NVIDIA VGPU™ DEPLOYMENT GUIDE FOR VMWARE HORIZON 7.5 ON VMWARE VSPHERE 6.7

TABLE OF CONTENTS

Chapter 1. About this Document 1
1.1 Formatting Conventions 1
1.2 Related Documentation 3
1.3 Support Contact Information 3
1.4 Acronyms 4
1.5 The Lab 5
Chapter 2. Getting Started 7
2.1 Why NVIDIA vGPU? 7
2.2 NVIDIA vGPU Architecture 8
2.3 Supported GPUs 10
2.4 TESLA GPU Specifications 15
2.5 Supported Graphics Protocols 16
2.6 Homogenous Virtual GPUs 17
2.7 Before You Begin 18
2.7.1 Building an Evaluation Plan 18
2.7.2 Size Your Environment 18
2.7.3 Choosing Your Hardware 19
2.7.4 General Prerequisites 20
2.7.5 Current Builds 22
2.7.6 Pre-Installation Preparation 22
2.7.7 Server BIOS Settings 22
Chapter 3. Installing vSphere 6.7 25
3.1 Choosing Install method 25
3.2 Preparing USB Boot Media 26
3.3 Installing vSphere 6.7 27
3.4 Initial Host Configuration 33
Chapter 4. Installing VMware vCenter 6.7 Server 37
4.1 Installing vCenter 6.7 Server Appliance 37
4.1.1 About VCSA 37
4.1.2 vCenter Server Appliance (VCSA) Installation 40
4.2 POST Installation 53
4.2.1 Adding Licenses to Your vCenter Server 53
4.2.2 Adding a Host 58
4.2.3 Setting the NTP Service on a Host 62
4.2.4 Setting a vCenter Appliance to Auto-Start 63
4.2.5 Mounting an NFS ISO Data Store 66
Chapter 5. Building Horizon 7 68
5.1 Installing Horizon 7 68
5.2 Registering vCenter Server 80
5.3 Installing View Composer Server 82

5.4 Registering View Composer Server 96
5.5 Installing the Horizon Client 102
Chapter 6. NVIDIA VGPU Manager VIB Installation 106
6.1 Preparing the VIB For Install 106
6.1.1 Uploading VIB in vSphere 107
6.2 Installing the VIB 108
6.3 Updating the VIB 112
6.4 Verifying the Installation of the VIB 113
6.5 Uninstalling VIB 115
6.6 6.0 Only: Configuring Suspend and Resume for VMware vSphere 116
6.7 Changing the Default Graphics Type in VMware vSphere 6.5 and Later 117
6.8 Changing vGPU Scheduling Policy 119
6.8.1 vGPU Scheduling Policies 120
6.8.2 RmPVMRL Registry Key 120
6.8.3 Changing the vGPU Scheduling Policy for All GPUs 121
6.8.4 Changing the vGPU Scheduling Policy for Select GPUs 122
6.8.5 Restoring Default vGPU Scheduler Settings 123
Chapter 7. Selecting Correct vGPU Profiles 124
7.1 The Role Of The vGPU Manager 124
7.2 The Full List of vGPU Profiles 124
7.3 Bringing Quadro to Virtualization 145
7.4 Matching Profiles to User Needs 145
7.5 Creative and Technical Professionals 146
7.6 Knowledge Worker Profiles 147
Chapter 8. Application Based VDI Host sizing 148
8.1 How Many Users Can I Get On A Host? 148
8.2 An Example Application: Siemens NX 149
8.3 Application Based Sizing 149
8.3.1 Recommended System Requirements 150
8.4 Testing Methodology 150
Chapter 9. Creating your first vGPU Virtual Desktop 157
9.1 Creating a Virtual Machine 157
9.2 Installing Windows 164
9.3 Installing VMware Tools on the VM 168
9.4 Adding the Golden Master to the Domain 172
9.5 Installing Horizon Agent 175
9.6 Installing Horizon Direct Connection 180
9.7 Optimizing Windows 183
9.8 Additional Virtual Machine Settings 187
9.9 Enabling the NVIDIA vGPU 187
9.10 Switching The Mode on a Tesla M60 or M6 192
9.11 Disabling ECC Memory 193
9.12 Installing NVIDIA Driver in Windows Virtual Desktop 195
9.13 Installing NVIDIA Driver in Linux Virtual Desktop 200
9.14 Licensing NVIDIA vGPU 202

9.14.1 Licensing NVIDIA vGPU on Windows 203
9.14.2 Licensing NVIDIA vGPU on Linux 205
9.15 Finalizing the Installation 205
Chapter 10. Creating a Horizon vGPU Pool 211
10.1 Creating a Template 212
10.2 Creating a Customization Specification file 213
10.3 Provisioning a Single Virtual Machine 219
10.4 Creating Full Clone Desktop Pools 222
10.5 Creating Linked Clone Desktop Pools 235
10.6 Entitling Users Access to Desktop Pools 246
10.7 Connecting VMware Horizon Client to the View Connection Server 248
Chapter 11. Proof of Concept, Monitoring, & Benchmarking 251
11.1 Proof of Concept 251
11.2 Testing Virtual Vs Physical Desktops 252
11.3 NVIDIA System Management Interface 252
11.3.1 NVIDIA VIRTUAL GPU in the VMware Ecosystem 253
11.4 Links to Apps and Tools 259
11.4.1 Browsers, Applications, & Tools 259
Chapter 12. Troubleshooting 261
12.1 Forums 261
12.2 Filing a Bug Report 262
Chapter 13. Using MobaXterm 264
13.1 Connecting to the Host Server 265
13.2 Using the MobaXterm Text Editor 266
13.3 Uploading a .VIB File 267
Chapter 14. Using WINSCP 270

LIST OF FIGURES

Figure 2-1. NVIDIA vGPU Platform Solution Architecture 9

Figure 2-2. NVIDIA vGPU Internal Architecture 10

Figure 2-3. Graphic Example of a vGPU Configuration with Tesla M60 ..... **Error! Bookmark not defined.**

LIST OF TABLES

Table 2-1. Virtual GPU Types 11

Table 2-2. Tesla GPU Hardware Specifications 15

Table 7-1. NVIDIA vGPU Types per GPU ..... **Error! Bookmark not defined.**

Table 8-5. VM Configurations 131

Chapter 1.
ABOUT THIS DOCUMENT

This section contains the formatting conventions and information contained in this deployment guide. There is also a quick list of acronyms used. Lastly, we detail the lab environments used to test the contents of this guide and capture the screenshots used.

1.1 FORMATTING CONVENTIONS

This document uses several formatting conventions to present information of special importance.

Lists of items, points to consider, or procedures that do not need to be performed in a specific order appear in bullet format. If there are links that are associated with a bullet point, it will be listed below that bullet:

- Item 1

  Hyperlink for Item 1

- Item 2

  Hyperlink for Item 2

Procedures that must be followed in a specific order appear in numbered steps:

1. Perform this step first.

2. Perform this step second.

Specific keyboard keys are bolded and capitalized, for example: **ESC**. If more than one key should be pressed simultaneously, it is displayed as **KEY1+KEY 2**, for example **ALT+F4**.

Interface elements such as document titles, fields, windows, tabs, buttons, commands, options, and icons appear in bold or italics text, for example "**Configuration**" or "*Configuration*".

Menus and submenus have the notation **Menu→Submenu**. For example, "Select **File→Save**" means that you should first open the **File** menu, and then select the **Save** option.

Specific commands appear in standard Courier font. Sequences of commands appear in the order in which you should execute them and include horizontal or vertical spaces between commands. The following additional formatting also applies when discussing command-line commands:

Plain-text responses from the system appear in bold **Courier** font.

This manual also contains important safety information and instructions in specially formatted callouts with accompanying graphic symbols. These callouts and their symbols appear as follows throughout the manual:

CAUTION: Cautions alert you to the possibility of a serious error, data loss, or other adverse condition

Notes provide helpful information

1.2 RELATED DOCUMENTATION

Refer to the NVIDIA Virtual GPU (vGPU) resources page http://www.nvidia.com/gridresources for additional information about NVIDIA vGPU technology, including:

- NVIDIA Virtual GPU Technology

  https://www.nvidia.com/en-us/design-visualization/technologies/virtual-gpu/
- Purchasing Guide for NVIDIA vGPU Solutions

  https://www.nvidia.com/en-us/design-visualization/buy-grid/
- NVIDIA Tesla GPU Datasheets

  http://www.nvidia.com/object/grid-enterprise-resources.html#datasheet
- Application Deployment Guides and Solution Overviews

  http://www.nvidia.com/object/grid-enterprise-resources.html#guides
- Customer Success Stories

  http://www.nvidia.com/object/grid-enterprise-resources.html#case
- White Papers

  http://www.nvidia.com/object/grid-enterprise-resources.html#whitepapers
- Videos

  http://www.nvidia.com/object/grid-enterprise-resources.html#videos

1.3 SUPPORT CONTACT INFORMATION

For technical support there are several resources to assist you:

For community support, please post questions (and answers!) on our respective forums:

- VMware vGPU Community:
  - *https://communities.vmware.com/community/vmtn/vmware-nvidia-direct-access-program*

In addition, you should reach out to your local VMware Horizon and NVIDIA vGPU teams for

guidance.

Should you find a bug please follow the steps in Section 12.2, page 262, to create a bug report and then submit to the VMware vGPU Community site: *https://communities.vmware.com/community/vmtn/vmware-nvidia-direct-access-program*

For support when architecting your solution, your VMware Horizon and NVIDIA vGPU teams are available to assist. Please be sure you are in touch with them and keep them up to date with your progress. If you do not know your correct account management teams, please reach out to the appropriate email below:

- NVIDIA vGPU team: gridteam@nvidia.com

The NVIDIA vGPU resources page includes additional contact methods to help you get the answers you need as soon as possible.

1.4 ACRONYMS

CLI	"Command Line Interface" – an ASCI text-based interaction with an application or operating system
GPU	"Graphics Processing Unit" – a single chip processor with integrated transform, lighting, triangle setup/clipping, and rendering engines that is capable of processing a minimum of 10 million polygons per second. https://www.nvidia.com/object/gpu.html
GUI	"Graphical User Interface" – a visual, windowed based means of interacting with an application or operating system
OS	"Operating System" – a base level installation such as Windows that controls the underlying physical or virtual hardware
POC	"Proof of concept"

UI	"User Interface" – a tool used to enter data or interact with an application or operating system
vDGA	"Virtual Dedicated Graphics Accelerator" - allow guest operating systems to directly access an I/O device, bypassing the virtualization layer.
VDI	"Virtual Desktop Infrastructure" – a general term for desktop operating systems running as a guest operating system on a host server
vGPU	"Virtual Graphics Processing Unit" – a virtual GPU provided to the hypervisor and managed by the NVIDIA vGPU Manager to provide a virtual machine access to a time-sliced physical GPU
vib	"vSphere Installation Bundle"
VM	"Virtual Machine" – an operating system running as a guest on a host server
vSGA	"Virtual Shared Graphics Acceleration" - a VMware technology to share a GPU with guest operating systems using graphic API translation methods
WebUI	"Web User Interface" – a web-based means of interacting with an application or operating system

1.5 THE LAB

The following equipment was used to install the hypervisor, Virtual GPU Manager and virtual machines to confirm the procedures and collect screen shots for this guide:

NVIDIA Technical Marketing Lab:

- Management Cluster
  - Hosts:
    - Supermicro 1U servers, local storage, 1GB NICs
  - Guests:
    - Microsoft Windows 2012 Domain Controller
    - Microsoft SQL Server 2012
    - vCenter Server Appliance (VCSA), PhotonOS based
    - Microsoft Windows 2012 Servers for running:
      - View Composer
      - Connection Server on Windows
- Production Cluster
  - Hosts:
    - NVIDIA certified 2U servers, 2x vGPU enabled Tesla GPUs, vSAN (SSD & spindle HDD), 1GB NICs, 10G NICs for iSCSI to Pure Storage SAN
  - Guests:
    - Microsoft Windows 10 Professional x64
- SAN Storage
  - Pure Storage M50 (10G iSCSI)

- Networking
    - Arista 10G core switch
        - No LAG or similar
    - Juniper & Cisco 1GB distribution switches
- Load Generation
    - An NVIDIA custom designed benchmark engine
- Monitoring
    - An NVIDIA custom designed monitoring engine

It's not necessary to have all the equipment we used for building out a POC/trial environment, in fact it can be done on a single physical server. This is not, however, representative of a recommended enterprise deployment and can impact performance. To ensure the most accurate results and performance you intend to deliver to your end users, build your POC/trial as close as possible to your production deployment. Further, by separating the management and production (or virtual desktop) clusters, we were able to reinstall new versions of code while retaining foundational services and exported virtual machines.

Chapter 2. GETTING STARTED

NVIDIA vGPU enables multiple virtual machines (VMs) to have simultaneous, direct access to a single physical GPU, using the same NVIDIA graphics drivers that are deployed on non-virtualized operating systems. This gives VMs unparalleled graphics performance and application compatibility, together with cost-effectiveness and scalability brought about by sharing a GPU among multiple workloads.

This chapter covers how NVIDIA vGPU solutions fundamentally alters the landscape of desktop virtualization and enables users and applications of all levels of complexity and graphics requirements to utilize said solutions. It also describes the NVIDIA vGPU architecture, the Tesla GPUs recommended for virtualization, the three virtual GPU software editions, and key standards supported by NVIDIA virtual GPU technology.

2.1 WHY NVIDIA vGPU?

The promise of desktop virtualization, realized for server workloads years ago, is flexibility and manageability. Initially, desktop virtualization was used where flexibility and security were the primary drivers due to cost considerations. The democratization of technology over the years has reduced the total cost of ownership of desktop virtualization. This, along with advances in storage and multi-core processors make for a reasonable and/or advantageous cost to ownership.

The big remaining challenge for desktop virtualization is providing a cost effective yet rich user experience. There have been attempts to solve this problem with shared GPU technologies like vSGA that are cost effective, but those technologies do not support the rich application support needed to be successful and ensure end user adoption. This compared to dedicated

GPU pass-through with vDGA that does provide 100% application compatibility, but only for the highest end use cases due to the high cost with limited density of virtual machines per host server.

Due to the lack of scalable, sharable, and cost effective per user GPUs that provide 100% application compatibility, providing a cost-effective rich user experience has been challenging for broad use cases in desktop virtualization. Meanwhile, high-end 3D applications simply did not work in a virtualized environment or were so expensive to implement with vDGA it was reserved for only the most limited of circumstances.

Today, this is no longer true to thanks to NVIDIA vGPU technology combined with VMware Horizon. NVIDIA vGPU technology allows the flexibility where multiple virtual desktops share a single physical GPU that may reside on a single physical PCI card. This provides the 100% application compatibility of vDGA pass-through graphics, but with lower cost of multiple desktops sharing a single graphics card to provide a rich, yet more cost-effective user experience. With Horizon you are able to centralize, pool, and more easily manage traditionally complex and expensive, distributed workstations and desktops. Now all your user groups can take advantage of the promise of virtualization.

2.2 NVIDIA vGPU ARCHITECTURE

Physical NVIDIA GPUs can support multiple *virtual* GPUs (vGPUs) that can be assigned directly to guest VMs under the control of NVIDIA's Virtual GPU Manager running in a hypervisor. Guest VMs use the NVIDIA vGPUs in the same manner as a physical GPU that has been passed through by the hypervisor. An NVIDIA driver loaded into the VM's guest OS leverages direct access to the GPU for performance and critical fast paths. Non-critical performance management operations use a para-virtualized interface to the NVIDIA Virtual GPU Manager. Figure 2-1 shows the above NVIDIA vGPU system architecture.

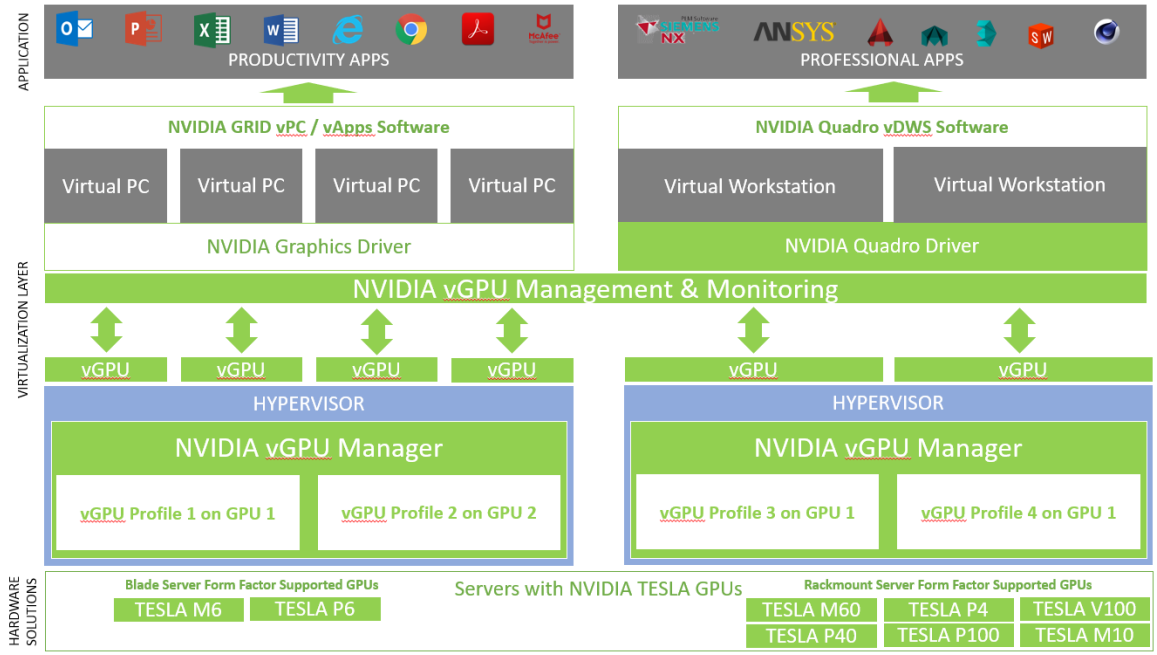


Figure 2-1. NVIDIA vGPU Platform Solution Architecture

NVIDIA vGPUs are comparable to conventional GPUs in that they have a fixed amount of GPU frame buffer and one or more virtual display outputs or *heads*. Multiple heads support multiple displays. Managed by the NVIDIA vGPU Manager installed in the hypervisor, the vGPU frame buffer is allocated out of the physical GPU frame buffer at the time the vGPU is created. The vGPU retains exclusive use of that frame buffer until it is destroyed.

Note that these are virtual heads, meaning on Tesla GPUs there is no physical connection point for external physical displays.

All vGPUs resident on a physical GPU share access to the GPU's engines, including the graphics (3D), and video decode and encode engines. Figure 2-2 shows the vGPU internal architecture.

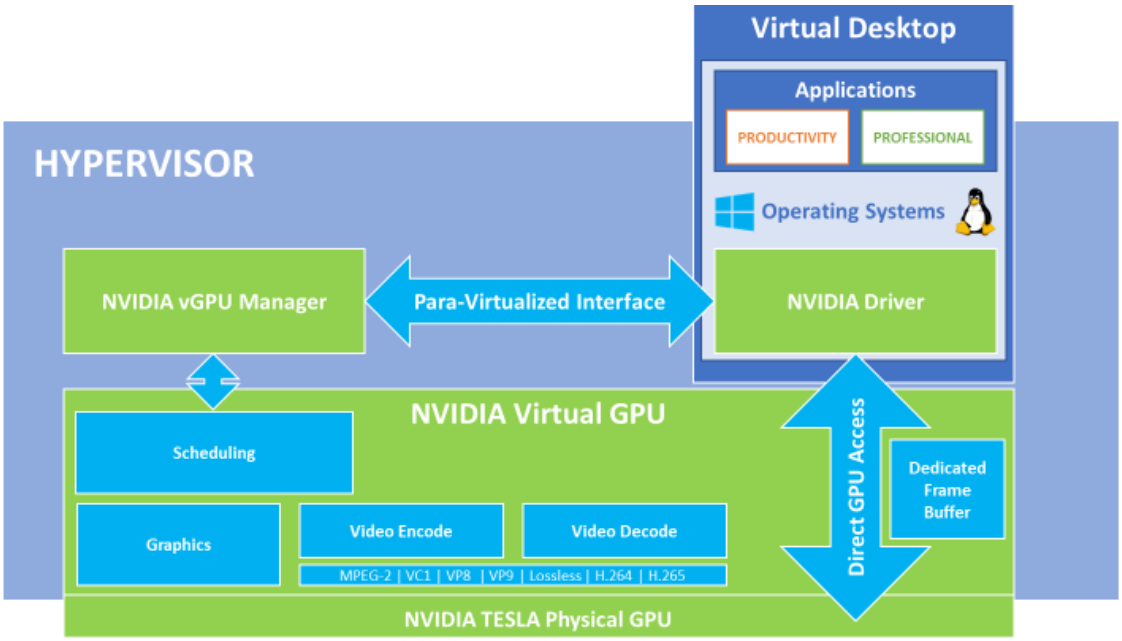


Figure 2-2. NVIDIA vGPU Internal Architecture

2.3 SUPPORTED GPUs

NVIDIA virtual GPU software is supported with NVIDIA Tesla GPUs. Determine the Tesla GPU best suited for your environment based on whether you're optimizing for performance or density, and whether the GPUs will be installed in rack servers or blade servers.

- HIGHEST PERFORMANCE – The following Tesla GPUs are targeted at users with demanding workflows, compressed design cycles, and complex simulations. Datasheets can be downloaded by clicking on the links below.

NVIDIA TESLA V100

NVIDIA TESLA P100

NVIDIA TESLA P40

NVIDIA TESLA P4

NVIDIA TESLA M60

- BEST FOR HIGH DENSITY

NVIDIA TESLA M10

- BEST FOR BLADE SERVERS

NVIDIA TESLA P6

NVIDIA TESLA M6

NVIDIA virtual GPU software enables the delivery of graphics-rich virtual desktops and workstations accelerated by NVIDIA® Tesla® GPUs, the most powerful data center GPUs on the market today. NVIDIA virtual GPU software divides Tesla GPU resources so the GPU can be shared across multiple virtual machines running any application.

The portfolio of NVIDIA virtual GPU software products includes:

- NVIDIA Quadro® Virtual Data Center Workstation (Quadro® vDWS)
- NVIDIA GRID® Virtual PC (GRID® vPC)
- NVIDIA GRID® Virtual Applications (GRID® vApps)

To run these software products, you will need an NVIDIA Tesla GPU and software license that addresses your specific use case.

Table 2-1 lists the virtual GPU software editions and recommended Tesla GPUs for each.

Table 2-1. Virtual GPU Types

NVIDIA VIRTUAL GPU SOFTWARE FEATURES		Quadro vDWS	GRID vPC	GRID vApps
Configuration & Deployment	Desktop Virtualization	Y	Y	
	RDSH App Housing	Y[4]	Y	Y
	RDSH Desktop Hosting	Y[4]	Y	Y
	Windows OS Support	Y	Y	Y
	Linux OS Support		Y[5]	
	GPU Pass-Through Support[6]	Y		Y

	Bare Metal Support[7]	Y		Y
	NVIDIA Graphics Driver	Y[4]	Y	Y
	NVIDIA Quadro Driver	Y		
	Guaranteed Quality of Service Scheduling[8]	Y	Y	Y
Display	Maximum Active Displays	Four 4K	Four QHD / Two 4K[18]	One[9]
	Maximum Display Resolution	4096 x 2160	4096 x 2160	1280 x 1024
Data Center Management	Host, Guest, and Application Level Monitoring[10]	Y	Y	Y
	Live Migration[5]	Y	Y	Y
Support	NVIDIA Direct Enterprise-Level Technical Support	Y	Y	Y
	Maintenance Releases, Defect Resolutions, and Security Patches for up to 3 years[11]	Y	Y	Y
Advanced Professional Features	Independent Software Vendor (ISV) Certifications	Y		
	CUDA / OpenCL	Y[12]		Y[13]
	NVENC	Y	Y	

Graphics Features & APIs	OpenGL Extensions, Including WebGL	Y	Y	Y
	DirectX	Y	Y	Y
	Vulkan Support	Y		
	Quadro Performance Features and Optimizations	Y		
Profiles	Max Frame Buffer Supported	32 GB	2 GB[5]	24 GB
	Available Profiles[14]	0Q, 1Q, 2Q, 3Q, 4Q, 6Q, 8Q, 12Q, 16Q, 24Q, 32Q[19]	0B, 1B, 2B[5]	24A, 16A, 12A, 8A, 6A, 4A, 3A, 2A, 1A
Recommended GPUs for Different Use Cases and Workflows	High-end professional graphics users, includes uses for double precision compute workloads (ie. Running 3D models and design workflows, intensive CAE simulations)	TESLA V100[15], TESLA P100		
	Mid to high-end professional graphics users, includes use for single precision compute workloads (ie. Rendering & creating complex designs)	TESLA P40		
	Entry to mid-level professional graphics users, including deep learning inference workloads and Pascal specific GPU architecture features	TESLA P4		
	Entry to mid-level professional graphics users, including	M60		

	support for heterogeneous profiles and larger number of OEM server certifications			
	Best density and performance for the knowledge worker		M10	
	Software developers using 2D Electronic Design Automation (EDA) tools and Linux apps; healthcare providers and financial traders needing increased productivity with multiple high-res monitor support		M10	
	Best for blade form factor	M6, P6	M6, P6	M6, P6

[1] Assumes cost of subscription, NVIDIA GRID software, and hardware, with 3-year amortization of hardware of two Tesla M10 cards supporting 87 vApps users 64 vPC users

[2] Lakeside Software, Inc. "Elevating User Experience Through GPU Acceleration: A Windows 10 versus Windows 7 Analysis." Lakeside Software White Paper. 2017

[3] Data from Lakeside Software Community Program 2017

[4] With packaged GRID vApps license

[5] Supported starting with NVIDIA virtual GPU software Spring 2018 release (version 6.0)

[6] Only supported on 1:1 profiles

[7] Only NVIDIA Tesla M6 Hardware supported as primary display device

[8] Scheduling options include fixed share, equal share, and best effort/time slicing

[9] GRID vApps supports one 1280x1024 display from the GPU card. However, XenApp renders to an offscreen buffer, so it can support multiple software rendered displays at higher resolutions

[10] Application level monitoring only available starting with the NVIDIA virtual GPU August 2017 release (version 5.0)

[11] Available with active Support, Updates, and Maintenance and Subscriptions (SUMs) contract

[12] Supported on 8GB 1:1 profile on Maxwell and all profiles on Pascal

[13] Only supported for Maxwell 8A profile on GRID 4.x and earlier releases

[14] Profiles supported have dependency on GPU selected. For vGPU profiles, "Q" stands for Quadro, "B" stands for Business, and "A" stands for Apps. For more information, read the Virtual GPU Software User Guide at docs.nvidia.com/grid/latest/grid-vgpu-user-guide/index

[15] V100 support only available with NVIDIA virtual GPU software Spring 2018 release (version 6.0)

[16] Gartner. April 25, 2017. "Gartner Survey Shows 85 Percent of Enterprises Will Have Started Windows 10 Deployments by End of 2017." [Press Release]. Retrieved from www.gartner.com/newsroom/id/3690917

[17] John Peddie Research. October 16, 2017. Jon Peddie Research: Multiple Displays Can Increase Productivity By 42%. Retrieved from www.jonpeddie.com/press-releases/jon-peddie-research-multiple-displays-can-increase-productivity-by-42 Supports up to two 4K displays or four 2560x1600 displays on 2B profile.

[18] Supports up to four 2560x1600 displays on 1B profile. Support for two 4K displays starts with NVIDIA virtual GPU software release 6.0, and support for four 2560x1600 displays on 2GB profile starts with NVIDIA virtual GPU software release 6.2

[19] 32Q profile available with Tesla V100

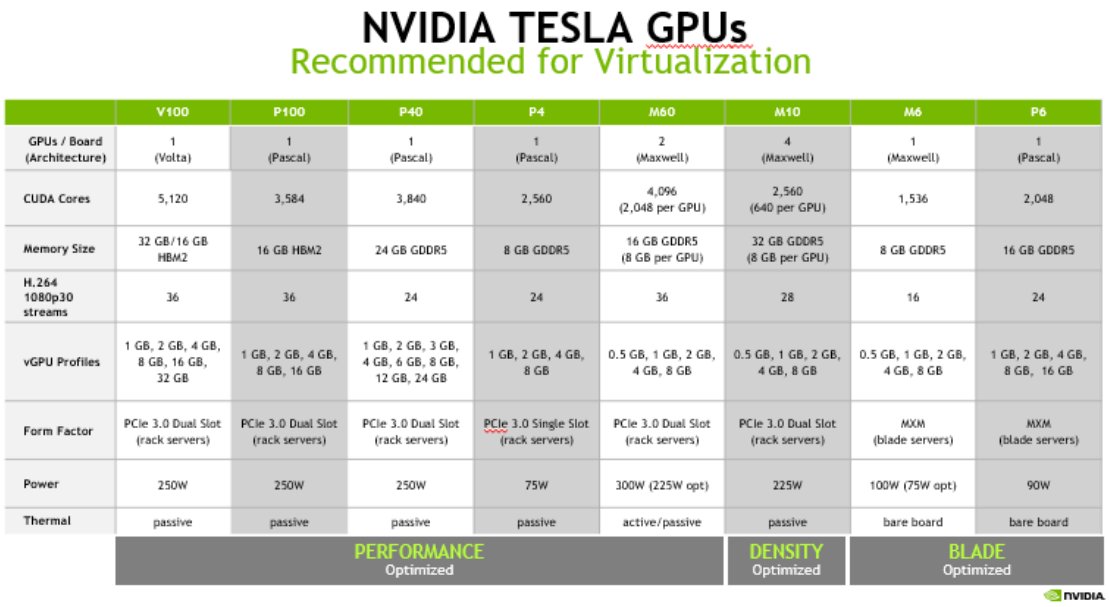
The maximum number of vGPUs that can be created simultaneously on a physical GPU is defined by the amount of frame buffer per virtual desktop, and thus how many virtual desktops can share that physical GPU. For example, a Tesla M10 physical GPU can support up to 64 GRID vPC 0B profiles (32 GB total with 512MB per user) or 32 GRID vPC 1B profiles (32 GB total with 1GB per user). You cannot oversubscribe frame buffers and they must be shared equally for each physical GPU however, you may select different profiles for each GPU.

NVIDIA Tesla V100, P100, P40, M60 (passive and active cooling), and M10 are full height, full length, double width PCI-e cards, and are passively (with the exception of the M60) cooled (no onboard fan) requiring certified servers designed to properly power and cool them. Tesla P4 is a single slot 75W GPU meaning you can accommodate two Tesla P4s in the same physical space you could fit a single double width card. Tesla P6 and M6 are designed to fit into blade server form factors. Refer to http://www.nvidia.com/buygrid for a list of certified server platforms for NVIDIA Tesla GPUs. Cross-reference the NVIDIA certified server list with the VMware vSphere HCL to find servers best suited for your NVIDIA vGPU and VMware vSphere environment. Each card requires auxiliary power cables connected to it (except Tesla P4), see chart below for specific pin counts. Most industry standard servers require an enablement kit for proper mounting the of the NVIDIA Tesla cards. Check with your server OEM of choice for more specific requirements.

