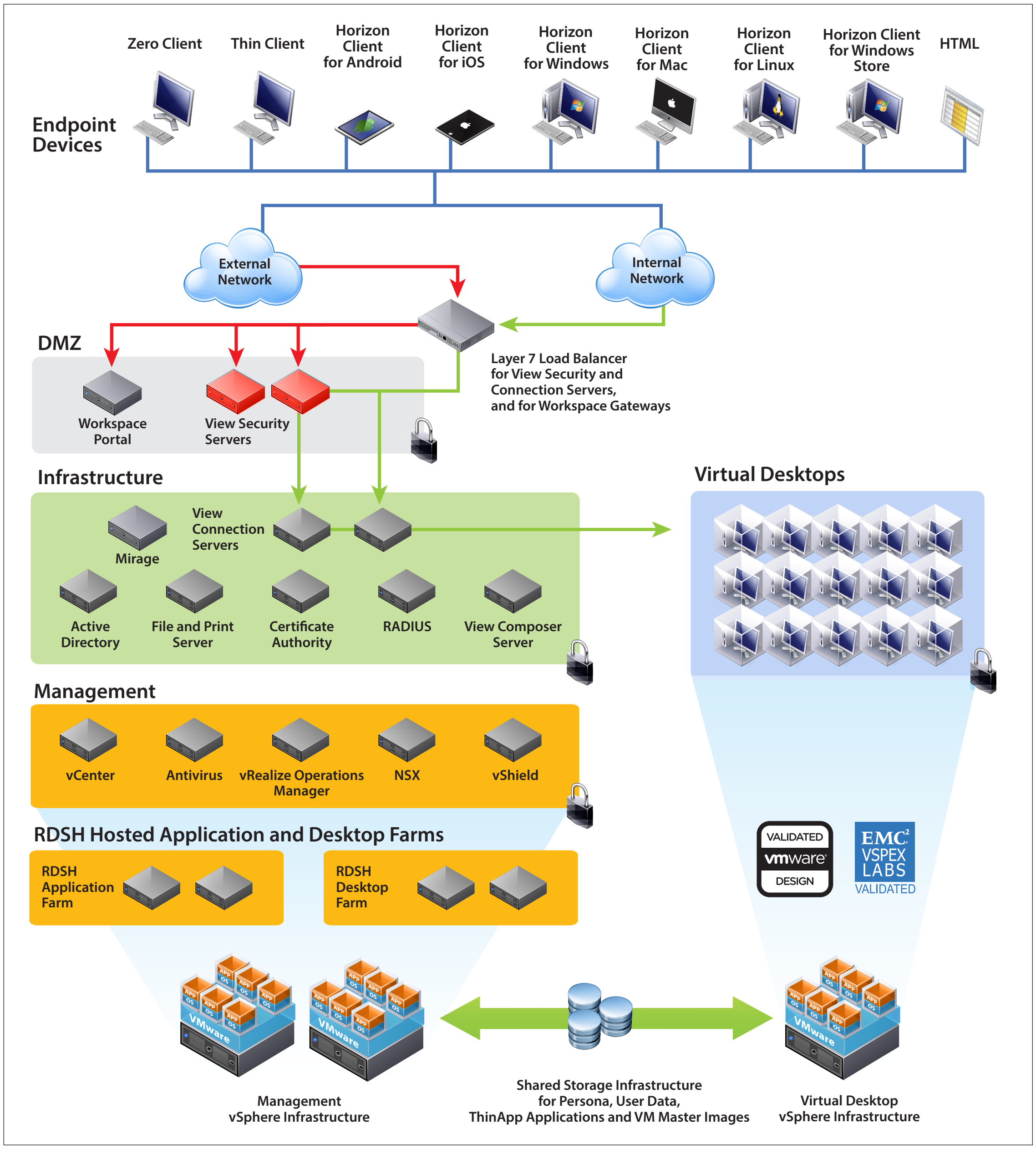



**Figure 1:** Horizon 6 Architecture

Each Horizon component has its own storage requirements. Mirage and View require separate storage and should not share the same backplane. Large-scale deployments in production have proven that View and Mirage designs sharing the same backplane are likely to experience performance and configuration difficulties.

# Capacity and Sizing Considerations

The primary storage considerations in an end-user computing infrastructure have two dimensions: performance and capacity, which are the focus of this paper.

## The Importance of IOPS

Input/Output Operations per Second (IOPS) is the performance measurement used to benchmark computer storage devices devices such as hard disk drives (HDD), solid-state drives (SSD), and storage area networks (SAN). Each disk type discussed in this document has a different IOPS performance statistic and should be evaluated independently.

When you consolidate multiple virtual machines and other user workloads on a hypervisor, you should understand the typical storage performance expected by a single operating system. This requires an understanding of the added contention for access to the storage subsystem that accompanies every subsequent guest operating system that you host on that hypervisor. Although IOPS cannot account for all performance requirements of a storage system, this measure is widely considered the single most important statistic. All the virtual assessment tools offered by VMware partners capture granular IOPS data, giving any IT architect the ability to optimize the storage accurately for end-user-computing workloads.

## The Impact of Latency

Latency can definitely affect performance and in some cases might actually have a greater impact than IOPS. Even if your storage can deliver a million IOPS, it does not guarantee your end users an enjoyable virtual desktop or workspace experience.

When assessing latency, always look up and down the storage stack to get a clear understanding of where latency can build up. It is always good to start at the top layer of the storage stack, where the application is running in the guest operating system, to find the total amount of latency that the application is seeing. Virtual-disk latency is one of the key metrics that influences good or bad user experience.

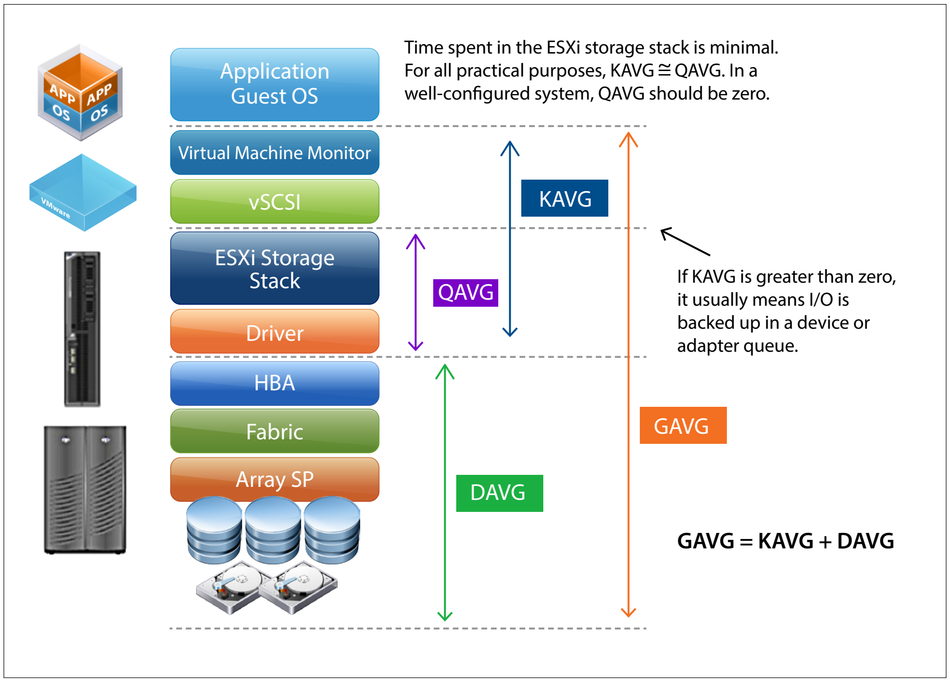



**Figure 2:** Storage Stack Overview

However, if you are using VMware vSphere®, VMware ESXi™ cannot see application latency because it is above the ESXi stack. What ESXi can do is detect three types of latency that are also reported back into esxtop and VMware vCenter™.

- Average guest latency (GAVG) has two major components: average disk latency (DAVG) and average kernel latency (KAVG).
- DAVG is the measure of time that I/O commands spend in the device, from the driver host bus adapter (HBA) to the back-end storage array.
- KAVG is how much time I/O spends in the ESXi kernel. Time is measured in milliseconds. KAVG is a derived metric, which means that there is no specific calculation for it. To derive KAVG, subtract DAVG from GAVG.

In addition, the VMkernel processes I/O very efficiently, so there should be no significant wait in the kernel, or KAVG. In a well-configured, well-running VDI environment, KAVG should be equal to zero. If KAVG is not equal to zero, then the I/O might be stuck in a kernel queue inside the VMkernel. When that is the case, time in the kernel queue is measured as the QAVG.

To get a sense of the latency that the application can see in the guest OS, use a tool such as Perfmon to compare the GAVG and the actual latency the application is seeing. This comparison reveals how much latency the guest OS is adding to the storage stack. For instance, if ESXi is reporting GAVG of 10 ms, but the application or Perfmon in the guest OS is reporting storage latency of 30 ms, then 20 ms of latency is somehow building up in the guest OS layer, and you should focus your debugging on the guest OS storage configuration.

## Floating or Dedicated Desktops

You can configure desktop pools to give users either dedicated or floating assignment to the desktops within the pool.

In a dedicated (persistent) assignment, VMware View Manager™ assigns each entitled user to one desktop in the pool. When a user connects to the pool, that user is always presented with the same desktop. The user's settings and data are saved between sessions, and no other user in the pool can access that desktop.

For a floating (stateless) assignment, View Manager dynamically assigns desktops in the pool to entitled users. Users connect to a different desktop each time they log in. When a user logs out, the desktop is returned to the pool.

IT architects and engineers need to choose either floating or dedicated desktops for their end users. Much of the decision process rests on differences in virtual desktop operation and support. Key considerations should include the advantages of operating a linked-clone desktop pool as opposed to a full-clone pool, specifically with respect to patching, updating, and lifecycle management.

## Full, Linked, and Array-Based Clones

If virtual desktops will be the primary platform, you must decide which kind of virtual machine best suits your requirements. For View deployments, you can choose full, linked, or array-based clones of your master virtual machine image. There are several factors to weigh before deciding on a strategy.

