The VMware® Reference Architecture for Stateless Virtual Desktops on Local Solid-State Storage with VMware View™ 5
Table of Contents

Introduction .............................................................. 5
Architecture Highlights .................................................. 6
VMware View 5 .............................................................. 8
Stateless Solution Overview ........................................... 10
VMware Reference Architecture Overview ......................... 11
VMware View 5 Stateless Reference Architecture Goal ............ 11
VMware View 5 Stateless
Reference Architecture Design ........................................ 12
VMware View Logical Design .......................................... 13
The Stateless Building Block ............................................ 14
Virtual Infrastructure Configuration ..................................... 15
  Physical Component Configuration .................................. 15
  Compute Configurations .............................................. 15
  Infrastructure Virtual Machines .................................... 15
  Solid-State Drive Configuration ..................................... 16
  Shared Storage Configuration ........................................ 17
  Access Infrastructure .................................................. 18
  Physical Network Details ............................................. 18
  VMware View Connection Servers .................................. 18
  VMware View Pod and Building Block Details ................. 19
  VMware View Connection Server Configuration ................ 19
    Detailed Solid-State Drive Pool Configuration ............... 23
  Virtual Machine Image ............................................... 24
Validation Methodology .................................................. 25
  Client Access Devices ................................................ 25
  Workload Description ............................................... 27
  High Availability ..................................................... 27
  Unplanned Downtime .................................................. 28
  Planned Downtime .................................................... 28
Validation Results ....................................................... 32
  Application Response Times ........................................ 34
  Solid-State Drive Statistics ........................................ 35
Conclusion ............................................................... 38
About the Author ........................................................ 38
Acknowledgements ...................................................... 38
References ............................................................... 38
Trademarks ............................................................... 38
Appendix D. VMware View Components .................................................. 51
  VMware View Connection Server ...................................................... 52
  VMware View Security Server .......................................................... 52
  VMware View Administrator .............................................................. 52
  VMware View Agent ......................................................................... 52
  VMware View Client .......................................................................... 52
  VMware View Portal .......................................................................... 52
  VMware View Composer .................................................................... 53
  VMware View Transfer Server .......................................................... 53
  VMware vCenter Server .................................................................... 53

Desktop and Pool Management ............................................................. 54
  Individual Desktops ........................................................................... 54
  Dedicated Pools ................................................................................ 54
  Floating Pools .................................................................................. 54
  Microsoft Terminal Services Desktop Pool ........................................ 54

Desktop Persistence ............................................................................. 55

Storage Flexibility .............................................................................. 55

Capacity Savings ............................................................................... 55

Storage I/O Contention ...................................................................... 57

Storage I/O Flexibility ...................................................................... 58
Introduction

This document provides the reference architecture for Stateless Virtual Desktops with VMware® View™ 5 on local solid-state storage. A stateless desktop architecture is ideally suited for standard desktop environments where the desktop image is consistent from user to user; however with proper application design can be used in broad use cases.

VMware View 5 allows specific placement of different types of disks to be re-directed, even to local storage, the I/O requirements on shared storage can be drastically reduced.

The VMware View 5 Reference Architecture for Stateless Virtual Desktops involves high-performing, solid-state drives within a physical host to offload the majority of desktop virtualization IOPS needed, while providing a stateless desktop virtualization design for planned and unplanned downtime.

This use of local solid-state drives (SSDs) is a completely new approach to desktop virtualization storage. SSDs are critical to achieving a low per-desktop cost because they remove the need for operating system and application storage requirements on the SAN. By decentralizing these storage needs, the desktop environment for CPU and memory resources scales linearly, with extremely low latency.

The simplicity of the architecture that is enabled by View 5’s tiered storage capability allows for a new era in the evolution of desktop virtualization cost modeling.

Increasingly, organizations are turning to virtual desktop technologies to address the operational and strategic issues related to traditional corporate desktop environments. VMware View 5 provides not only a virtual desktop environment that is secure, cost effective, and easy to deploy; but now can also provide comprehensive storage flexibility.

VMware View 5 modernizes the desktop experience to deliver private cloud capabilities to users for a consistent experience across the global enterprise. This Desktop-as-a-Service model is possible with unmatched scalability in a stateless design.

The challenges of traditional desktop administration, especially at scale, range from lost laptops with corporate data, security issues related to viruses or hackers, or simply ensuring IT resources can maintain a service level agreement (SLA) appropriate for specific end users. In addition to the challenges of operational management, IT must also consider the implication of broader system-wide issues such as compliance, corporate governance and business continuity strategies.

This whitepaper provides full descriptions of the test environment as well as performance metrics captured during test validation.
Architecture Highlights

The stateless tiered storage reference architecture validation shows results for both 96 and 120 Windows 7 virtual machines per VDI host server. Additional highlights of the reference architecture IOPS, datastore latency and CPU usage are shown in the figures below. Detailed performance data can be found in the Validation Results section.

Figure 1: Maximum Application Response Time of 3.3 Seconds, Even at 120 Workloads

Figure 2: Maximum of 10 IOPS Recorded per Virtual Machine
The VMware Reference Architecture for Stateless Virtual Desktops on Local Solid-State Storage with VMware View 5

Figure 3: Maximum 4ms Latency to the Local SSD Datastores

Figure 4: Boot Storm and Steady State Well Within Host Hardware Limits
The VMware Reference Architecture for Stateless Virtual Desktops on Local Solid-State Storage with VMware View 5

VMware View 5

VMware View 5 is a desktop virtualization solution that simplifies IT manageability and control while delivering one of the highest fidelity end-user experiences across devices and networks.

The VMware View solution helps IT organizations automate desktop and application management, reduce costs and increase data security through centralization of the desktop environment. This centralization results in greater end-user freedom and increased control for IT organizations. By encapsulating the operating systems, applications and user data into isolated layers, IT organizations can deliver a modern desktop. It can then deliver dynamic, elastic desktop cloud services such as applications, unified communications and 3D graphics for real-world productivity and greater business agility.

Unlike other desktop virtualization products, VMware View is built on, and tightly integrated with, vSphere, the industry-leading virtualization platform, allowing customers to extend the value of VMware infrastructure and its enterprise class features such as high availability, disaster recovery and business continuity.

View 5 includes many enhancements to the end-user experience and IT control. Some of the more notable features include:

- **PCoIP Optimization Controls**—deliver protocol efficiency and enable IT administrators to configure bandwidth settings by use case, user or network requirements and consume up to 75 percent less bandwidth
- **PCoIP Continuity Services**—deliver a seamless end-user experience regardless of network reliability by detecting interruptions and automatically reconnecting the session
- **PCoIP Extension Services**—allow Windows Management Instrumentation (WMI)-based tools to collect more than 20 session statistics for monitoring, trending and troubleshooting end-user support issues
- **View Media Services for 3D Graphics**—enable View desktops to run basic 3D applications such as Aero, Office 2010 or those requiring OpenGL or DirectX—without specialized graphics cards or client devices
- **View Media Services for Integrated Unified Communications**—integrate voice over IP (VoIP) and the View desktop experience for the end user through an architecture that optimizes performance for both the desktop and unified communications
- **View Persona Management** (View Premier editions only)—dynamically associates a user persona with stateless floating desktops. IT administrators can deploy easier-to-manage stateless floating desktops to more use cases while enabling user personalization to persist between sessions
- **View Client for Android**—enables end users with Android-based tablets to access View virtual desktops

Support for VMware vSphere 5 leverages the latest functionality of the leading cloud infrastructure platform for highly available, scalable and reliable desktop services.

For additional details and features available in VMware View 5, see the release notes.
Typical VMware View 5 deployments consist of several common components (illustrated below) which represent a typical architecture. It includes VMware View components as well as other components commonly integrated with VMware View.
Stateless Solution Overview

VMware View 5 enables the centralization of the virtual desktop environment by leveraging flexible storage architecture. This architecture can be deployed in a modular and cost-effective fashion. This design provides linear scalability across both compute and storage regardless of scale.

Figure 6: VMware View 5 Architecture for Stateless Virtual Desktops

The high-level compute infrastructure consists of:

- Intel® Xeon® Processor X5675 (12M Cache, 3.06 GHz, 6.40 GT/s Intel® QPI)
- 192GB RAM per host
- Intel® SSD 320 Series 300GB SATA Solid State Drive
- 10Gbit Converged Networking
- Windows 7 virtual machine with 1vCPU and 1GB vRAM
VMware Reference Architecture Overview

VMware reference architectures are built and validated by VMware and supporting partners. They are designed to address common use cases; examples include enterprise desktop replacement, remote access, business process outsourcing and disaster recovery. A reference architecture describes in detail the environment and workload used to simulate realistic usage, and draws conclusions based on that particular reference architecture.

This guide is intended to help customers—IT architects, consultants and administrators—involved in the early phases of planning, design and deployment of VMware View–based solutions. The purpose is to provide a standard, repeatable and highly scalable design that can be easily adapted to specific environments and customer requirements.