2.4 TESLA GPU SPECIFICATIONS

Table 2-2 lists the Tesla GPU hardware specifications.

Table 2-2. Tesla GPU Hardware Specifications



2.5 SUPPORTED GRAPHICS PROTOCOLS

This version of NVIDIA vGPU software includes support for:

- Full DirectX 12, Direct2D, and DirectX Video Acceleration (DXVA)
- OpenGL 4.5
- NVIDIA vGPU SDK (remote graphics acceleration)
- Vulkan 1.0
- OpenCL and CUDA applications **WITHOUT** Unified Memory are supported on the following virtual GPUs:
    - The 8Q vGPU type on Tesla M6, Tesla M10, Tesla M60 GPUs
    - All Q-series vGPU types on the following GPUs:
        - Tesla P4
        - Tesla P6
        - Tesla P40
        - Tesla P100 SXM2
        - Tesla P100 PCIe 16GB
        - Tesla P100 PCIe 12GB

- Tesla V100 SXM2
- Tesla V100 PCIe
- Tesla V100 FHHL

NOTE: Unified Memory and CUDA tools are **NOT** supported on NVIDIA vGPU

The following are NOT supported in this release of vGPU:

- Open CL and CUDA applications WITH Unified Memory NOTE: For CUDA and OpenCL we recommend using vDGA (not covered in this guide).

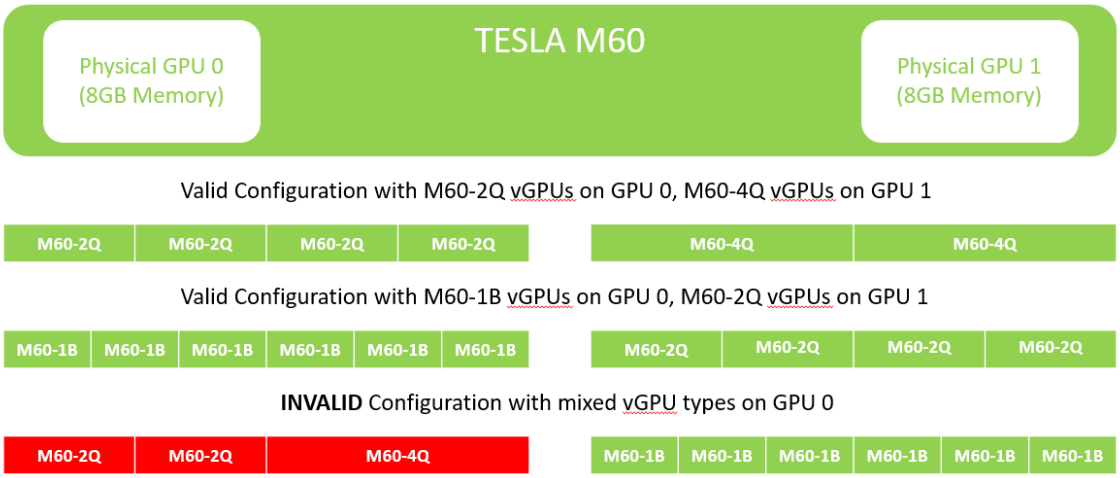
2.6 HOMOGENOUS VIRTUAL GPUS

This release of NVIDIA vGPU software supports only homogeneous virtual GPUs. This means, at any given time, the virtual GPUs resident on a single physical GPU must be all of the same type. However, this restriction doesn't extend across physical GPUs on the same card. Different physical GPUs on the same card may host different types of virtual GPUs at the same time, provided that the vGPU types on any one physical GPU are the same.

For example, a Tesla M60 card has two physical GPUs, and can support several types of virtual GPUs. Figure2 - 3 shows the following examples of valid and invalid virtual GPU configurations on Tesla M60:

- A valid configuration with M60-2Q vGPUs on GPU 0 and M60-4Q vGPUs on GPU 1
- A valid configuration with M60-1B vGPUs on GPU 0 and M60-2Q vGPUs on GPU 1
- An invalid configuration with mixed vGPU types on GPU 0

Figure 2-3: Graphic Example of a vGPU Configuration with Tesla M60:



2.7 BEFORE YOU BEGIN

This section describes the general prerequisites and some general preparatory steps that must be addressed before proceeding with the deployment.

2.7.1 Building an Evaluation Plan

To be successful, we highly recommend you start with an evaluation plan consisting of the following:

- List of your business drivers and goals
- List of all the user groups, their workloads and applications with current, and future projections in consideration
- Current end-user experience measurements and analysis
- ROI / Density goals

2.7.2 Size Your Environment

Based on your evaluation plan, we recommend sizing an appropriate environment for each user group you are trying to reach with your evaluation.

For example, here are three examples of Dassault Systèmes SOLIDWORKS use cases and their computing needs:

- Entry Level Engineer / Design Reviewer
  (Small Objects / Assemblies / Approximately 10% of Users)
  - 4 GB RAM
  - 2 vCPUs (2.6 GHz)
  - 2-4 GB video RAM
  - Two 2560x1600 displays
- Mid-Level Engineer
  (Avg to Large Objects / All App Features / Approximately 80% of Users)
  - 8 GB RAM
  - 4 vCPUs (2.6 GHz)
  - 4-8 GB video RAM
  - Four 2560x1600 displays and/or One 3840 x 2160
- Advanced Engineer
  (Extremely Large objects / All App Features / Approximately 10% of Users)
  - 16 GB RAM
  - 8 vCPUs (3.2 GHz)
  - 8-16 GB video RAM
  - Four 2560x1600 displays and/or Two 3840 x 2160

Based upon your needs, make sure to include enough RAM, CPUs, and GPUs to meet your sizing requirements.

2.7.3 Choosing Your Hardware

The following elements are required to install and configure vGPU software on VMware vSphere:

- NVIDIA certified servers with NVIDIA Tesla cards (see http://www.nvidia.com/object/grid-certified-servers.html for a list of certified NVIDIA servers. It will be helpful to cross check this list with the VMware HCL to ensure compatibility for your deployment: https://www.vmware.com/resources/compatibility/search.php) The following specifications are recommended:
  - 2.6GHz CPU or faster (Intel Xeon E5-2600 v4 Family Product)
  - High-speed RAM
  - Fast networking
  - If using local storage IOPS plays a major role in performance. If using VMware for Virtual SAN, see the VMware Virtual SAN requirements website for more details.

- High performance storage with Virtual SAN or high IOPS storage system
- Higher-performance end points for testing access

2.7.4 General Prerequisites

The following elements are required to install and configure vGPU software on VMware vSphere 6.7

- Evaluation plan consisting of all the following:
  - List of your business drivers and goals
  - List of all the user groups, their workloads and applications with current, and future projections in consideration
  - Current end-user experience measurements and analysis
  - ROI / Density goals
  - Review VMware's Horizon with View Getting Started section which includes courses and guidance on potentially any current configuration that you may already have:
    - http://www.vmware.com/products/horizon-view/gettingstarted
- NVIDIA certified servers with NVIDIA Tesla cards (see http://www.nvidia.com/object/grid-certified-servers.html for a list of certified servers. Cross check this list with the VMware HCL to ensure compatibility with your environment: https://www.vmware.com/resources/compatibility/search.php) that meet the following recommended specifications:
  - 2.6GHz CPU or faster (Intel Xeon E5-2600 v4, Intel Xeon Scalable Processor Family)
  - High-speed RAM
  - Fast networking
  - If using local storage IOPS plays a major role in performance. If using VMware for Virtual SAN, see the VMware Virtual SAN requirements website for more details.
- Intel Xeon E5-2600 v4, Intel Xeon Scalable Processor Family Higher-performance end points for testing access
- Free 60-day evaluation for VMware vSphere and vSphere with Operations Management can be located here: https://my.vmware.com/en/web/vmware/evalcenter?p=vsphere-eval&src=WWW_us_VMW_XfK4FDRuqSPE01Y7isJw
- Select the appropriate NVIDIA Tesla GPU for your use case. Please reference Table 2-1 or 2-2 for a better understanding of which GPU suits your deployment requirements. For additional guidance, please contact your NVIDIA and VMware reps.
- VMware vSphere 6.7 build
  - vSphere and vCenter Server are available only from the VMware website (see link above).
  - You may deploy vCenter Server 6 on a Windows server or as an OVA Appliance.

- VMware Horizon 7 software. If needed, you may register for a trial to obtain the license keys required for various elements to deploy and manage Horizon at: https://my.VMware.com/group/VMware/evalcenter?p=horizon.
- NVIDIA vGPU software:
  - NVIDIA vGPU manager VIB
  - NVIDIA WDDM guest driver (32-bit and 64-bit versions)

The vGPU Manager VIB is loaded similar to a driver in the vSphere hypervisor, and is then managed by the vCenter Server.

- Microsoft software:
  - Windows Server 2012R2 (recommended) or Server 2008R2 (minimum)
  - Windows 10 Professional or Enterprise VL (for use as the guest operating system)
- Your choice of one of the following CLI/SSH/SCP tools installed on your Windows-based toolbox PC:
  - MobaXterm (SSH and SPC) is available from www.mobaxterm.mobatek.net. This is the recommended tool. See Chapter 13, *Using MobaXterm*, on page 264 for information on using MobaXterm
  - Putty (SSH) and WinSPC (SPC), available from www.putty.org and www.winscp.net, respectively. See Chapter 14, *Using WINSCP*, on page 270 for information on using WinSCP.
- Licenses:
  - From the VMware Horizon website:
    - VMware vSphere 6.7
    - VMware Horizon 7
  - Microsoft (volume licenses recommended)
- Testing and Benchmarking:
  - NVIDIA System Management interface (NSMI)
  - OPTIONAL: VMware vRealize Operations for Horizon (V4H)
  - OPTIONAL: Lakeside Systrack 7 with GPU monitoring
  - User satisfaction survey
- Server where you will install View Connection Server 6:
  - Windows Server 2012R2
  - Server must be joined to a domain
  - Server must have a static IP address assigned

2.7.5 Current Builds

As of the publication date of this manual (see the back cover), the current builds are:

- NVIDIA-VMware_ESXi_6.7_Host_Driver-390.72-1OEM.670.0.0.8169922.vib
- NVIDIA-VMware_ESXi_6.7_Host_Driver-390.72-1OEM.670.0.0.8169922-offline_bundle.zip
- 391.81_grid_win8_win7_32bit_international.exe
- 391.81_grid_win8_win7_server2012R2_server2008R2_64bit_international.exe
- 391.81_grid_win10_32bit_international.exe
- 391.81_grid_win10_server2016_64bit_international.exe
- NVIDIA-Linux-x86_64-390.75-grid.run

2.7.6 Pre-Installation Preparation

Perform the following procedure prior to installation:

1. Determine how vSphere will run on the physical hosts. Consider booting from a thumb drive since this is an Early Access build.

Download and install any of the following for SSH and SCP:

- SSH tool (such as PuTTY) and an SCP tool (such as WinSCP).
- MobaXterm, which handles both SSH and SCP functions.

Set the BIOS on the physical hosts as appropriate. See 2.7.7, *Server BIOS* Settings on page 22.

Install Virtual Clone Drive or similar to easily mount and un-mount ISO images within the virtual machines.

2.7.7 Server BIOS Settings

Configure the BIOS as appropriate for your physical hosts, as described below:

2.7.7.1 Cisco

- CPU Performance: Set this parameter to either **Enterprise** or **High Throughput**.

- Energy Performance: Set this parameter to **Performance**.
- PCI Configuration: Set the following parameters:
  - MMCFG BASE: Change this value from **Auto** to **2 GB**.

> **CAUTION: FAILURE TO CHANGE THIS VALUE MAY CAUSE THE VSPHERE INSTALLAITON TO FAIL WHILE BOOTING THE ISO.**

  - Memory Mapped I/O above 4-GB: Set this parameter to **Enabled**.
- VGA Priority: Set this parameter to **Onboard VGA Primary**.
- Refer to the *Cisco Unified Computing System BIOS Settings* page at the Cisco website for additional Cisco-specific BIOS settings.

2.7.7.2 Dell

- System Profile: Set this parameter to Performance.
- Memory Mapped I/O Above 4 GB: Set this parameter to Enabled

> If NVIDIA card detection does not include all the installed GPUs, set this option to Enabled

- Embedded Video Controller: Set this parameter to Enabled.
- Refer to the Dell BIOS Settings page at the Dell website for additional Dell-specific BIOS settings.

2.7.7.3 HP

- HP Power Profile: Set this parameter to Advanced.
- HP Power Regulator: Set this parameter to HP Static High-Performance Mode.
- Energy/Performance Bias: Set this parameter to Maximum Performance.
- Memory Power Savings Mode: Set this parameter to Maximum Performance.
- Video Options: Set this parameter to Embedded Video Primary, Optional Video Secondary.

2.7.7.4 Supermicro

- Power Technology: Set this parameter to either Maximum Performance or Energy Efficient.

- Energy/Performance Bias: Set this parameter to either Performance or Balanced Performance.
- Above 4G Decoding: Set this parameter to Enabled.
- MMCFG base: Set this parameter to 0xC0000000.

Chapter 3.
INSTALLING vSPHERE 6.7

This chapter covers the following vSphere 6.7 installation topics:

- Choosing an install method
- Preparing the USB boot media
- Installing vSphere 6.7 from the USB media
- Initial host configuration
- Assigning a host license

NOTE: This deployment guide assumes you are building an environment as a proof of concept and is not meant to be a production deployment, as a result, choices made are meant to speed up and ease the process. See the corresponding guides for each technology, and make choices appropriate for your needs, before building your production environment.

3.1 CHOOSING INSTALL METHOD

With the ability to install from and onto a SD card or USB memory stick, vSphere offers flexibility verses local hard drive install. Please see vSphere documentation regarding best practices for logs when booting from USB or similar. In our main lab we used Supermicro's IPMI and virtual media to boot from ISO file and install on local storage. In home labs USB was used to quickly move from one version to another.

3.2 PREPARING USB BOOT MEDIA

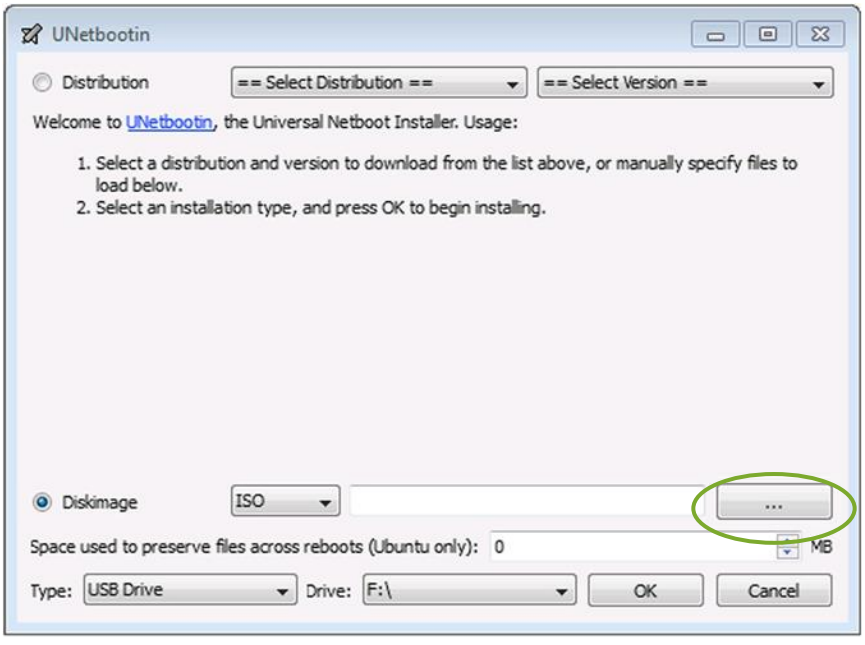
Reference VMware KB: “Installing ESXi 5.x on a supported USB flash drive or SD flash card (2004784)”

Booting vSphere 6.7 from a USB drive is useful if your host has an existing vSphere Version 5.5 or earlier installation that you want to retain.

Use the following procedure to prepare a USB drive for booting:

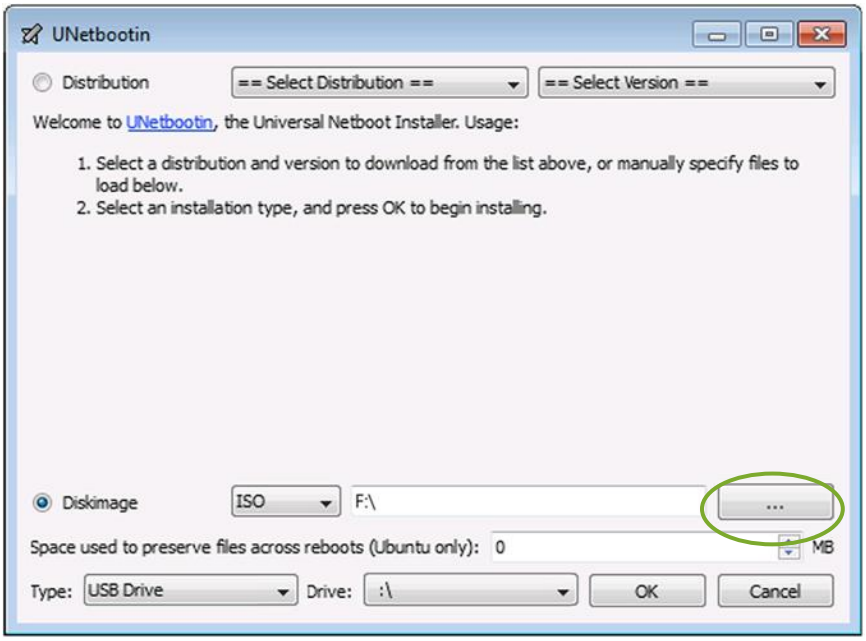
1. Download **UNetbootin** from http://unetbootin.sourceforge.net/.
   The Windows version of the application does not include an installer; however, the OSX version is packaged in a **.DMG** file that you must mount. You must also copy the application to the **Applications** folder before launching. Alternatively, you can use YUMI, which allows booting multiple installation images on one USB device plus the option to load the entire installation into RAM. The download link is http://www.pendrivelinux.com/yumi-multiboot-usb-creator/.

Start the application, select **Diskimage**, and then click the **…** icon to browse for the installation **.ISO** file.



Navigate to the location that contains the installation **.ISO** file and then select **Open**.

Select the mounted USB drive on which to perform the installation and then select **OK.**
The copying process begins, and a series of progress bars are displayed.



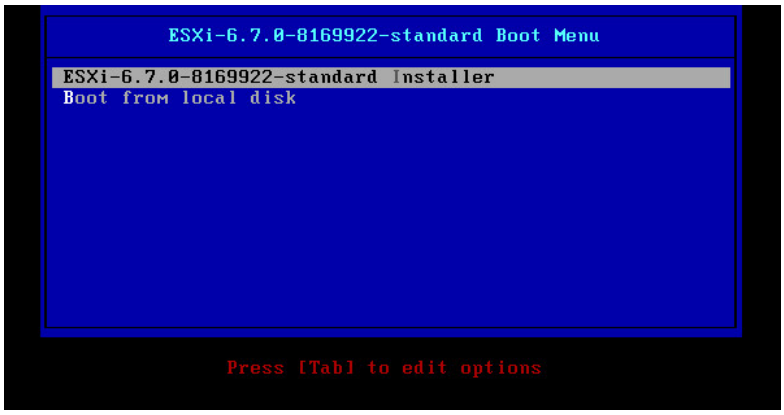
When the copying process is complete, click Exit and then remove the USB drive.

To install from this USB drive, insert into the host using either an internal or on motherboard USB port, then set that as the primary boot source or select from the boot menu on power up.

3.3 INSTALLING VSPHERE 6.7

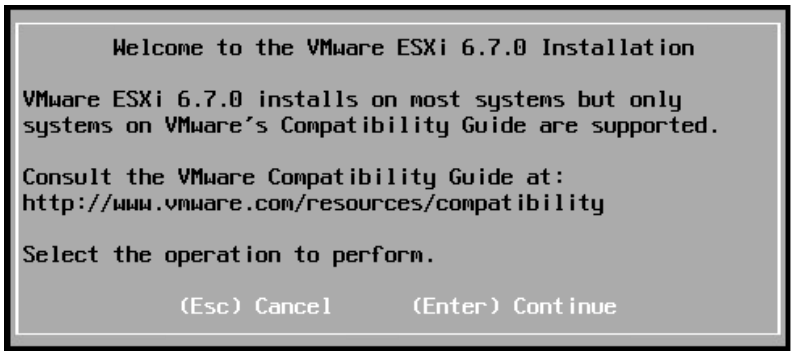
Use the following procedure to install the vSphere 6.7 regardless of boot source. Select the boot media with the vSphere ISO on your host's boot menu.

Apply power to start the host.
The following menu displays when the host starts up:

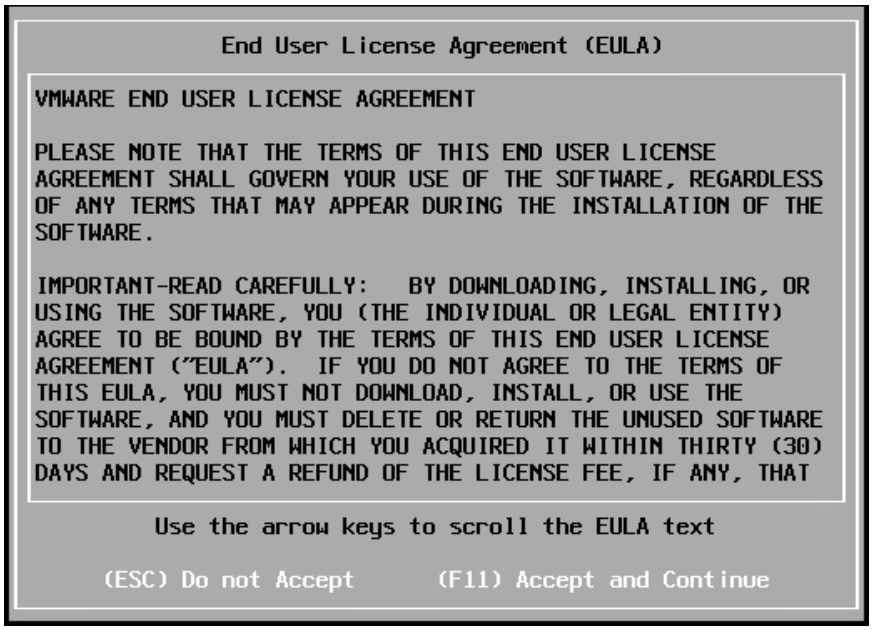


Select the installer using the arrow keys and then press **[ENTER]** to begin booting the vSphere 6.7 installer.

A compatibility warning display:

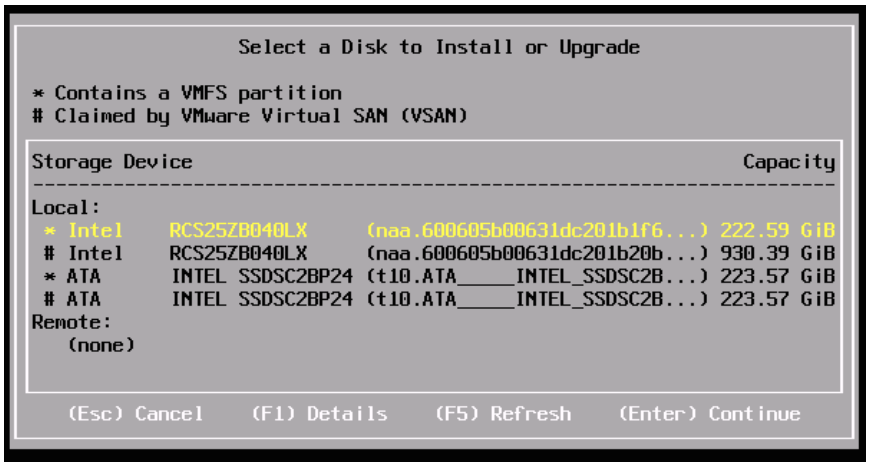


Press **[ENTER]** to proceed.
The End User License Agreement (EULA) displays:



Read the EULA and then press **[F11]** to accept it and continue the installation. The installer scans the host to locate a suitable installation drive:

It should display all drives available for install:

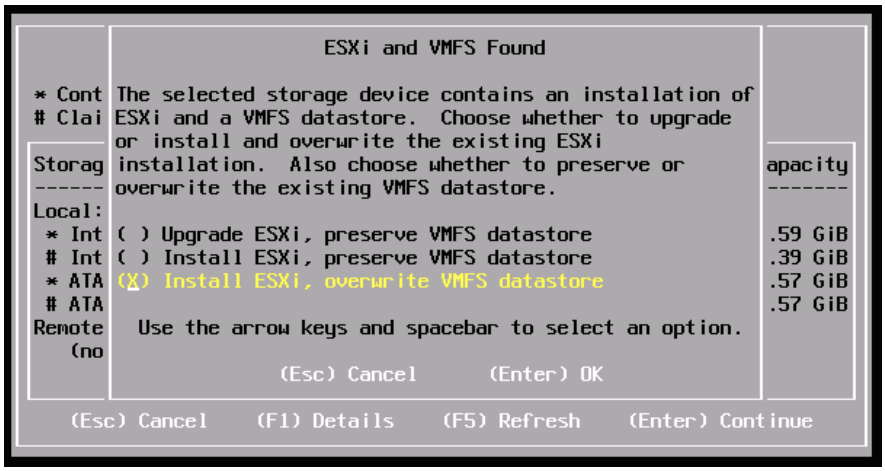


Use the arrow keys to select the drive you want to install vSphere 6.7 and then press **[ENTER]** to continue.

You can install vSphere 6.7 to a USB drive and then boot and run the system from that USB drive. This sample installation shows vSphere being installed on a local hard drive.

The installer scans the chosen drive to determine suitability for install:

The **Confirm Disk Selection** window displays:



Press **[ENTER]** to accept your selection and continue. (For this EA2 release, Upgrade ESXi is not a supported selection.)

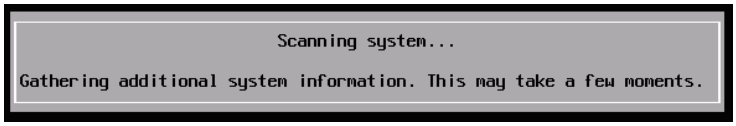
The **Please select a keyboard layout** window displays:

Select your desired keyboard layout using the arrow keys and then press **[ENTER].**
The **Enter a root password** window displays.

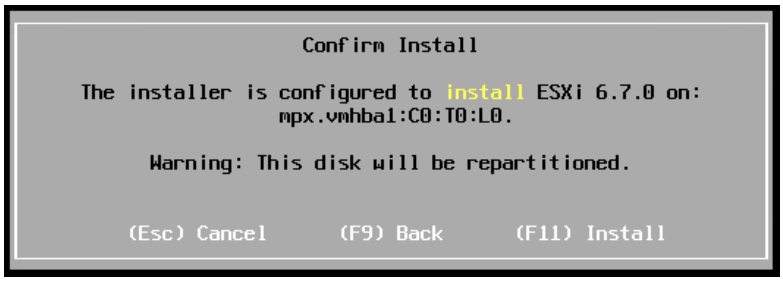
Enter a root password in the Root password field.

CAUTION: TO PREVENT UNAUTHORIZED ACCESS, YOUR SELECTED ROOT PASSWORD SHOULD CONTAIN AT LEAST EIGHT (8) CHARACTERS AND CONSIST OF A MIX OF LOWERCASE AND CAPITAL LETTERS, DIGITS, AND SPECIAL CHARACTERS.

Confirm the password in the **Confirm password** field and then press **[ENTER]** to proceed. The installer rescans the system:



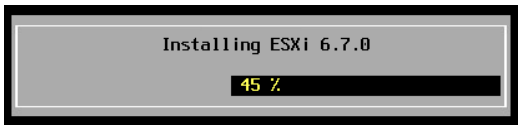
Then displays the **Confirm Install** window:



Press **[F11]** to proceed with the installation.

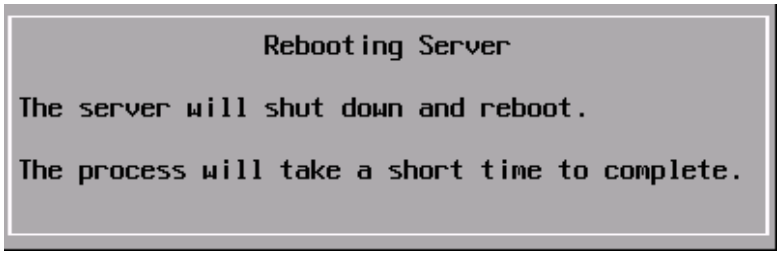
CAUTION: THE INSTALLER WILL REPARTITION THE SELECTED DISK. ALL DATA ON THE SELECTED DISK WILL BE DESTROYED

The vSphere 6.7 installation proceeds:



The **Installation Complete** window displays when the installation process is completed:

Press **[ENTER]** to reboot the system. (Make sure your installation media has been ejected and your bios set to the boot disk.)



The installation is now complete.

3.4 INITIAL HOST CONFIGURATION

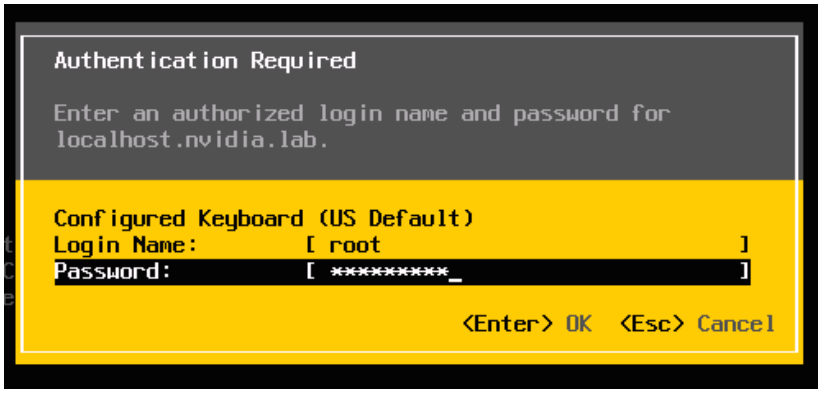
A countdown timer displays when you first boot vSphere 6.7. You can wait for the countdown to expire or press **[ENTER]** to proceed with booting. A series of notifications displays during boot.