### Full Clones

One approach is to use full clones, specifically full-clone operations initiated by vSphere (see Array-Based Clone Virtual Machines ). Each full-clone virtual desktop is deployed as a full image copy of the base template machine.

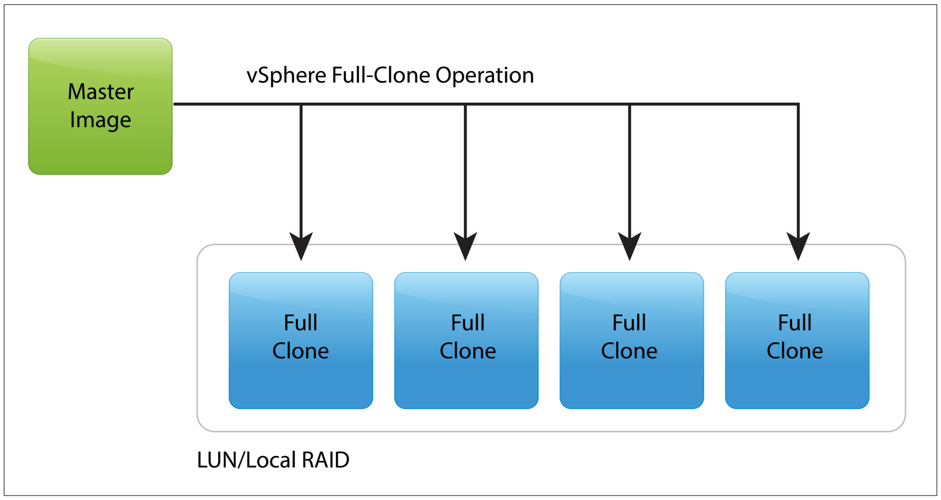



**Figure 3:** VMware vSphere Full Clones

Full-clone virtual machines ensure a consistent user environment. Each end user gets a dedicated virtual desktop that is not refreshed, recomposed, or rebuilt by any automated tools or processes. Certain software applications might require this type of deployment methodology.

For years, full clones were considered the simplest, although generally the most expensive, virtual desktop deployment. Full clones require the most raw storage capacity, but most storage solutions are now built specifically to minimize the storage needs of full clones, making full clones a more cost-effective deployment method for virtual desktops.

### Linked Clones

Virtual desktop deployments with View often leverage VMware View Composer™ linked clones for their desktop pools. View Composer uses VMware linked-clone technology to optimize desktop storage space and improve image control. Linked clones act as unique pointers for each user to a single virtual machine master. Although linked clones have traditionally been non-persistent, features such as SE Sparse can enable persistent linked clones.

Each linked-clone virtual machine has a unique identity and can be powered on, suspended, or reconfigured independently of the master image. Linked clones can be refreshed at the administrator's discretion, without affecting user data or settings, for efficient operating system management and optimization of storage resources. Maintaining the user data and any user-installed applications, however, requires additional technology. VMware offers both Persona management and VMware App Volumes™ to handle these user requirements.

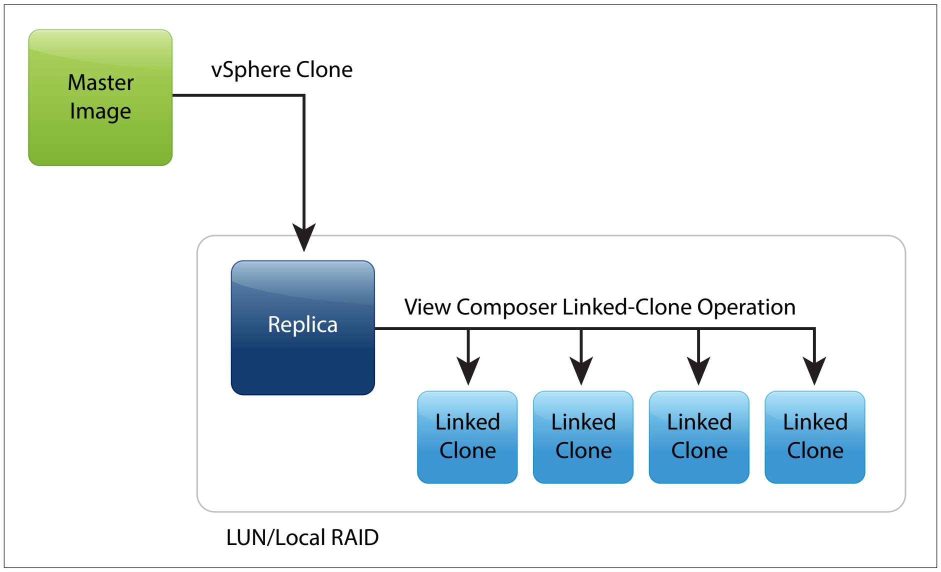



**Figure 4:** View Composer Linked Clones

Linked clones offer several storage benefits. Linked-clone virtual machines share a common replica image, so a linked clone can require as much as 70 percent less capacity than the parent image, greatly reducing the overall storage requirements for a given deployment. For real-world sizing, you can assume a 50 percent reduction in the linked-clone size compared to the parent or replica.

Horizon 6 with View can support up to 32 hosts on a VMFS- or NFS-based datastore with vSphere 5.1 or later, which means that you can create much larger linked-clone desktop pools than in previous versions. However, linked clones can expand in size over time, so you must give appropriate consideration to the expected growth relative to your refresh or recompose schedule.

### Array-Based Clones

The vSphere API for Array Integration (VAAI) establishes communication between ESXi hosts and storage devices, enabling ESXi hosts to offload certain storage operations to the array.

VAAI helps storage vendors speed up I/O operations that can be handled more efficiently in the storage hardware than in software. Some storage array vendors also offer an alternative cloning method, outside of vSphere, called block-based array cloning. In this method, the storage processors in the array efficiently clone the parent virtual machine. Most of these storage-based solutions use a native, block-level operation.

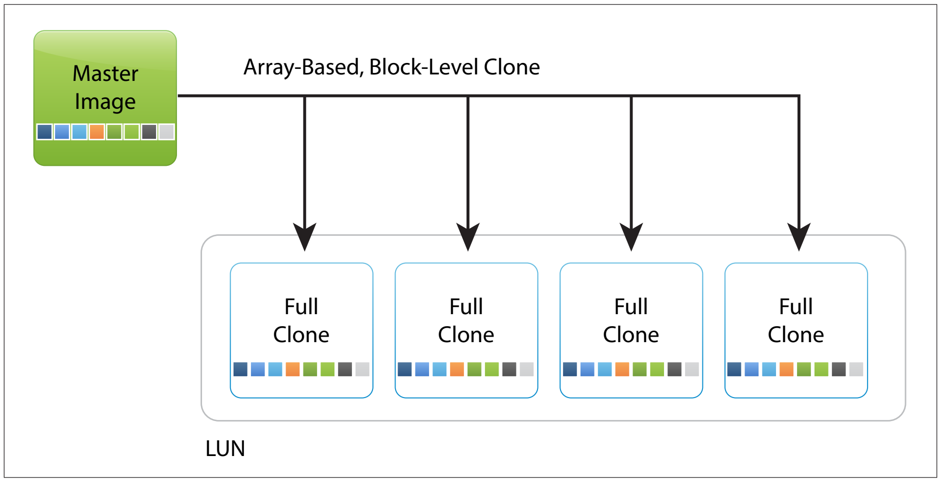



**Figure 5:** Array-Based Clones

Array-based clones are typically full clones of the parent virtual machine. However, in some cases, the cloning operations are designed to be more efficient than vSphere-based cloning. When the clones are created, they are powered on and customized with a process analogous to the physical process. For Windows, an unattended Sysprep answer file customizes particular information and generates a security identifier (SID) for each desktop virtual machine. These newly created virtual machines are fully independent—they have no dependency on the master image—and operate like standard full clones.

These cloning operations are independent of the View servers, so they require the creation of a manual desktop pool, with the new clones added to it. In many cases, the storage vendors have automated this process. There are several use cases where providing array-based clones might be advantageous, particularly in environments where large pools of virtual machines are provisioned and then removed. A good example of this is a lab or classroom environment that experiences highly transient and disparate desktop virtual machine configurations.

### VCAI Virtual Machine Linked Clones

The View Composer Array Integration (VCAI) feature offloads linked-clone tasks to a network-attached storage (NAS) device, leveraging the VAAI capabilities of vSphere to interact with NAS devices that have native snapshot capability. In other words, you can create linked clones that are to be offloaded to a storage array, with View Composer still in control of provisioning operations. VCAI is used in conjunction with linked-clone desktop pools and NFS datastores that are exported by NFS array vendors, as illustrated in Figure 6.

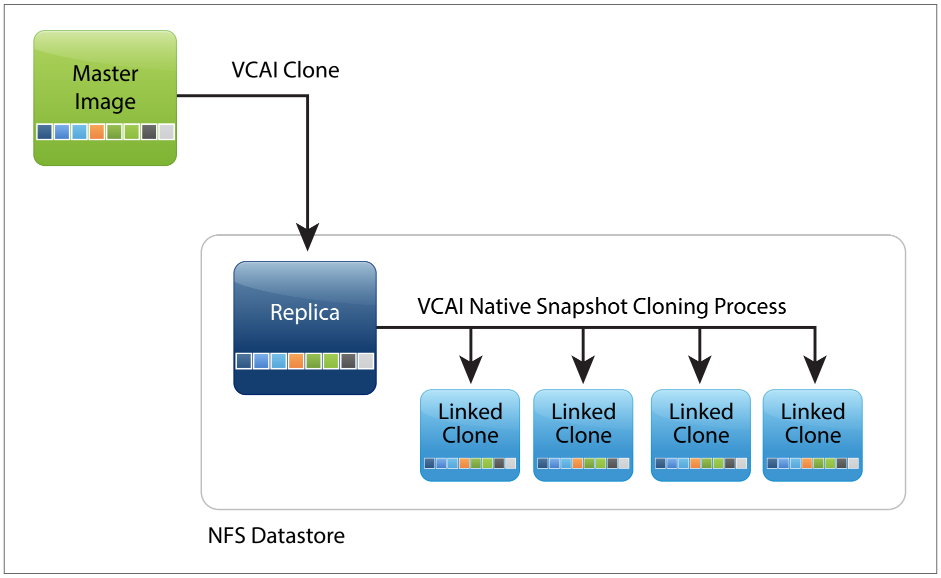



**Figure 6:** VCAI Linked Clone

View Composer manages the cloning processes, while the VCAI-capable array offloads cloning tasks to the storage processors.