The reference architecture “building block” approach uses common components to minimize support costs and deployment risks during the planning of large-scale, VMware View 5–based deployments. The building block approach is based on information and experiences from some of the largest VMware View deployments in production today. While drawing on existing best practices and deployment guides pertinent to many of the individual specific components, the reference architectures are tested and validated in the field and described in detail.

Some key features that can help an organization get started quickly with a solution that integrates easily into existing IT processes and procedures include:

• Standardized, validated, repeatable components
• Scalable designs that allow room for future growth
• Validated and tested designs that reduce implementation and operational risks
• Quick implementation, reduced costs and minimized risk

VMware View 5 Stateless Reference Architecture Goal

This reference architecture provides a proven and tested stateless architecture for enterprise desktop deployments. The goal was to design and validate a standardized building block, consisting of components capable of supporting 1,440 virtual desktops, specifically for stateless desktop pools running on local solid-state storage. The overall design also includes the infrastructure components necessary to integrate seven (7) building blocks to support a 10,000-user VMware View “pod” that can be managed as a single entity.

The architecture uses common components and a standardized design to reduce the overall cost of implementation and management. All infrastructure components used to validate the reference architecture are interchangeable, which allows components from an organization’s vendor of choice to be incorporated and unique features that enhance the value of the overall solution to be utilized.
VMware View 5 Stateless Reference Architecture Design

For this reference architecture, a building block-based, stateless desktop solution capable of supporting up to 10,000 virtual desktops was designed.

The solution includes the typical hardware components necessary to integrate 10,000 users into a VMware View “Pod,” which can be managed as a single entity, leveraging a 1,440-user “building block.”

This architecture uses common components and a standardized design across the building blocks to reduce the overall cost of implementing the solution. All physical infrastructure components used to validate this reference architecture are interchangeable. This means that an organization can use a vendor of choice for any physical component of the architecture, including network, storage and server components. Various vendors might offer or include unique features that further enhance the value of the overall VMware View offering for a particular enterprise.

As the architecture is stateless in nature, local direct attached SSD disks are leveraged as the virtual machine datastores. User data resides on a shared storage platform.

The stateless architecture provides a linear scalability from a single host to hundreds of hosts, and allows a modular approach for customers to scale out their virtual desktop infrastructure.

This design is intended to be a general guide that addresses common design questions for stateless desktops.

A single-user role type was chosen to demonstrate scalability with a typical use case. By virtualizing the application set, a single pool type could be deployed within an environment with multiple user roles.

The abstraction of the applications from the base image allows each user to leverage the same base OS and multiple, different applications, reducing the administrative overhead required to maintain different user roles in the same VDI architecture.
VMware View Logical Design

The figure below shows the logical design layout of the stateless architecture.
The Stateless Building Block

The overall pod contains seven (7) building blocks of 1,440 users each, the focus of this reference architecture. Each building block contains fifteen (15) physical servers, each configured with two (2) local solid-state drives and three (3) clusters containing five (5) VDI host servers each. A centralized shared storage device is used for user profile data, virtual desktop ".vswap" files and the core infrastructure server data.

The blade server chassis used for this validation was capable of supporting 16 blade servers. However, any type of server with the same hardware specification can also be used. One consideration when using blades is the port consolidation of each blade server-to-uplink module. Port consolidation is a common factor when leveraging blade servers. Because of the ability to leverage 10Gbit Ethernet, extensive bandwidth was available to the cluster tested, and easily scaled to accommodate 16 blades within the chassis.

Each building block shares items such as the physical networking layer (as did the core infrastructure environment).

---

Figure 8: Stateless Building Block

The physical compute layout is shown in the figure below.

---

Figure 9: Stateless Compute Cluster Physical Layout
Virtual Infrastructure Configuration

The virtual infrastructure at the core of each building block is comprised of physical servers and VMware vSphere 5.0. The overall virtual infrastructure is designed to support 10,000 desktop users across 105 hosts. Depending on the environment and desktop workload, results vary.

The overall VMware View 5 Stateless pod is designed to be supported by five VMware vCenter 5 servers. The VMware vCenter server databases were hosted on a single Microsoft SQL 2008 server. Both VMware vCenter and the Microsoft SQL server were implemented as virtual machines. The vSphere hosts running these infrastructure virtual machines were part of a standard VMware vSphere High Availability (HA) cluster to protect them from any physical server failures. This is a common approach for hosting desktop infrastructure services that helps provide the highest level of availability.

Each VMware View building block contains three VMware vSphere 5.0 clusters of five physical servers each. A five-node cluster is designed to host 480 virtual desktops. The cluster was configured with HA and vMotion disabled. In a stateless design, these features are not only unnecessary but also will not function properly without shared storage.

Although the overall building block is designed with a dedicated VMware vCenter server that includes its own dedicated database server, it might be desirable to consolidate the database servers into one or a few larger database servers. However, database consolidation is not covered as part of this validation effort.

Physical Component Configuration

Compute Configurations
High-performance commodity components were chosen to maximize the per-host value of the reference architecture.

<table>
<thead>
<tr>
<th>VMware View Desktop 1,440-User Cluster</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Quantity</td>
<td>Description</td>
</tr>
<tr>
<td>15</td>
<td>Compute Servers – vSphere ESXi 5.0 Build 469512</td>
</tr>
<tr>
<td>2</td>
<td>Intel® Xeon® Processor X5675 (12MB Cache, 3.06 GHz, 6.40 GT/S Intel® QPI)</td>
</tr>
<tr>
<td>12</td>
<td>192GB RAM total per host, comprised of Kingston 8GB DDR3-1333, Part #KTH-PL313/8G</td>
</tr>
<tr>
<td>2</td>
<td>Intel® SSD 320 Series 300GB SATA Solid State Drive</td>
</tr>
<tr>
<td>2</td>
<td>Extreme Networks NetXtreme II 57711E 10Gbit Ethernet</td>
</tr>
</tbody>
</table>

Infrastructure Virtual Machines
A separate set of servers was used to host the common infrastructure components needed for an enterprise desktop environment, such as Microsoft Active Directory, DNS, DHCP and the VMware View Connection Server. The VMware vCenter server also ran within this cluster. Each desktop infrastructure service was implemented as a virtual machine running Windows 2008 Server R2. The infrastructure servers also hosted the load generation virtual machines used in the application performance testing.
All required infrastructure services were configured as virtual machines as per the diagram below:

![Infrastructure Virtual Machines](image)

**Solid-State Drive Configuration**

There are several types of Intel® SSDs that are available.

Single-level cell (SLC) NAND flash-based SSDs have the highest endurance rating and are made from ~100k cycle capable NAND with an MTBF of 2 million hours. Multi-level cell (MLC) NAND–based SSDs are made from ~5k cycle NAND with an MTBF of 1.2 million hours. Choice of SLC- or MLC-based SSDs mainly depends on the usage model and cost. The more write-intensive usage models tend to use SLC-based SSDs.

Each server was configured with two drives, and because of the stateless design the drives were configured as a striped array (RAID zero). While the traditional thinking around drive level redundancy was considered, the additional capacity combined with the increased reliability associated with SSD resulted in a different path. Given the stateless design, it is simply unnecessary to have a mirrored array of disk with expensive SSD.

If a host, drive, power supply or similar component fails, the high-level design provides the redundancy necessary to provide a user with a new desktop as detailed in the high-availability section of this reference architecture document.

Capacity is the center point of the design, as the ability to use the local SSD for various components of the desktop is only limited by today’s technology.

To make this architecture cost effective today, a capacity of 600GB is more than sufficient to support a properly optimized Windows 7 desktop at a sufficient quantity.

However, higher capacity drives will expand the design to allow for additional use cases on the same host, as well as provide additional performance benefits. The third generation Intel® SSD 320 Series Solid State Drives based on 25nm MLC NAND technology has a current capacity range from 40GB to 600GB.
The desktop tiered storage components are detailed in the image below. User data, the parent base image and the .vswp's associated with the individual desktops were on shared storage. Local SSD contained only the replica base image and linked clones.

![Figure 11: VMware View 5.0 Tiered Storage Components](image)

To ensure realistic results with a new solid-state drive, an SSD requires special attention when being used for benchmarking. A “new” SSD is faster with writes than it will be after heavy use. This is especially prominent with MLC-based solid-state. While this is a known item that is easy to address, without preconditioning the results from the local SSD drives would be misleading.

For production usage, preconditioning is not required. The process that was followed is detailed below:

- Each solid-state drive was preconditioned individually. Initialize the disk and format as NTFS within Windows.
- With the tool IOMeter, run a continuous 128k sequential write test to the entire drive’s address space. This forces all blocks on the disk to have some type of content so the SSD does not contain an unusually large reserve space for writes
- Delete the .tst file from the drive
- Duplicate the test above three times to use up any reserve space, each time deleting the .tst and re-running the test
- With IOMeter, a continuous 128k read-write test was then run for four hours against the entire drive's address space to further condition the drive to ensure accurate final results

**Shared Storage Configuration**

Shared storage is not necessary for the placement of desktop images. Therefore, the actual requirement for more expensive shared storage infrastructure is reduced using this architecture.