The VMware vSphere 6.7 screen displays once the boot completes:

Figure 3-1. VMware vSphere 6.7 Screen

Use the following procedure to configure the host:

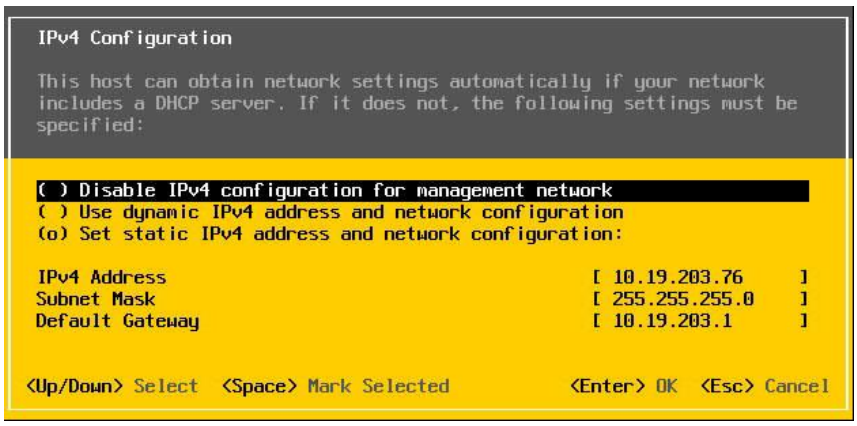
1. Press **[F2]**.
   The **Authentication Required** window displays:



Enter the root account credentials that you created during the installation process and then press **[ENTER].** The **System Customization** screen displays.

Scroll down to select **Configure Management Network** and then press **[ENTER].**
The **Network Adapters** window appears:

Use the arrow keys to select the adapter to use as the default management network and then press **[ENTER].** The **IPv4 Configuration** window displays:



Use the arrow keys to select **Set static IPv4 address and network configuration** and then enter the IPv4 address, subnet mask, and default gateway in the respective fields.

Press **[ENTER]** when finished to apply the new management network settings.
The **Confirm Management Network** popup displays.

Press **[Y]** to confirm your selection.
The **DNS Configuration** window displays.

Add the primary and (if available) secondary DNS server address(es) in the respective fields.

Set the host name for this vSphere host in the **Hostname** field.

Press **[ENTER]** when finished.

Select **Test Management Network** on the main vSphere 6 screen to open the **Test Management Network** window.

Perform the following tests:

- Ping the default gateway.
- Ping the DNS server.
- Resolve a known address.

Return to the main vSphere 6.7 screen when you have completed testing, and then select **Troubleshooting Options**. The **Troubleshooting Mode Options** window displays.

In order to install the NVDIA VIB in a later step, you will need to enable the vSphere Shell. This can be accomplished by selecting **Enable vSphere Shell**.

Press **[ENTER]** to toggle **Enable vSphere Shell** on.
The window on the right displays the status: **Enable vSphere Shell Disabled**

Enable SSH by selecting **Enable SSH** and press **[ENTER]** to toggle this option on.
The window on the right displays the status: **SSH is Enabled**

Chapter 4.
INSTALLING VMWARE vCENTER 6.7 SERVER

This chapter covers installing VMware vCenter 6.7 Server, including:

- An installation walk-through
- The initial vCenter Server Configuration
- Adding a host
- Setting a vCenter appliance to auto-start
- Mounting an NFO ISO data store

Review the prerequisites in section 2.7.4, *General Prerequisites* on page 18 before proceeding with these installations.

NOTE: This deployment guide assumes you are building an environment for a proof of concept. Refer to VMware best practice guides before building your production environment.

4.1 INSTALLING vCENTER 6.7 SERVER APPLIANCE

4.1.1 About VCSA

The VCSA is a pre-configured virtual appliance built on Project Photon OS. Since the OS has been developed by VMware it benefits from enhanced performance and boot times over the previous Linux based appliance. Furthermore, the embedded vPostgres database means VMware have full control of the software stack, resulting in significant optimization for vSphere environments and quicker release of security patches and bug fixes. The VCSA scales

up to 2000 hosts and 35,000 virtual machines. A couple of releases ago the VCSA reached feature parity with its Windows counterpart and is now the preferred deployment method for vCenter Server. Features such as Update Manager are bundled into the VCSA, as well as file-based backup and restore, and vCenter High Availability. The appliance also saves operating system license costs and is quicker and easier to deploy and patch.

Software Considerations

- VCSA 6.7 must be deployed to an ESXi host or vCenter running v5.5 or above. However, all hosts you intend to connect to vCenter Server 6.7 should be running ESXi 6.0 or above, hosts running 5.5 and earlier cannot be managed by vCenter 6.7 and do not have a direct upgrade path to 6.7.

- You must check compatibility of any third-party products and plugins that might be used for backups, anti-virus, monitoring, etc. as these may need upgrading for vSphere 6.7 compatibility.

- To check version compatibility with another VMware products, see the Product Interoperability Matrix.

- The points above are especially important since at the time of writing vSphere 6.7 is new enough that other VMware and third-party products may not have released compatible versions. Verify before installing vSphere 6.7 and review the Release Notes and Important information before upgrading to vSphere 6.7 KB.

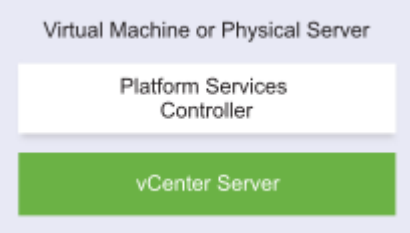
Architectural Considerations

- When implementing a new vSphere 6.7 environment you should plan your topology in accordance with the VMware vCenter Server and Platform Services Controller Deployment Types.

- A series of videos covering vCenter Server and Platform Services Architecture can be found here. If you require further assistance with vCenter planning see also the vSphere Topology and Upgrade Planning Tool here,

- Most deployments will include the vCenter Server and PSC in one appliance, following the embedded deployment model, which I will use in this guide.

- Greenfield deployments of vSphere 6.7 can take advantage of Embedded PSC with Enhanced Linked Mode, providing native vCenter Server HA support, and removal of SSO site boundaries.



Other Considerations

- The VCSA with embedded PSC requires the following hardware resources (disk can be thin provisioned)
    - Tiny (up to 10 hosts, 100 VMs) – 2 CPUs, 10 GB RAM.
    - Small (up to 100 hosts, 1000 VMs) – 4 CPUs, 16 GB RAM.
    - Medium (up to 400 hosts, 4000 VMs) – 8 CPUs, 24 GB RAM.
    - Large (up to 1000 hosts, 10,000 VMs) – 16 CPUs, 32 GB RAM.
    - X-Large (up to 2000 hosts, 35,000 VMs) – 24 CPUs, 48 GB RAM – new to v6.5.
- Storage requirements for the smallest environments start at 250 GB and increase depending on your specific database requirements. See the Storage Requirements document for further details.
- Where the PSC is deployed as a separate appliance this requires 2 CPUs, 4 GB RAM, 60 GB disk.
- Environments with ESXi host(s) with more than 512 LUNs and 2048 paths should be sized large or x-large.

- The ESXi host on which you deploy the VCSA should not be in lockdown or maintenance mode.

- All vSphere components should be configured to use an NTP server. The installation can fail or the vCenter Server Appliance vpxd service may not be able to start if the clocks are unsynchronized.

- FQDN resolution should be in place when deploying vCenter Server.

- A list of Required Ports for vCenter Server and PSC can be found here.

- The configuration maximums for vSphere 6.7 can be found here.

- In vSphere 6.7 TLS 1.2 is enabled by default. TLS 1.0 and TLS 1.1 are disabled by default, review the Release Notes for more information.

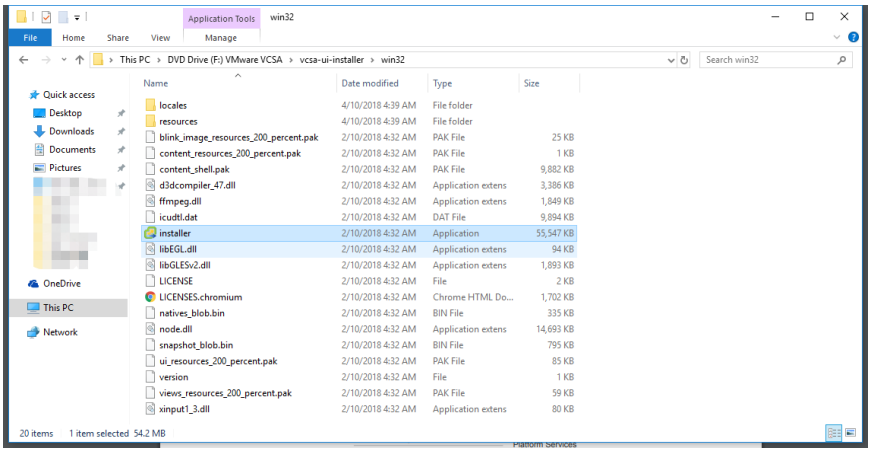
- There are a number of Intel and AMD CPUs no longer supported with vSphere 6.7, review the Release Notes for a full list of unsupported processors.

4.1.2 vCenter Server Appliance (VCSA) Installation

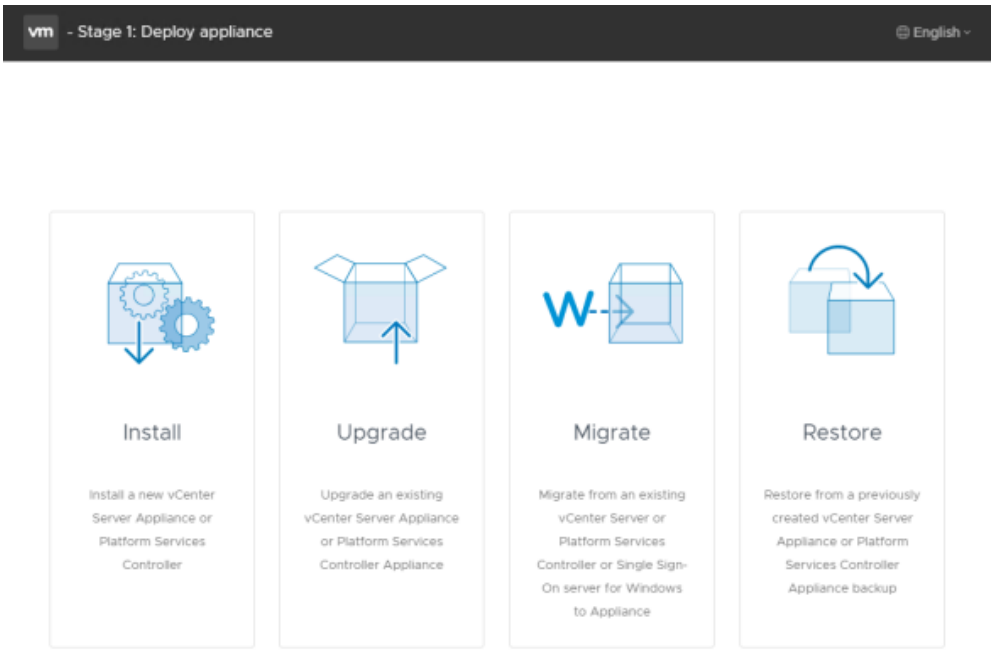
Download the VMware vCenter Server Appliance 6.7 ISO from VMware downloads: v6.7.0.

1. Mount the ISO on your computer. The VCSA 6.7 installer is compatible with Mac, Linux, and Windows.

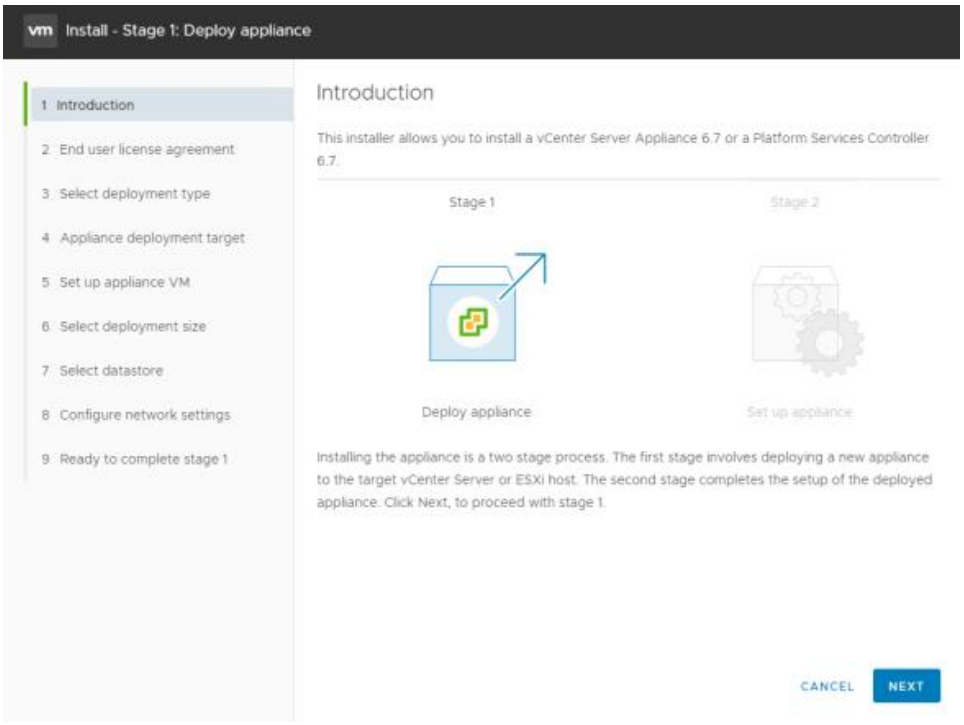
2. Browse to the corresponding directory for your operating system, e.g. \vcsa-ui-installer\win32. Right click **Installer** and select **Run as administrator**.



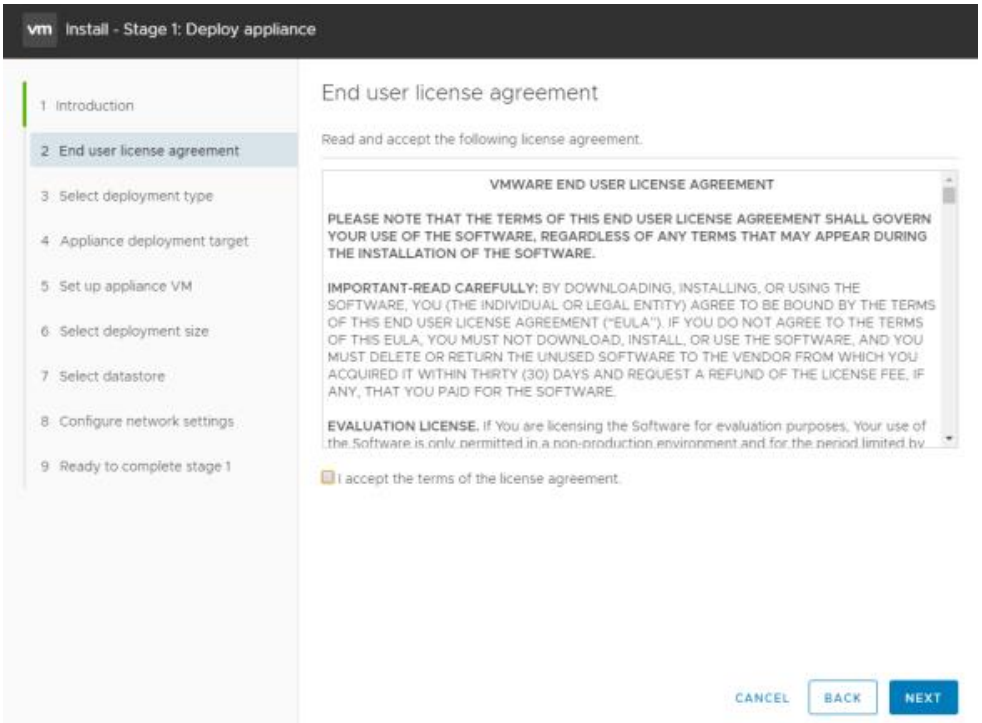
3. As we are installing a new instance click **Install**.



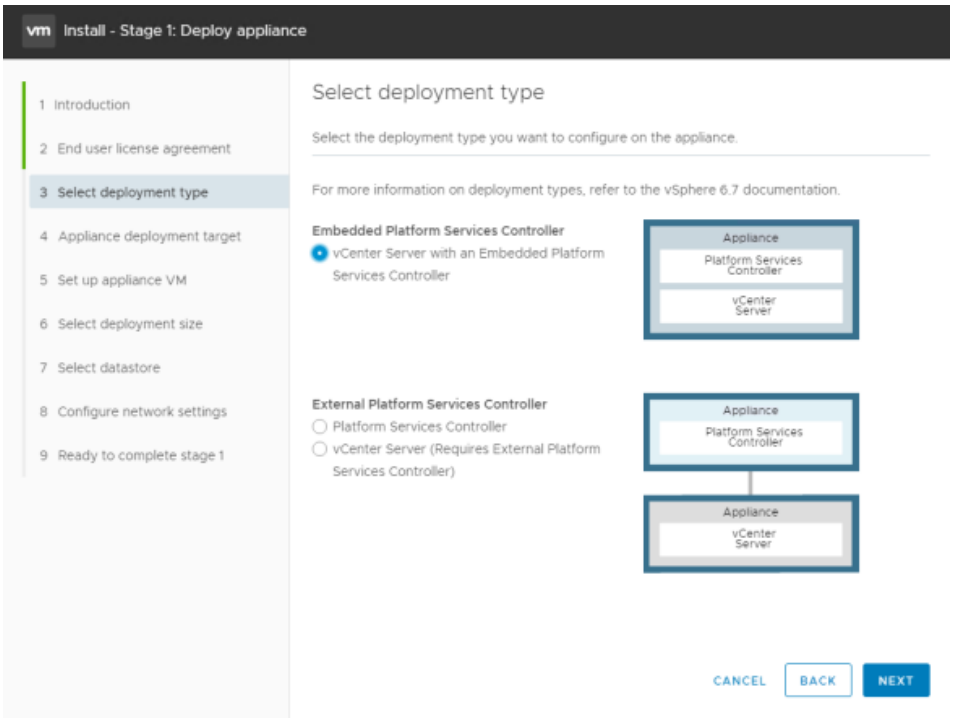
4. The installation is split into 2 stages, we begin with deploying the appliance. Click **Next**.



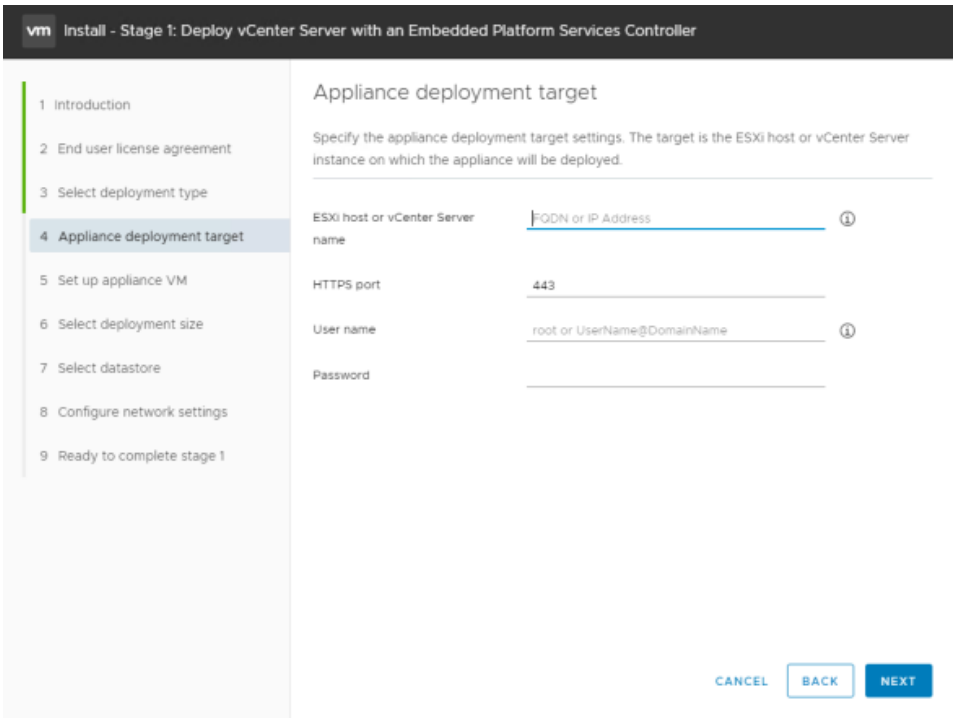
5. Read and then accept the EULA, click **[Next]** to continue:



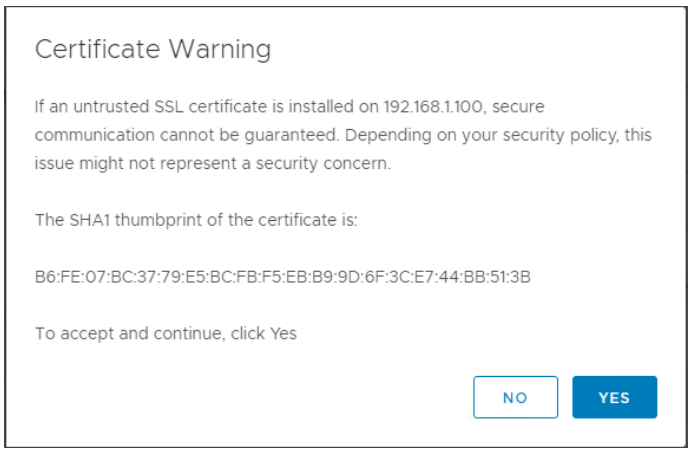
6. Select the deployment model, in this example we will be using an embedded deployment combining the vCenter Server and Platform Services Controller in one appliance, click **Next**.



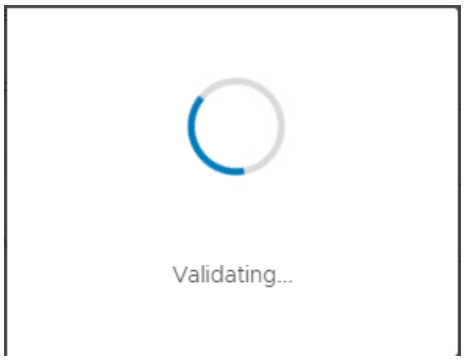
7. In this step you are selecting the vSphere host to install the VCSA on as a guest, this can be on a host running ESXi 5.5 or later. It is recommended that the vCenter server (Windows or appliance based) run on a separate management cluster from the one designated for VDI workloads. Enter the IP address or FQDN of the chosen host, then its root user name and password and click **[Next]**:



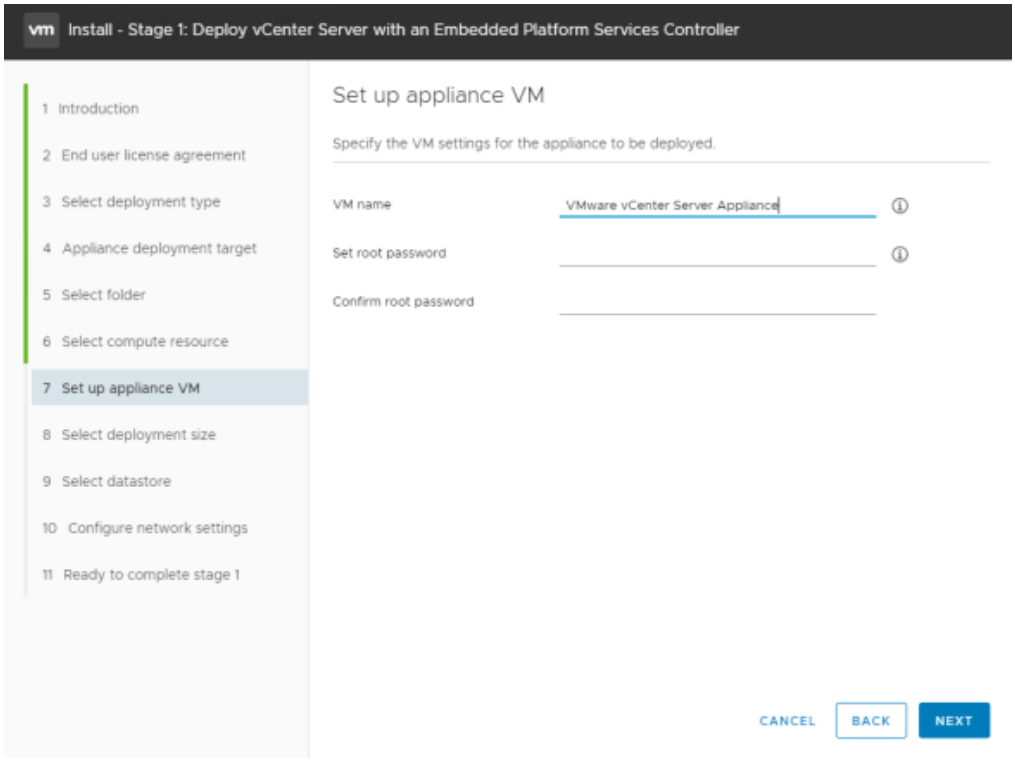
8. If your desktop can reach the host then you should see a certificate warning as it connects, this is due to the use of a self-signed cert. If you are using signed certificate, then you will not see this warning. Click **[Yes]** and continue:



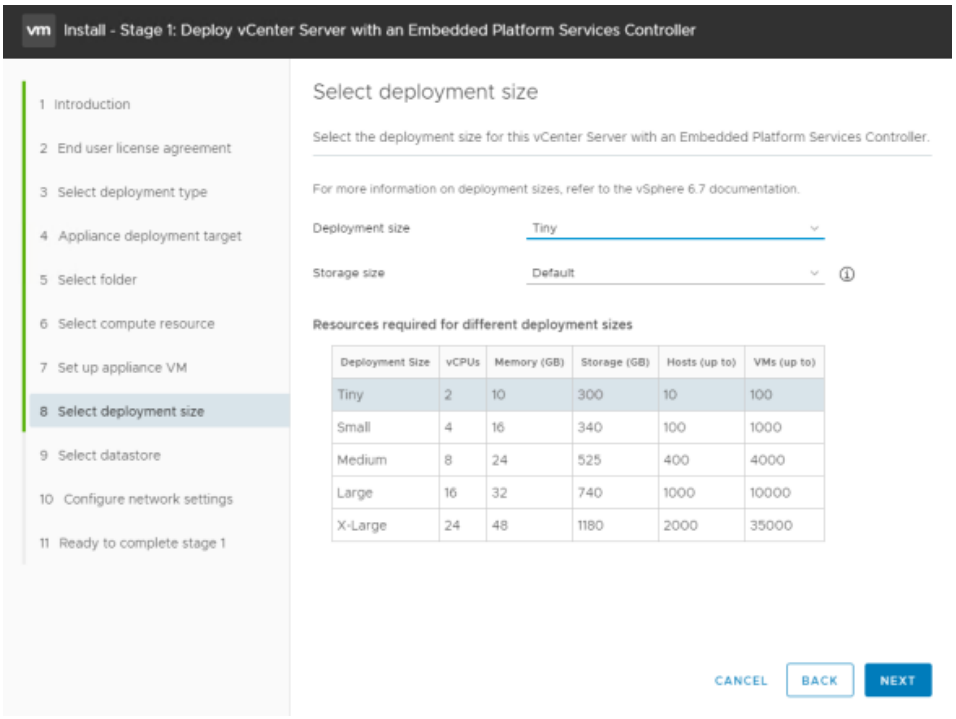
9. The credentials you provided are validated:



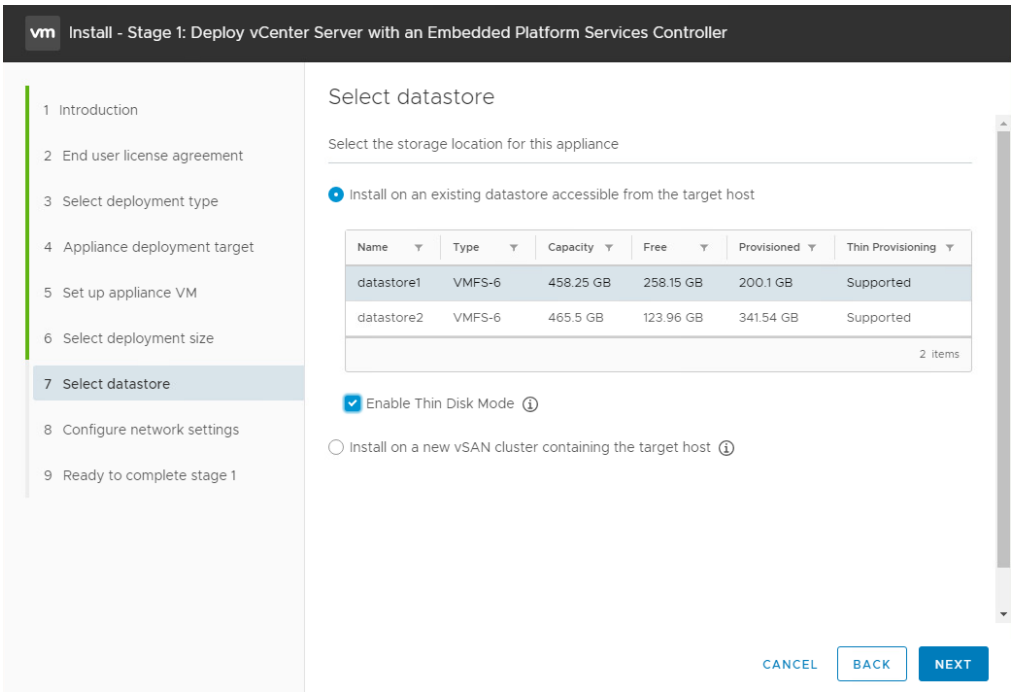
10. With a successful connection you are now prompted to name the appliance, then enter a root password for the appliance, enter it twice and click **Next**:



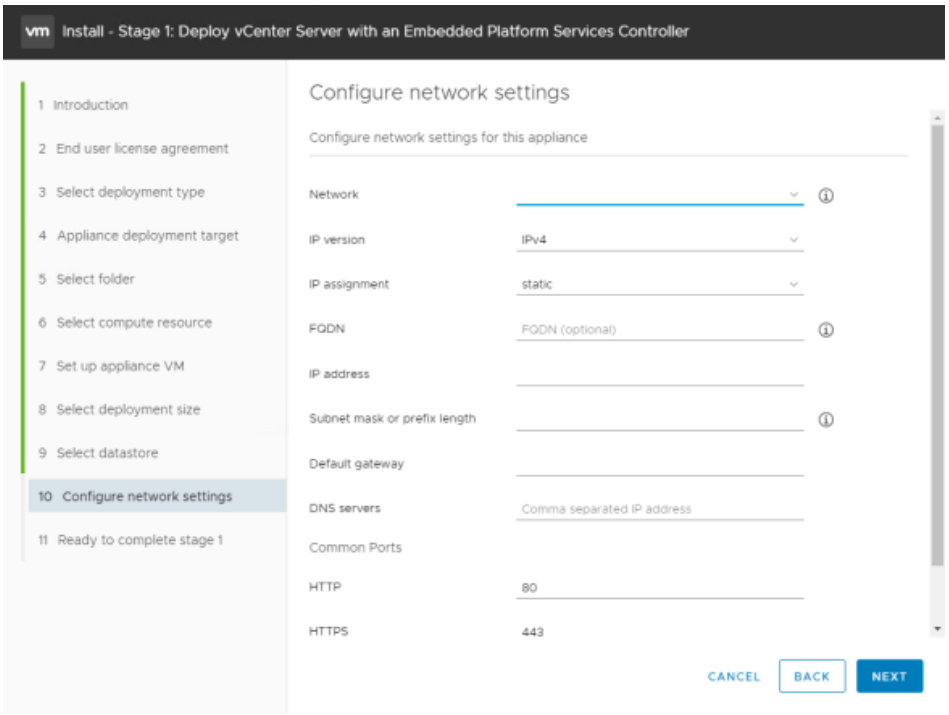
11. Select the deployment size in line with the number of hosts and virtual machines that will be managed, click **Next**.