VCAI is supported on NFS-attached storage hosted on specific devices— storage arrays on the VMware Hardware Compatibility List and listed in View Composer API for Array Integration (VCAI) support in VMware Horizon View—and requires the NFS VAAI Virtual Installation Bundle (VIB) Plug-in for vSphere. Always evaluate the suitability of VCAI for your storage design requirements on a case-by-case basis.

# Sizing Guidelines

Most storage vendors now offer VDI-specific solutions that provide additional storage savings and benefits for both full- and linked-clone virtual desktops.

## Virtual Machine Swap

Another dimension of per-virtual-machine storage calculations is the virtual machine swap file. There are two types of swap files to consider. The first type is the vRAM or virtual machine swap file (VSWP). The VSWP file has a `.vswp` extension and is stored with the virtual machine. Its size is equal to the amount of non-reserved vRAM allocated, or 100 percent of the allocated vRAM without memory reservations. A secondary swap file, or overhead swap file, is created to accommodate operations when the host is under memory pressure.

For the virtual machine swap file, consider reserving some or all of the allocated vRAM in the virtual machine to reduce the capacity overhead produced by the swap file. For example, for a virtual machine with 2 GB vRAM, a 1 GB reservation reduces the swap file size by 50 percent; a 2 GB reservation requires no swap file.

One more feature to consider when evaluating storage requirements and VSWP is the ability to store the VSWP file on local ESXi storage. This separates the virtual desktop from the VSWP file, and helps to reduce the shared storage required for a given solution. This is especially important when evaluating any backup or replication feature of a virtual desktop solution.

When moving the VSWP file to local storage, provide local storage that is capable of a large number of IOPS and throughput. SSDs are ideal and more commonly available today, although many solutions are still using 15 K legacy drives without major performance degradation. VMware recommends reviewing these two metrics—swap size and storage capacity—to ensure that users are getting the best possible performance.

## Full-Clone Virtual Machine Calculation

To calculate the storage capacity requirements for a full-clone virtual machine, use the following formula:

$$Full\ Clone + VSWP + Overhead$$

With the resulting value, you can calculate the total usable storage required for any deployment of full clones.

## Linked-Clone Virtual Machine Calculation

To calculate the storage capacity requirements for a linked-clone virtual machine, use this more detailed formula:

$$Replica\ per\ LUN + Linked\ Clone + Growth + VSWP + Overhead$$

The replica size, which is equal to the master image size, is taken into account on a per-LUN basis. A replica of the master image is automatically placed into each LUN that is selected to host the linked-clone virtual machines in a View desktop pool.

Capacity per linked-clone virtual machine begins with the linked clone itself (50 percent of the replica size is a good estimate). The initial size of the linked clone is equal to the size of the Windows page file, which could be 150 percent of the configured memory, in addition to virtual swap, and 200 MB customization data.

Add the anticipated amount that the linked clone can grow between refresh or recompose operations. (A good estimate is 20 percent of the linked clone size, but consider a higher percentage if you anticipate less frequent refresh or recompose operations.) Finally, add the virtual machine swap file size and overhead.

When the persistent and disposable disk features are used, they must also be included in the overall capacity calculation.

## Persona, User, and Shared Data Considerations

Consider the final dimensions of the entire capacity, which include Persona or profile, user and corporate data, and the Windows page file. Unless you are deploying a completely new infrastructure, there might already be shared files and data, which lie beyond the scope of a virtual desktop storage design. If you are deploying a solution that uses either standard Windows roaming profiles or a proprietary solution such as View Persona Management, be sure to take the added capacity requirements into account. This applies in both full- and linked-clone architectures, as suggested in Figure 7.

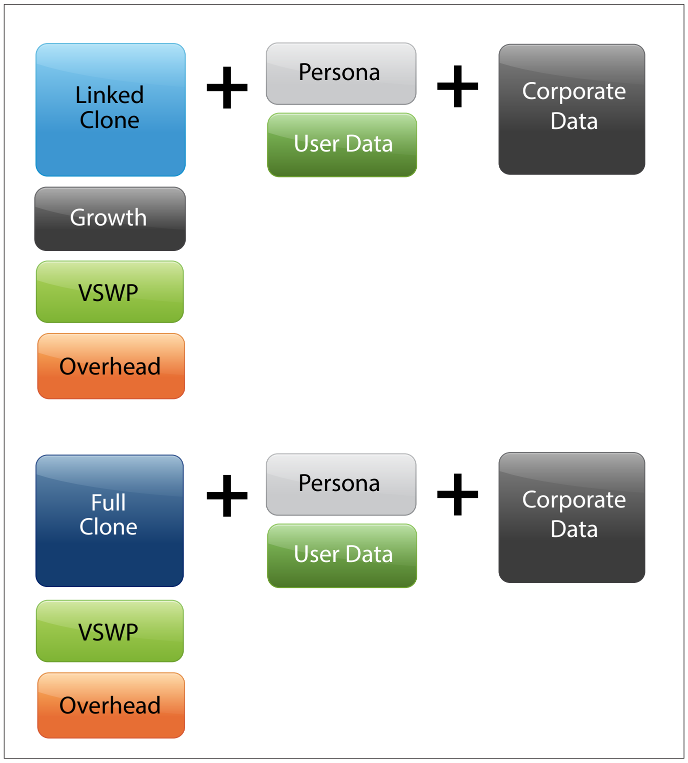



**Figure 7:** Make Allowances for Added Capacity Requirements

In most deployments, IT architects incorporate storage tiers for specific data types, for example to use high-performance VMFS datastores for the virtual machine operating system and swap data, while using NFS for shared and user data. This setup does not imply that NFS performance is inferior to block-based storage.

Rather, the management and operation of shared and user data might be better suited to an NFS file system. This flexibility highlights one of the many advantages of Horizon with View: the ability to assign the most suitable storage for each type of data.

Any final design needs to take performance, capacity, and operational requirements into account. The example in Figure 8 shows a linked-clone virtual machine data layout with Persona Management, user data, and corporate shared data tiered on different datastores.

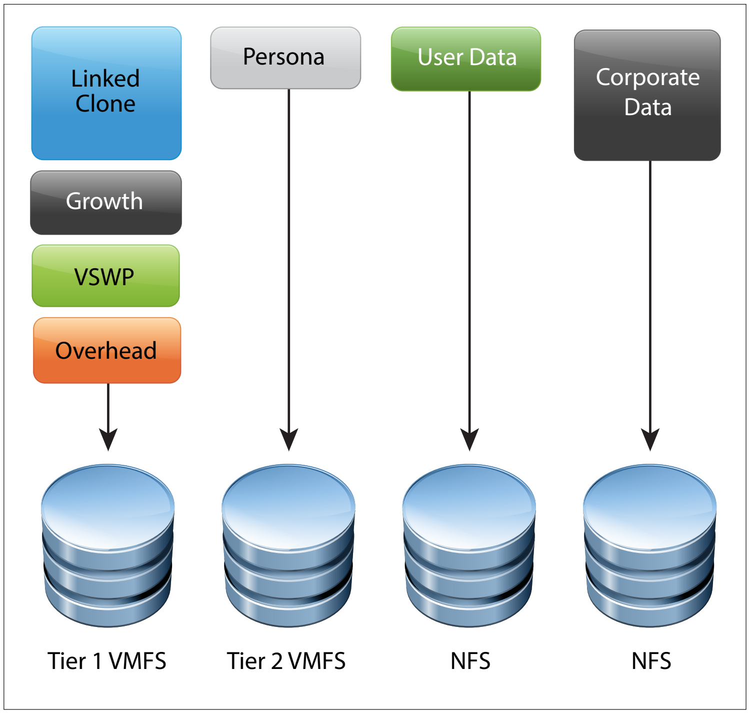



**Figure 8:** Linked-Clone Virtual Machine Data Layout

# Storage Platform Considerations

Virtual desktop workloads require the storage architect to balance performance and capacity requirements carefully in order to realize the best possible performance within a given budget. After determining the correct sizing guidelines, look at options for storage technology.

The storage landscape for virtual desktop workloads is one of the fastest changing areas of enterprise IT. The traditional choice of a standard, spinning, hard disk–based storage array is now considered inferior and insufficient for end-user workloads. In fact, nearly every configuration today includes host-side caching, solid-state storage, or some kind of hybrid technology that includes both flash (SSD) and conventional spinning disks (HDD). Cost-effective hybrid solutions are now available from most storage vendors.

When designing any architecture, consider using converged appliances, virtual storage appliances, and flash architectures in direct-attached storage (DAS), now more commonly known as local storage. Figure 9 illustrates the deployment options available today.

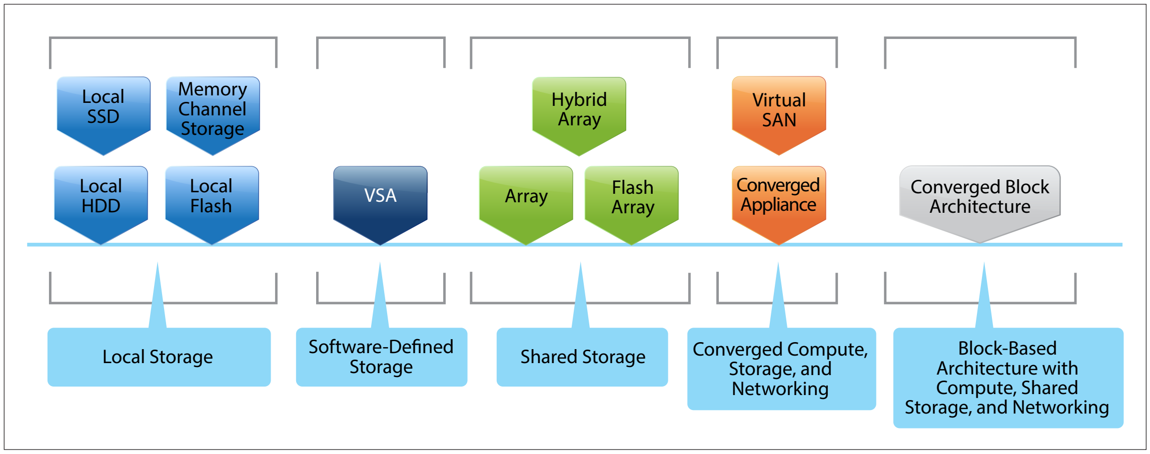



**Figure 9:** Deployment Options

## Local Storage Considerations

Local storage architectures are increasingly popular in virtual desktop design, largely due to the adoption of the reference architecture described in VMware Reference Architecture for Stateless Virtual Desktops on Local Solid-State Storage, which leverages local SSD disks for the virtual machine linked clones and swap space.