The usage of the shared storage is completely outside the actual building block. Technically, it is possible to provide individual hosts without any type of shared storage whatsoever; however, for simplicity with templates as well as user data it was included in the design. An actual production environment would base the shared storage device around user requirements.
Access Infrastructure
The physical networking was implemented with a redundant network core of full 10Gbit Ethernet. The network core also load balances incoming requests across VMware View Connection Servers, where user requests were routed to the appropriate building block for each virtual desktop session.

Realistically, it would be possible to provide networking at 10Gbit speeds for 105 servers (in blade chassis) with a pair of switches; however this reference architecture used core networking as it would be more typical in a datacenter.

Physical Network Details
The table below details the components and configuration of the physical network.

<table>
<thead>
<tr>
<th>POD CORE NETWORKING COMPONENTS</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Quantity</td>
<td>Description</td>
</tr>
<tr>
<td>2</td>
<td>Modular Core Networking</td>
</tr>
<tr>
<td>8</td>
<td>10 Gigabit Ethernet Modules</td>
</tr>
<tr>
<td>1</td>
<td>Load Balancing Module</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>BUILDING BLOCK NETWORK COMPONENT (COMMON ACROSS BUILDING BLOCKS)</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>10Gbit Network Switches</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>VLAN ID</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>16</td>
<td>VMware View Desktops – Infrastructure – 802.11q Tagged</td>
</tr>
<tr>
<td>17</td>
<td>Infrastructure Servers – 802.11q Tagged</td>
</tr>
<tr>
<td>20</td>
<td>Management Needs – 802.11q Tagged</td>
</tr>
<tr>
<td>23</td>
<td>Storage – iSCSI and NFS – 802.11q Tagged</td>
</tr>
<tr>
<td>24</td>
<td>vMotion – 802.11q Tagged</td>
</tr>
</tbody>
</table>

VMware View Connection Servers
For this test, two VMware View Connection Servers were implemented with load balancing to provide redundancy for the building block. For a 10,000-user pod, VMware View Connection Servers should be deployed in the standard 5 + 2 configuration to support added capacity and provide the highest level of performance and redundancy.

As per VMware best practices, the VMware View Connection Servers were configured as a replica group, with a load balancer in front of the broker providing client network traffic routing. This allows a single View Connection Server to fail and have zero impact to the end-user base.
VMware View Pod and Building Block Details

The tables below detail the components of the VMware View pod and building block.

### 10,000-User VMware View Pod

<table>
<thead>
<tr>
<th>Quantity</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>5+2</td>
<td>VMware View Connection Server Connection Servers</td>
</tr>
<tr>
<td>7</td>
<td>VMware View Building Block Clusters</td>
</tr>
<tr>
<td>2</td>
<td>VMware View Pod Network Core Components</td>
</tr>
<tr>
<td>5</td>
<td>VMware vCenter Server 5 – View Composer Installed</td>
</tr>
<tr>
<td>1</td>
<td>Microsoft 2008 R2 SQL Server</td>
</tr>
<tr>
<td>1</td>
<td>Shared Storage Component</td>
</tr>
</tbody>
</table>

### 1,440-User VMware View Cluster

<table>
<thead>
<tr>
<th>Quantity</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>VMware vSphere ESXi 5 Clusters</td>
</tr>
<tr>
<td>15</td>
<td>VMware vSphere ESXi 5 Hosts</td>
</tr>
</tbody>
</table>

### VMware View Connection Server Configuration

Details for each VMware View Connection Server Connection server were as shown in the following images:

![Table showing VMware View Global Settings](image-url)

*Figure 12: VMware View Global Settings (Included for Clarity)*
The VMware Reference Architecture for Stateless Virtual Desktops on Local Solid-State Storage with VMware View 5

Figure 13: View Connection Server Settings

Figure 14: View Connection Server Authentication Settings
Session Management

Desktop Pools were configured to match what a stateless deployment would be in production. Utilizing this type of design, only a single pool is necessary per cluster. Three pools were created per 1,440-user building block.

Individual user accounts were created in Active Directory and assigned to specific groups. Each Group was entitled to a VMware View Connection Server desktop pool.

<table>
<thead>
<tr>
<th>VIEW MANAGER POOL CONFIGURATIONS - 1,440 USERS</th>
</tr>
</thead>
<tbody>
<tr>
<td>VMs</td>
</tr>
<tr>
<td>600</td>
</tr>
<tr>
<td>600</td>
</tr>
<tr>
<td>600</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>VIEW MANAGER ACTIVE DIRECTORY GROUPS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Group Name</td>
</tr>
<tr>
<td>Entitle_RA_Testing_1</td>
</tr>
<tr>
<td>Entitle_RA_Testing_2</td>
</tr>
<tr>
<td>Entitle_RA_Testing_3</td>
</tr>
</tbody>
</table>

Detailed VMware View Pool Configuration

Once a pool was created, the configuration for each was as follows:

![Figure 15: Pool Settings](image-url)
Provisioning settings were as indicated in the following screen:

![Provisioning Settings](image)

**Figure 16: Provisioning Settings**
Detailed Solid-State Drive Pool Configuration

The datastores for a stateless design are configured in the same way as for a standard shared storage infrastructure. Each set of local SSDs was named hostname_SSD1 so that it would be clear when configuring within the VMware View Connection Servers.

VMware View automatically manages the datastores; no further configuration is required.

At the pool level, go to vCenter Settings, and select browse. Check the datastores necessary for the servers within that pool.

![Figure 17: Configuration of Datastores](image-url)
Virtual Machine Image

Each virtual desktop in the reference architect environment was based on a single optimized Microsoft Windows 7 32-bit template. To reduce the IOPS required per virtual machine image, VMware best practice recommends optimizing the base OS.

An optimized image can expect to have drastic decreases in processor, memory and disk utilization.

All testing in this reference architecture was performed in detail with an optimized image following the VMware View Windows 7 Optimization Guide.

The template configuration is as follows:

<table>
<thead>
<tr>
<th>VIRTUAL MACHINE CONFIGURATION</th>
<th>Quantity</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1,440</td>
<td></td>
<td>Windows 7 32-bit Virtual Machines</td>
</tr>
<tr>
<td>1GB</td>
<td></td>
<td>RAM</td>
</tr>
<tr>
<td>1vCPU</td>
<td></td>
<td>CPU</td>
</tr>
<tr>
<td>24GB</td>
<td></td>
<td>Virtual Disk</td>
</tr>
</tbody>
</table>

Applications contained with the template are:

<table>
<thead>
<tr>
<th>VIRTUAL MACHINE CONFIGURATION</th>
</tr>
</thead>
<tbody>
<tr>
<td>Adobe Acrobat Reader 9</td>
</tr>
<tr>
<td>Microsoft Internet Explorer 8</td>
</tr>
<tr>
<td>Windows Media Player 11</td>
</tr>
<tr>
<td>Microsoft Word 2007</td>
</tr>
<tr>
<td>Microsoft Excel 2007</td>
</tr>
<tr>
<td>Microsoft PowerPoint 2007</td>
</tr>
</tbody>
</table>
**Validation Methodology**

When validating VMware View reference architecture designs, it is important to simulate a real world environment as closely as possible. For this validation, each component was built and validated for the virtual infrastructure necessary in a 1,440-user cluster using a simulated workload. The networking, load balancing and core common infrastructure components were also implemented for the access infrastructure necessary to support a 10,000-user VMware View Pod.

The following diagram provides a high-level overview of the environment used.

![Diagram of Physical Layout](attachment://physical_layout_diagram.png)

**Figure 18: Overview of Physical Layout**

Testing was conducted in two phases. In the first phase, desktop pools were created provisioning the virtual machines to a single server. Each pool was created manually, just as it typically would be created in a normal environment. Testing was performed to find a realistic limit of the single server, which for a 12-core server allowed 120 desktops.

The second phase of the validation included session establishment, logon and execution of a workload across the cluster. Each client access device established individual VMware View session connections to assigned View desktops using the VMware View Client. Once a session was established, the workload was run to simulate typical user activity.

Each session worked in the environment as a standard user throughout the test, during which time the overall system statistics were collected from several components of the architecture. The following sections explain in more detail how each layer was implemented and used as part of the validation and how the workload was implemented.

**Client Access Devices**

To test and validate each layer of the architecture, simulated client access devices were deployed to simulate a real-world environment using VMware Desktop Reference Architecture Workload Simulator (RAWC). The simulated users connect from a client access device through the supporting infrastructure and establish network-based sessions with the View desktops. The RAWC client access devices were implemented separate from the actual building blocks and other infrastructure components, such as Active Directory, DNS, DHCP and file and print services, to allow for realistic simulation of end users that are outside the datacenter.