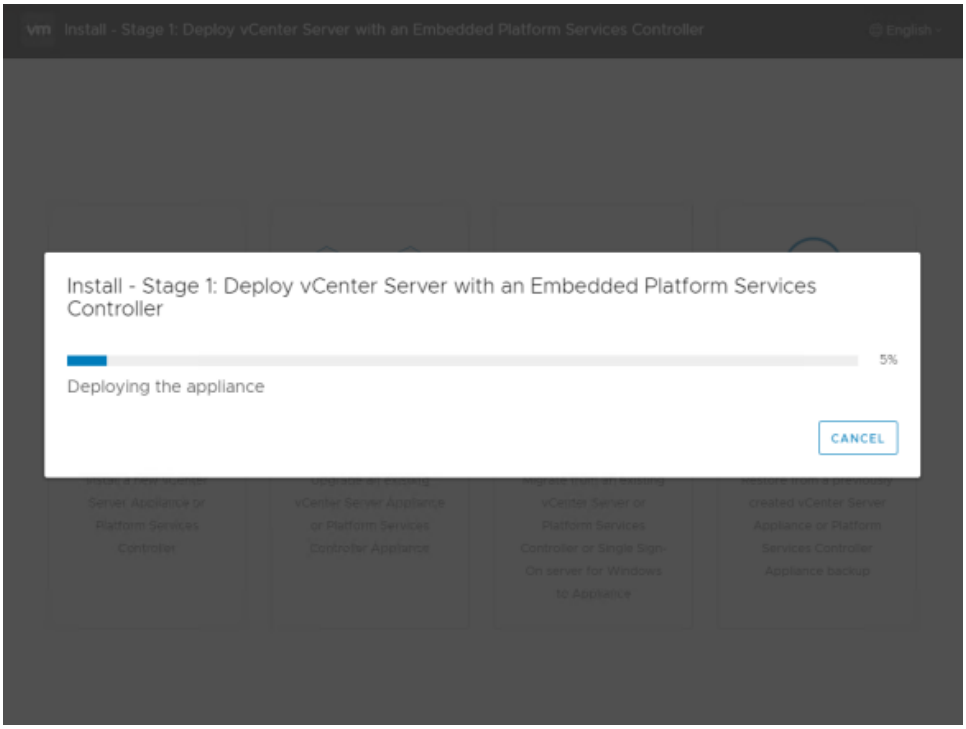
12. Select the datastore where the VCSA will be deployed, select thin provisioning if required, and click **Next**. Configure the network settings for the appliance and click **Next**.



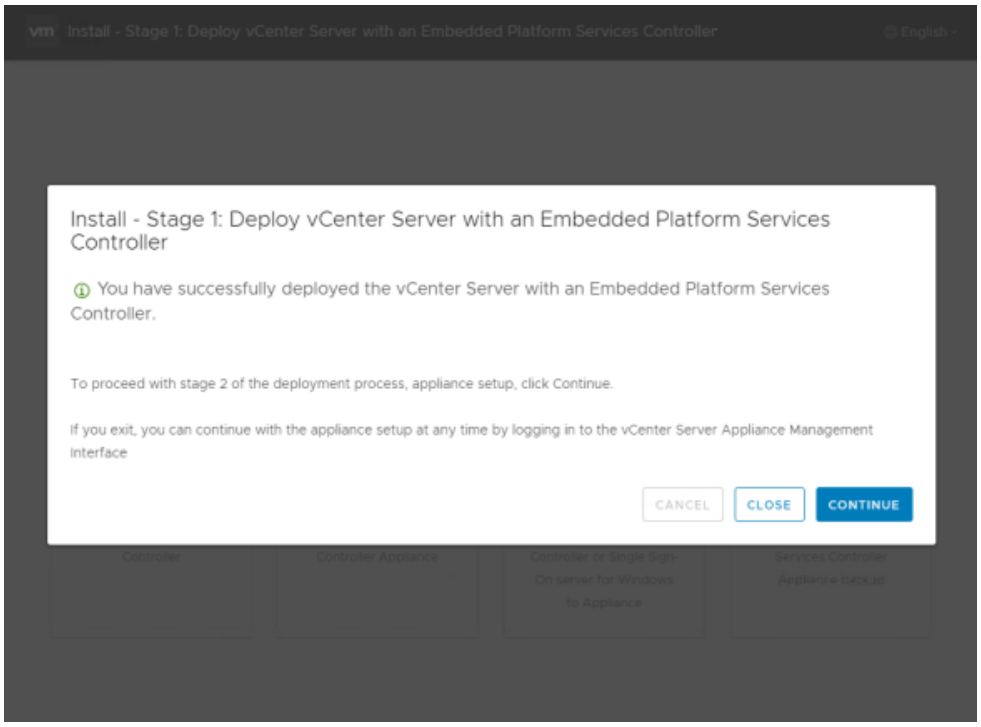
13. Configure network settings, this is a long page and will require scrolling down to see all settings. Before configuring these settings, choose an appropriate static IP address and enter that into local DNS (e.g. on the Domain Controller). Once you can resolve the address, enter that IP address, host name, and then scroll down for remaining entries:



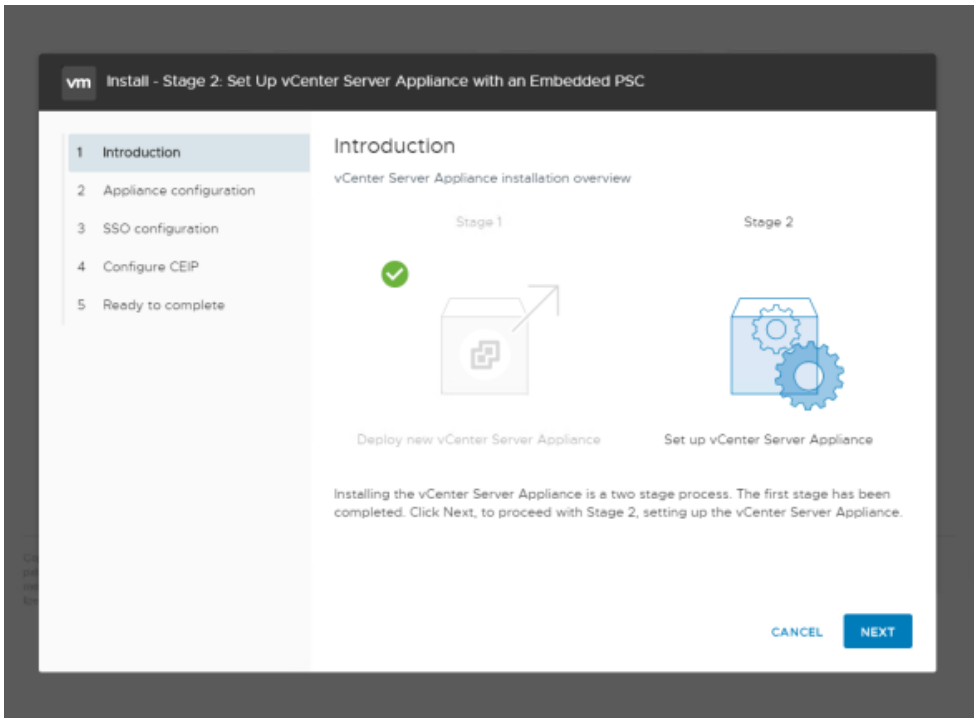
14. On the summary page click **Finish**. The appliance will now be deployed.



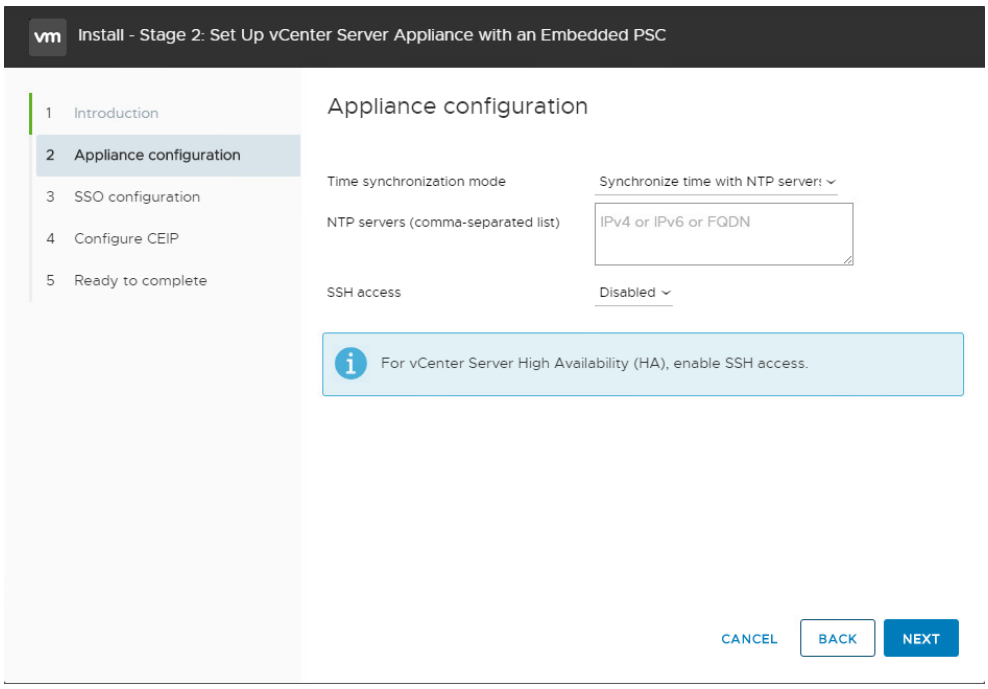
15. With the VCSA now deployed we can move on to stage 2, click **Continue**.



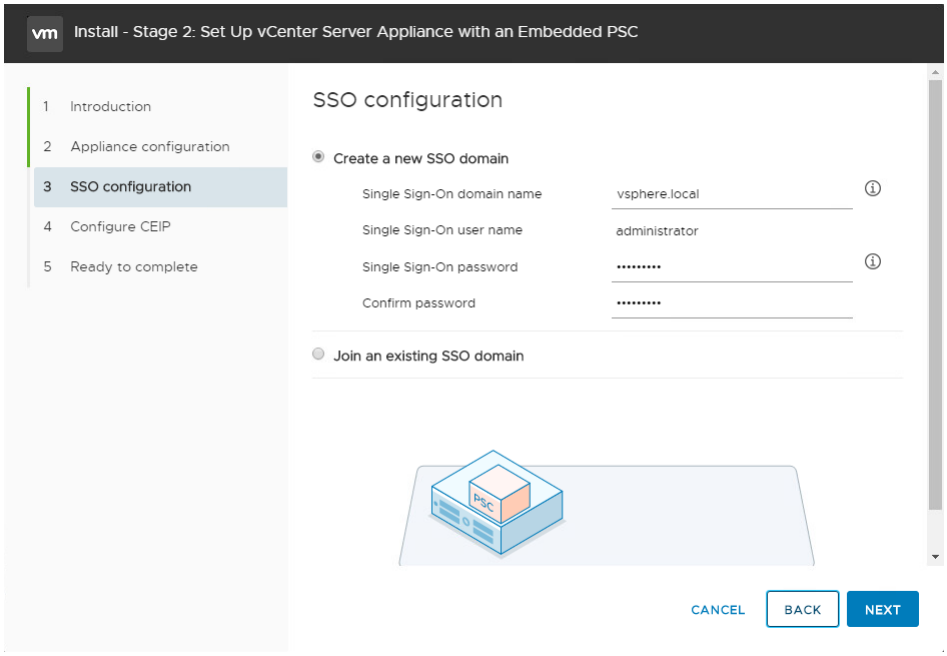
16. Click **Next** to begin the VCSA setup.



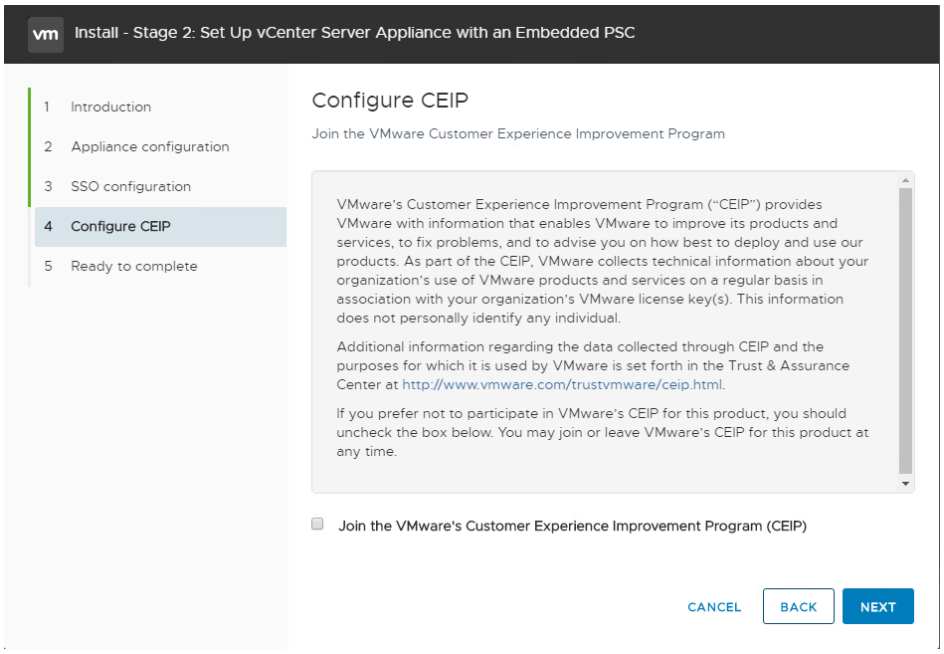
17. Configure the NTP servers, enable SSH access if required, and click **Next**.



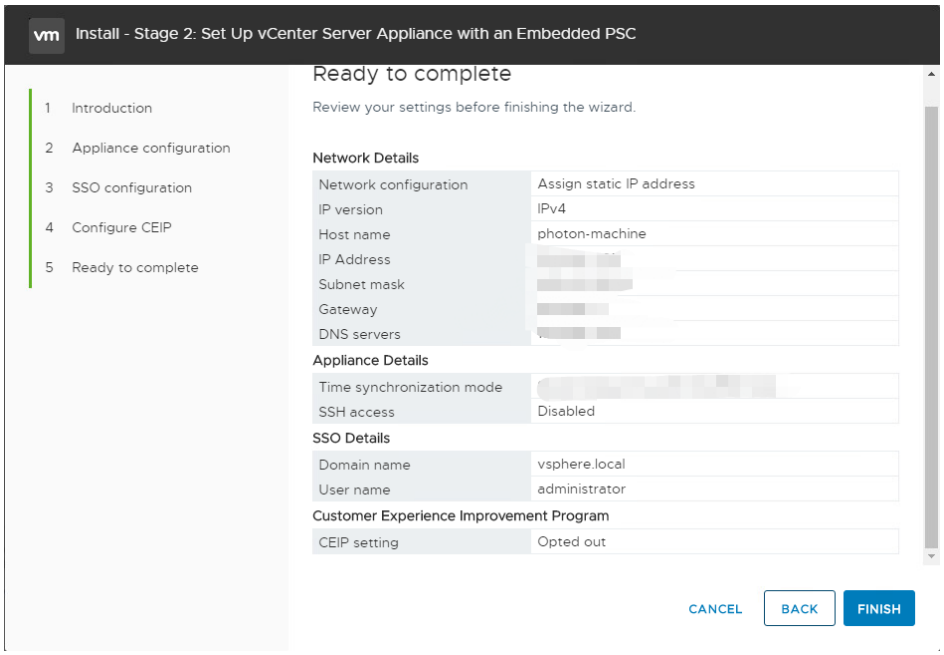
18. Enter a unique SSO domain name, the default is vSphere. Local. **The SSO domain name should not be the same as your Active Directory Domain**. Configure a password for the SSO administrator, click **Next**.



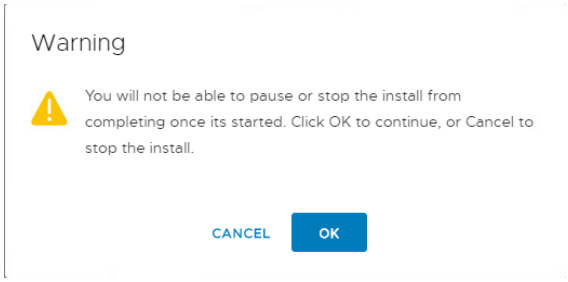
19. Select or deselect the customer experience improvement program box and click **Next**.

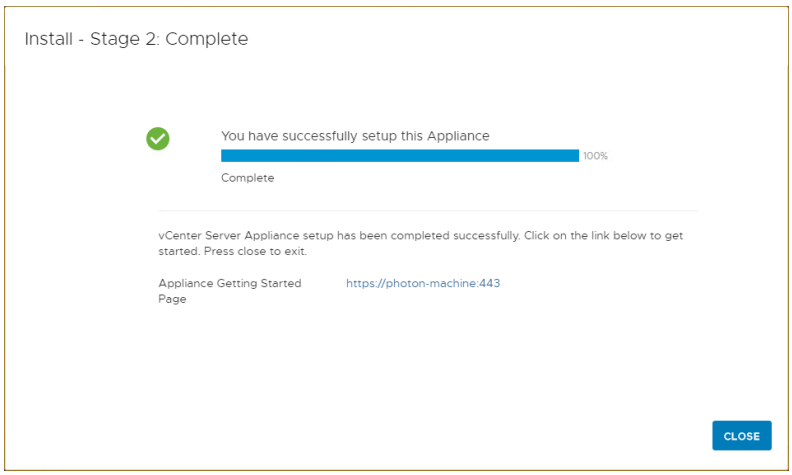


20. Review the details on the summary page and click **Finish**.



21. Click **Ok** to acknowledge that the VCSA setup cannot be paused or stopped once started. When the installer is complete click **Close** to close the wizard.





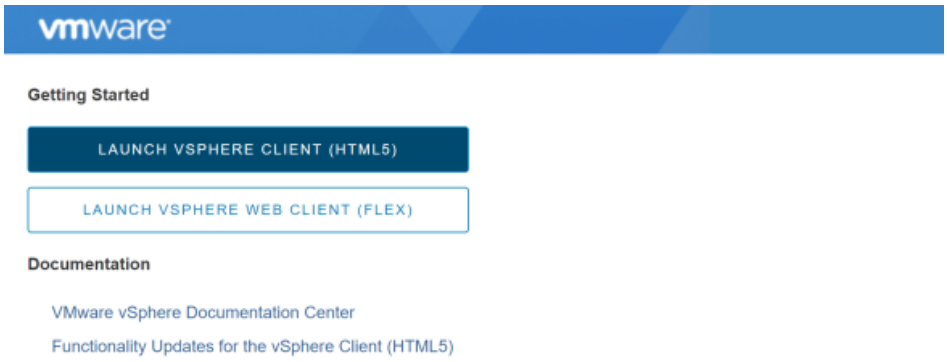
4.2 POST INSTALLATION

This section describes post install and configure vCenter 6.7 Server.

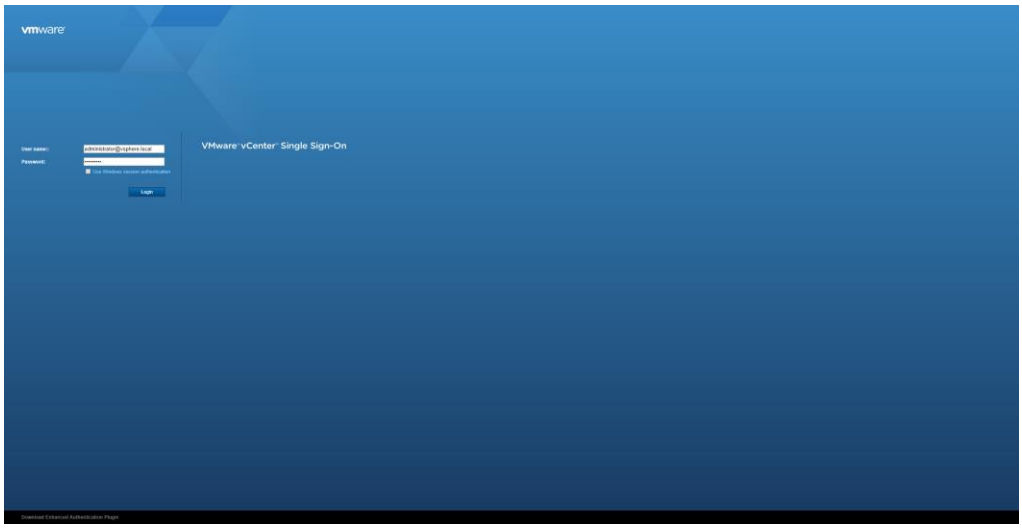
4.2.1 Adding Licenses to Your vCenter Server

Use the following procedure to configure vCenter 6.7:

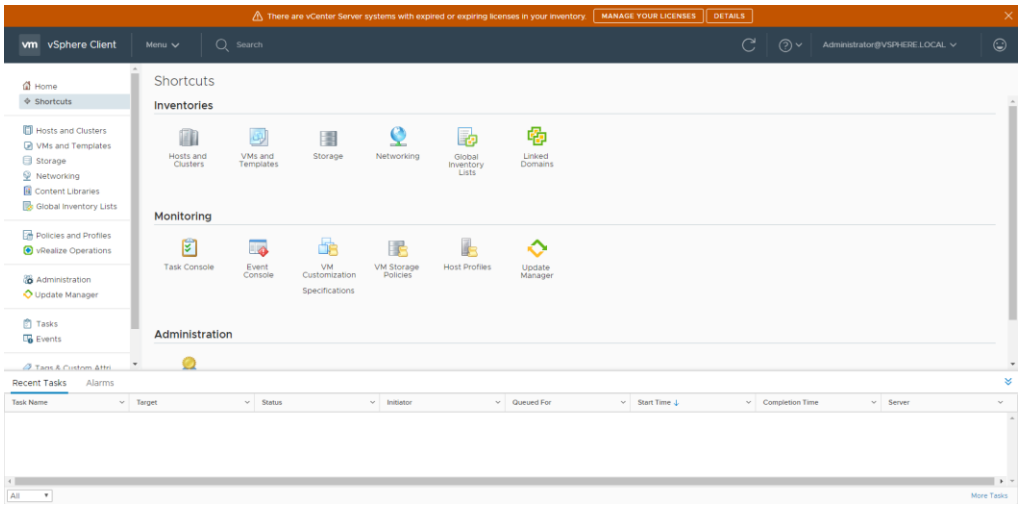
1. Connect to the vCenter post install using the IP or FQDN of the vCenter. Access vSphere by clicking either **Launch vSphere Client (HTML5)** or **Launch vSphere Web Client (FLEX)**. As the web client will be depreciated in future versions, and the HTML5 client is now nearly at full feature parity, we will use the HTML5 vSphere client.



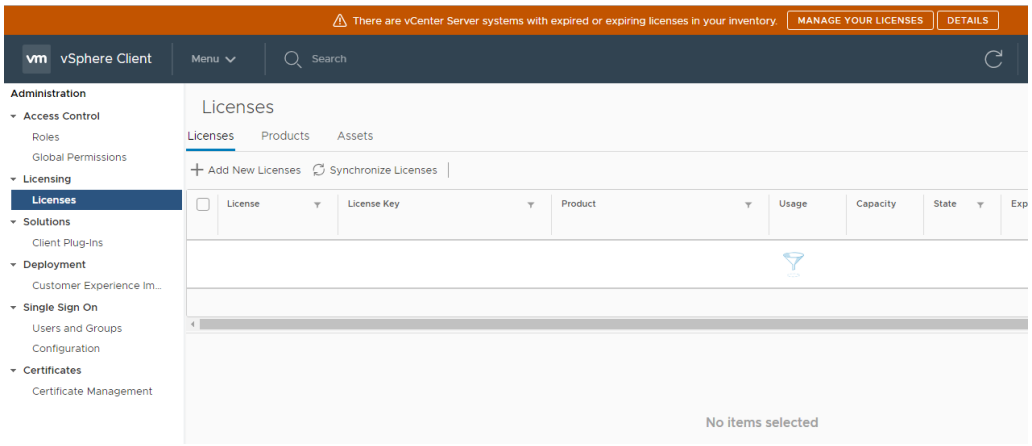
The *VMware Single Single-On* page displays:



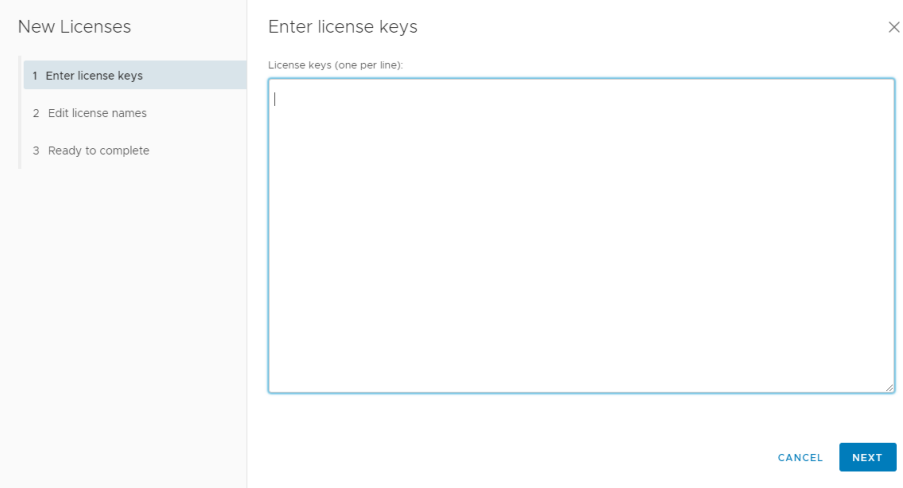
2. Enter the username and password that you specified during installation, and then click the **Login** button. The *VMware vSphere Web Client* page displays.



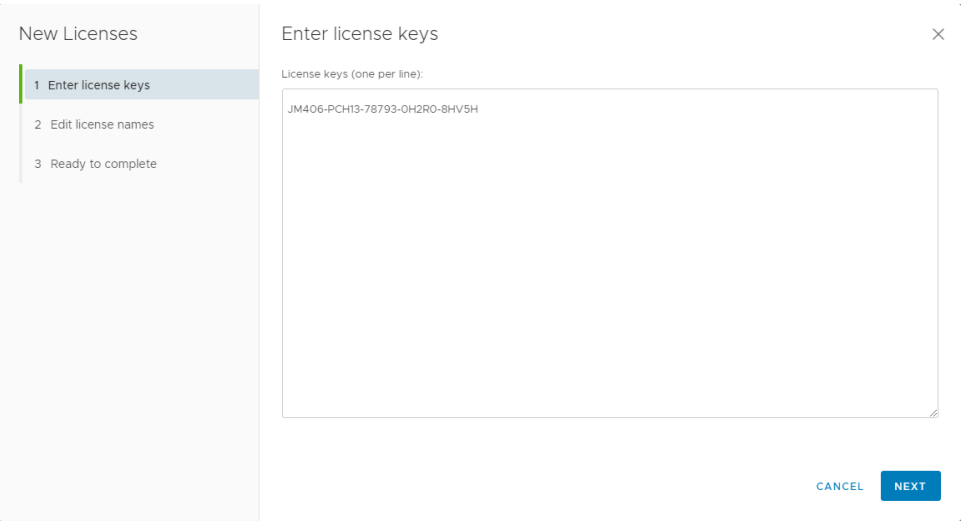
3. You must apply a new vCenter license key within 60 days. If you have purchased vCenter Server then log into your licensing portal here. If the license key does not appear then check with your VMware account manager. Log in to the vSphere Web Client using the SSO administrator login. From the **Menu** drop-down click **Administration**,



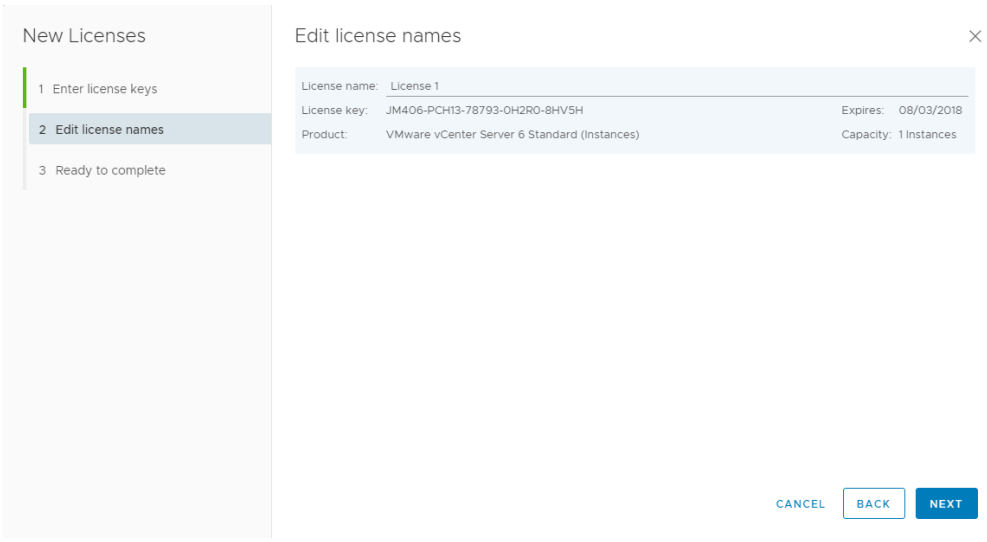
4. Select Licenses from the left-hand menu and then select the Licenses tab to open the Licenses tab. Click **Add New Licenses** to open the New Licenses popup.



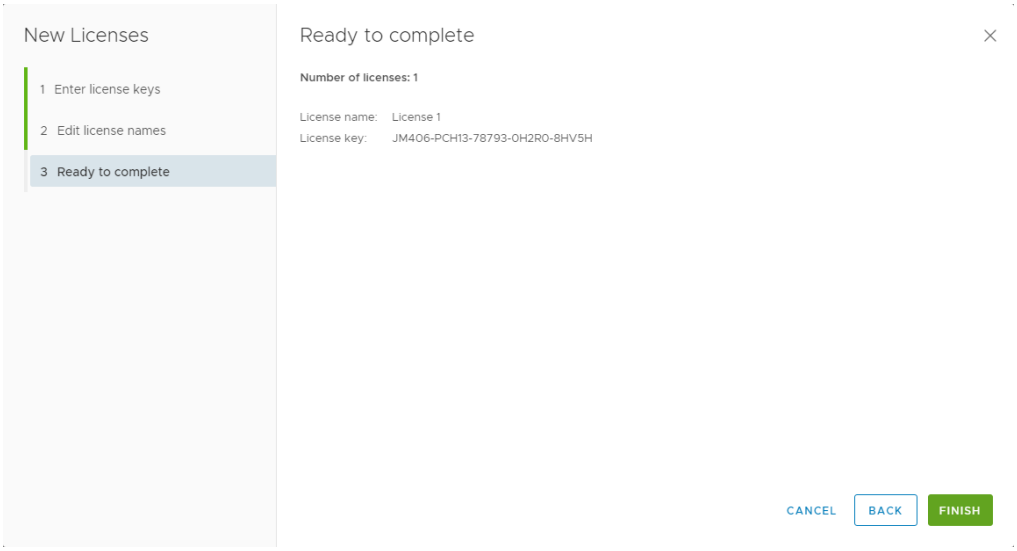
5. Enter the Center Server Standard license key provided at the vSphere beta program website



6. Enter a unique name for the license in the License Name field and then click **Next**.



7. Review your selections and then click **Finish** to close the Enter New License popup and return to the VMware vSphere Web Client page.