In a local storage architecture, there are tradeoffs with respect to hypervisor resource scheduling and high-availability. Specifically, if you use vSphere, then you must consider that VMware vSphere Distributed Resource Scheduler™, VMware vSphere Storage DRS™, and VMware vSphere High Availability (HA) all require *shared* storage.

In a virtual desktop design, these considerations are not as critical as they would be for server workloads because of differences in how the infrastructures operate.

With a virtual infrastructure handling server workloads, capacity demands tend to be more dynamic, requiring more resources during specific times of the day, week, or month. Administrators can shift resources appropriately to accommodate this fluctuating behavior.

In a virtual desktop infrastructure, demand for resources is usually more predictable, so IT tends to design for 100 percent concurrency in terms of compute and storage in steady state. It is easier to achieve a balanced design, with appropriate consideration for fault tolerance and failover capacity, using *local* storage. Local storage also keeps the I/O boundary within the server host, adopting a modular building-block methodology to simplify sizing and scale-out operations.

Enhanced VMware vSphere vMotion® capabilities in vSphere 5.x also allow you to move a file without the need for shared storage architectures.

### Flash Storage

When evaluating different storage types, it is important to compare flash-based storage to a spinning disk. Flash can provide significantly more IOPS than a standard spinning disk, but flash memory and solid-state disks (SSD) have unique performance characteristics that must be considered.

Flash-based memory systems are in many respects similar to standard random-access memory (RAM), but with two important differences. First, unlike RAM, flash memory is nonvolatile, so data is not subject to loss in the event of a power cycle or similar event. Second, RAM can be written and erased an almost infinite number of times. This is referred to as the program/erase (P/E) cycle. Flash has a finite P/E count, typically around 100,000 cycles. Simply put, flash memory eventually wears out, but it can take a long time for this to happen; it depends on the workload.

### Local HDD

It is always important to consider regular, spinning hard disk drives (HDD) as a local storage deployment option. All traditional servers are available with local disks, so the cost benefit is obvious. It is easy to leverage enterprise-class server drives and RAID controllers to provide high-performance block storage for virtual machines. Capacity and performance requirements are calculated as for a shared storage design.

### Local SSD

Like the local HDD design, a local SSD design replaces the spinning disk with an SSD disk. The SSD can provide significantly more IOPS than a magnetic disk, and is the most common type of local storage architecture in virtual desktop deployments.

Dependence on the local storage controller is a key consideration. For faster-throughput storage such as SSD, the controller can become an I/O bottleneck. An SSD can provide amazing read/write performance, but you must account for the limitation of the controller.

Occasionally, a magnetic disk can outperform low-cost SSD drives. Lower-end and less-sophisticated flash-memory-based SSDs can perform poorly for small random write I/O operations that are characteristic of the Windows OS. If you embark on a local SSD architecture, it is imperative to perform the appropriate load simulation at your target steady-state workload to ensure that the performance requirements have been satisfied.

### Local Flash

In this document, *local flash* refers to PCIe-attached flash memory. (PCIe is an acronym for Peripheral Component Interface Express.) These devices benefit from the speed of the PCIe interconnect to the host hypervisor, which is faster than a standard local storage controller. In general, they can support much more I/O throughput than local SSD. In some cases, the number can be exponential—millions of IOPS instead of thousands. PCIe-based flash is typically presented to the hypervisor as a disk device, through a custom driver implementation installed separately from the hypervisor binaries.

Performance considerations for PCIe-based flash and SSD are similar, but some operational considerations differ. Upgrades, maintenance, and swap-out operations are more disruptive with PCIe, because the device must be physically removed from or replaced on the server host. Although several storage vendors are working on technology to mitigate these issues, removing a local flash normally involves a full ESXi host power off. In contrast, an SSD disk can be swapped with the host still running. With a supported controller and resilient disk RAID configuration, you can swap the SSD disk without disruption to the host, except for degraded service while the RAID group is being reconstructed.

**Note:** Most local SSD implementations do not include RAID, so the preference in most production environments is to maximize usable disk space, thus sacrificing the operational flexibility of local SSD. This type of decision highlights the importance of a complete evaluation of the benefits and compromises for each storage technology, especially in terms of the impact on end-user workloads.

### Memory Channel Storage

The mainstream adoption of local flash and SSDs has helped drive down the price of these higher-performing solutions, but there is another technology that provides even higher performance. Memory channel storage (MCS) uses the existing memory subsystem to directly present nonvolatile memory to the hypervisor. This design allows for low-latency, high-performance, persistent storage directly on the processor bus.

The first major advantage of MCS is the improved performance enabled by its direct access to the processor bus. This allows for exponentially faster latency times than flash and SSD. The second advantage comes from the large number of dual in-line memory module (DIMM) slots in today's physical servers that allow for terabytes of storage capacity. Persistent storage in the form of a DIMM also uses less power than other local storage options.

With disadvantages similar to PCIe flash, MCS is becoming increasingly popular for a large number of use cases. The only added drawback for MCS is that a change in DIMM configuration might require a corresponding BIOS change, increasing the administration work required to support this technology.

There are only a few MCS vendors today, but as with virtual storage appliances (VSA) and flash storage, options in the marketplace tend to increase as use cases increase. The technology has existed for several years, but the capacity required to supplant traditional HDD, SSD, and local flash did not become cost-effective until recently.

### VMware Virtual SAN

Horizon 6 with View is Virtual SAN-aware. It creates storage policies for virtual machines based on the pool type, with each pool type having different default settings for availably, capacity, and performance. These policies are set at a vCenter level, when the first pool of a given type, such as a floating linked-clone pool, is created.  Administrators can modify the default policies for linked-clone desktops, for example changing the **Failure to Tolerate** settings. However, if a different policy is applied, it is overwritten when the pool is refreshed or recomposed.  When policies are applied to full-clone desktops, they are not modified by View after they are provisioned.

## Shared Storage Considerations

Shared storage options for a virtual deployment are described in the following sections.

### Virtual Storage Appliances

VSAs are becoming increasingly more common. They augment a virtual desktop storage design by bringing further features and performance in a layer above shared and direct-attached, or local, storage architecture.

VSA devices often work with both a local storage and a shared storage infrastructure, and can add features such as acceleration, deduplication, and replication. A VSA device can present local or shared storage available to the hypervisor as an iSCSI or NFS target. In some cases, a VSA device can present local storage on each host in a cluster as an aggregated shared volume.

The most common deployment use case addresses existing shared storage that does not perform adequately. Adding VSAs helps to provide the performance or availability that is required in the design. More advanced virtual storage appliances can utilize available hypervisor RAM and present it as a datastore for hosting virtual desktops, providing extremely high levels of I/O performance. However, putting all of this new technology together can make capacity calculation complex. VMware recommends reaching out to the VSA vendor for help with these calculations.

Often, VSAs can be deployed on virtual desktop hosts or in a dedicated cluster, to provide replication and high-availability features from commodity shared or local storage that would otherwise not have these features. This arrangement enables enterprise-level functionality on commercial-priced solutions.

### Storage Area Networks

A storage area network (SAN) is the original deployment architecture, and the one that most IT architects are the most familiar with. In enterprise architectures, a SAN is often a data center resource that is shared across both physical and virtual deployments.

The seemingly endless permutations of SAN design available today are beyond the scope of this document, but you must still consider the storage protocol (such as Fibre Channel, iSCSI, or possibly others) and back-end I/O requirements to ensure that the right levels of performance and capacity are available for steady-state operations.

Almost all top-tier storage vendors augment their existing magnetic disk arrays with caching technologies to improve performance. The assist mechanisms available vary from front-end cache to in-line, single-level cell (SLC) memory, with automatic tiering or placement of hot data.

Consider array-based deduplication for any design. Data deduplication, whether inline as the data is written, or as a background or batch process, can significantly reduce capacity requirements for virtual machine storage as well as for user and application data.

### Hybrid Storage Arrays

As traditional SAN technology continues to evolve, the storage landscape has seen several new vendors providing hybrid storage platforms. These new vendors and their solutions cater specifically to virtual desktops or similar high, sustained IOPS workloads. These hybrid arrays often use a blend of flash and magnetic disk, along with proprietary file and block management at the controller level, to provide built-in tiering and high-performance characteristics.

Compared to a traditional approach with SAN and magnetic disk arrays, these hybrid arrays can often achieve the same or better performance with the internal tiering and caching mechanisms described above, with a smaller controller and disk footprint. The tradeoff with hybrid arrays is typically on the operational side of the infrastructure. Storage architects and administrators need to learn a new set of tools and standards when these new solutions are introduced, so give careful consideration to whether the advantages of operating and managing another storage solution in the data center are outweighed by the need for alternate sizing and scaling methodology.

### Flash Arrays

Flash-based arrays provide a standards-based interconnect, enabling all the high-availability features of the hypervisor while providing exceptional performance to the workloads hosted on them. However, this performance comes at a significantly higher cost than traditional HDD-based shared storage systems.

Flash-based arrays represent a real shift in the design and operation of virtual infrastructures. The performance requirements of virtual desktop and other Tier 1 application workloads need extensive design and validation to accommodate flash technology. The game-changing aspect of a flash-based SAN is that you can begin to stack workloads on this platform with minimal concern for performance degradation, because the flash array can provide exponential improvement in IOPS performance.