Each RAWC client access device was implemented using a Windows 2008 R2 Server running the VMware View 5 Client. Over 100 client access devices were deployed, and each was used to establish ten unique VMware View sessions. Each session was established using a unique individual user account that was entitled to use the
desktop pool. During the tests, virtual clients were powered on at regular intervals until all the virtual clients had been powered on. Each simulated client’s 70 sessions were automatically started after being powered on, with a five-second delay implemented between sessions.

The exact RAWC configuration is show below:
**Workload Description**

Each virtual machine was equipped to run a workload that simulates typical user behavior, using an application set commonly found and used across a broad array of desktop environments. The workload has a set of randomly executed functions that perform operations on a variety of applications. Several other factors can be implemented to increase the load or adjust the user behavior, such as the number of words per minute that are typed and the delay between applications being launched.

The workload configuration used for this validation included Office 2007 (including Word, Excel and PowerPoint) and Internet Explorer 8, Adobe Acrobat Reader and Windows Media Player.

During the execution of the workload, multiple applications were opened at the same time and windows were minimized and maximized as the workload progressed, randomly switching between each application. Individual application operations that were randomly performed included:

- **Microsoft Word 2007**
  Open/minimize/close, write random words/numbers, save modifications

- **Microsoft Excel 2007**
  Open/minimize/close, write random numbers, insert/delete columns/rows, copy/paste formulas, save modifications

- **Microsoft PowerPoint 2007**
  Open/minimize/close, conduct slide show presentation

- **Adobe Acrobat Reader**
  Open/minimize/close, browse pages in PDF document

- **Internet Explorer 8**
  Open/minimize/close, browse page

- **Windows Media Player 11**
  Open and view a sample video

Based on the “think time” and words per minute used for this validation, this workload could be compared to that of a high-end task worker or lower-end knowledge worker. Real results would absolutely depend on the environment’s actual usage, but in general the results in this reference architecture are designed to be conservative.

With this workload, it was validated that at least 1,440 users were maintained by this architecture using the provided server, network, storage resources and configuration. In addition to the ability to sustain 1,440 users with fast application response time, the necessary resources were also available to accommodate a host failure within each cluster, as well as to accommodate a moderate amount of growth or unpredicted increase or change in user workload. Depending on the specific environment, additional changes or features implemented and the workload characteristics of your users, you may be able to accommodate more or fewer users.

**High Availability**

A stateless design has different requirements for high availability compared to a standard desktop virtualization deployment. In the past, the design would be based on the redundancy provided by a shared storage device, and would utilize features within VMware vSphere to provide High Availability (HA), Distributed Resource Scheduling (DRS) and live migration with vMotion. With stateless usage of local datastores, these features cannot perform as designed.

In this design, HA is not necessary, as the failure of a host will be seen by the VMware View Connection Servers. In case of server and/or storage failure, the broker will simply allocate a new desktop for the user after successful authentication.

Much thought was given to how the removal of vMotion and DRS would affect the architecture. Depending on the desktop environment and organizational needs, the cost savings simply outweigh the flexibility provided by shared storage. VMware View Connection Servers can address both planned and unplanned downtime at the broker level, which was tested extensively as a part of this reference architecture.
Unplanned Downtime

A standard host-level outage is easily mitigated by careful pool planning. By provisioning a surplus of virtual desktops across the cluster, equal to the number of failed virtual desktops on a given host, the failure of a host is contained. In the design, 120 desktops will be created on each host; however, only 96 would be allocated to users by the broker unless there was a host/storage failure.

### Pool Details

<table>
<thead>
<tr>
<th>Quantity</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1,440</td>
<td>Maximum # of users that can be entitled to the pool</td>
</tr>
<tr>
<td>1,800</td>
<td>Maximum # of virtual desktops the pool can provision</td>
</tr>
</tbody>
</table>

### Per-Host Desktop Load

<table>
<thead>
<tr>
<th>Quantity</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>96</td>
<td>Number of users that exist per host in normal operation</td>
</tr>
<tr>
<td>120</td>
<td>Number of users that exist per host with a failed host</td>
</tr>
</tbody>
</table>

Because the Connection Server will automatically sense that the desktops are no longer available, if a user that was on the failed server attempts to log back in the user will receive a new desktop from another host. This effectively provides “HA,” but at the Connection Server layer.

Planned Downtime

Required scheduled downtime for host maintenance or similar standard planned downtime is addressed by modification within the VMware View Connection Server. Similar to putting a host in maintenance mode, VMware View can also be configured to provide “maintenance mode” for this stateless architecture.

Once a maintenance window has been scheduled by the organization’s standard change control processes, the following steps are undertaken:

1. In a VMware View Connection Server, edit the pool.

   Under vCenter Settings, browse the datastores seen by the pool, and uncheck the datastore for the host on which maintenance is required. This removes it from the process that provisions new virtual desktops, effectively isolating the host for future need.

![Figure 20: Unchecking the Datastore]
2. Within Pool Settings, modify the **Delete or refresh desktop on logoff** field to the “Delete Immediately” setting. This will remove desktops in the environment as each user logs off, and will then redeploy them if the provisioning settings are not already met. Because of the setting in step 1, this will be only applied to the remainder of hosts in the cluster.

3. In **Provisioning Settings**, modify the **Max number of desktops** to equal the cluster size less one server in total VM count. In this architecture there are 600 desktops normally deployed, so the removal of one host **Max number of desktops** would be set to be exactly 480. This will prevent over-provisioning on the four remaining hosts.
4. Place the host into maintenance mode once all desktops have been siphoned off the target host. These actions are undertaken before the maintenance window is scheduled to begin and should allow enough time for users to have logged off once during the time period where the settings were changed. The individual host can be put into maintenance as there would be no desktops powered on.

5. To exit Planned Downtime:
   - Ensure the host is out of maintenance mode
   - Edit the Pool Settings
   - In vCenter Settings > Select Datastores, select the host that was previously excluded for maintenance

![Reselect the Datastore](image)
• For **Provisioning Settings**, increase the **Max number of desktops** to the prior limit of 600.

![Figure 24: Increase the Number of Desktops](image)

• Within **Pool Settings** change the **Delete or refresh desktop on logoff** setting back to “Refresh Immediately.”

![Figure 25: Pool Settings](image)

VMware View will then repopulate the empty datastore to create the new virtual desktops for the pool. As users log in, they will automatically begin using the host that exited maintenance mode and joined the cluster.

The deletion of desktops at logoff can create intensive demands on the infrastructure. Because of the local SSDs, however, this is quite reasonable. (In addition, if the VMware View building block cluster is **clear of all users** at the time of maintenance, steps 1 and 4 can be skipped entirely.) The host can be removed via the other steps, and maintenance mode can be obtained by simply powering off the desktops on the target host.
Validation Results

This section details the results and observations that were concluded during the validation of stateless tiered storage of desktop virtualization with VMware View 5. The results are aligned and specific to the design, which does not require shared storage and thus it is not detailed. Because of this, the results are focused on the individual host.

All validation included several test iterations to ensure data consistency after a ramp-up period; for clarity multiple iterations are not shown.

Testing was performed with an optimized Windows 7 image following best practice detailed here. All data includes the initial period of massive user logons ramping up the test to demonstrate that even when pushed heavily the environment can provide an acceptable range of performance.

The graphs below show the per-host processor and memory usage.

Figure 26: Host CPU Usage in GHz
Figure 27: Host Memory Utilization in GB
Application Response Times

The figures below show the consolidated application response time results for both 96 and 120 workloads on a single host, as well as the individual results for each workload scenario.

Figure 28: Consolidated Application Response Time Results

Figure 29: Application Test Results with 96 Virtual Machines
Solid-State Drive Statistics

Even a single, local solid-state disk is capable of several orders of magnitude higher IOPS (tens of thousands) than seen below; however, more than one was used to provide the capacity necessary.
The initial ramp-up period shows higher latency, but as mentioned earlier this was included to show how well the infrastructure deals with a massive influx of user logons all at once.

Figure 32: SSD Datastore Latency (ms)
Capacity of the SSDs was tracked closely to ensure the environment would have no issue supporting the necessary user counts. Even after multiple test runs, the linked clones grew only modestly. The architecture should allow for a weekly refresh period.
Conclusion

The VMware View 5 architecture for stateless desktops discussed in this reference document significantly reduces the hardware infrastructure costs of desktop virtualization environments. The architecture provides a stateless desktop virtualization environment designed with VMware View 5 tiered storage that can scale from a few hosts to hundreds of hosts, while providing the lowest cost per desktop in the industry. In summary, this design provides linear scalability across both compute and storage, regardless of scale.