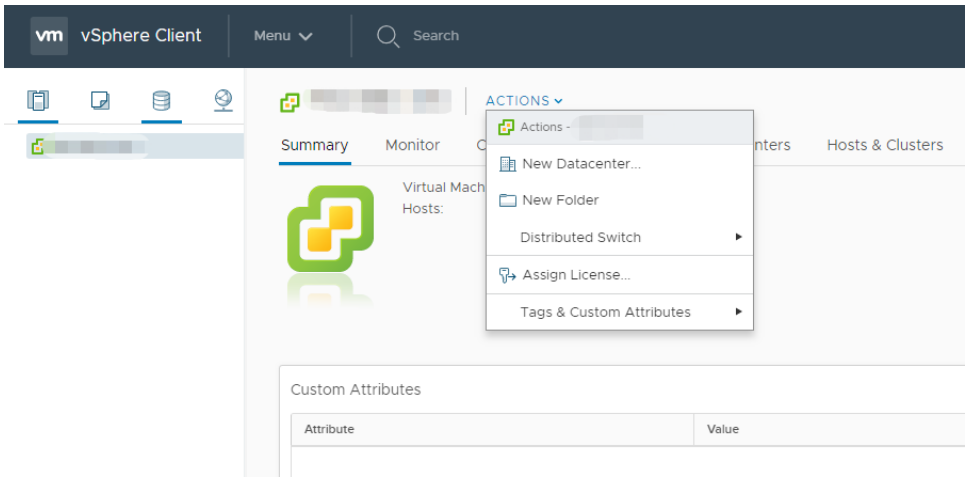
4.2.2 Adding a Host

Use the following procedure to add a host in vCenter 6.7:

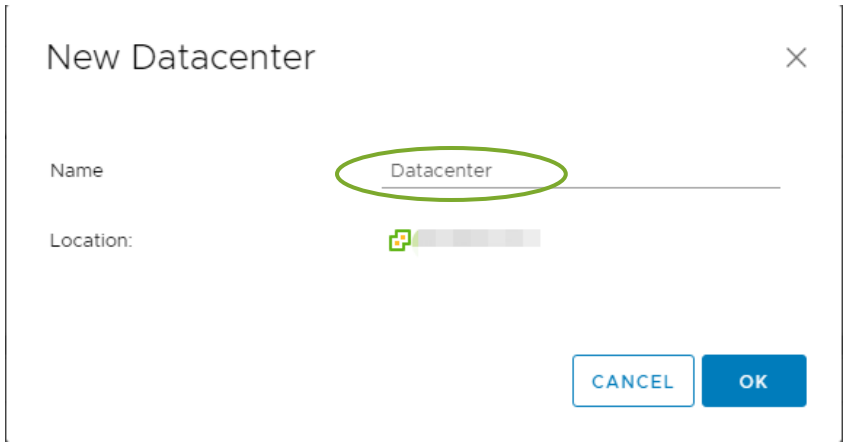
1. Select the **Home** icon (house) on the *VMware vSphere Web Client* page

2. Select **Hosts** and **Clusters**.

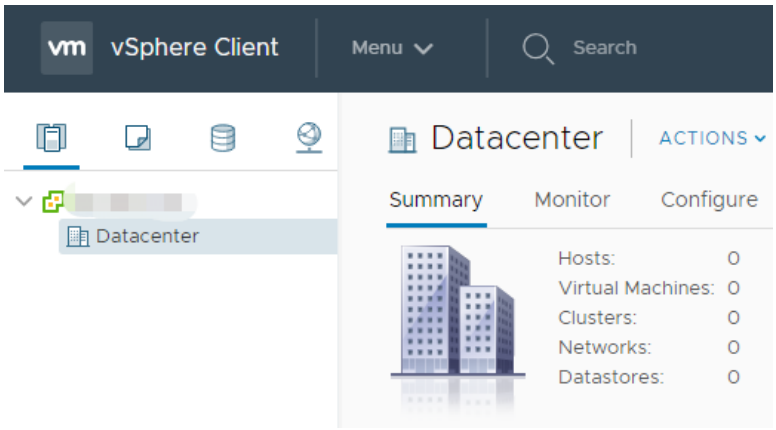
3. Drop down **ACTIONS** and select **New Datacenter.**



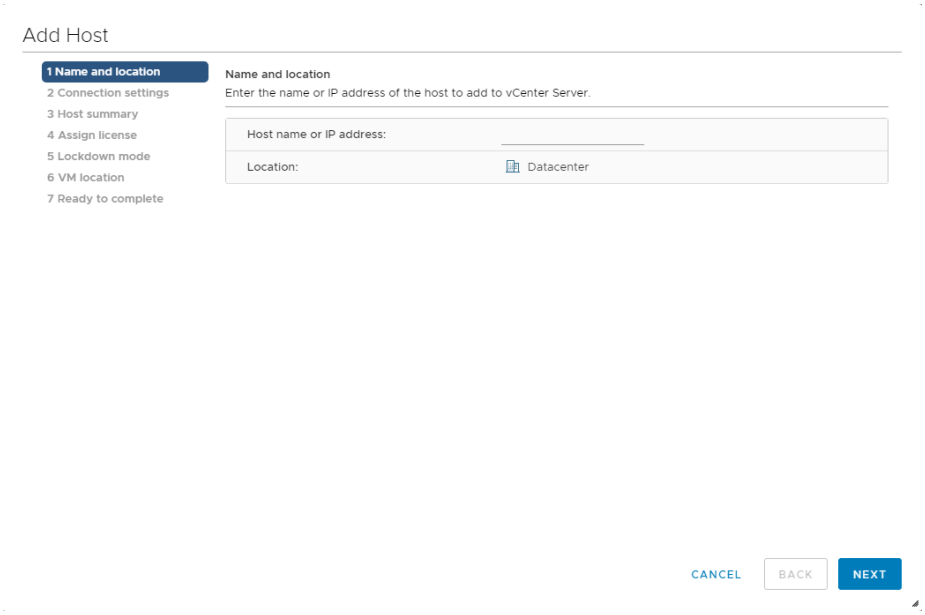
The New Datacenter popup Displays



4. Enter a name for the datacenter in the **Datacenter name** field and click **OK**.
The new datacenter is visible in the left panel of the *vSphere Web Client*.

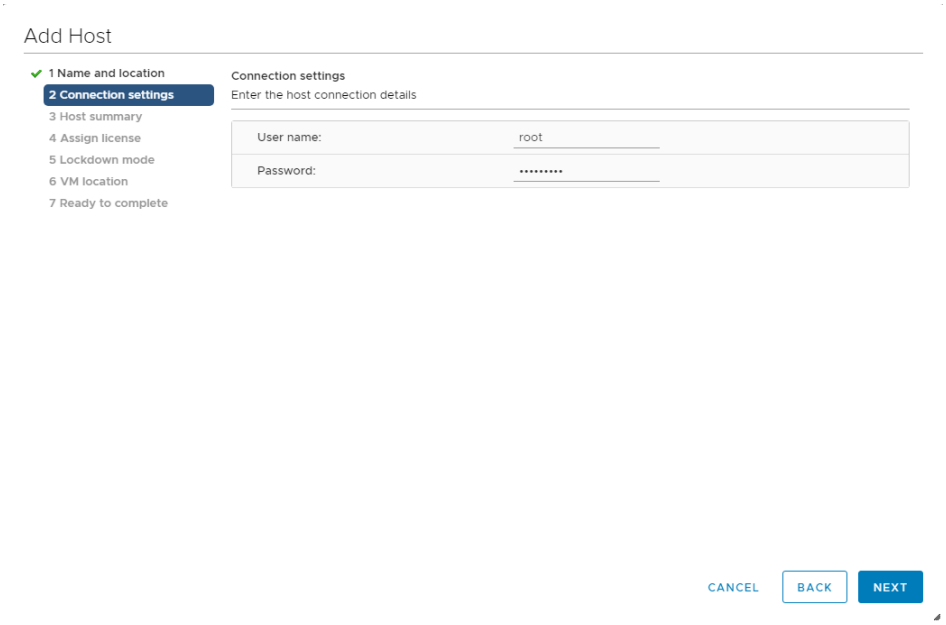


5. Drop down **ACTIONS** and select **Add a Host**.
The *Name and location* dialog open.

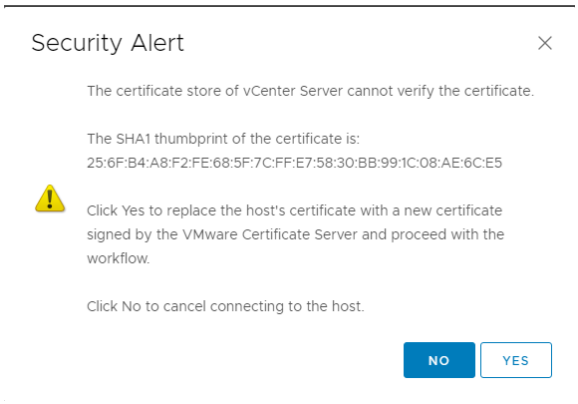


Enter the host name or IP address of the vSphere host and click **Next**.

The ***Connection settings*** dialog displays.



6. Enter the administrator account credentials in the **User name** and **Password** fields and click **Next**. The *Security Alert* popup displays.



Click **Yes** to replace the host certificate.
The *Host summary* dialog displays.

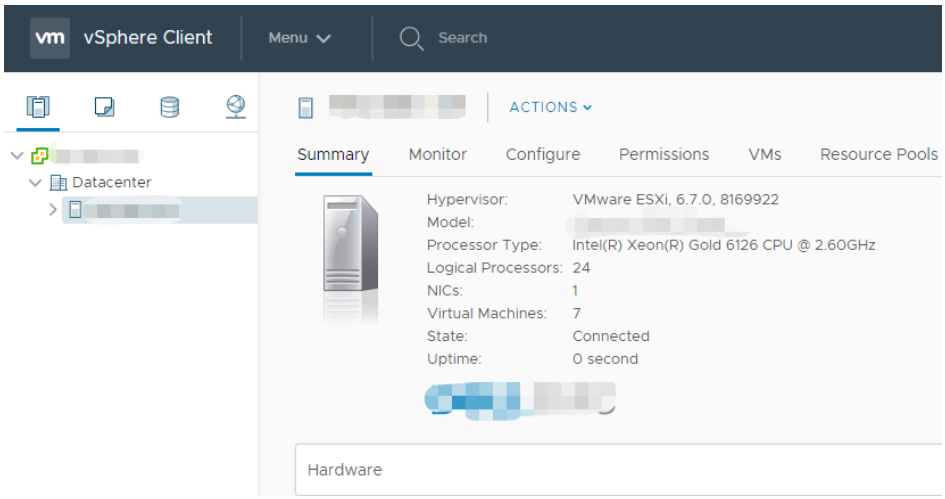
Review the settings and click **Next** to proceed.
The *Assign license* dialog displays.

Confirm the license selection and click **Next**.
The *Lockdown mode* dialog displays.

Accept the default setting (Disabled) and click **Next**.
The *VM location* dialog displays.

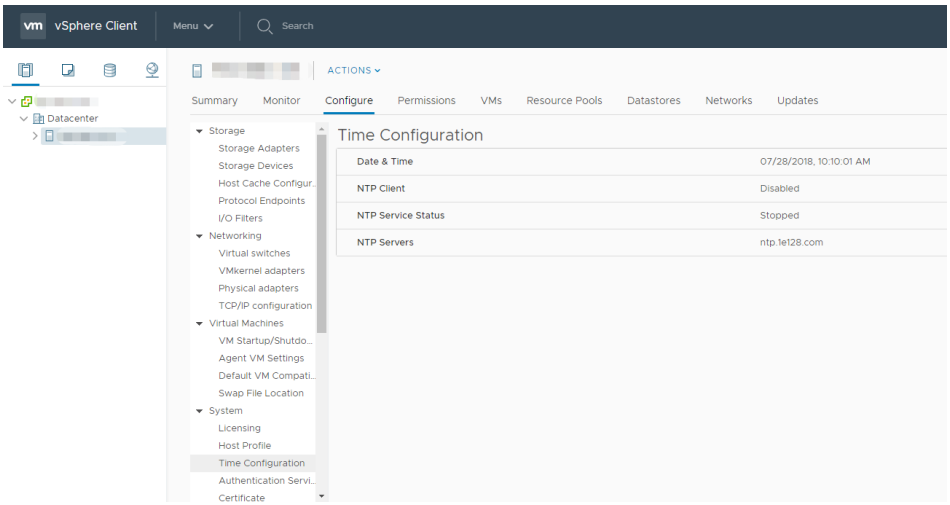
Select a cluster or accept the default option and click **Next** to proceed.
The *Ready to complete* dialog displays.

7. Click **Finish** to complete adding the new host.
The new host is now visible in the left panel when you click the datacenter name.

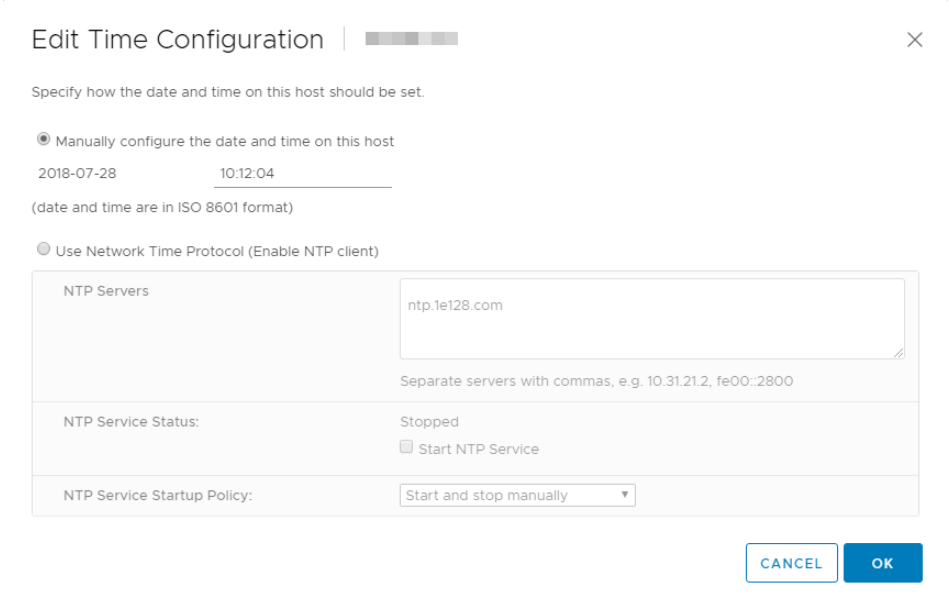


4.2.3 Setting the NTP Service on a Host

1. Click a host object in the menu on the left, click **Configure→System→Time Configuration→Edit**



2. Enter a valid time server and click **OK**:

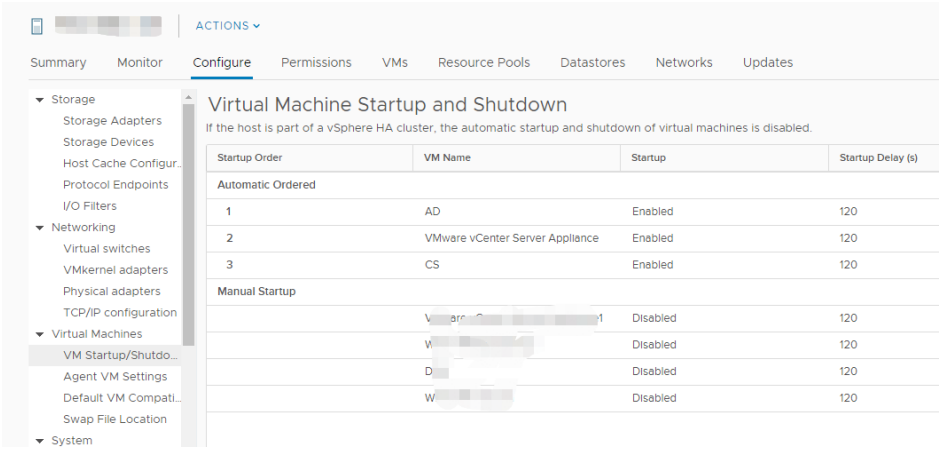


3. Do this for each host to ensure time is accurate for all guests.

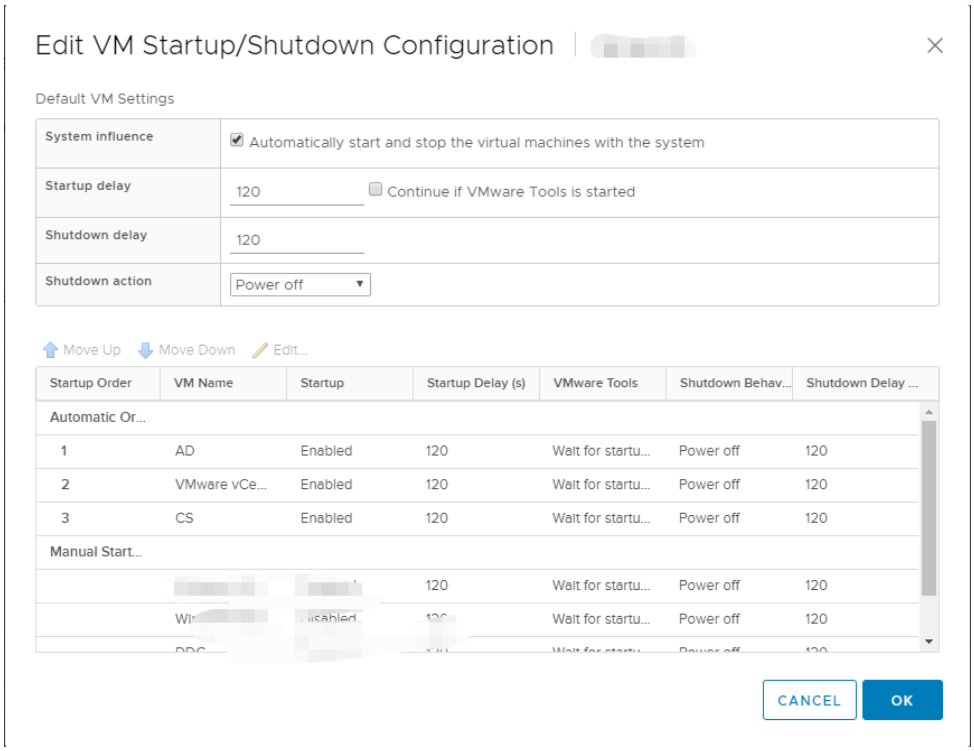
4.2.4 Setting a vCenter Appliance to Auto-Start

Use the following procedure to set a vCenter Appliance to start automatically:

1. In the vSphere Web Client, select the host then select **Configure→ Virtual Machines→ VM Startup/Shutdown.**



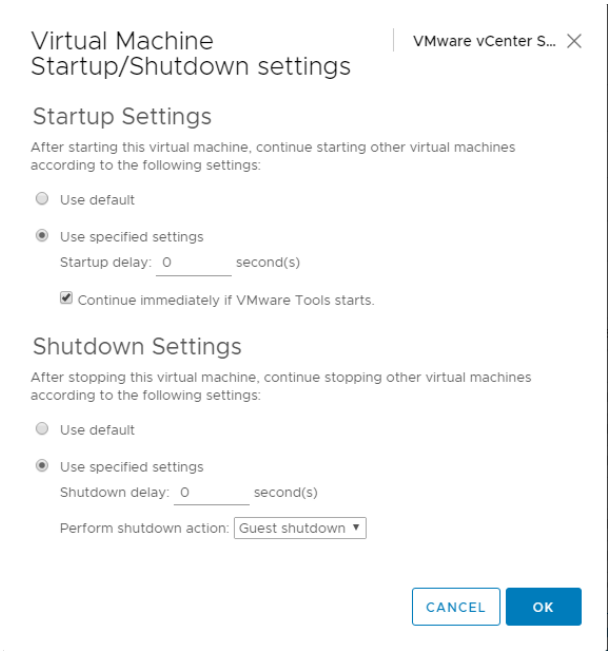
2. Click the **Edit** button.
   The *Edit VM Startup and Shutdown* window displays.



Select the **vCenter Appliance** and click the **Up** arrow to move that virtual machine up to the **Automatic Startup** section. Click the **Edit** button.

Select the following options:

- Set Startup Behavior to Use specified settings and select Continue immediately if VMware Tools starts
- Set Startup Delay to 0
- Set Shutdown Behavior to Use specified settings
- Set Shutdown Delay to 0
- Select Guest Shutdown



Click **OK** to apply the configuration.

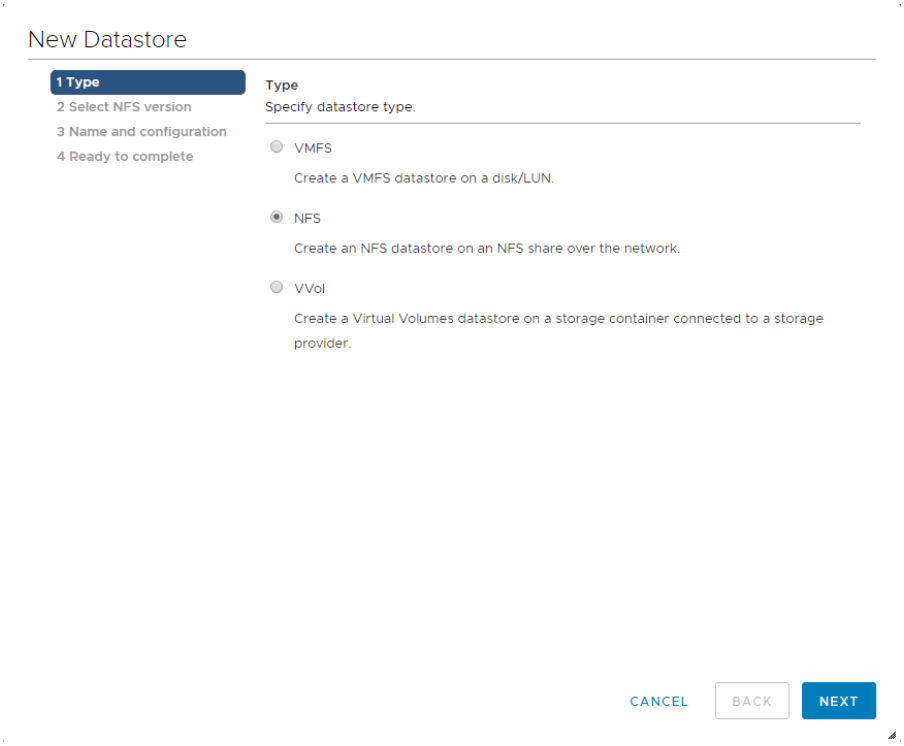
The vCenter Web Client may not reflect these configuration changes immediately. Either click the **Refresh** icon or different configuration group and return to the current setting.

4.2.5 Mounting an NFS ISO Data Store

Use the following procedure to mount an NFS ISO data store:

1. In the main *vSphere Web Client* window, select **Hosts and Clusters** and select the host.

   Select **Storage**→**New Datastore** from the **Actions** drop-down menu.
   The *New Datastore* window displays with the **Type** tab selected.



Select **NFS** and click **Next** to proceed.
The **Select NFS version** tab displays.

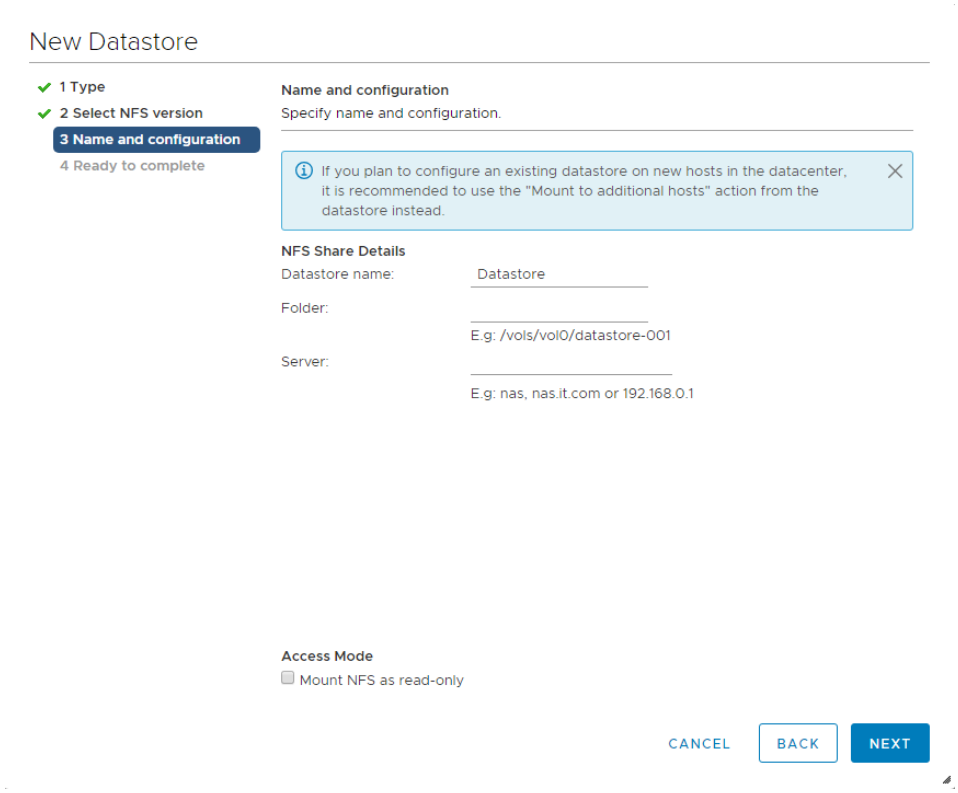
Select the correct NFS version and click **Next** to proceed.
The *Name and configuration* tab displays.

Enter the NFS exported folder path and the NFS server address in the **Folder** and **Address** fields, respectively.
Since this an ISO data store, consider mounting it as read-only by checking the **Mount NFS** as read-only checkbox.

Click **Next** to proceed.
The **Host accessibility** tab displays.

Select the host that will use the new data store,

Select **Next** to proceed.
The **Ready to complete** tab displays.



Review the settings.

Click **Finish** to complete adding the NFS ISO data store.
This data store is now accessible as an installation source for virtual machine CD drives.

Chapter 5. BUILDING HORIZON 7

This chapter covers installing VMware Horizon 7, including:

- Building Horizon 7
- Registering the license
- Registering a vCenter Server
- Installing a Horizon View Composer Server
- Registering a Horizon View Composer Server

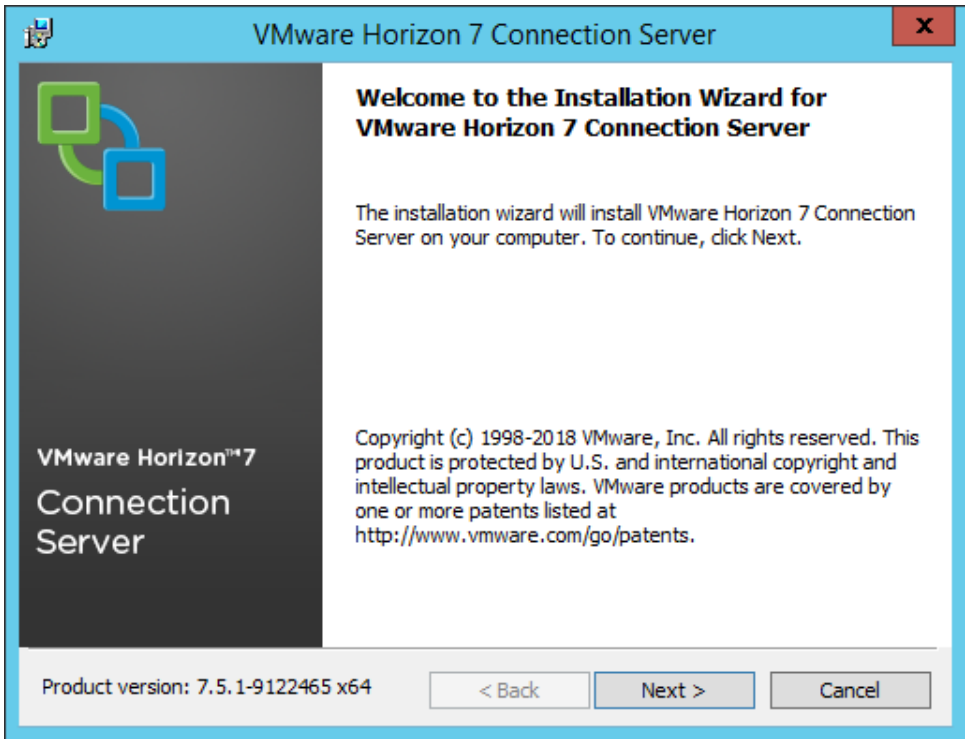
5.1 INSTALLING HORIZON 7

The server on which you are installing Horizon View must meet the requirements listed in “General Prerequisites” on page 20. As of the publication date of this manual the current version of the View Connection Server is:

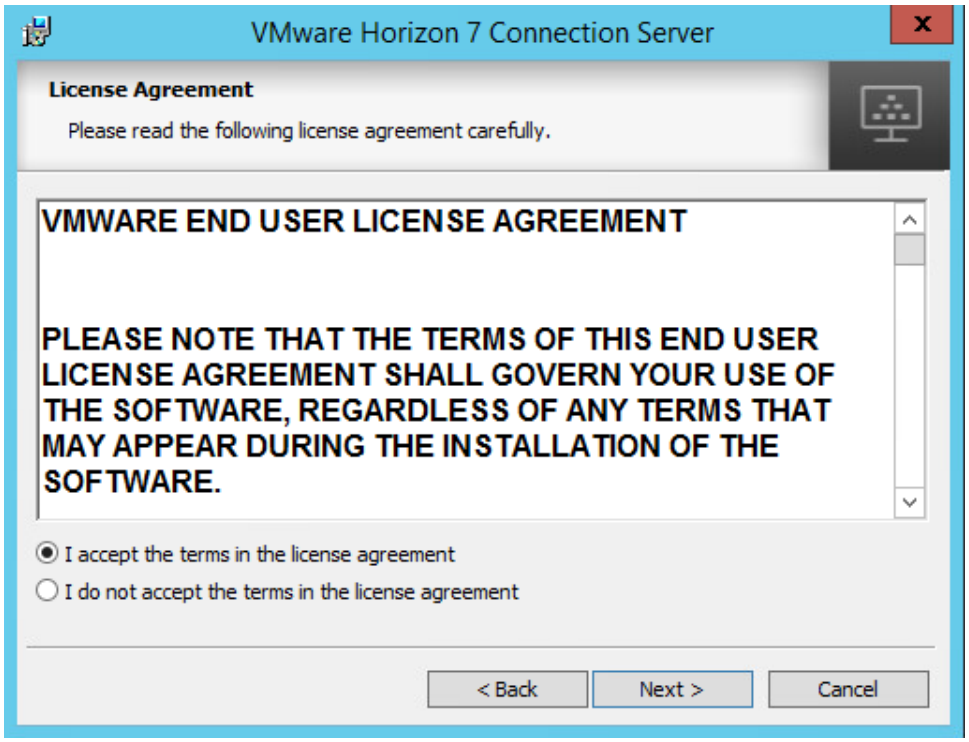
VMware-viewconnectionserver-x86_64-7.5.1-9122465.exe

Use the following procedure to install View 7.5:

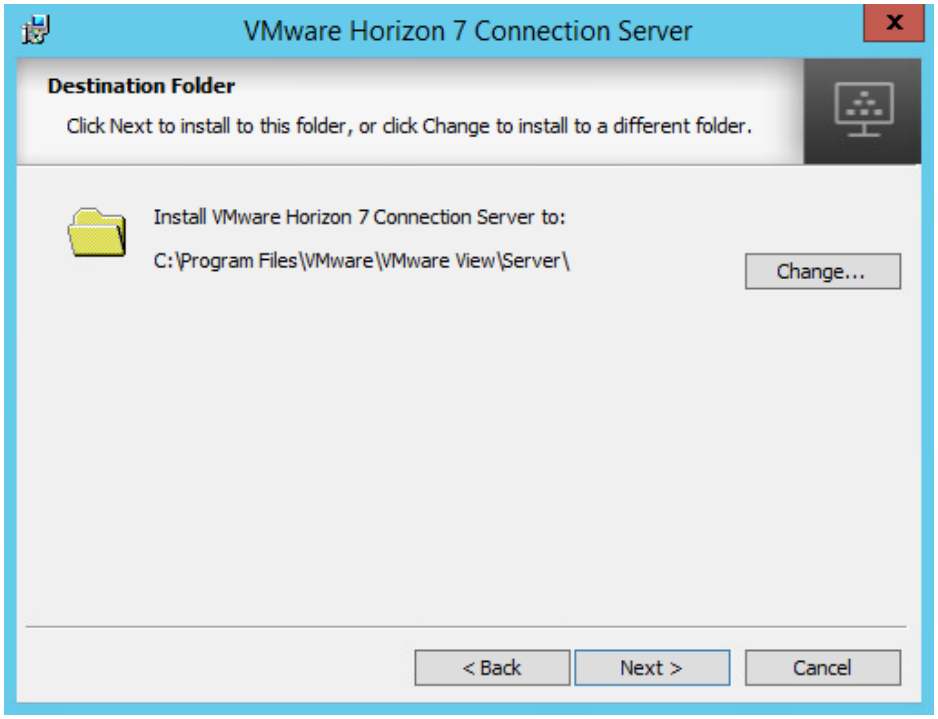
2. Launch the installer to display the ***Welcome to the Installation Wizard*** window.