Although flash solutions are not as simple as “set it and forget it” operationally, they are altering the workload virtualization landscape. Cost is, of course, a factor with flash-array-based designs, but more flash-only solutions are being deployed as the acquisition cost decreases.

### Converged and Hyper-Converged Infrastructures

The IT industry is witnessing the maturation of the converged infrastructure marketplace. A converged appliance aggregates the compute, storage, and networking elements of traditional enterprise architecture and combines them in a pre-sized, pre-validated form factor. With more vendors entering the converged and hyper-converged market, their solutions have become more mainstream and more reliable. A mix of startups and established enterprise vendors has helped to validate the impact of this converged technology.

Typically, a converged appliance has at least one compute node plus shared storage and networking, all within one chassis and shared among the compute nodes. The networking stack is often 10GbE or InfiniBand to provide a high-bandwidth, high-capacity interconnection among nodes and to the outside infrastructure. The storage subsystem might comprise RAM, PCIe flash, SSD, and magnetic disk. Automatic caching, tiering, and staging of I/O through the appliance go through the proprietary drivers and software integrated with the hypervisor.

An appliance-based approach simplifies the design process. Instead of breaking a large problem into its component parts and solving them individually, you can use an appliance as a scale-out building block for your virtual desktop infrastructure. Sizing is straightforward: Appliances are typically validated to provide a certain number of IOPS for a specific number of users in a single building block. To determine how many blocks you need, divide the total number of users by the number of users that can be supported in a single block. It is a good idea either to allow for a few spares or start at a smaller number and scale out as you prove the concept.

Often, the operational overhead is acceptable, given the simplicity and high performance of these architectures. In some cases, converged appliance vendors have gone to great lengths to ensure the simplicity of their solutions. Appliance-based deployments are increasingly popular and provide a low barrier to entry in the design and deployment of a virtual infrastructure.

#### Converged Block Architecture

In contrast to the converged appliance use case, which bundles all components in a single chassis, block architecture traditionally consists of separate, integrated components for compute, networking, and storage. These converged blocks are constructed in the same rack or racks and sized for a particular type of workload, such as virtual desktops.

Block-based architectures are typically based on high-end, enterprise-grade components. Compared to appliances or local storage architecture, they are usually sized for a larger number of initial users.

Like other options, blocks have evolved and typically include array-based acceleration, caching, and data deduplication technologies. They typically incorporate Tier 1 shared storage arrays in their block architecture, so they share many of the features, as well as the flexibility, found in component arrays. A typical block provides the ability to swap components required for a given design without drastically affecting the overall sizing of the block.

Block components are also validated together, and the vendor often pre-assembles and integrates all the components before shipping the complete solution. Pre-assembly provides a comprehensive, production-ready deployment that is ready to be plugged in to the data center.

It is important to consider the flexibility and maturity of these configurations when looking at an all-in-one approach in any design. As usual, the operational aspects must be considered alongside the advantages of any approach.

# VMware App Volumes Storage Considerations

VMware introduced App Volumes in 2014 as a major addition to the VMware EUC portfolio. App Volumes is a real-time application-delivery and lifecycle management tool. Enterprises can use App Volumes to build dynamic application-delivery systems that ensure that all applications are centrally managed. Applications are delivered to desktops through virtual disks. There is no need to modify desktops or applications themselves, and the App Volumes solution can be scaled out easily and cost-effectively, without compromising end-user experience.

The App Volumes architecture shown in Figure 10 highlights the dependency of App Volumes on the underlying storage platform.

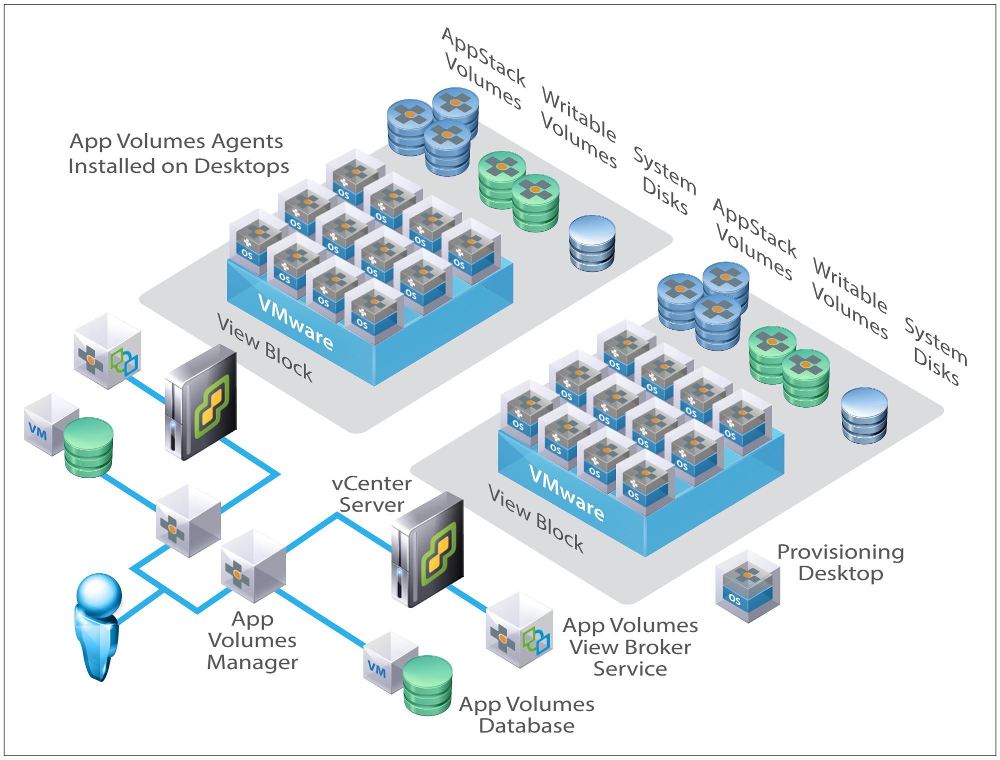



**Figure 10:** App Volumes Architecture

Three specific components live on storage and can be affected by the storage performance.

- **App Volumes Database** – A Microsoft SQL (production) or SQLExpress (non-production) database that contains configuration information for AppStacks, writable volumes, users, machines, entitlements, and transactions.
- **AppStack Volume** – Read-only volume(s) containing any number of Windows applications. Multiple AppStacks can be mapped to an individual system or user. An individual AppStack can also be mapped to more than one system or user.
- **Writable Volume** – A user-specific volume where the user is allowed to preserve data. Writable volumes can be used to store user-installed applications and local profile information, such as application settings, application licensing information, and data. A user can be assigned only one writable volume at a time.

Although the database component generally resides in a standard enterprise SQL environment, the AppStack and writable volumes reside on a VMware datastore.

### AppStack and Writable Template Placement

When new AppStacks and writable volumes are deployed, the templates are used as the copy source. Administrators should place these templates on a centralized shared storage platform. As with all production shared storage objects, the template storage should be highly available, resilient, and recoverable.

### Free-Space Considerations

Accurate AppStack sizing and writable volume sizing are critical for success. AppStack volumes should be large enough to allow installation of applications and should also account for application updates that can be anticipated from time to time.

Writable volumes should also be sized sufficiently to accommodate all users' writeable volumes. Storage platforms that allow for volume resizing are especially helpful when there is not sufficient data for an accurate volume calculation.

Disk space is not consumed immediately because all AppStacks and writable volumes are vSphere VMFS thin-provisioned. It is important to follow VMware best practices when managing thin-provisioned storage environments. Free-space monitoring is essential in large production environments.

### Writable Volumes Delay Creation Option

There are two policy options that can complicate free space management for writable volumes:

- The option to create writable volumes on the user's next login bases storage processes and their effects on capacity on user login behavior.
- The option to restrict writable volume access (and initial creation) to one or a group of desktop virtual machines can also allow user login behavior to dictate when a writable volume template is copied.

In a large App Volumes environment, it is *not* a good practice to allow user behavior to dictate storage operations and capacity allocation—it can lead to boot storms, among other problems. Instead, create writable volumes at the time of entitlement.

### Mount Local

If you use local storage, you can place AppStacks and writable volumes on the datastore where virtual machines reside. App Volumes points to local storage before referring to shared storage, so that specified AppStacks or writable volumes on local storage are mounted instead of duplicate copies on shared storage.

For the Mount Local feature to work, all vSphere hosts must have the same root-level user credentials.

## Performance

When assessing application performance, remember these three commandments:

1. **Know thy app.**

   Understanding how the application behaves is the first step to understanding the storage requirements for a satisfactory user experience.

2. **Know thy user.**

   How a user manipulates an application can have an even bigger impact on application performance. It is imperative to understand user expectations.

3. **Test, test, and test again.**

   - Test the app initially.
   - Test the app during each Proof of Concept and pilot phase.
   - Test the app as you scale out to the full user base.

When testing an application, use a performance analytics tools such as VMware vRealize™ Operations Manager™. Gather virtual machine, host, network, and storage performance information for use when AppStacks are operated on a larger scale, and remember that user feedback is fundamentally important for assessing the overall performance of an application.

AppStack volumes are read-only. Depending on utilization patterns, the underlying shared storage platform might be subjected to significant read I/O activity. Consider using flash and hybrid-flash storage technologies for AppStack volumes.

Writable volumes, in contrast, are read-write. Storage utilization patterns are largely influenced by user behavior with regard to desktop logins and logouts, user-installed applications, and changes to local user profiles. Group each set of similar users into use cases, and evaluate performance based on peak average use.

Host configurations have significant impact on performance at scale. Consider all ESXi best practices during each phase of scale-out. To support optimal performance of AppStacks and writable volumes, give special consideration to the following host storage elements:

- Host storage policies
- Storage network configuration
- HBA or network adapter (NFS) configuration
- Multipathing configuration

## Queue Depth Configuration

For best results, follow the recommendations of the relevant storage partner when you configure hosts and clusters.

# Workspace Portal Storage Considerations

VMware Workspace™ Portal provides an easy way for users to access applications and files on any device and enables IT to deliver, manage, and secure these assets centrally. For end users, the result is true mobility: anytime, anywhere access to everything they need to work productively. For IT, the result is better control over corporate data across devices.