About the Author

Mac Binesh is a Group Product Manager in the End User Computing group at VMware. Mac is responsible for design, development, validation and publication of VMware View Reference Architectures. Mac also works with VMware partners (DELL, NetApp, HP, Cisco, EMC, VCE) on the validation of co-branded VMware View Reference Architectures.

Mac has over 30 years of experience in software engineering, software quality assurance, systems engineering and network design working for companies such as Auspex, Veritas, ZeitNet, Adaptec, Sun/iPlanet, Oracle and most recently at Microsoft in Engineering and Lead Technical roles focusing on Network Security, Firewalls, Enterprise Portals, Identity Management and Virtualization.

Acknowledgements

VMware would like to acknowledge the following individuals for their contributions to this paper and help with the test setup and analysis, as well as providing the lab infrastructure, design guidance, content review and building the solution:

VMware—Alexander Kobuzhatsky, Lebin Cheng, Matt Eccleston, Fred Schimscheimer, Mason Uyeda, Rory Clements and Tristan Todd

Intel—Howard Jacob, Animesh Lal and David Blunden

All vSphere performance charts and data were generated using Xangati Performance Management for VMware View. For more information, please visit www.xangati.com/products/virtual-desktop-infrastructure-vdi-dashboard/.

References

VMware View
http://www.vmware.com/products/view/

VMware vSphere 5
http://www.vmware.com/products/vsphere/

Trademarks

Intel, Intel Xeon, and the Intel logo are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.
Appendix A. Design Flexibility

When building a highly scalable, flexible and cost-effective architecture, it is important to view each area of a virtual desktop separately. User data, applications and desktop operating systems must be thought of as dynamic, flexible entities. Each entity must be independent of another entity if the environment is to have the highest level of scalability and cost effectiveness.

In particular for the cost effectiveness of virtualized desktops, it is extremely important to separate user data, as it is the lynchpin to allow floating desktop images necessary to leverage a stateless architecture.

User information should be handled by redirecting key folders to network-based file shares, or by utilizing profile management software such as VMware View Persona Management.

The actual desktop should be viewed as disposable. Stateless architectures, like the reference architecture described in this document, are only able to obtain their benefit when the user data is appropriately abstracted from the virtual desktop.

While user profile management has been perceived as being problematic in the past, with proper design it can be stable and be successfully leveraged on a very large scale. A key to successful roaming profiles is keeping the size of the profiles as small as possible. Using extensive folder redirection, especially beyond the defaults that are found in standard group policy objects, roaming profiles can be stripped down to a bare minimum of files.

Examples of recommended folder redirections include:
• Application Data
• My Documents
• My Pictures
• My Music
• Desktop
• Favorites

Locking down the desktop (specifically not allowing the user to save data locally) is especially important with stateless desktops as the data on the desktop could be eliminated entirely when a user logs out of the system.
Application flexibility can take on many forms. In this specific reference architecture, the application architecture was kept as simple as possible to ensure the results could be easily duplicated, and there was not any lack of clarity in the results. A stateless design using a single floating image does not remove the possibility that multiple use cases can be provided by using application design. Because of the need to keep this reference architecture concise, application packaging, including VMware ThinApp or other deployment methods, is outside the scope of this specific whitepaper.

The Importance of a Phased Approach

A successful virtual desktop strategy must deliver the end user a comparable experience to that replaced by the new solution. If the user experiences degradation in performance, either real or perceived, the acceptance of a new solution is reduced and it may be rejected entirely by the user population.

Users expect the experience that they have on their local, traditional desktops. This is a core reason that many environments with a full, server-based computing model have continued to have a client device running a full desktop operating system at each user’s desk. In the past, a server-based computing model coupled with a thin client device delivered a slower cadence of work when multiple applications were involved, and was often rejected.

Users and devices must be examined in order to create a successful design. The evaluation of usage patterns and the discovery of work requirements are vital. There are key areas to observe in order to establish user needs:

- Physical location of the user
- Connection type from client device to desktop
- Job requirements specific to the application set
- Client device type
- Mobility requirement

Physical Location

The proximity of the user to the datacenter (and thus their desktop) can determine what options are available to the end user. A user that is in the same building as an organization’s datacenter might be provided a different desktop deployment method. If a user is in a branch office, often there will be additional desire to provide that user with a desktop that is centralized so that the desktop can be more easily maintained. Because of the centralization, the application itself is now within the datacenter, potentially providing the perception of performance increase to the end user.

Connection Type

The connection to which a user has access is often directly related to their physical location. A user that is connected to the datacenter by an extremely fast link can often be provided a much richer desktop experience compared to a user that is in a branch office with a limited Internet connection. If a user does have a limited connection, more attention must be paid to graphically intense applications, large print jobs and procedures like scanning. Any process that requires a large amount of data to traverse between the client and the back-end desktop operating system will need scrutiny. In the case where the connection is extremely limited or only available for certain periods of time, an offline desktop will be mandatory for that user.

Application Sets

The applications in an organization must be tested to see how they function across a remote desktop with a given connection type. Certain graphically intense applications will not work as desired on some limited connections, and even a higher speed local connection may still have latency issues that cause a failure to provide an acceptable end-user experience. Only complete and detailed analysis and testing will provide test answers for some applications.

Client Device Type

In certain scenarios, the client device hardware can be a factor in what model to choose. While even the most basic client device can handle nearly all remote desktop methods of accessing a desktop, there are methods
of offloading graphics from the back-end infrastructure to the client device to provide a better user experience. These methods often require specific client device requirements.

**Mobility Requirements**
Similar to connection type, the ability of a user to work offline will lead to that user needing an offline desktop operating system.

**Defining Roles**
This selection procedure establishes and defines user demographic roles or “use cases.” Because every environment is different, it is difficult to segregate users into quantified standard groups. However, several example core roles might include:

- **Task User**—A task user is characterized by the need for just a few basic applications, requiring minimal computing resources. Often connected to the organization’s network, the task user needs only to have internal access to a desktop. These users are data-entry clerks, call center employees, manufacturing personnel, or fill similar roles.

- **Standard User**—The standard user is the “general all-purpose” user in an environment. These users typically need a wide range of business applications, sometimes including more graphically intensive items such as training videos. Most commonly the application set is not intense, however it can be demanding from a manageability perspective. Standard users are normally connected and online in the office; however, they are sometimes granted remote access to the corporate network. This can include remote access to the entire desktop, or to items such as web applications, web email, or other remote technologies.

The standard user is the focus of this reference architecture.

- **Mobile User**—A mobile user is often similar to the standard user, however they need to be completely disconnected from the network at certain times. Because of this, offline technologies must be available for the user for the desktop, applications and user data. Laptop users are the key example of this role.

- **High-end Graphics/Computing User**—This type of user is characterized by an application or application set. A high-end graphics user works with an application and requires extremely low-latent screen refreshing, and often the ability to have hardware-accelerated 3D graphics in their end-user device. This same role also includes users with extremely high computational applications that require dedicated resources to function effectively. Users that need CAD, engineering, financial and certain high-end health care applications are examples of this role.

**Virtual Desktop Feasibility Assessment**
Applications, roles and overall environments differ, so it is important to perform a feasibility assessment as well as a proof of concept (POC) within the actual environment being considered for desktop consolidation. The feasibility assessment is the process by which an examination of user needs is defined, and the determination of appropriate action is detailed.

The actual use of resources from a traditional desktop is particularly important for virtual desktop infrastructure and should be collected for a period of time, typically at least two weeks. Application use cases should then be closely examined, especially those with known extensive graphical needs. In addition, the amount of local data necessary within the desktop is also important to collect for ROI modeling. This data can also be collected within the assessment process.

Typically, once the high-level data has been collected, several user demographic roles will clearly appear. Each of the roles should have a proof-of-concept performed to determine how the concept will actually work for that type of end user’s needs. The desktop virtualization solution’s manageability benefits can also be easily determined with the POC. Often, specific end users will work in a POC environment regularly for day-to-day computing needs to truly “try out” the model. This POC usage can be provided in addition to their previous desktop experience because of the flexibility in end-user devices that VMware View can provide.
Typically, the desktop virtualization architectural design is modified based on the feedback from the end-user acceptance testing from a POC phase, providing a considerably more accurate overall roll-out plan and reducing costs.

Sometimes, the POC phase is performed prior to an assessment to showcase the idea of virtual desktops. However, the assessment phase is still the recommended first step to obtain overall ROI metrics and discover the proper user roles to provide a detailed design.

**Virtual Desktop Cost**

One of the goals of a virtual desktop feasibility assessment should be to determine the cost per virtual machine for each user demographic role identified. There are many variables for the various products that constitute desktop designs.

The actual cost of a desktop varies depending on the products used to comprise the solution and cannot be calculated without further environment-specific research and the desire to look at all the costs involved – including the more difficult to quantify “soft” costs, such as labor savings.