Select **Next** to display the ***End User License Agreement*** window.

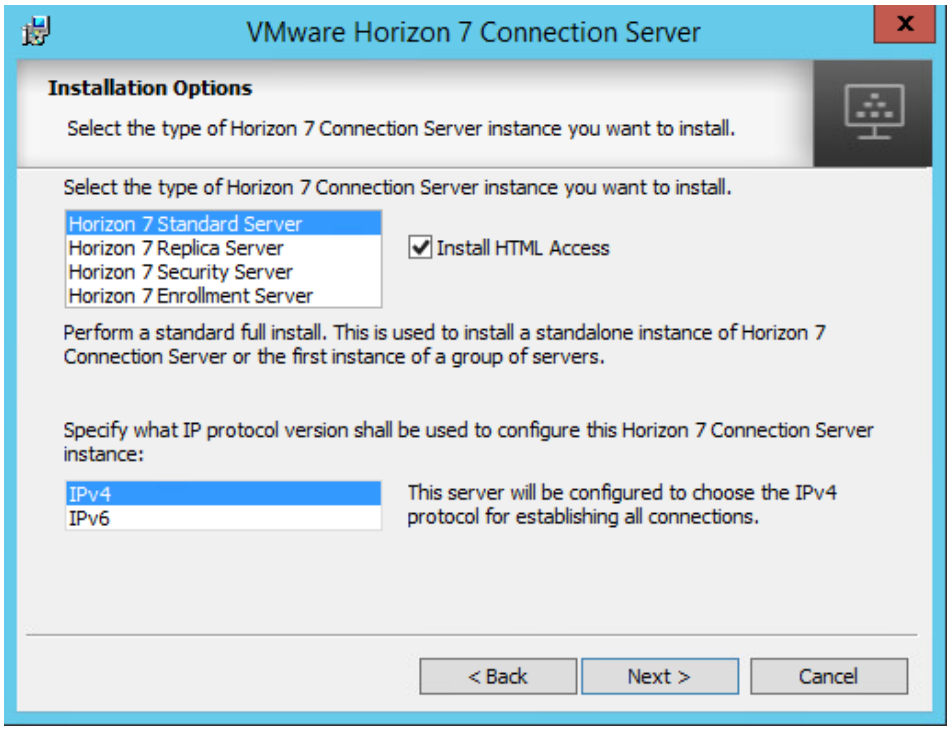


Check **I accept the terms of the license agreement** radio button to accept the agreement. Select **Next** to continue.



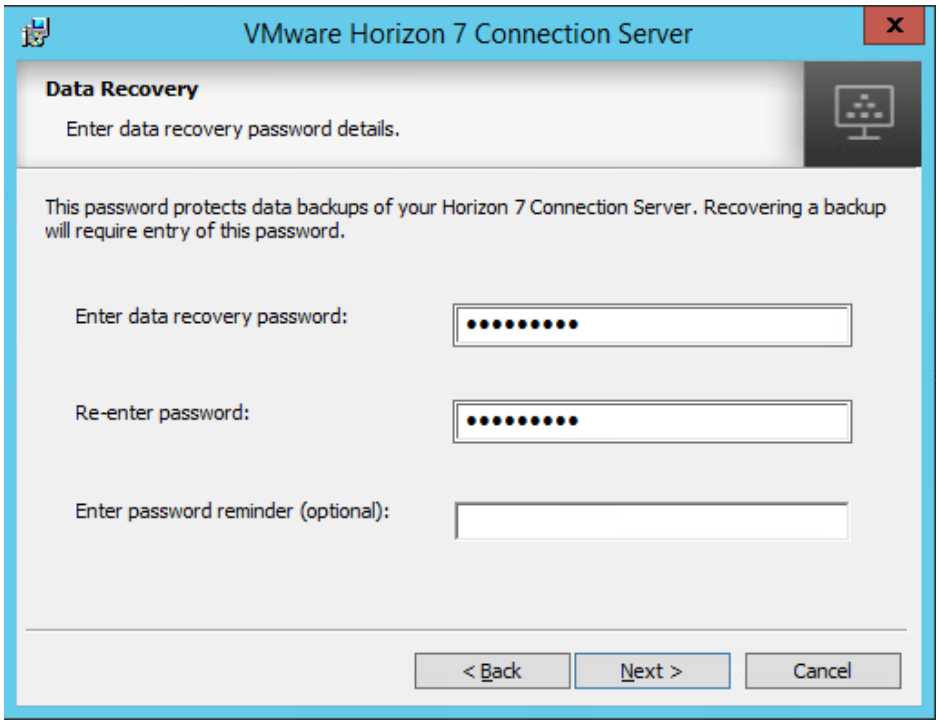
The Destination Folder window allows you to choose an install location and click **Next**.

Select **View Standard Server** with the **Install HTML Access** checkbox checked and **IPv4** selected:



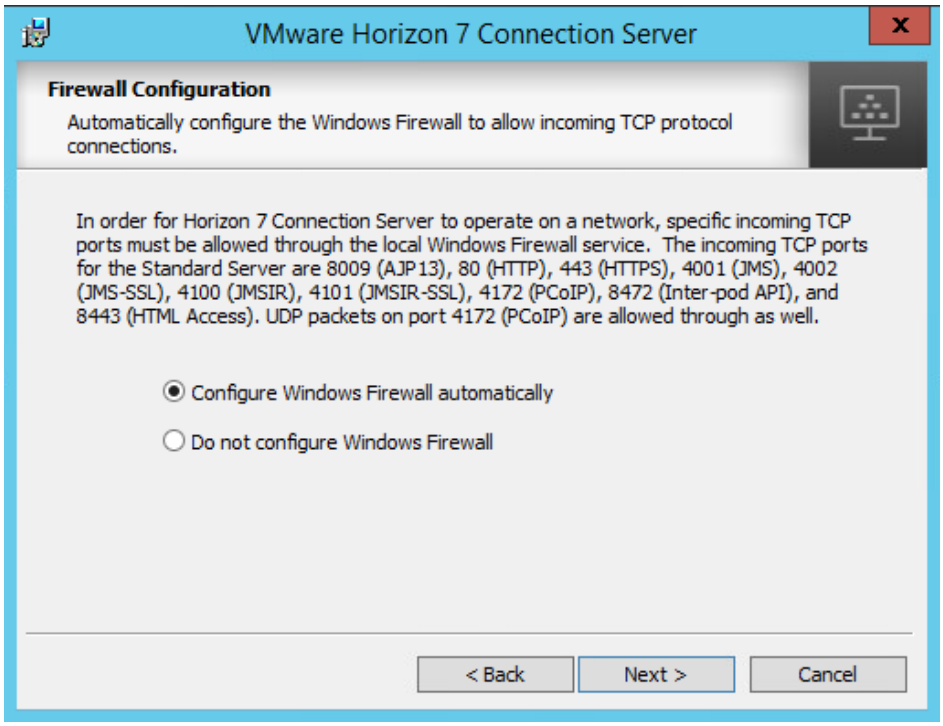
Select **Next** to continue.

The ***Data Recovery*** window displays.



Enter a password in the **Enter data recovery password** and **Re-enter password** fields,

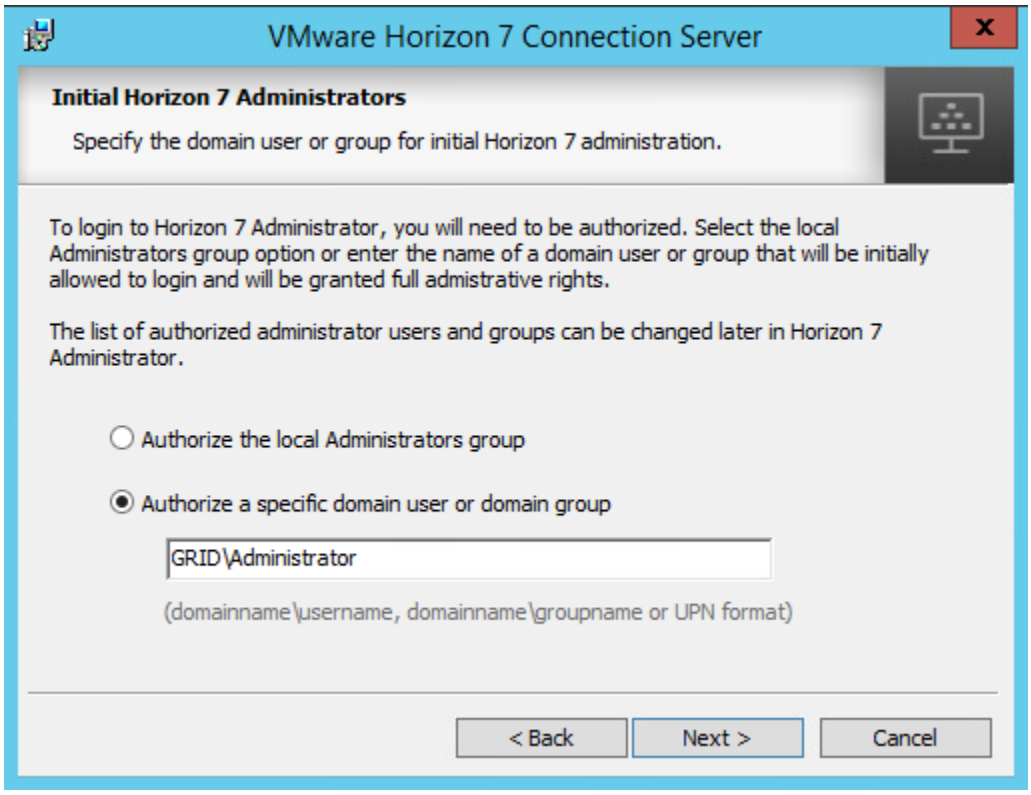
Select **Next** to continue.
The Firewall Configuration window displays.



CAUTION: TO PREVENT UNAUTHORIZED ACCESS, YOUR SELECTED ROOT PASSWORD SHOULD CONTAIN AT LEAST EIGHT (8) CHARACTERS AND CON- SIST OF A MIX OF LOWERCASE AND CAPITAL LETTERS, DIGITS, AND SPECIAL CHARACTERS.

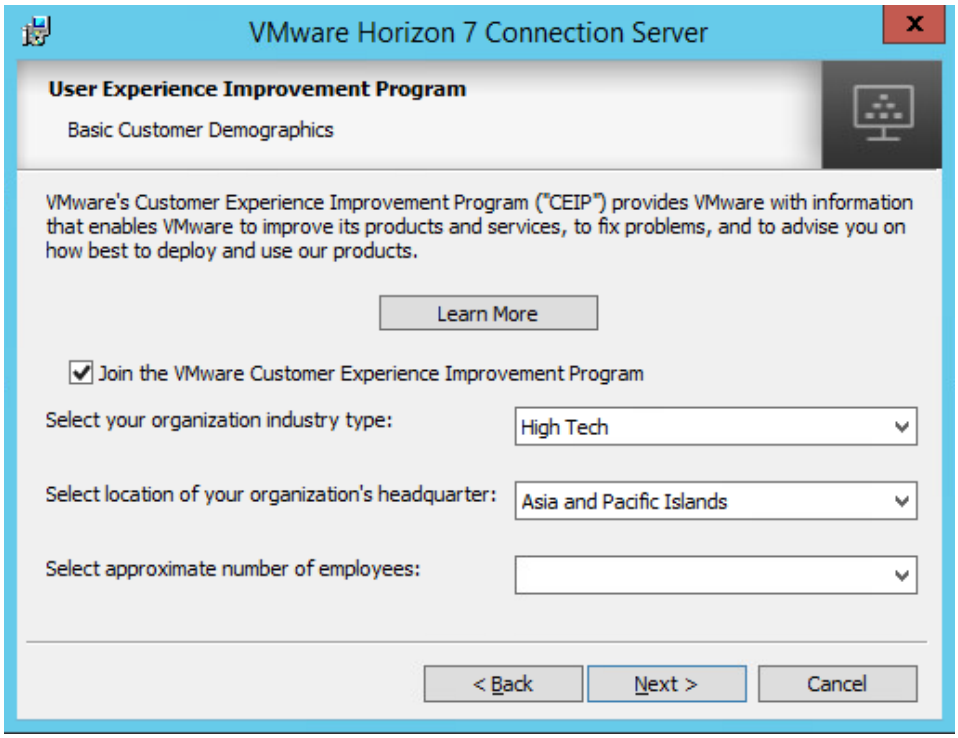
Check the **Configure Windows Firewall automatically** checkbox

Select **Next** to continue.
The ***Initial View Administrators*** window displays.



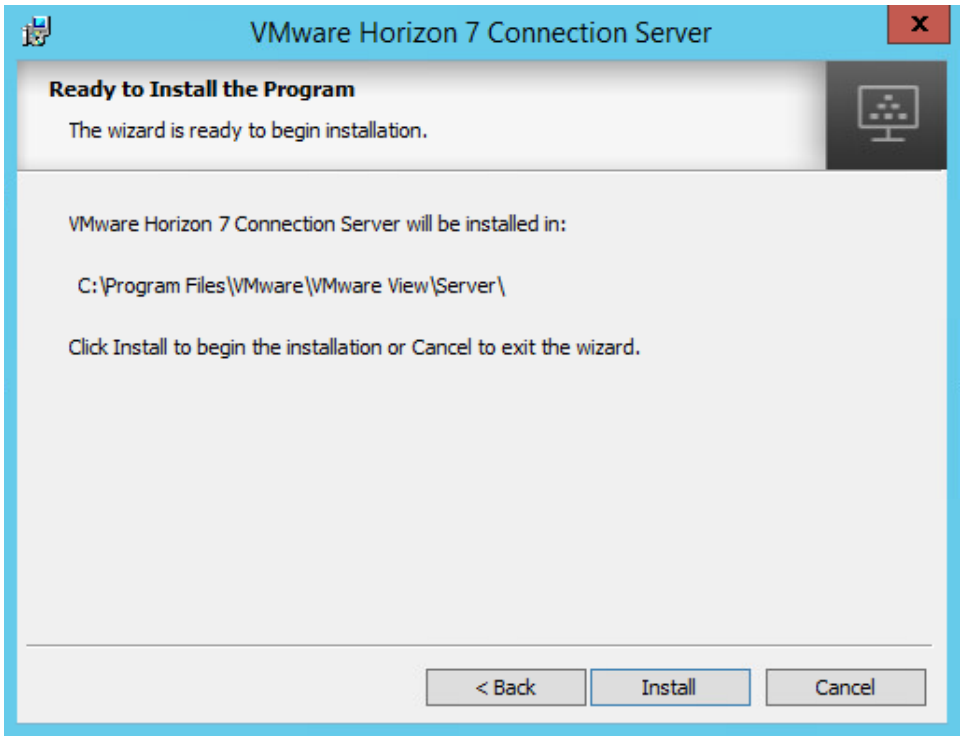
Select either the local Administrators group or a specific domain user or group, as appropriate.

Select **Next** to continue.
The ***User Experience Improvement Program*** window displays



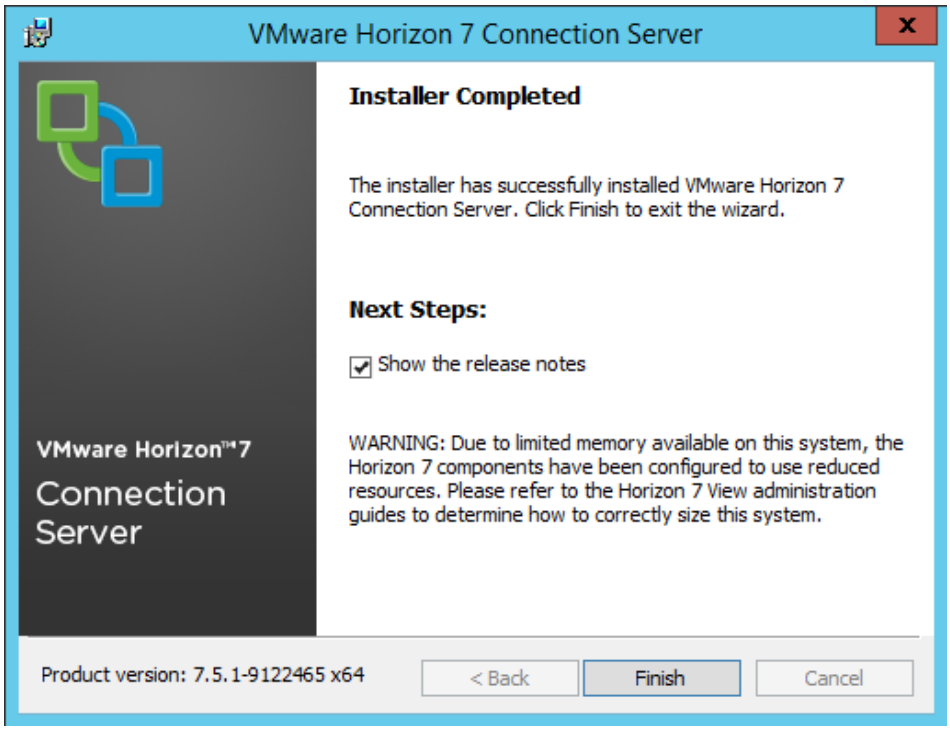
Choose your option for the **Participate anonymously in the user experience improvement program** checkbox.

Select **Next** to continue.
The ***Ready to Install the Program*** window displays.

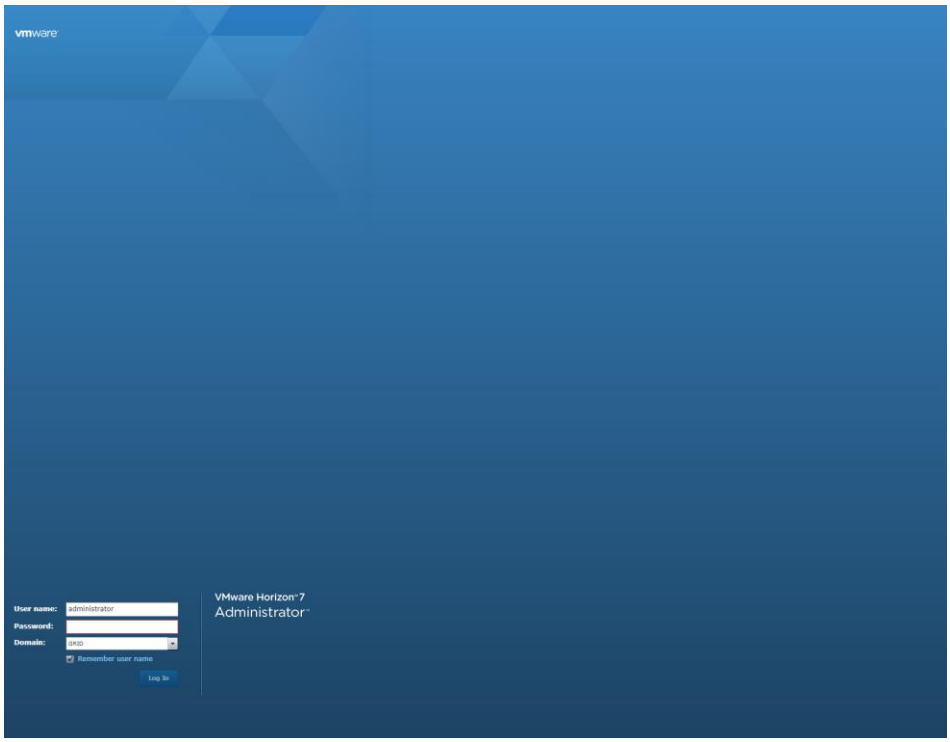


Select **Install to complete the installation**.

Select **Finish** when the ***Installer Completed*** window display.



Browse to the IP address or hostname of the server you installed on using **https://<IP address>/admin**:

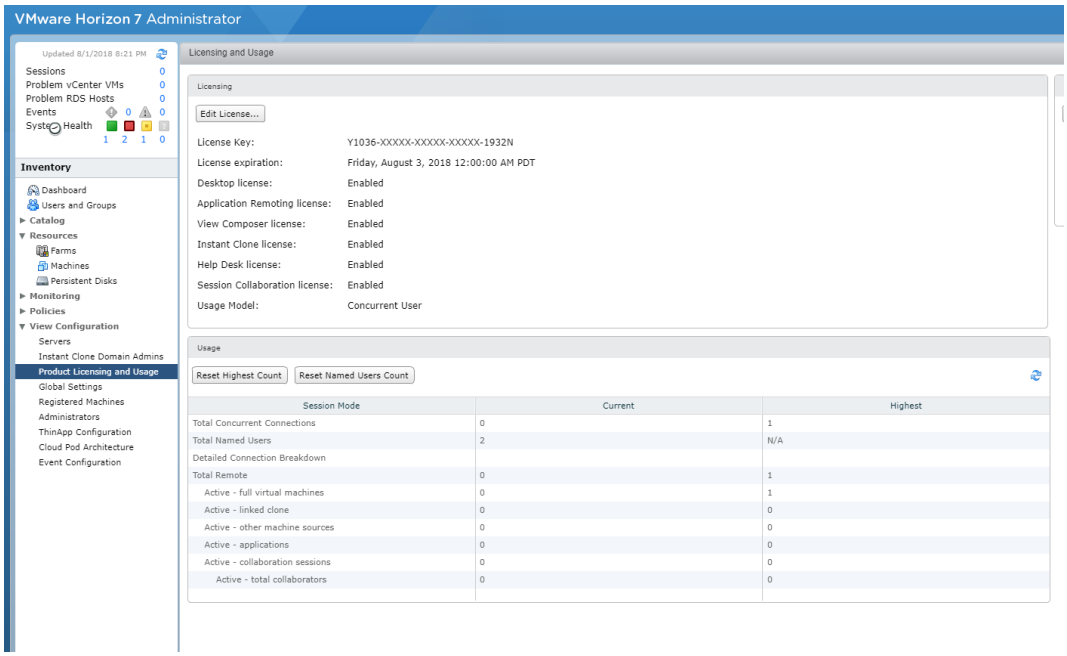


Registering the View License

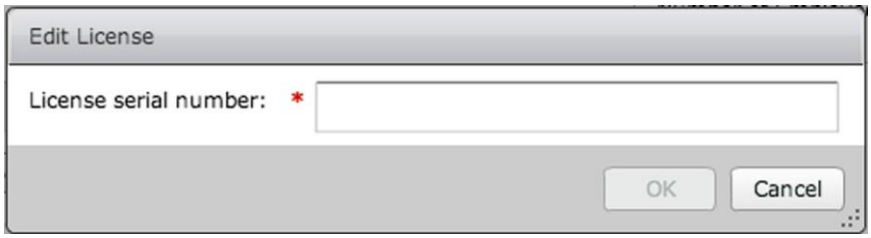
Use the following procedure to register the View license:

3. Connect to the View web console.
   The ***View Administrator*** page displays.

Select **View Configuration→Product Licensing and Usage** in the left pane.
The ***Licensing and Usage*** window displays.



Select **Edit License**.
The ***Edit License*** popup displays.



Enter your license key in the **License serial number** field

Click **OK** to decode and apply the license.

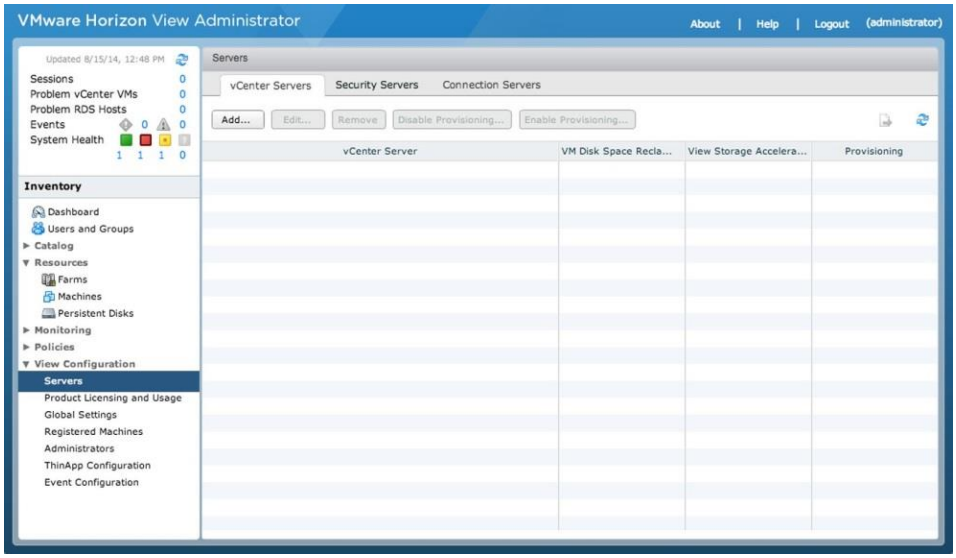
You must obtain the license from the VMware Beta site where you downloaded the software.

Verify that the license is enabled by looking for the **Enabled** messages in the **Licensing** area of the ***Licensing and Usage*** window.

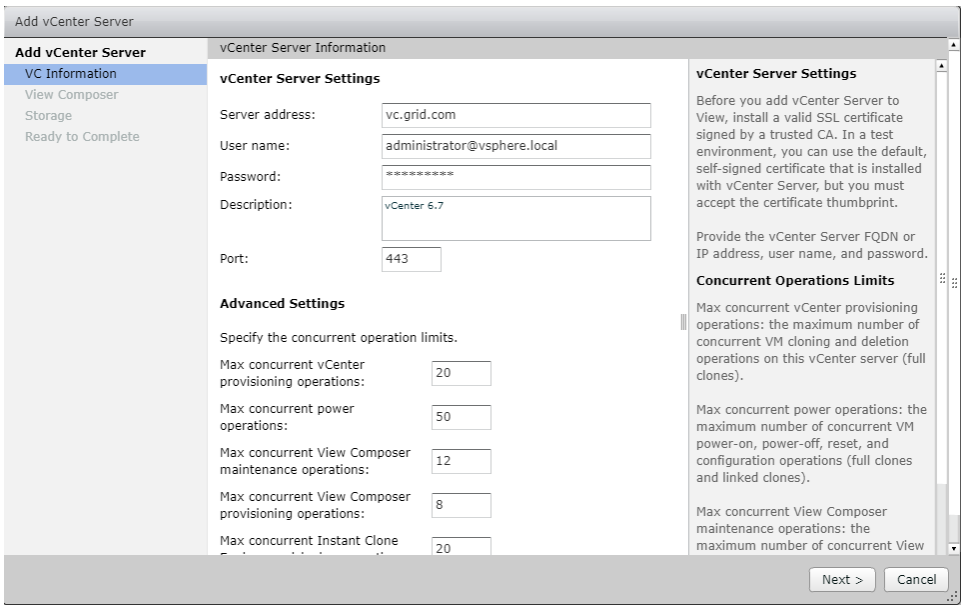
5.2 REGISTERING VCENTER SERVER

Use the following procedure to register a vCenter server:

4. Select **Horizon Configuration→Servers** in the left pane of the ***Horizon Administrator*** page to display the ***Servers*** window.



Select **Add...** to register a new vCenter server.
The ***Add vCenter Server*** popup displays with the **VC Information** tab selected.



Enter the vCenter server address and vCenter (not local desktop) user credentials in the appropriate fields.

Select **Next** to display the ***View Composer Settings*** tab.

Select **Do not use View Composer** and

Select **Next** to display the ***Storage*** tab.