## Virtual Appliance Storage Requirements

With the introduction of Workspace Portal 2.1, the vApp no longer includes multiple virtual appliances by default. One virtual appliance can now support the Workspace Portal environment, serving all management functions for the deployment. However, to provide high availability, you should provision a second virtual appliance.

The Workspace Portal vApp should be stored on Tier 2 storage. All virtual machines should be thin provisioned to help keep the disk space usage low. The disk requirement for each Workspace Portal virtual appliance is one 36 GB disk. Place the vApp on a datastore capable of supporting the Workspace Portal IOPS requirements. An example of a supported storage configuration is detailed in Figure 11.

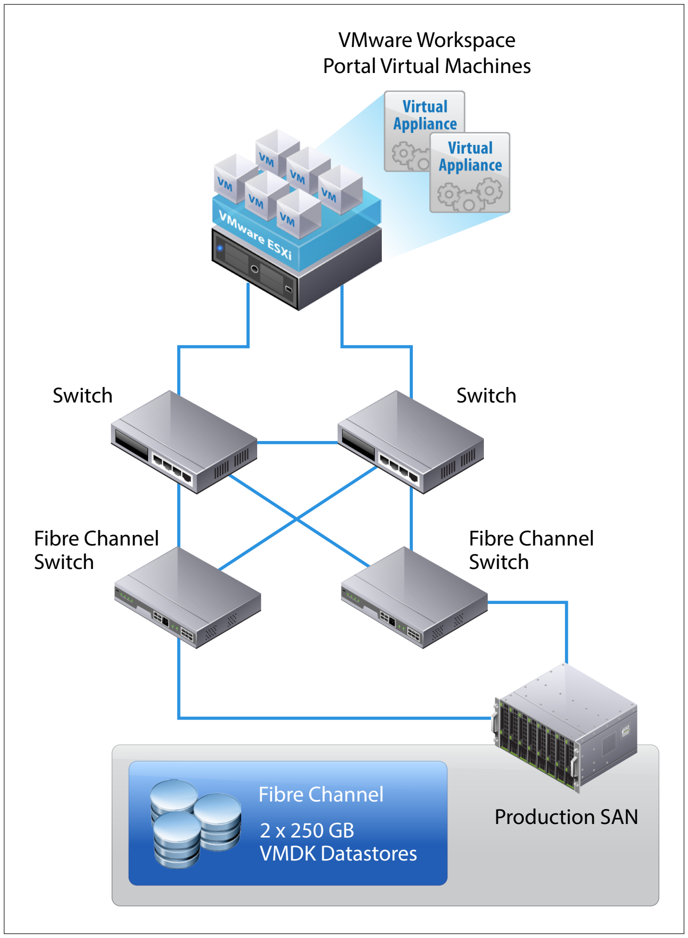



**Figure 11:** A Supported Storage Configuration

### Database Sizing Requirements

A PostgreSQL database is included in the `workspace-va` configuration, and you can use an external database server. For information about specific database versions and service pack configurations supported with Workspace, see the VMware Product Interoperability Matrixes.

External database sizing requires 64 GB for the first 100,000 users and 20 GB for each additional 10,000 users.

# Mirage Storage Considerations

VMware Mirage divides desktop images into logical layers. IT can create and manage standardized layers that are stored in the data center and applied to user endpoints or let users install their own applications and add their user data and settings. Mirage provides centralized control and management of desktop images, whether your organization owns the computers, or users bring their own devices (BYOD).

IT-managed elements and user-controlled elements are all rolled into one Mirage desktop image in the data center. Figure 12 shows the layers that Mirage uses to build a delivered desktop.

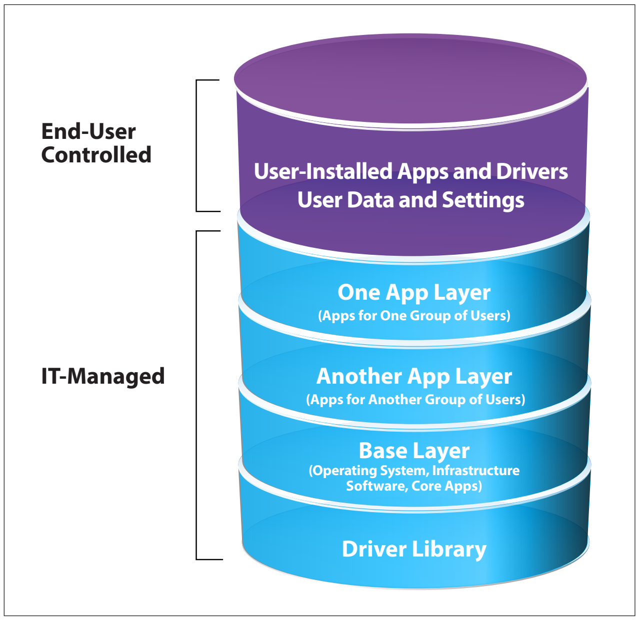



**Figure 12:** Layers in a Mirage Desktop

With all of those layers residing on shared storage, be sure to consider performance for each type of layer. The IT-managed layers are read-only and therefore predictable. The main storage component of the Mirage architecture consists of the Windows file shares that hold the base Mirage layers. There is also a database cluster that keeps track of all other Mirage components and assignments.

Networking connectivity speeds, Quality of Service and Class of Service implementations, bandwidth utilization, and network latency all affect Mirage storage performance. Operational concurrency is the main driver for performance evaluation. High latency with a single operation is easy to maintain, but high latency with 500 concurrent operations magnifies performance issues for all of these technologies. The faster the data moves to the Mirage server, the faster information can be written to disk, which increases the potential need for additional storage IOPS. Even with ample network bandwidth available to Mirage clients, other factors, such as network latency, can reduce the amount of data that can be sent from a Mirage client, decreasing the IOPS demand for backend storage.

Mirage throttles connections if a storage bottleneck is detected, but throttling affects the time it takes to complete centralization or other Mirage-related operations. Endpoint throttling can reduce the amount of data sent to the Mirage server, ultimately extending the time it takes to complete Mirage operations. This can also decrease IOPS required on the backend storage.

For additional information on concurrency of operations and related topics, see the VMware Mirage Large-Scale Reference Architecture.

It is important to monitor the storage performance for throughput and overall capacity on an ongoing basis. There are simple, effective methods for managing storage capacity challenges, such as adding storage IOPS capacity, adjusting upload change intervals, and scheduling certain Mirage operations for times when system utilization is lower.

### Mirage Server Nodes

The Mirage server nodes, located in the data center, manage the storage and delivery of base layers, app layers, and Centralized Virtual Desktops (CVD) to clients, and consolidate monitoring and management communications. You can deploy multiple servers as a server cluster to manage endpoint devices for large enterprise organizations. It is good practice to keep the server on a dedicated machine or a virtual machine.

The Mirage server nodes use a local cache to store data served to the users, so these nodes are the most susceptible to positive or negative storage performance. With an estimate of 15 GB of unique data per CVD, each Common Internet File System (CIFS) share can support hundreds of CVDs. CVD size should generally include 30 percent for deduplication, 20 percent for snapshot overhead, and 20 percent for growth. Additional storage can be added along with additional Mirage server and endpoint capacity. Architects and administrators should evaluate the performance of the storage subsystem to ensure there is ample IOPS capacity and acceptable performance for each environment.

### Mirage Database and Recovery

Mirage servers and the management server share access to the backend Mirage database, which contains configuration and operational information used by the Mirage system components. Each CVD needs around 2 MB in the Mirage database, which can support up to 20,000 CVDs per Mirage Instance, for a total of 40 GB.

VMware recommends using vSphere HA for database server resiliency. Although this effectively protects the database servers from most hardware failures, it is still advisable to use backups for additional security.

With this recovery model, logs continue to grow until you back up the database and transaction logs. If you set the recovery model to **Full**, you must back up the database and transaction logs regularly. Otherwise, the transaction logs can continue to grow without limit.

# Storage Enhancements

There are several standard features that you can leverage when deploying a Horizon with View virtual desktop infrastructure.

## View Storage Accelerator

It is important to consider both steady and storm conditions with respect to I/O for a given storage subsystem. The most common example of a storm condition is a virtual machine boot storm. In this example, 1,000 virtual machines are powered on, generating significantly more requests for IOPS than in steady state.

View Storage Accelerator absorbs this read I/O at the host level, thereby reducing the IOPS requirement at the local or shared-storage-array level. It is implemented as an in-memory cache of common blocks and is applicable to floating (stateless) as well as dedicated (persistent) desktops. It is completely transparent to the guest virtual machine. During peak events such as boot and login, which are read-intensive, the performance improvement is measured as a net reduction in IOPS to a centralized shared storage array.

Figure 13 shows the effect of View Storage Accelerator on a 1,000-virtual machine boot storm on a shared storage array. The graph shows around 13,000 IOPS without VSA enabled, and around 3,600 with the feature turned on, a reduction of approximately 70 percent.