Here are key items that must be considered for desktop consolidation:

- Client device costs, either a thin client or thick client
- Server-side hardware costs
- Storage needs, focusing on capacity and IOPS
- Back-end licensing costs
- Virtualization Hypervisor licensing costs
- Desktop Virtualization connection broker technologies
- Operating System licensing costs of the desktop operating system
- Desktop operating licensing needs, especially the requirement for a Microsoft Virtual Desktop Access license when running Windows desktop operating systems as a virtual desktop

There is a need to include the ongoing soft costs which might include:

- Costs for maintenance (patching)
- Resources required for upgrades (for example upgrading to a new OS)
- Security (cost of lost data from a laptop, hard drive corruption data loss, etc.)
- Lost worker productivity in desktop failure

Regardless of user demographic role, each of the items warrants careful consideration in the overall desktop cost. There are hidden costs often overlooked in desktop consolidation solutions that may or may not be outweighed by the better security, manageability and flexibility of the environment.

**Ensuring Success**

Proper planning and working with experienced desktop virtualization specialists can increase the success of a desktop virtualization project. Poor performance, extremely high cost estimates and component failure could be the result of an evaluation done internally without experienced help. A “false start” with any technology or solution can be very difficult to recover from, resulting in user and management doubts. There are simply too many immediate benefits with desktop virtualization to risk any type of false start.

With any new technology project, organizational changes can slow progress. Potential organizational changes should also be considered to help increase the success of a desktop virtualization project. Because of where the desktop operating system is located, and changes with what is being managed where, it is completely normal
to wonder “who will be responsible for what?” It is common for server staff to be accustomed to supporting the equipment that is in the datacenter, and for the desktop administrators not to be included with items located in the datacenter.

In some cases, it may be necessary to review organizational group member responsibilities for the desktop infrastructure. Often, a new group is formed comprised of both server and desktop teams to support a new more diverse desktop computing environment. Change control policies should be reviewed. Modifications should be addressed and documented.
Appendix B. VMware View Reference Architecture Design Approach

All VMware View reference architectures leverage basic design principles and best practices. Using a building block approach allows for the flexibility of creating a comprehensive virtual desktop infrastructure that performs or exceeds desired goals and functionality while maintaining a logical, straightforward architecture.

VMware View reference architecture begins with the user, or Client Access Devices layer, and continues through the Session Management layer (see the figure below). This approach provides a clear definition of services necessary to allow each functional area to be defined independently from the others, while still providing a cohesive structure for addressing the interdependency of all solution components.

![Figure 36: View Reference Design](image)

The VMware View reference architecture addresses the following commonly required aspects of an enterprise class solution:

**Standardization**
By using a building block approach and common components, the VMware View Reference Architecture offers IT organizations the ability to implement a predictable, familiar solution.

**Repeatability**
The building block approach is designed to be repeatable across differing lines of business and departments.

**Scalability**
The building block approach allows organizations to scale out to meet growing demand for thousands of users predictably.

**Availability**
Each layer of the VMware View Reference Architecture is designed to offer the highest level of availability and resiliency.
**Security**

VMware View Reference Architecture takes security into account in each layer to offer a secure solution.

**Integration**

The VMware View Reference Architecture addresses the integration with components commonly found in today’s enterprise.

This VMware View Reference Architecture also references and includes several deployment guides that provide detailed instructions on deploying many of the components used to validate the architecture.

**VMware View Design Approach Component Mapping**

To help better understand how the various overall desktop virtualization components map to the VMware View design approach, see the figure below.

Each layer has specific items that must be considered in detail.

*Figure 37: View Component Sub-Layers*

**Client Access Devices**

This layer includes the physical devices that provide users access to their virtual desktop.

Sub-layers include:

- Client device
- Client software
- Peripheral support
Access Infrastructure
Networking and connectivity components designed to facilitate client communication are addressed here.

Sub-layers include:
• Local and wide area networking
• VMware View Connection Server
• Network load balancing and optimization

Virtual Infrastructure
This layer defines the components and technology used to host the virtual desktop operating systems and supporting VMware View infrastructure.

Sub-layers include:
• Host infrastructure
• Virtual and physical network infrastructure
• Storage infrastructure, including local SSD technology

View Desktops
This layer defines the components and configuration of the virtual machines assigned to and accessed by users.

Sub-layers include:
• Virtual hardware configuration
• Virtual desktop guest operating system
• Application deployment methodology
• VMware vCenter with View Composer configurations

Session Management
This layer defines the deployment and management of a large number of virtual desktops to an end-user community. It also includes the integration with existing desktop infrastructure services, such as Active Directory, for maintaining user and computer accounts. Components and sub-layers control user authentication, virtual desktop provisioning, and deployment as well as user entitlement to desktop resources.

Sub-layers include:
• Desktop provisioning and pool management
• Session monitoring
• Active Directory integration
• Virtual printing
Appendix C. VMware View Reference Architecture Components

Client Access Devices

The client access device layer is comprised of the hardware and software components needed to deliver a PC-like experience. The process for choosing the appropriate client device varies from deployment to deployment; mixed client environments are common in VMware View deployments. Typically, users are segmented based on their needs and requirements during the planning and design phases, and business requirements and goals are also taken into consideration and mapped to the needs of the user segments.

For example, an organization might have PCs that are on a staggered depreciation schedule. Depreciated assets can be replaced immediately by thin client devices, but assets that have not fully depreciated are often converted into unmanaged endpoints, typically by converting them to PXE-booted clients using a Linux-based solution. An alternative is to tightly lock down the currently installed Windows OS and repurpose them as-is. Either approach can offer the flexibility to gain the highest return on PC hardware investment. Some key considerations for an IT organization during this process are:

• Will the client endpoint fulfill user requirements?
• Do reduced management benefits balance the effort to convert existing PCs to repurposed PCs?
• Are there any licensing benefits to keeping the current client endpoints?
• VMware View supports a wide range of client endpoint devices. This offers IT organizations a great deal of flexibility and deployment options when rolling out a VMware View-based solution.

Typical client devices include:

Physical PCs
Using the VMware View Client, users can easily access their virtual desktop from any standard Windows-based PC from within the enterprise network or, if given access, from a home PC.

Repurposed PCs
It is not uncommon for an IT organization to repurpose PCs as VMware View Universal Clients. In some cases, the existing Windows OS is tightly locked down, so only the VMware View Client for Windows can be run to access a virtual desktop. Some organizations convert the PCs into PXE boot clients that use a minimal version of Linux and VMware View Web Access to access the virtual desktop instances.

Thin Clients
VMware View supports a wide array of Thin Client devices. The most currently qualified devices are listed in the VMware View Thin Client Compatibility Guide.

Mobile User Devices
Mobile users typically have laptops as their client devices, and they often have limited or no network connectivity. VMware View offers a unique way to address such users, with a feature called View with Local Mode. Using local mode, an entitled user can check a VMware View virtual desktop out to the laptop device and use it offline. Once network connectivity is restored, the local mode desktop synchronizes the changes in the background. Alternatively, the user can initiate a check-in that synchronizes the virtual desktop with the virtual datacenter.
Access Infrastructure

The access infrastructure provides network connectivity between client access devices and the virtual infrastructure that hosts the virtual desktop sessions, including the components that manage, or broker, user connection requests to entitled desktops. This is a critical layer with respect to the overall user experience; a properly designed, sized and functioning access infrastructure is vital to a successful VMware View implementation. Undersized or underperforming access infrastructures can result in poor performance and a less than satisfactory user experience.

Local and Wide Area Networking

Due to its reliance on network communication, a VMware View session is affected by the capabilities of the underlying network infrastructure more than a traditional PC. The solution covered in this document focuses on large-scale, LAN-based deployments. WAN-based deployments are covered within separate reference architectures.

Remote Desktop Protocol Considerations

The display protocol most commonly used to access a VMware View 5 environment is VMware’s PC-over-IP (PCoIP) protocol, which delivers an end-user experience equal to the current experience of using a physical PC. As a starting point, to maximize use of available network bandwidth and enhance each user’s experience, IT organizations should follow the recommendations in the VMware View 5 with PCoIP Network Optimization Guide.

The VMware View Client also allows a large number of customized settings. These settings can be centrally controlled and managed through the use of Microsoft group policy objects, and are covered in detail in the VMware publication, VMware View Connection Server Installation and Administration Guide.

In addition to PCoIP and standard Microsoft Remote Desktop Protocol (RDP), VMware View also supports protocols provided by other manufacturers, such as the HP RGS for HP physical machines.

Network Layer Performance Recommendations

To achieve a user experience similar to that of a traditional PC, VMware View requires an optimal network infrastructure, so both network bandwidth and latency should be taken into consideration in the design phase. Studies have shown that the minimum bandwidth for a useable PCoIP Session is approximately 100Kbps. Streaming multimedia content using multimedia redirection increases bandwidth requirements.