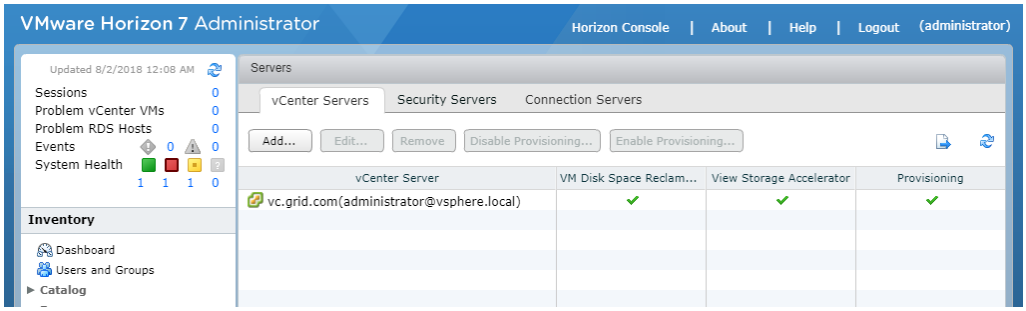
Alternatively, if you wish to use View Composer and have already configured your environment to support Horizon Composer, then select **View Composer** co-installed with vCenter Server and click **Next**. Continue performing the Horizon Composer setup.

Accept the default storage settings.

Select **Next** to display the ***Ready to Complete*** window.

Click **Finish**.

The vCenter server is now added and will display in the View configuration.



5.3 INSTALLING VIEW COMPOSER SERVER

The server on which you are installing View must meet the requirements listed in “General Prerequisites” on page 20. You will also need access to a SQL server. For the purposes of our POC/trial, we installed SQL Express 2012 on a separate Windows Server 2012R2 guest on our

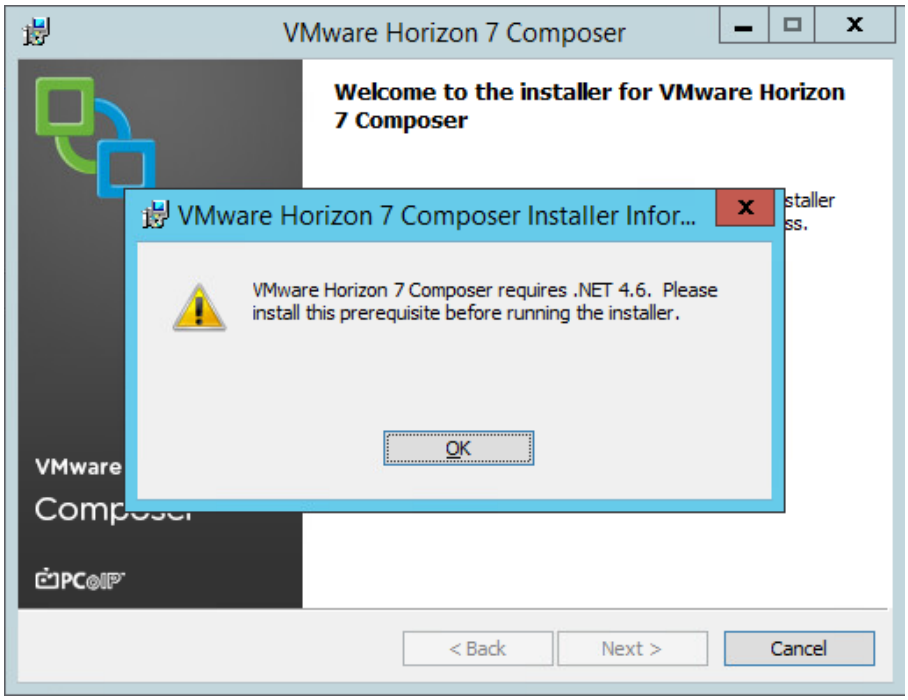
management cluster. As of the publication date of this manual the current version of the View Composer Server is:

VMware-viewcomposer-7.5.1-8971623.exe

The purpose of the Composer Server is to create Linked Clones, the requirements are:

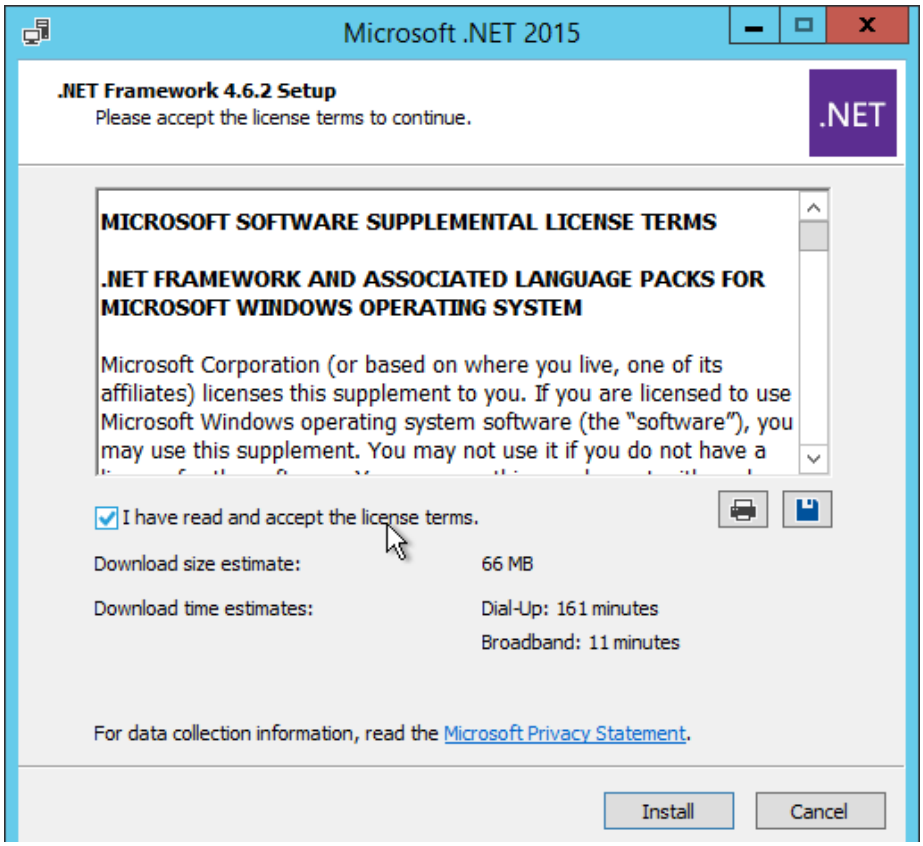
- 1 Microsoft Windows Server 2012R2 servers for Horizon View Composer
- 1 Microsoft Windows Server 2012R2 servers for SQL 2012

Use the following procedure to install View Composer 7.5:

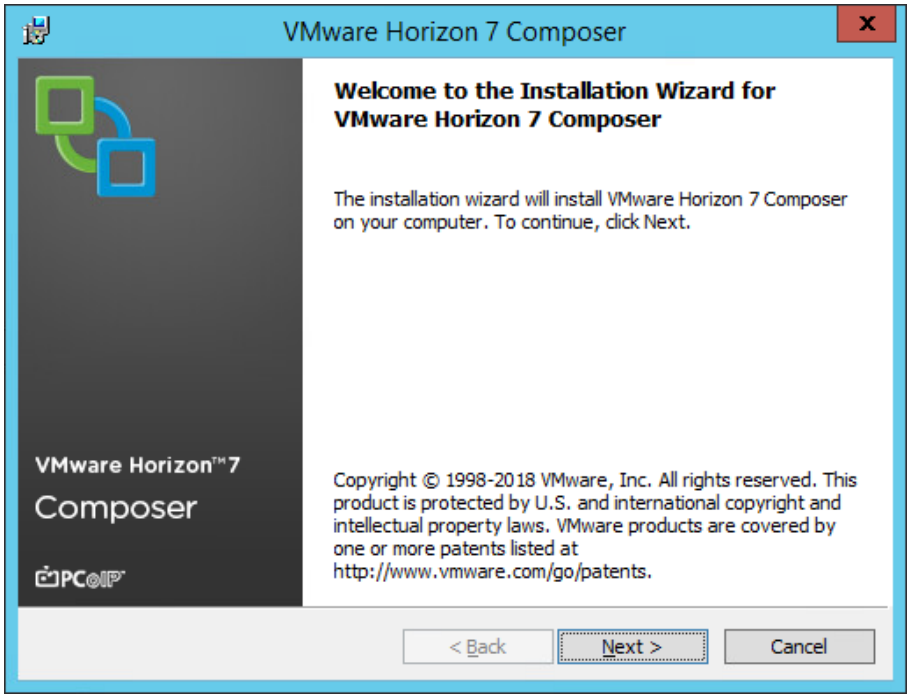


Horizon 7 Composer requires .NET 4.6 to run

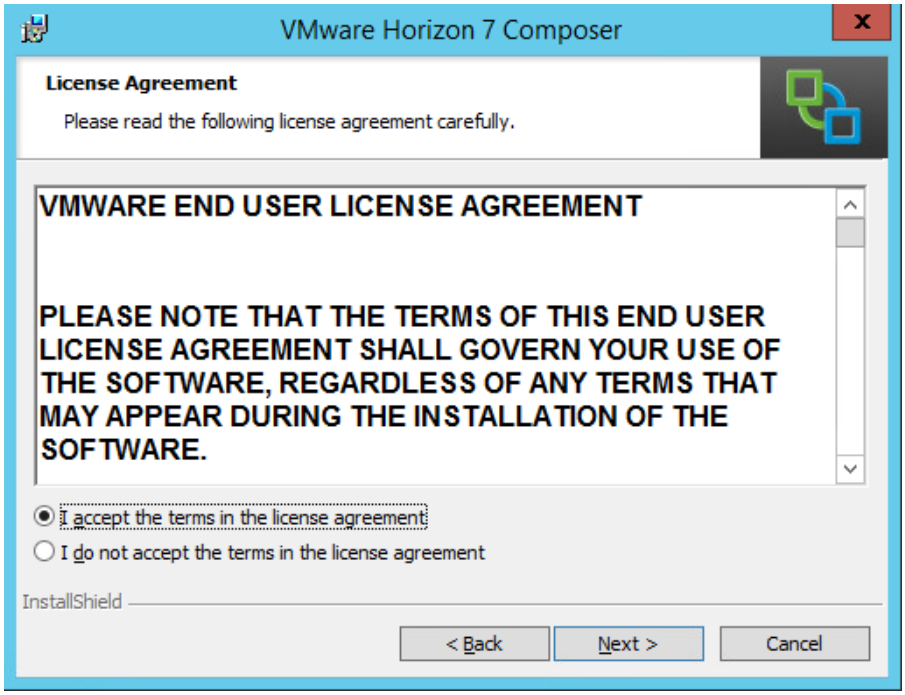
Check the box next to I have read and accept the license terms and click Install.



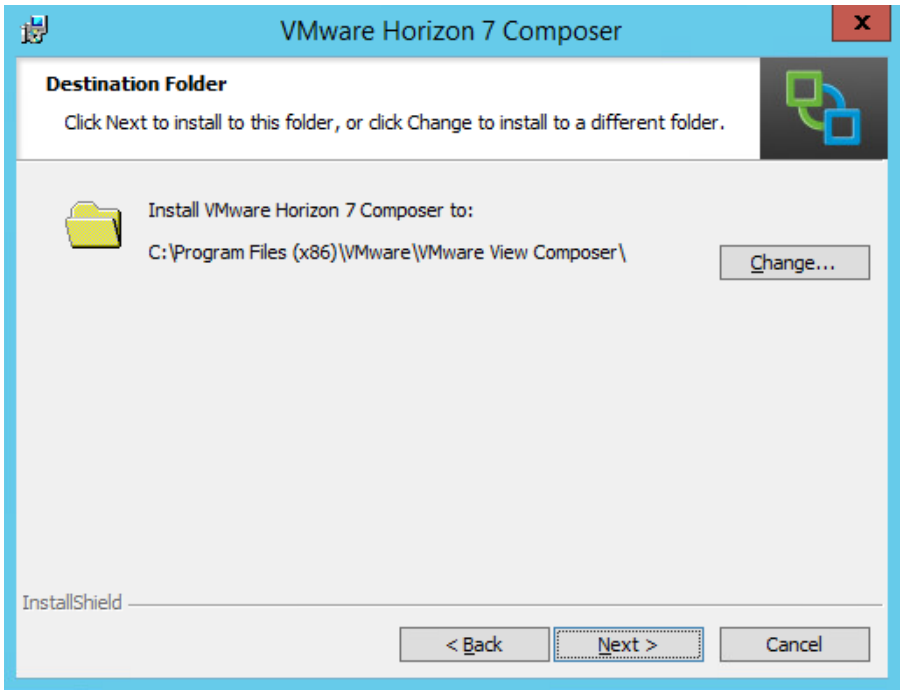
The VMware Horizon View Composer installer begins, click **Next**.



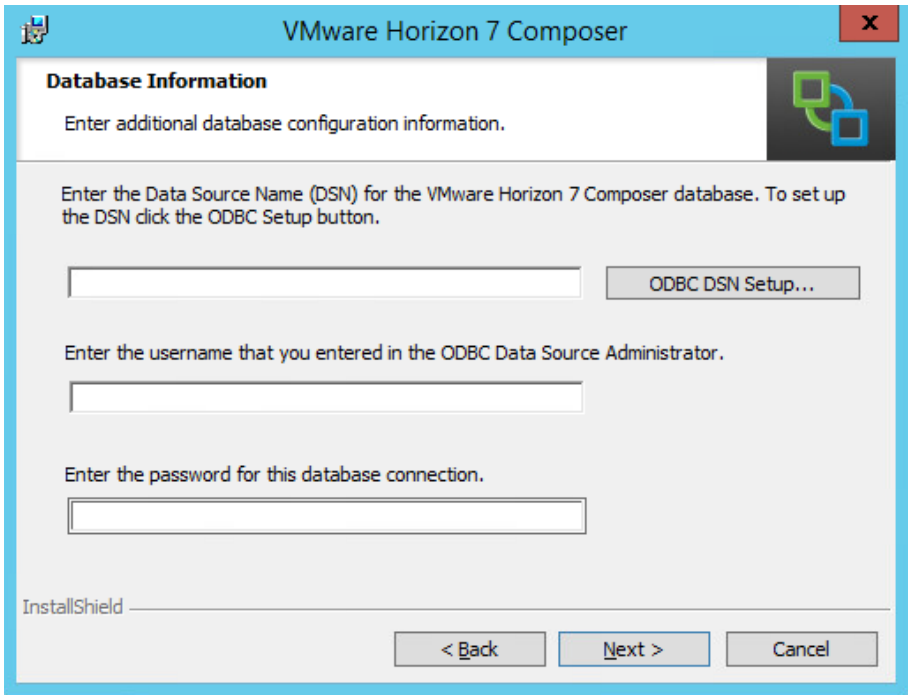
Check **I accept the terms of the license agreement** radio button to accept the agreement. Select **Next** to continue.



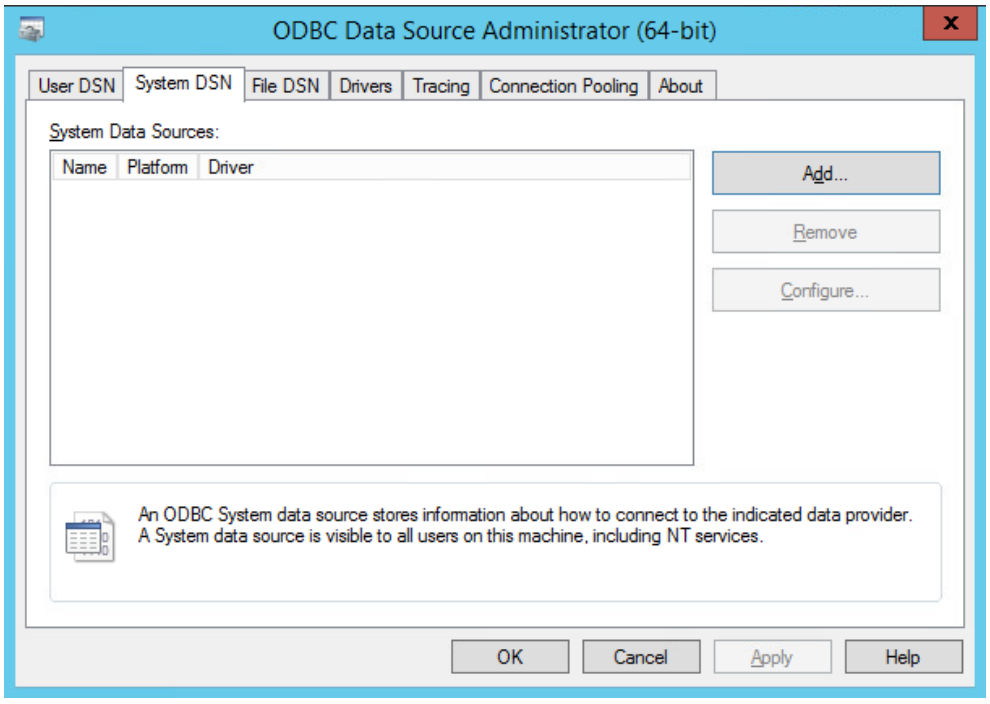
The Destination Folder window displays, choose an install location and click **Next**.



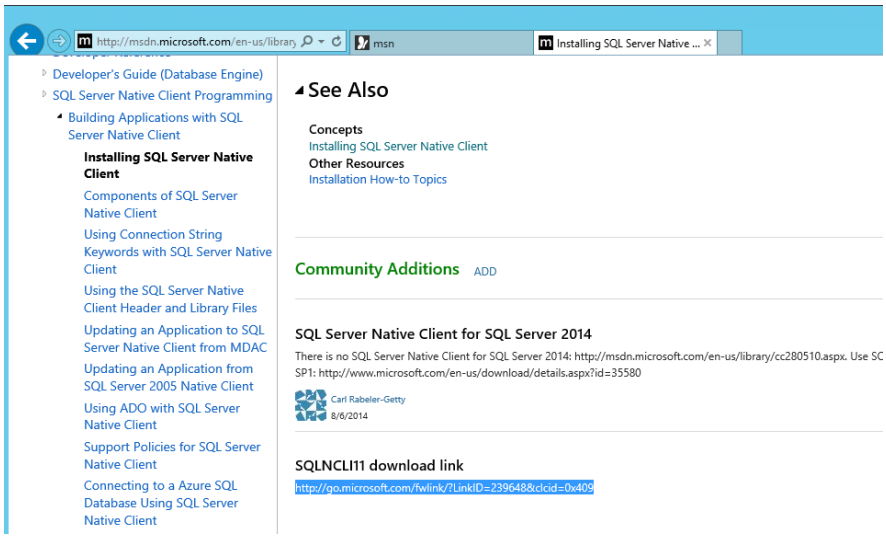
Next set up a connection to a SQL server. Click the **ODBC DSN Setup…** button.



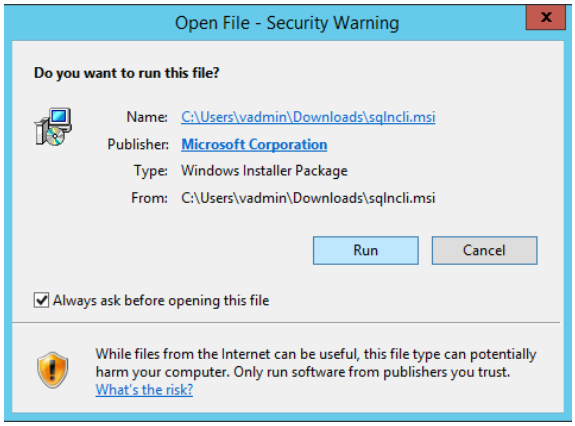
Click the **Add…** button.



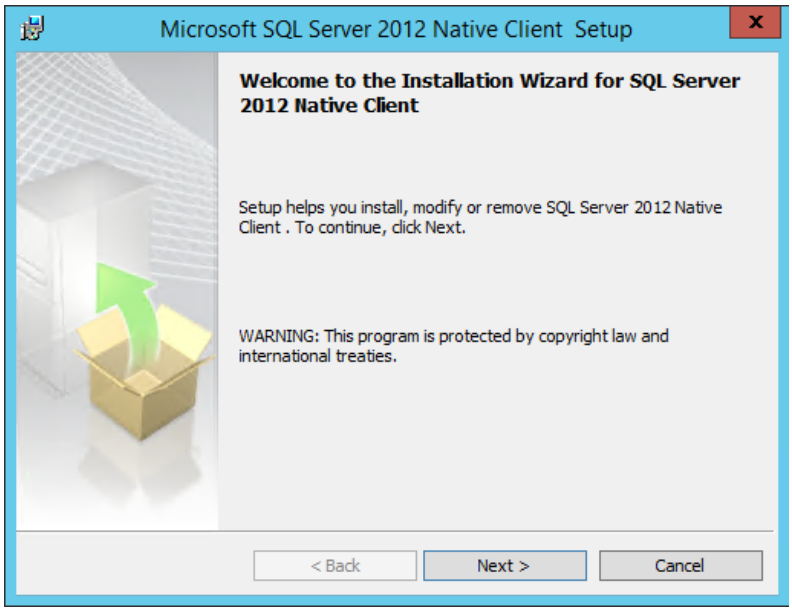
You may need to download and install the SQL Server Native Client.



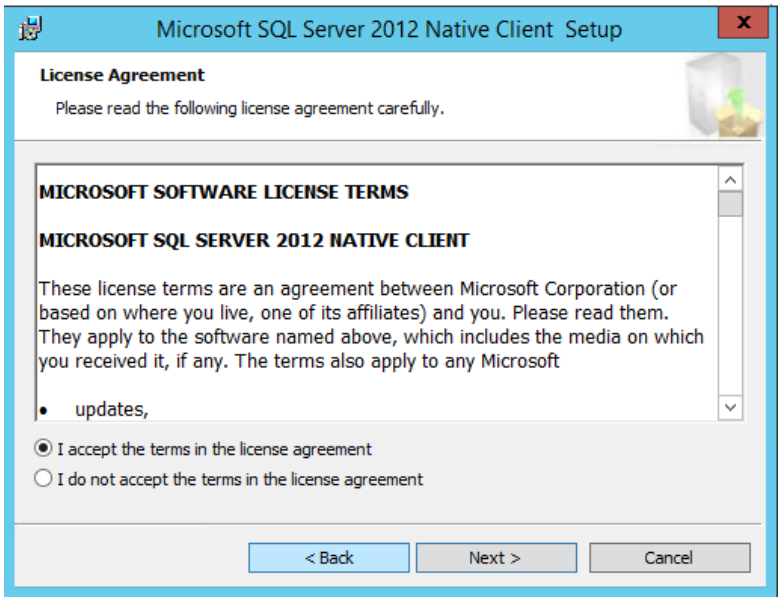
Accept any security warnings after you down and begin install.



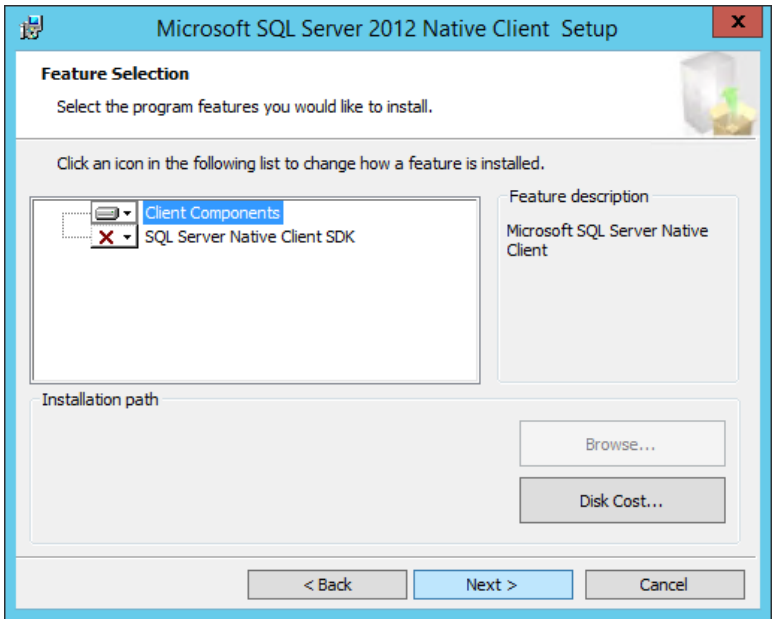
Follow all the defaults to install.



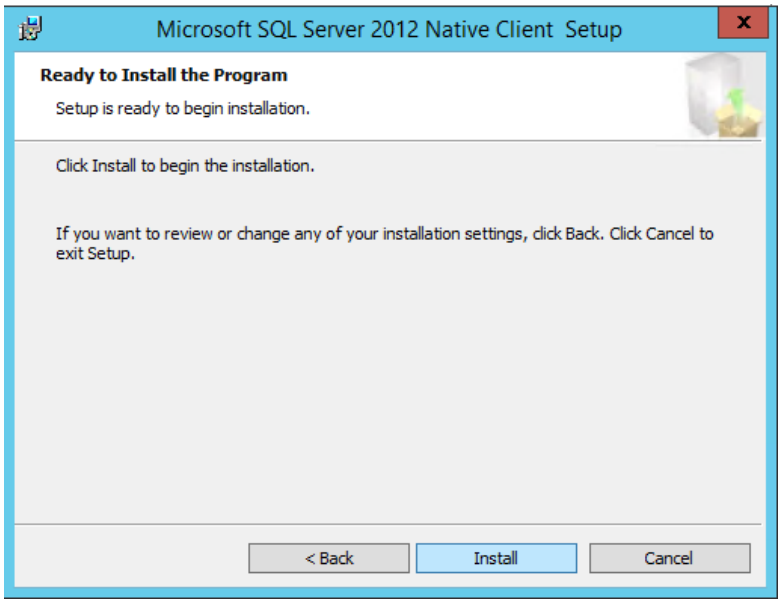
Accept the license agreement and click **Next**.



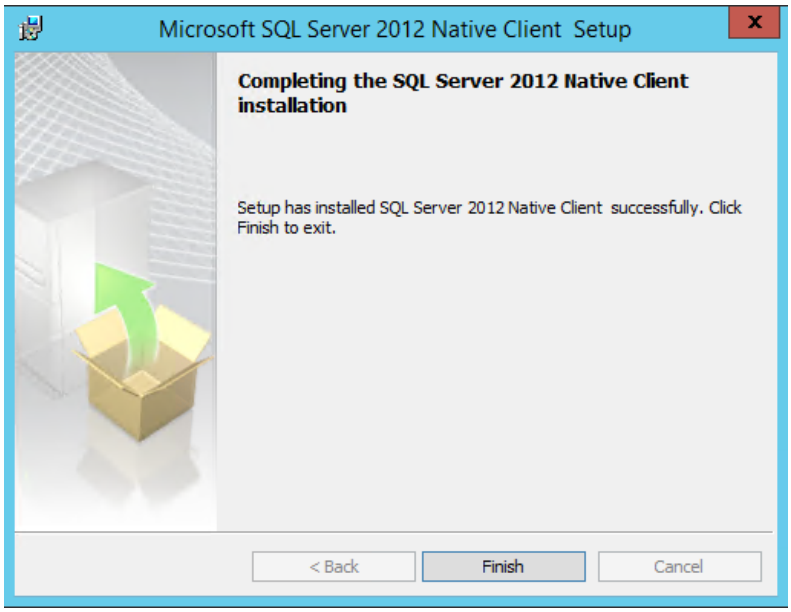
Accept defaults and click **Next**:



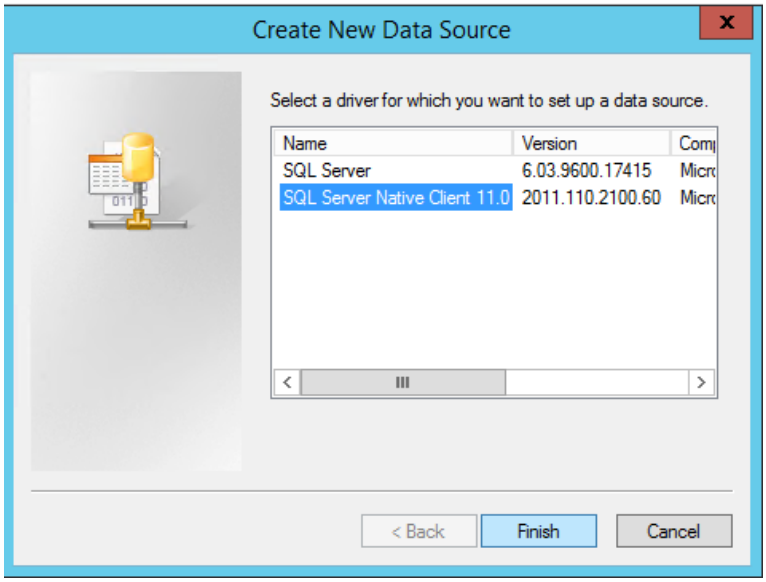
Click **Install**:



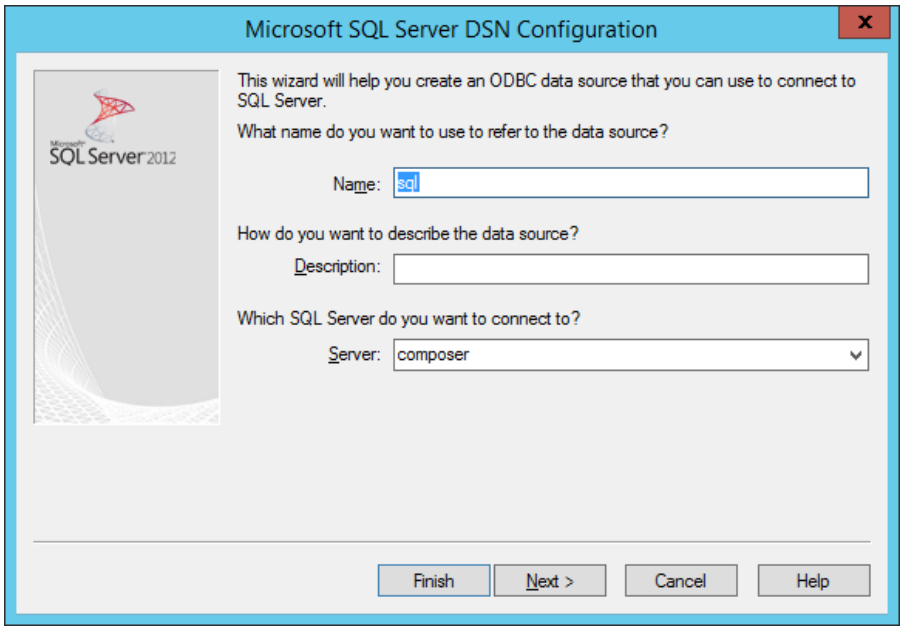
Once install is complete, click **Finish**:



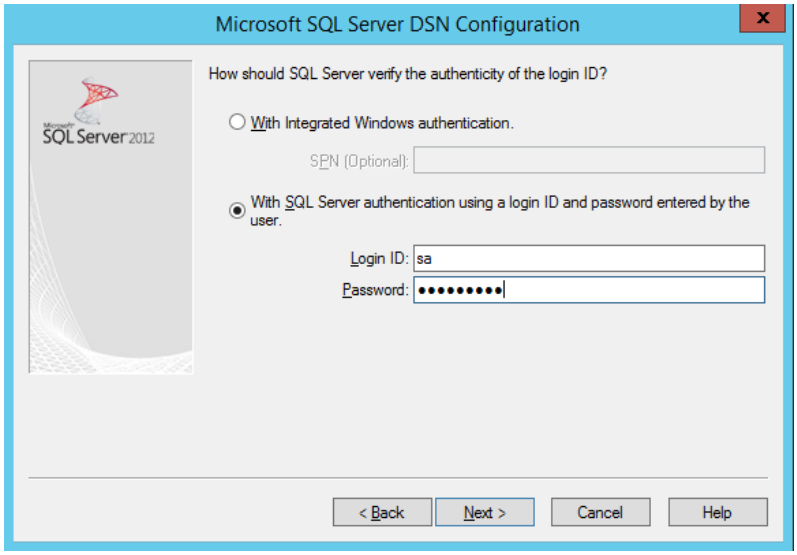
Return to (or restart) the **ODBC DSN Setup** for View Composer. Select the SQL Native Client 11 in the **Create New Data Source** wizard and click **Finish**.