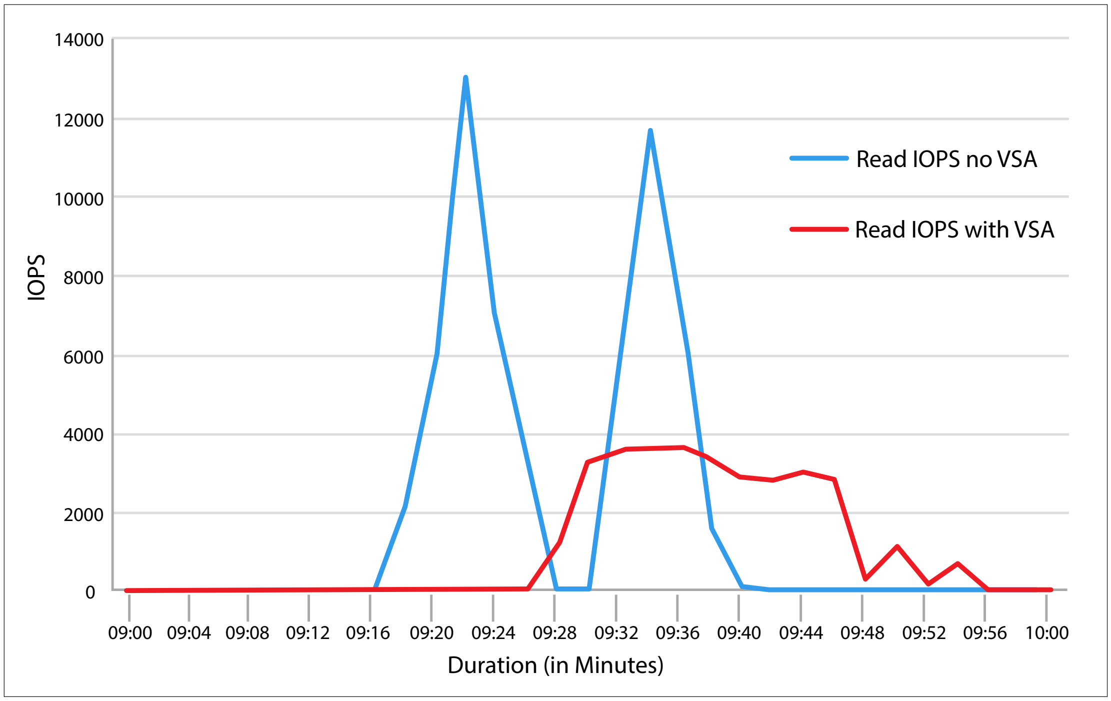



**Figure 13:** Significant Reduction of IOPS in Boot Storm

### Space-Efficient Sparse Virtual Disks (SE Sparse)

The View SE Sparse disk feature grants the ability to reclaim previously used space in the guest OS. If the desktop pools use linked clones, then deploying this capability can reclaim disk space, helping to restrain linked-clone growth by occupying only as much space as is currently being used.

There are two steps involved in space reclamation, as shown in Figure 14. The first is the *wipe* operation, which frees a contiguous area of space in the virtual machine disk. The second is the *shrink* operation, which remaps or truncates that area of free space to enable the physical storage to return to the free pool.

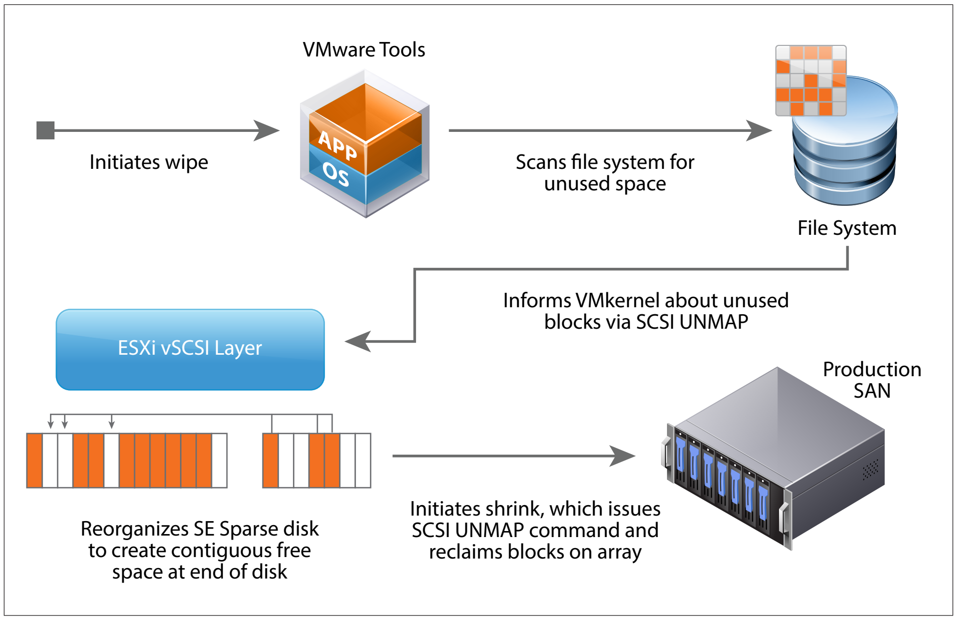



**Figure 14:** Space Reclamation

SE Sparse disks have a configurable block allocation size, which can be tuned to the recommendations of the storage array vendor. Consider SE Sparse when you use linked-clone pools to control linked-clone growth.

### Tiered Storage

The option for tiering linked-clone storage allows the administrator to select different datastores for the replica and linked-clone disk placement when a desktop pool is created.

For example, you can place replica virtual machines into a very high-performance, low-capacity datastore, and place the linked clones in a lower-performance but higher-capacity datastore. This strategy effectively splits the read/write I/O across the two datastores. It is just one example of how you can use tiered storage as part of the overall storage design.

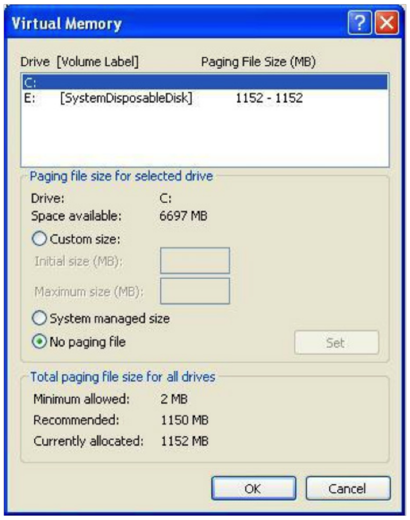
### Disposable Disks

A disposable disk redirects certain volatile files away from the operating system disk to help reduce the tendency of linked clones to expand in size.

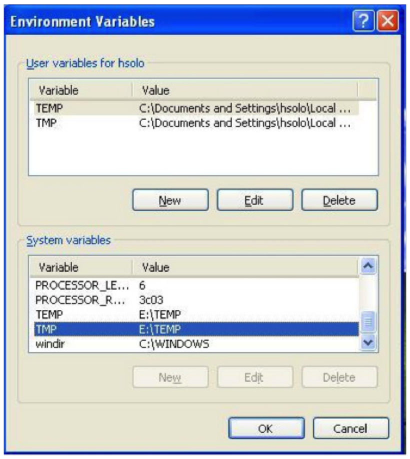
Much has been written on designs that use disposable disks, not all of it clear or accurate. To clear up any confusion or misunderstanding about what disposable disks do and exactly how they function, three elements require clarification: disposable disk redirection, location, and delete-and-refresh operations. Understanding disposable disks and their functionality enables architects and engineers to design their environments appropriately.

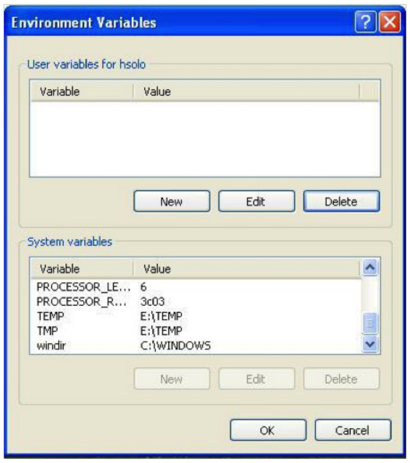
### Disposable Disk Redirection

By default, three elements are redirected to the disposable disk. The first is the Windows swap file. View Composer redirects the swap file from the C: drive to the disposable disk. Set the swap file to a specific size to make capacity planning easier.




The other two elements that are redirected are the *system* environment variables `TMP` and `TEMP`. The user environment variables `TMP` and `TEMP`, however, are *not* redirected by default. The use of the same environment variable names at the system and user levels can cause some confusion.




VMware recommends removing the user environment variables `TMP` and `TEMP`, to force Windows to use the *system* variables and redirect the user temporary files to the disposable disk.

### Disposable Disk Location

There is a common misconception that, like the user data disk, the disposable disk can be redirected to a different storage tier. This is not the case; the disposable disk is *always* stored with the operating system disk.

In Horizon 6 with View, the configuration allows for selection of the drive letter for the disposable disk, which helps to prevent conflicts with mapped drives, but this setting and the size are the only customizations you can make to the disposable disk.

**Disposable Disk Deletion and Refresh**

The question of how the disposable disk is refreshed tends to cause the most confusion. Some people believe the disposable disk is refreshed when the user logs out; others believe it is refreshed on reboot. The disposable disk is actually refreshed only when View powers off the virtual machine. User-initiated shutdowns and reboots, and vCenter power actions, do not affect the disposable disk.

The following actions cause the disposable disk to be refreshed:

- Rebalance
- Refresh
- Recompose
- VM powered off because the Pool Power Policy is set to **Always Power Off**

If the Pool Power Policy is set to any of the other settings (**Power On**, **Do Nothing**, or **Suspend**), then the disposable disk is *not* refreshed automatically.

**SE Sparse Disks**

The storage reclamation feature uses an SE Sparse disk for the operating system disk, which allows View to shrink the operating system disks and reclaim previously used space if files are deleted from within the guest OS. The operating system disk is created as an SE Sparse disk. User data disks and disposable disks are created as standard VMDK files. The key difference between SE Sparse disks and disposable disks is that the former rely on *files* being deleted from within the guest OS. In contrast, the entire disposable *disk* is deleted, along with all the files it contains, when View powers off the virtual machine.

SE Sparse disks are not currently supported on Virtual SAN. If disposable disks are configured, then the power cycle policy should be reviewed. It may be beneficial to add an operational task for administrators to change the **Power On** setting for the pool to **Power Off** periodically within a maintenance window to refresh the disposable disk. This operation is particularly important for persistent desktops, which have long refresh and recompose cycles.

# Horizon 6 Storage Enhancements

Major enhancements for View storage are described in the following sections.

## Virtual SAN

Virtual SAN is a hypervisor-converged, software-defined storage platform that is fully integrated with vSphere. Virtual SAN aggregates locally attached disks of hosts that are members of a vSphere cluster to create a distributed, shared storage solution. Because Virtual SAN sits directly in the I/O data path, it can deliver high levels of performance, scalability, and resilience without taxing the CPU with additional overhead. Virtual SAN enables the rapid provisioning of storage within VMware vCenter during virtual machine creation and deployment operations.