The amount of bandwidth needed per virtual desktop user varies, depending on the user workload and how active the use. As a starting point, 100-150Kbps is a good rule of thumb to use until environment-specific usage patterns can be determined. This range helps account for the peak usage of a typical VMware View user. As noted above, multimedia usage increases the amount of bandwidth needed, but it is content-specific and typically not needed on a regular basis.

Even more critical than bandwidth is network latency, an expression of how much time it takes for a packet of data to get from one designated point to another. It is unavoidable in a network environment, especially at long distances. Transmission and equipment delays typically create latencies that may be as low as 1ms on a Local Area Network (LAN), and are typically 50-100ms for U.S. domestic links on a Wide Area Network (WAN). For international links, latencies can range from 100-200ms, sometimes higher, while multi-hop satellite links can produce delays of over 2,000ms.

High network latency can contribute to a slow refresh of the desktop session and have an adverse impact on the user experience. For simple tasks such as typing, cursor motion and mouse selection response time should be less than 150ms. As latency approaches 200ms, the user experience is often impacted even further. It is often difficult to categorize how remote offices or users will experience a VMware View desktop under these conditions. Third-party solutions, such as WAN Optimization solutions from Cisco Systems, have been used successfully in combination with remote protocols to overcome limitations on high-latency networks. Both...
solutions integrate tightly with a VMware View solution. VMware recommends careful monitoring of both local and wide area connectivity during and after VMware View implementations.

From the VMware vSphere server, it is also important to consider that in addition to the display traffic, each virtual desktop also needs adequate speed for typical data traffic like email, file transfers and web browsing.

**VMware View Load Balancing**

The primary purpose of load balancing in a VMware View architecture is to optimize performance by distributing desktop sessions evenly across all available VMware View Connection Servers. It also improves serviceability and availability by directing requests away from servers that are unavailable, and improves scalability by distributing connection requests automatically to new resources as they are added to the VMware View environment.

Several approaches are available. For example, round-robin DNS, while technically the simplest load balancing solution to implement, has a significant disadvantage from a failover perspective. If a server fails, it must be manually removed from the DNS list of records corresponding to the load-balanced domain name. Another issue with a round-robin DNS approach arises in the remote-access use case, where VMware View clients are accessing their virtual desktops across the Internet through VMware View Security Servers. In this case, the responses of the master DNS server are cached in upstream DNS servers, and it can take several hours for a DNS name deletion to replicate to all Internet DNS servers. When a server is out of service, client connections can fail if they are directed to it during the time it takes for the cached record to expire across the Internet DNS servers.

Support for a redundancy and failover mechanism, typically at the network level, prevents the load balancer from becoming a single point of failure. For example, Virtual Router Redundancy Protocol (VRRP) communicates with the load balancer to add redundancy and failover capability. If the main load balancer fails, another load balancer in the group automatically starts handling connections.

To provide fault tolerance, a load-balancing solution must be able to remove failed VMware View server clusters from the load-balancing group. How failed clusters are detected may vary, but regardless of the method used to remove or blacklist an unresponsive server, the solution must ensure that new, incoming sessions are not directed to the unresponsive server. If a VMware View server fails or becomes unresponsive during an active session, users do not lose data. Instead, desktop states are preserved in the virtual desktop so that when users reconnect to a different VMware View connection server in the group, their desktop sessions resume from where they were when the failure occurred.

**Virtual Desktop Management Services**

The end user within a VMware View environment connects by logging into a VMware View Connection Server. A Connection Server integrates with Windows Active Directory, and provides access to a virtual desktop that is hosted on VMware vSphere, a blade or physical PC, or a Windows Terminal Server.

VMware View Connection Server is the primary component of VMware View 5. The main service that View Connection Server provides for the access layer is brokering incoming client requests to virtual desktop instances; that is, it helps to direct entitled users to their virtual desktops.

VMware View Connection Server includes advanced functions for provisioning and managing both new and existing virtual machines. These include workflow controls for provisioning new desktops and power control policies that can suspend and resume virtual desktops as desired. A large-scale implementation of VMware View requires such policy-based controls to reclaim unused memory and CPU resources when virtual desktop sessions are terminated or idle for long periods of time.
VMware View Connection Servers

The VMware View Connection Server performs the following functions:

• Validates the user name and provides a connection for that user

• Provides the ability for the user to access multiple virtual desktop pools. If the user is permitted to access a variety of pools, View prompts the user to select a pool at login time

• Monitors the activity level of a given virtual machine and set status to active or inactive

• Automatically provisions new virtual desktops to maintain availability levels

• Handles reassignment of a virtual desktop when a user disconnects
Appendix D. VMware View Components

Typical VMware View deployments consist of several common components, illustrated the figure below, which represents a typical architecture. It includes VMware View components as well as other components commonly integrated with VMware View.

![High-Level VMware View Architecture](image)

Figure 38: High-Level VMware View Architecture
**VMware View Connection Server**

The View Connection Server is a full-featured connection broker. It is the key element that directs traffic within the architecture as a whole. As referenced above, this is the component that provides access to the desktop to the end user client access device.

**VMware View Security Server**

Outside the corporate firewall, in a DMZ, you can install and configure View Connection Server as a security server. Security servers in the DMZ communicate with View Connection Servers inside the corporate firewall. Security servers offer a subset of functionality and are not required to be in an Active Directory domain.

**VMware View Administrator**

This Web-based application allows administrators to configure View Connection Server, deploy and manage View desktops, control user authentication and troubleshoot end-user issues.

When a View Connection Server instance is installed, the View Administrator application is also installed. This application allows administrators to manage View Connection Server instances from anywhere without having to install an application on their local computer.

**VMware View Agent**

A View Agent is installed on all virtual machines, physical systems and Terminal Service servers that you use as sources for View desktops. This agent communicates with View Client to provide features such as connection monitoring, virtual printing and access to locally connected USB devices.

If the desktop source is a virtual machine, a View Agent service is installed on a virtual machine and then used as a template or as a parent of linked clones. When you create a pool from this virtual machine, the agent is automatically installed on every virtual desktop.

The agent can be installed with an option for single sign-on. With single sign-on, users are prompted to log in only when they connect to a View Connection Server and are not prompted a second time to connect to a virtual desktop.

**VMware View Client**

The client software for accessing View desktops runs either on a Windows or Mac PC as a native application, or on a thin client if you have View Client for Linux.

After logging in, users select from a list of virtual desktops that they are authorized to use. Authorization can require Active Directory credentials, a UPN, a smart card PIN or an RSA SecurID token.

An administrator can configure View Client to allow end users to select a display protocol. Protocols include PCoIP, Microsoft RDP and HP RGS for View desktops that are hosted on HP Blades. The speed and display quality of PCoIP rival that of a physical PC.

View Client with Local Mode (formerly called Offline Desktop) is a version of View Client that has been extended to allow end users to download virtual machines and use them on their local systems regardless of whether they have a network connection.

**VMware View Portal**

From a Windows PC or laptop, end users can open a Web browser and use View Portal to download, install, update and start the Windows-based View Client. As of View 5, View Portal installs the full View Client for Windows, with or without Local Mode.

To use View Portal, end users open an Internet Explorer browser and enter the URL of a View Connection Server instance. View Portal provides a link for downloading the installer for the full View Client for Windows.
**VMware View Composer**

This software is installed on a vCenter Server instance that manages virtual machines. View Composer can then create a pool of linked clones from a specified parent virtual machine. This strategy reduces storage costs by up to 90 percent.

Each linked clone acts like an independent desktop, with a unique host name and IP address, yet the linked clone requires significantly less storage because it shares a base image with the parent.

Because linked-clone desktop pools share a base image, an administrator can quickly deploy updates and patches by updating only the parent virtual machine. End users’ settings, data and applications are not affected. As of View 5, linked-clone technology can be used for View desktops that are downloaded and checked out to use on local systems.

**VMware View Transfer Server**

This software manages and streamlines data transfers between the datacenter and View desktops that are checked out for use on end users’ local systems. View Transfer Server is required to support desktops that run View Client with Local Mode (formerly called Offline Desktop).

Several operations use View Transfer Server to send data between the View desktop in vCenter Server and the corresponding local desktop on the client system.

- When a user checks in or checks out a desktop, View Manager authorizes and manages the operation.
- View Transfer Server transfers the files between the datacenter and the local desktop.
- View Transfer Server synchronizes local desktops with the corresponding desktops in the datacenter by replicating user-generated changes to the datacenter.
- Replications occur at intervals that are specified in local-mode policies. Replication may also be initiated in View Administrator. Policies can be set that allow users to initiate replications from their local desktops.
- View Transfer Server keeps local desktops up-to-date by distributing common system data from the datacenter to local clients. View Transfer Server downloads View Composer base images from the image repository to local desktops.
- If a local computer is corrupted or lost, View Transfer Server can provision the local desktop and recover the user data by downloading the data and system image to the local desktop.