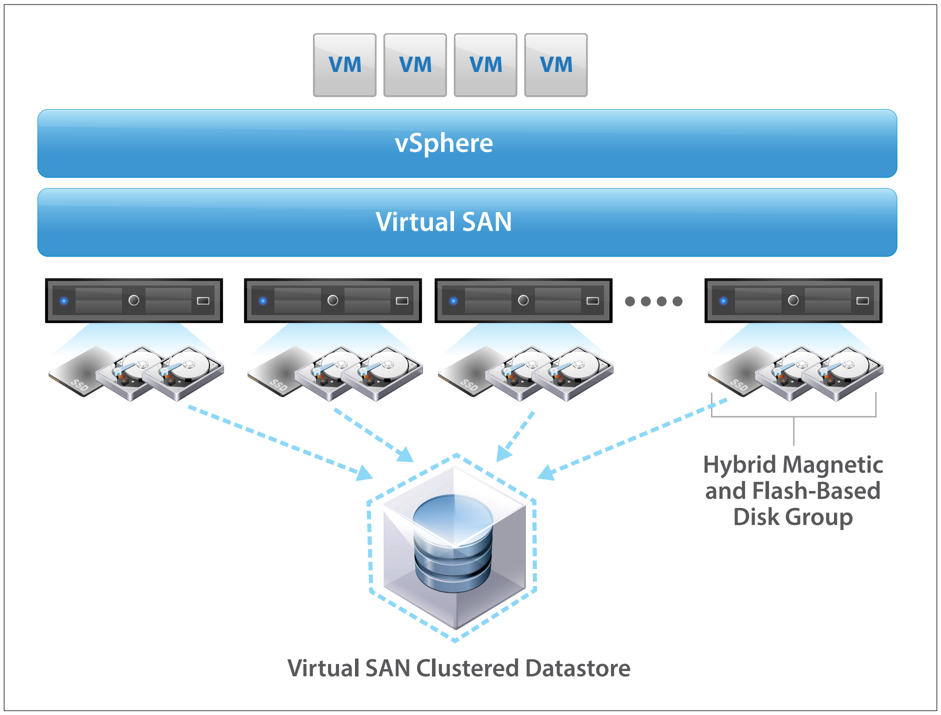



**Figure 15:** Virtual SAN Hybrid Disk Architecture

Virtual SAN hybrid disk architecture uses flash-based devices for performance and magnetic disks for capacity and persistent data storage. Its distributed datastore is an object-store file system that leverages the vSphere Storage Policy-Based Management (SPBM) feature to deliver centrally managed, application-centric storage services and capabilities. Administrators can specify storage attributes, such as capacity, performance, and availability, as policies on a per-virtual-machine basis. The policies dynamically self-tune and load-balance the system so that each virtual machine has the right level of resources.

The Virtual SAN storage platform for desktop workloads enables the solution to scale linearly, with each host capable of supporting approximately 100 users. The VMware Horizon with View and Virtual SAN Reference Architecture shows 400 desktops running on four ESXi hosts. Recent VMware testing, however, has shown that Horizon with View can scale up to 1,600 desktops on 16 hosts in a single cluster with Virtual SAN. These numbers are expected to continue to scale upward.


**Figure 16:** Summary of Test Results from Previous-Generation Virtual SAN Reference Architecture

Local SSDs and HDDs are used in conjunction with Virtual SAN technology to provide a scalable and enterprise-class virtual desktop storage solution. The local disk groups within a given vSphere cluster are combined to form a Virtual SAN datastore. This software-defined storage platform combines powerful and flexible hardware with advanced efficiency, management, and software-defined storage.

Horizon with View recognizes the Virtual SAN storage type and automates the creation of Virtual SAN storage policies based on the type of desktops being deployed. Virtual SAN policies can be applied from the View Administrator console.

### Cloud Pod Architecture

Cloud Pod Architecture (CPA) enables dynamic movement and location of View pods across multiple data centers for efficient management of end users in distributed locations. Although it does not affect storage design at a high level, CPA adds features that can be enhanced by storage technology.

When users can reach View pools from multiple data centers, architects need to evaluate how those users get their user-specific information. With such a large number of User Environment Management (UEM) tools available, there are even more solutions to this unique problem. VMware recommends reaching out to UEM vendors for more information on supported configurations, which usually involve some type of storage replication between locations. It is important to review these items before deploying them in any production environment.

## Summary

With the plethora of purpose-built technology available today, less effort is required to design a comprehensive solution than in the past. Some day, many of the steps in the design process might become unnecessary. Unfortunately, however, that day has not yet arrived. Fortunately, this paper can help acquaint the architect with the most essential considerations needed today.

As vendors and partners try to converge on solutions for virtual desktops and mixed high-performance workloads in the data center, there are quite a few pitfalls an architect can encounter. The most common pitfall is to underestimate performance requirements—including capacity and sizing requirements—when scaling.

To develop a successful virtual desktop storage design, it is important to take all of the technical elements, as well as the overall design concept, into account, with appropriate balance for performance, capacity, operational simplicity, and future requirements. Give proper consideration to the virtual machine overhead associated with the hypervisor and 3D features and the expected behavior of target workloads. Always allow for the effects of consolidating multiple instances onto a hypervisor and obvious, or perhaps not-so-obvious, capacity requirements for desktop virtual machines.

# References


An Introduction to Flash Technology

Best Practices for running vSphere on NFS Storage

Default cluster size for NTFS, FAT, and exFAT

Disk Defragmentation – Background and Engineering the Windows 7 Improvements

Enabling View Composer Array Integration (VCAI) on VMware Horizon View

IOps?

Optimize Storage Fast with VMware Virtual SAN!

Overhead Memory on Virtual Machines

Performance Best Practices for VMware vSphere 5.5

Server and Storage Sizing Guide for Windows 7

Should I defrag my Guest OS?

Storage Protocol Comparison White Paper

Tips for a Successful VMware Virtual SAN Evaluation

Understanding Memory Resource Management in VMware vSphere 5.0

Understanding View Disposable Disks

View Composer API for Array Integration (VCAI) support in VMware Horizon View

View Storage Accelerator – in Practice

Virtual Machine I/O – Fairness versus Performance

VMware App Volumes Deployment Guide

VMware App Volumes – What About Performance?

VMware Hardware Compatibility List

VMware Horizon with View and Virtual SAN Reference Architecture

VMware Horizon 6 View Configuration Maximums

VMware Horizon View 6 sizing limits and recommendations

VMware Mirage Large-Scale Reference Architecture

VMware Product Interoperability Matrixes

VMware Reference Architecture for Stateless Virtual Desktops on Local Solid-State Storage

VMware View 5.0 3D Reverse Engineered

VMware Virtual SAN Design and Sizing Guide for Horizon View Virtual Desktop Infrastructures

VMware vSphere 5.1 Storage Enhancements – Part 2: SE Sparse Disks

VMware vSphere 5.1 vMotion Architecture, Performance and Best Practices

VMX Swap Files

Windows Internals Book, 6th Edition

Windows 8 Virtual Desktop Infrastructure image client tuning guide

# About the Authors and Contributors

Kristopher Boyd, formerly a Product Manager with the VMware End-User Computing Solutions Engineering team, updated this paper for Horizon 6. Earlier papers on View storage considerations were written or updated by Rory Clements, Fred Schimscheimer, and Warren Ponder.

VMware would like to acknowledge the following people for their contributions to this document:

- Stephane Asselin, Senior End-User-Computing Architect, Technical Enablement, VMware
- Donal Geary, Desktop Virtualization Reference Architecture Engineer, VMware
- Ray Heffer, Senior End-User-Computing Architect, VMware
- Gary Sloane, Consulting Editor, VMware
- Tristan Todd, End-User-Computing Technical Enablement Architect, VMware
- Jeremy Wheeler, Professional Services Consultant, VMware
- Travis Wood, Senior Solutions Architect, VMware

To comment on this paper, contact the VMware End-User-Computing Technical Marketing team at euc_tech_content_feedback@vmware.com.

# Glossary of Acronyms

| | |
|---|---|
| BIOS | Basic Input/Output System |
| BYOD | Bring Your Own Device |
| CIFS | Common Internet File System |
| CPA | Cloud Pod Architecture |
| CPU | Central Processing Unit |
| CVD | Centralized Virtual Desktop |
| DAS | Direct-Attached Storage, also called local storage |
| DAVG | Average time, in milliseconds, that I/O commands spend in a device |
| DIMM | Dual In-line memory module |
| EUC | VMware End-User Computing |
| VDI | Virtual Desktop Infrastructure |
| GAVG | Average time, in milliseconds, that I/O commands spend in a guest OS |
| HA | High Availability |
| HDD | Hard Disk Drive |
| I/O | Input/Output |
| IOPS | Input/Output per Second |
| IT | Information Technology |
| KAVG | Average time, in milliseconds, that I/O commands spend in the ESXi kernel |
| LUN | Logical Unit Number |

| | |
|---|---|
| MCS | Memory Channel Storage |
| NAS | Network-Attached Storage |
| NFS | Network File System |
| OS | Operating System |
| PCIe | Peripheral Component Interface Express |
| QAVG | Average time, in milliseconds, that I/O commands spend backed up in a kernel queue |
| RAID | Redundant Array of Inexpensive Disks |
| RAM | Random-Access Memory |
| SAN | Storage Area Network |
| SE Sparse | Space-Efficient Sparse Virtual Disk |
| SID | Security Identifier |
| SLC | Single-Level Cell (or in-line) memory |
| SSD | Solid-State Drive |
| SPBM | Storage Policy-Based Management |
| SQL | Structured Query Language |
| Storage DRS | VMware Storage Distributed Resource Scheduler – Moves virtual machines between datastores for load balancing |
| 10GbE | 10 Gigabit Ethernet |
| UEM | User Environment Management |
| VAAI | vSphere API for Array Integration |
| VCAI | View Composer Array Integration |
| VIB | vSphere Installation Bundle |

| | |
|---|---|
| VMFS | Virtual Machine File System |
| vRAM | Virtual Random-Access Memory |
| VSA | Virtual Storage Appliance |
| vSphere DRS | vSphere Distributed Resource Scheduler – Moves virtual machines between hosts for load balancing |

**VMware, Inc.** 3401 Hillview Avenue Palo Alto CA 94304 USA Tel 877-486-9273 Fax 650-427-5001 www.vmware.com