**VMware vCenter Server**

This service acts as a central administrator for VMware vSphere servers that are connected on a network. VMware vCenter Server, formerly called VMware VirtualCenter, provides the central point for configuring, provisioning and managing virtual machines in the datacenter.

In addition to using these virtual machines as sources for View desktop pools, virtual machines can be used to host the server components of VMware View, including Connection Server instances, Active Directory servers, vCenter Server instances, etc.

You must install View Composer on the same server as vCenter Server to create linked-clone desktop pools. vCenter Server then manages the assignment of the virtual machines to physical servers and storage and manages the assignment of CPU and memory resources to virtual machines.
Desktop and Pool Management

A pool can be created with one or hundreds of virtual desktops. VMware View offers the ability to create and provision pools of desktops as a basis of centralized management. The VMware View Connection Server manages this session-layer component.

Virtual desktop pools are created from one of the following sources:

- A physical system such as a physical desktop PC or a Windows Terminal Services server
- A virtual machine that is hosted on a vSphere server and managed by vCenter Server
- A virtual machine running on VMware Server or another platform that supports View Agent

If a vCenter virtual machine is used as a desktop source, the process of making the necessary identical virtual desktops can be automated. The minimum and maximum number of virtual desktops to be generated for the pool can be set. Setting these parameters ensures that the View environment always has enough View desktops available for immediate use, but not so many that available resources are compromised.

Using pools to manage desktops allows applying settings or deploy applications to all virtual desktops in a pool. The following examples show some of the settings available:

- Specify which remote display protocol to use as the default for the View desktop and whether to let end users override the default
- Configure the display quality and bandwidth throttling of Adobe Flash animations
- If using a virtual machine, specify whether to power off the virtual machine when it is not in use and whether to delete it altogether
- Specify if the View desktop can or must be downloaded and run on a local client system

In addition, using desktop pools provides many conveniences.

Individual Desktops

The pool manager can control the power state of virtual desktops such as virtual machines, personal computers, or blade PCs available through VMware View Connection Server. Users with the proper entitlements can check their virtual desktops out for offline desktop use.

Dedicated Pools

Each user is assigned a particular View desktop and returns to the same virtual desktop at each login. Users can personalize their desktops, install applications and store data.

Floating Pools

The virtual desktop is optionally deleted and re-created after each use, offering a highly controlled environment. A floating desktop use case could be a computer lab or kiosk environment where each desktop is loaded with the necessary applications and all desktops have access to necessary data. The use of floating pools allows for lower cost per virtual machine, as well as increased overall flexibility long term for a desktop consolidation effort.

Microsoft Terminal Services Desktop Pool

With this type of pool, Terminal Services sessions can be managed by VMware View Connection Server and provided to VMware View users.
Desktop Persistence

Automated and manual desktop pools support two types of desktop persistence. Dedicated desktops are assigned to individual users and the desktop stays assigned to that user until an administrator makes a change. Properly entitled users may also check their desktops out for offline use. This type of pool is best for users who want to customize their desktops by installing additional applications and storing local data.

Floating, or “stateless,” desktops are allocated to users temporarily and used only for the current session. Once the user has logged off, the desktop goes back into the pool and becomes available for the next user. This type of pool allows for extreme flexibility between the operating system, application and user data layers when properly architected. When all desktop operating system images are exactly the same, heavy storage capacity and IOPS savings can be obtained within this type of desktop. This clean machine is provided for each user session, so the end user can only customize the desktop in a way that can be saved at a per-user level in a profile.

The VMware View Connection Server(s) also provides a set of power management policies that can be leveraged to further increase the dynamic nature of a VMware View solution. This gives administrators more control over the virtual machine behavior. Virtual machines that are idle and still consuming resources can be put into a suspended or powered off state, freeing up resources for active users. Power control policies can be applied to virtual machine–based individual desktops, automated desktop pools and manual desktop pools with the following power policies:

• Remain on—Once started, VMware View Connection Server will not power the machine down. If a virtual desktop is powered down, for example using the VMware vCenter client, VMware View Connection Server automatically starts it when needed.

• Always powered on—VMware View Connection Server ensures that any virtual desktop with this policy applied is powered on all the time. If a virtual desktop is powered down, VMware View Connection Server immediately powers it up again.

• Suspend when not in use—If a virtual desktop is not required, it is suspended. This policy is applied to individual desktops and assigned a dedicated virtual desktop when the user logs off. It is also applied to stateless virtual desktops when there are too many available virtual desktops. For example, this can be triggered by a virtual desktop being returned to the pool when a user has logged off.

• Power off when not in use—If a virtual desktop is not required, it is powered off. This is just like the “Suspend when not in use” policy, except that the virtual desktop is completely powered off.

Storage Flexibility

VMware View 5 provides two critical storage capabilities for desktop virtualization:

• Reduction of storage capacity usage with VMware View Composer, which creates a master desktop image to deploy across the environment. The master image ensures consistency across the infrastructure and simplifies management with ease of patching, updates and deployment. VMware View Composer allows storage reduction of 50 to 90 percent.

• The ability to direct various needs of the desktop image storage to different tiers of storage, including the usage of local host-based storage, for caching and I/O offload.

Capacity Savings

View Composer uses a base image, or parent virtual machine, to create a pool of linked-clone virtual machines. Each linked clone acts like an independent desktop, with a unique host name and IP address, yet the linked clone requires significantly less storage.

When a linked-clone desktop pool is created, a full clone is first made from the parent virtual machine. The full clone, or replica, and the clones linked to it are placed on the same datastore. If necessary, the rebalance feature can be used to move the replica and linked clones from one datastore to another. The paging and temp
files can also be redirected from a virtual machine’s operating system to a separate disk, allowing for flexibility with the storage I/O created by those type of “disposable” files.

When a dedicated desktop pool is created, View Composer also creates a separate user data disk for each virtual desktop. The end user profile and application data are saved on the user data disk. When a linked clone is deleted, the user data can be preserved. When an employee leaves the company, another employee can access the departing employee’s user data. A user who has multiple desktops can consolidate the user data on a single desktop. VMware recommends that you keep persistent user data drives on a separate datastore. The whole datastore that holds these virtual drives can then be part of a backup process.

Because linked-clone desktop pools share a base image, updates and patches can be deployed quickly by updating the parent virtual machine.

The recompose feature allows changes to be made to the parent virtual machine, take a snapshot of the new state and push the new version of the image to all, or a subset of, users and desktops.

This feature can be used for the following tasks:

- Applying operating system and software patches and upgrades
- Applying service packs
- Adding applications
- Adding virtual devices
- Changing other virtual machine settings, such as available memory
To disallow users from adding or removing software or changing settings, the refresh feature can be used to bring the desktop back to its default values. This feature also reduces the size of linked clones, which tend to grow over time.

Figure 40: Upgrading from SP1 to SP2

VMware View Connection Server supports a variety of desktop sources; therefore it is recommended that a robust profile management solution be implemented.

In a floating design, profile management of some type is mandatory to provide that majority of use cases. Profile management helps to ensure that personal settings will always be available to users who are entitled to multiple back-end resources, regardless of the system they are accessing. Profile solutions also help to reduce logon and logoff times and to ensure profile integrity.

Storage I/O Contention

The ability to reduce storage was first introduced with VMware View 3.0. Similarly, VMware View 5 reduces storage by leveraging a base disk, and using linked clone technology to direct individual desktops to that base image.

This reduction of storage creates intense density within shared storage, particularly during intensive I/O activities, including the boot up process, log-on process, anti-virus scans, or similar actions that are not optimized for a consolidated desktop environment.

There are multiple approaches that are being offered in the marketplace today to address this issue. The goal is to reduce or otherwise address the volume of I/O traffic generated by the desktop environment’s operating system and applications. These approaches have yet to show any significant cost improvements over traditional SAN-based desktop virtualization architectures while still providing a scalable, supportable and standardized infrastructure.

This reference architecture is unique as the design takes full advantage of new key features enabled by VMware View 5 tiered storage capability. Instead of centralized caching, this architecture uses solid-state drives as a local per-host “cache” to offload desktop virtualization storage traffic. This provides an increased amount of local inexpensive IOPS capability, but still provides a stateless design for a floating desktop pool. A stateless desktop enabled by a floating image is ideally suited for many standard users in a typical environment.
Storage I/O Flexibility

Because of the separation and flexibility of VMware View 5 tiered storage components, VMware View 5 now has the capability to profoundly affect the physical storage I/O usage that is associated with a typical virtual desktop. This is performed by directing specific types of desktop storage to different virtual disks, residing in various physical storage locations including local host drives.

Because of this flexibility of the VMware View 5 architecture, it can now utilize non-traditional means to reduce the IOPS requirements of a particular desktop.

Specifically, it is possible to provide a desktop using local storage as “cache” for the desktops, reducing the need for SAN-based storage and IOPS within that shared storage architecture.