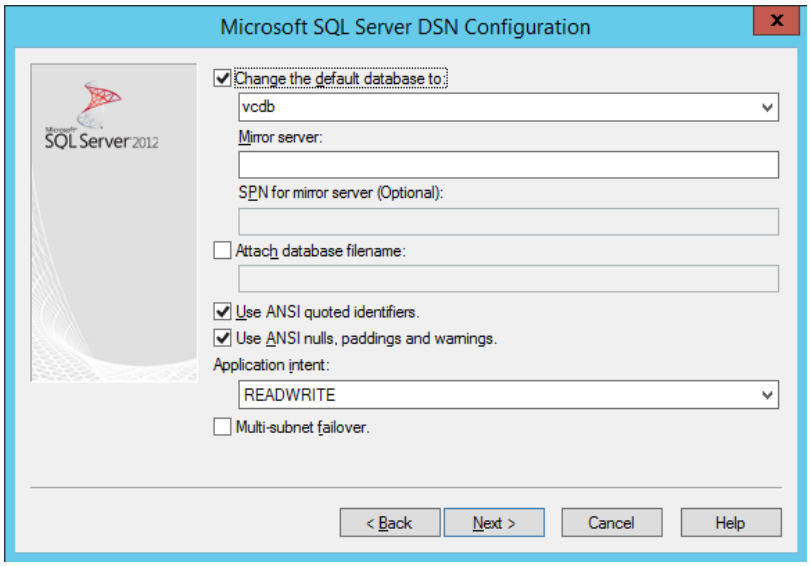
Name your New Data Source and locate the correct SQL server you want to connect. Click **Next** to continue



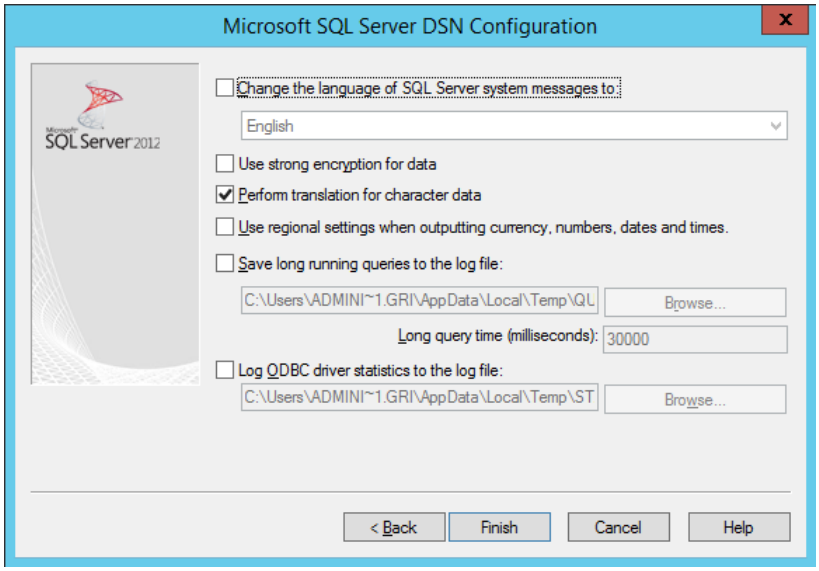
Enter the SQL account credentials (no Windows accounts) and click **Next**. For remote SQL databases, only SQL accounts will work. The SQL account must be **db_owner** of the database.



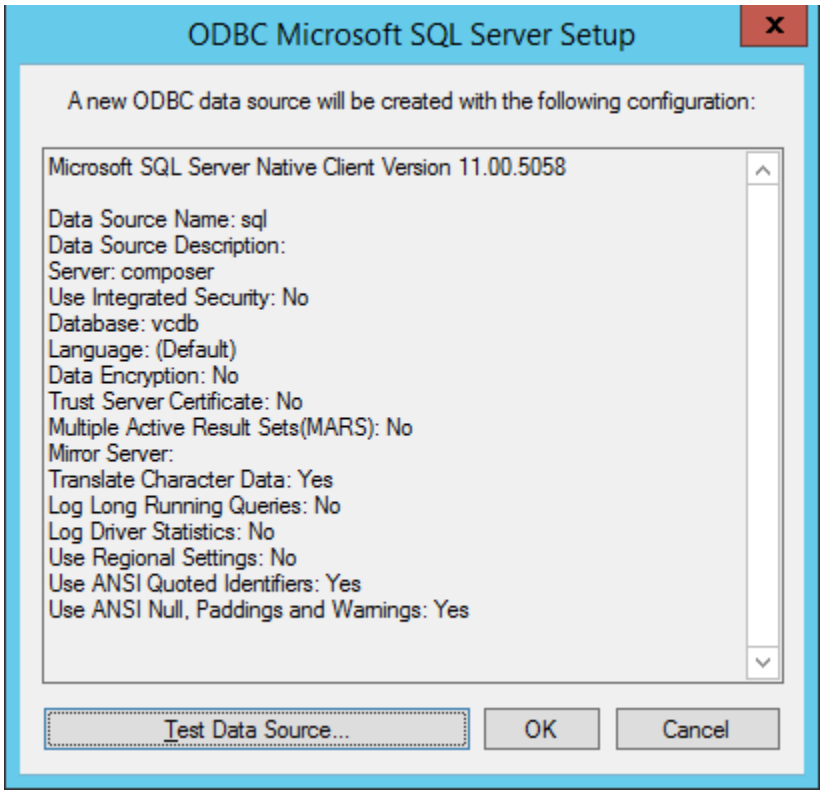
Select **Change the default name to:** and name your database for the View Composer. Click **Next** to continue.



Click **Finish** to continue.

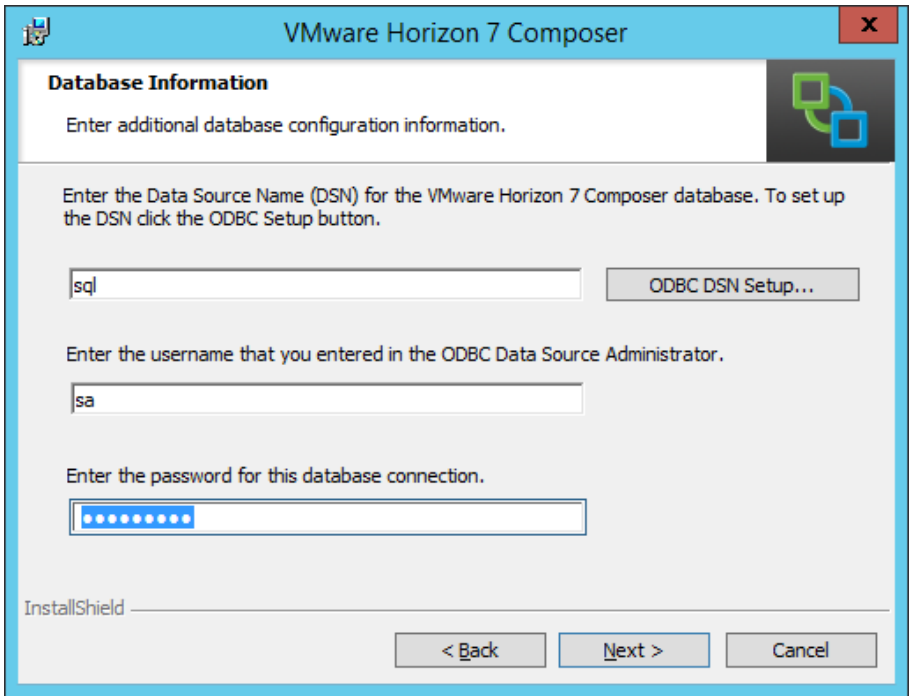


Review your configuration and click on **Test Data Source…**

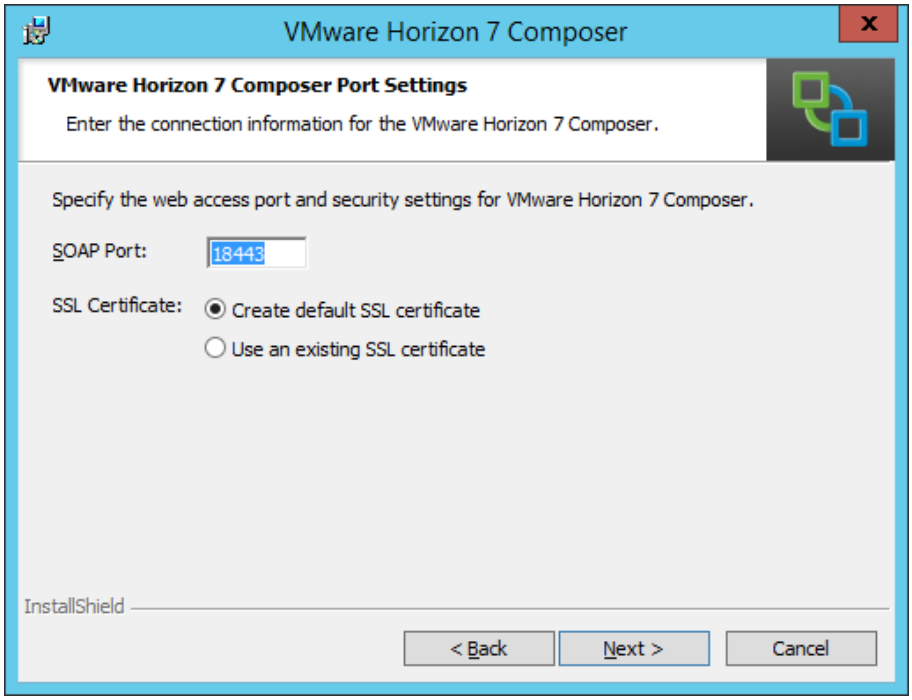


If all is good, you will get a successful configuration. If not, go back and review your settings.

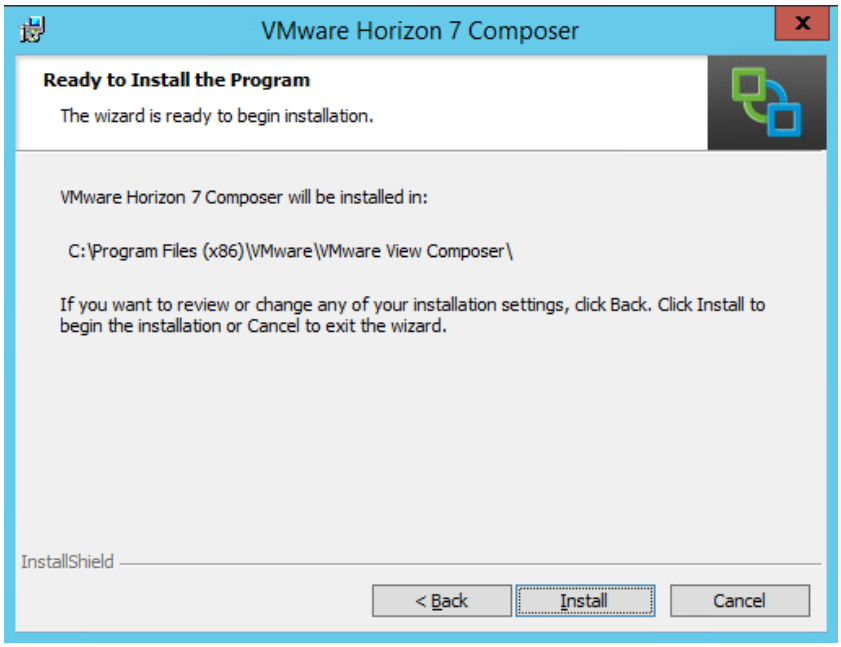
Return to the View Composer installation and enter the database name, SQL Admin account name and Password to complete installation. Select Next to continue.



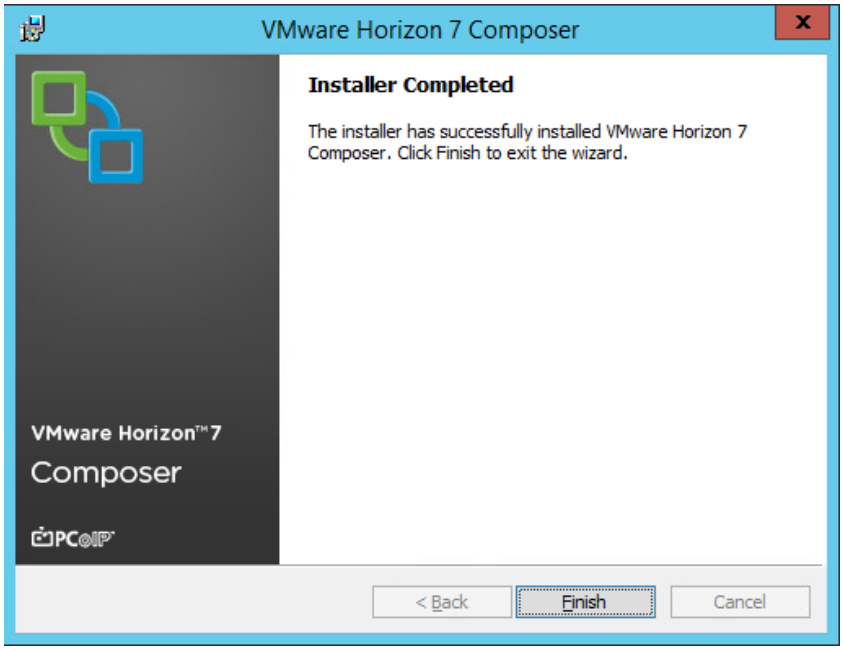
Select the web access port number. (Stay with default if possible.) Select Next to continue.



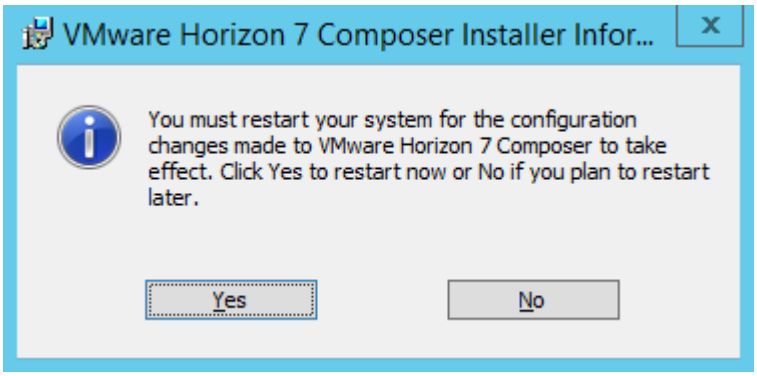
Select **Install** to continue.



The install completes, click **Finish**:



A full system reboot is required. Click **Yes** to complete the install and reboot the server.

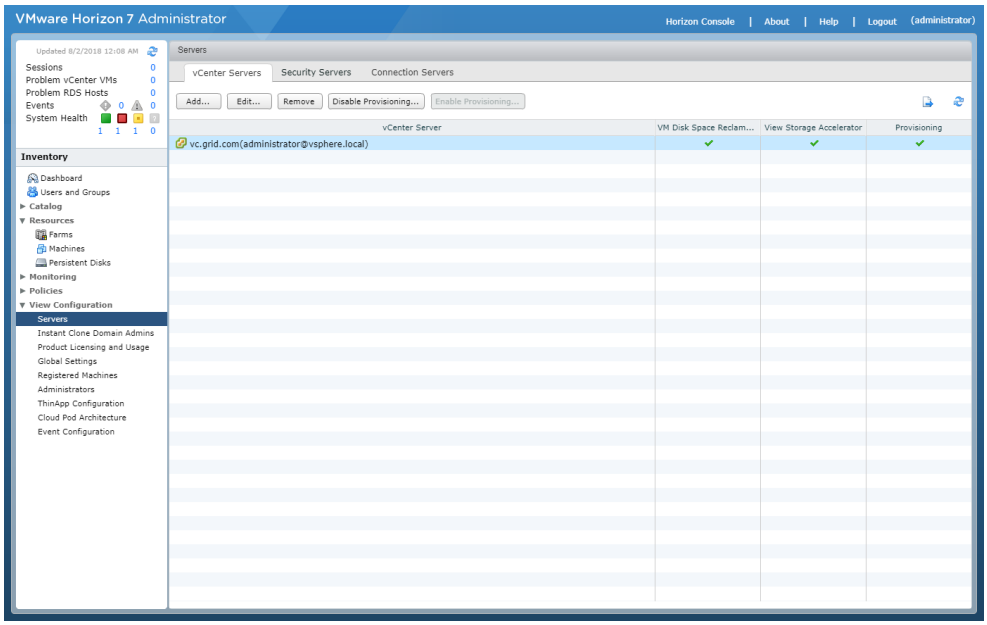


Once the server is back up, move on to register Composer to Horizon. (It may take a few minutes for all the services to start up, so be patient.)

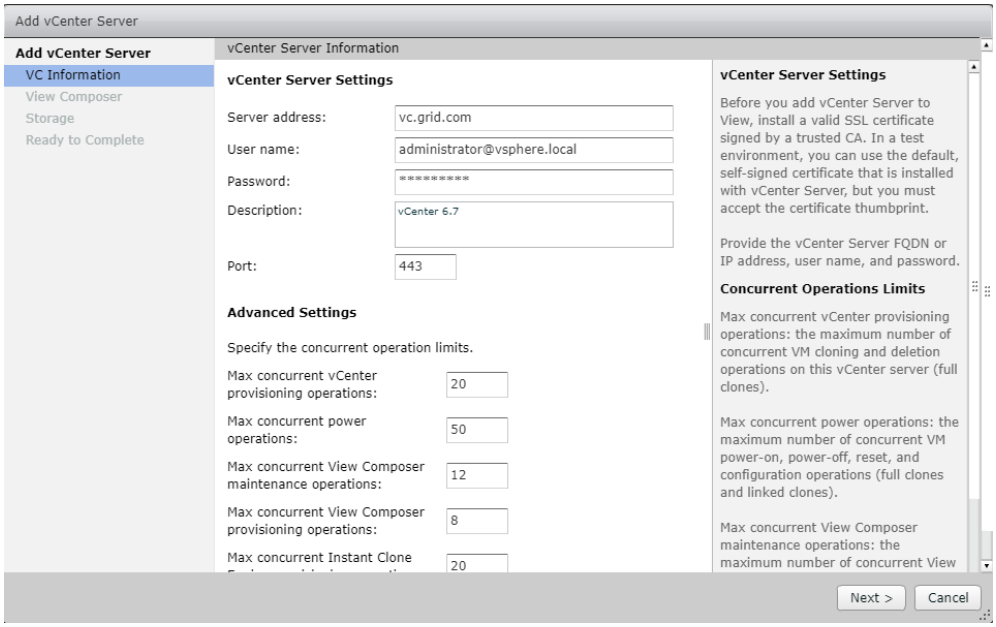
5.4 REGISTERING VIEW COMPOSER SERVER

1. Login to View Server to register the View Composer Server.

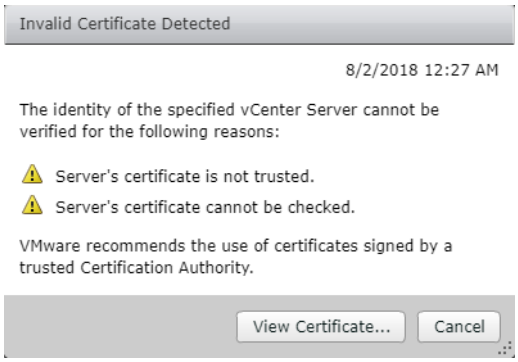
Select **Servers** under the **View Configuration** submenu in the left-hand window. Select **Edit** in the center window to add a View Composer to an existing vCenter Server or **Add** to add a new vCenter Server with a View Composer.



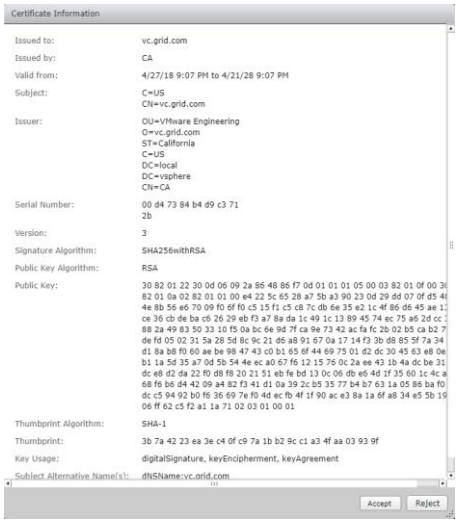
2. Enter the vCenter Server address, administrator account and password. Customize the configuration with the advanced settings if needed.



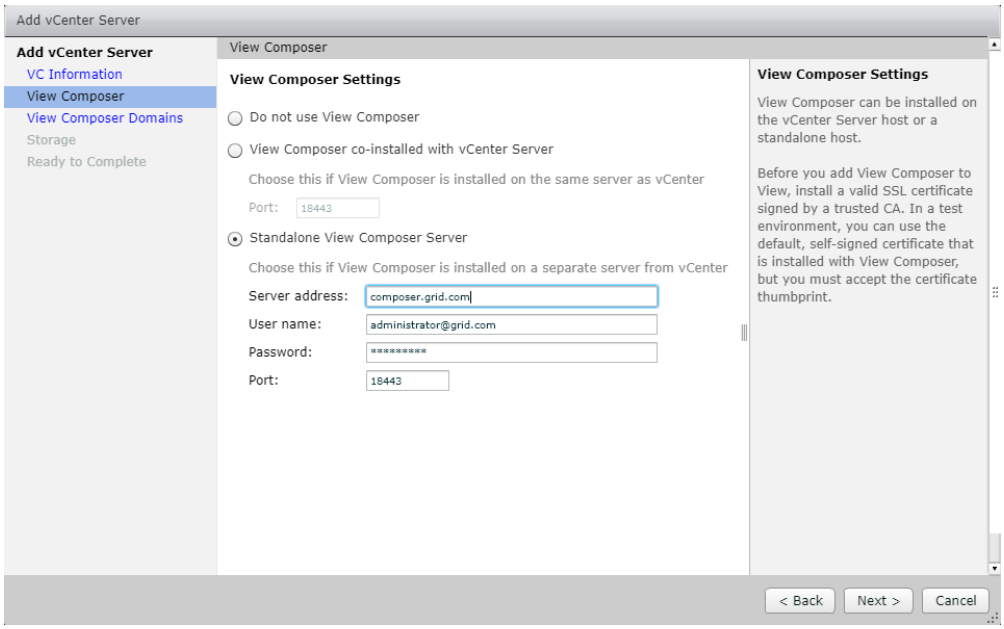
3. View the Invalid Certificate Detected by selecting **View Certificate**.



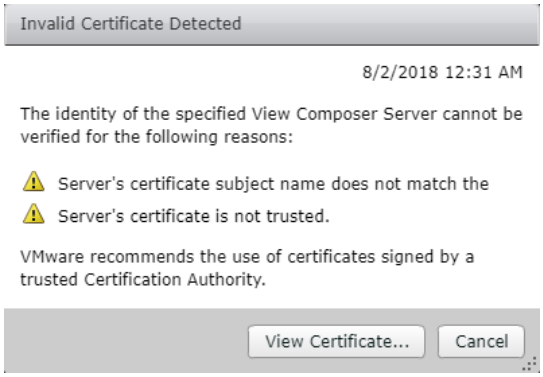
4. Select **Accept** to continue.



5. In this example, we used a standalone View Composer Server. Enter the server address, user account name and password. (If you co-installed with vCenter Server, select the View Composer co-installed option.)

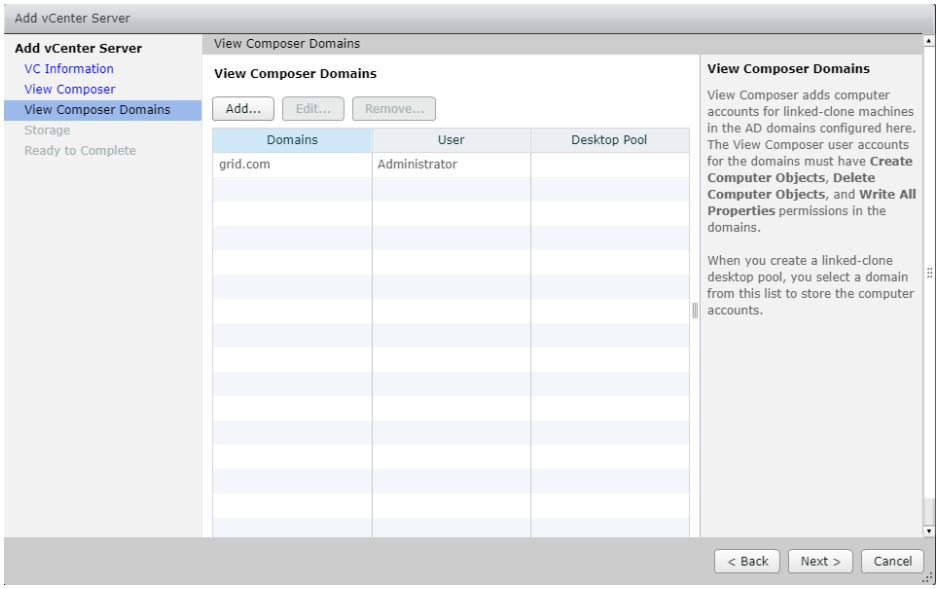


6. View the Invalid Certificate detected by selecting View Certificate.

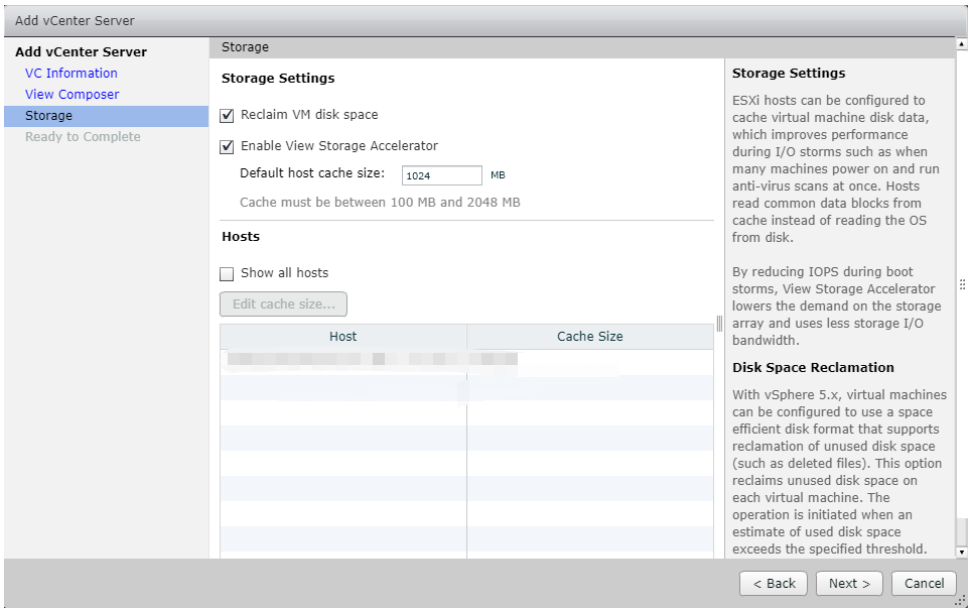


7. Accept the Certificate by selecting **Accept**.

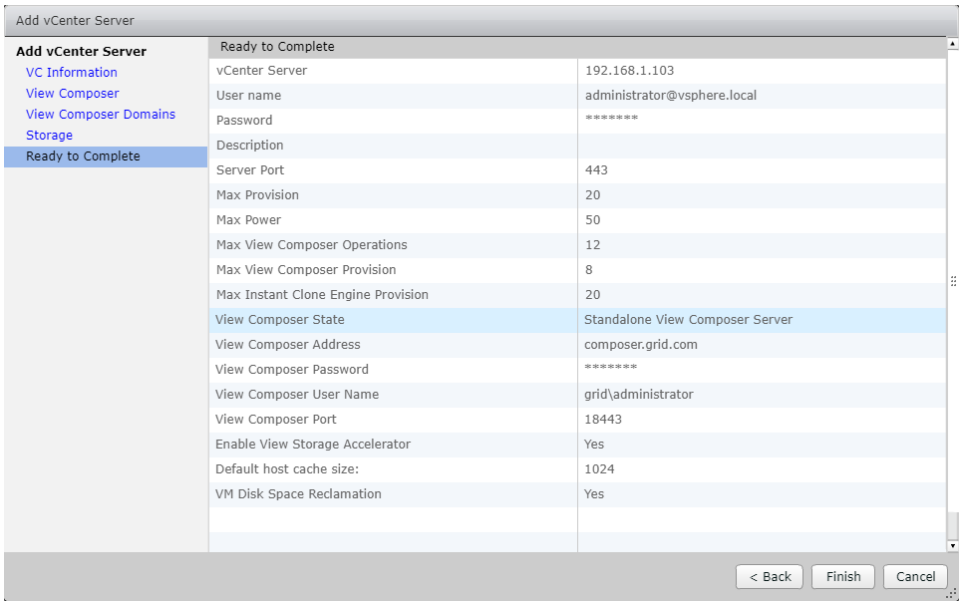
8. Select next to continue after viewing the View Composer Domains created.



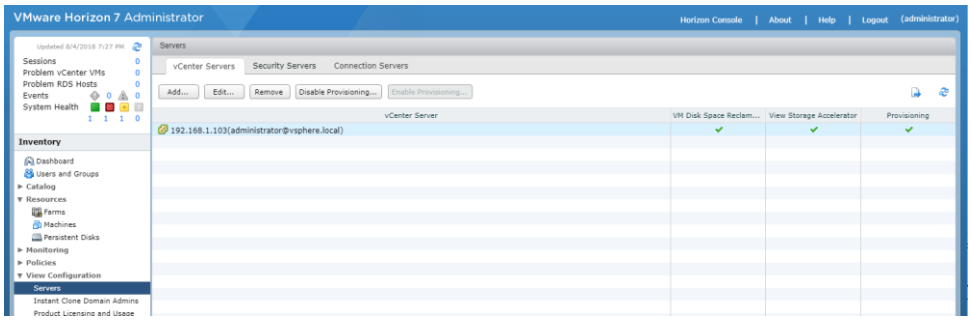
9. Select View Storage Settings. **Check Reclaim VM disk space**, **Enable View Storage Accelerator** and set the **Default host cache size**. You can customize the Cache Sizes of each ESXi host at this point if needed by selecting **Edit cache size.** When ready click **Next** to continue.



10. Review your configuration and select **Finish**.



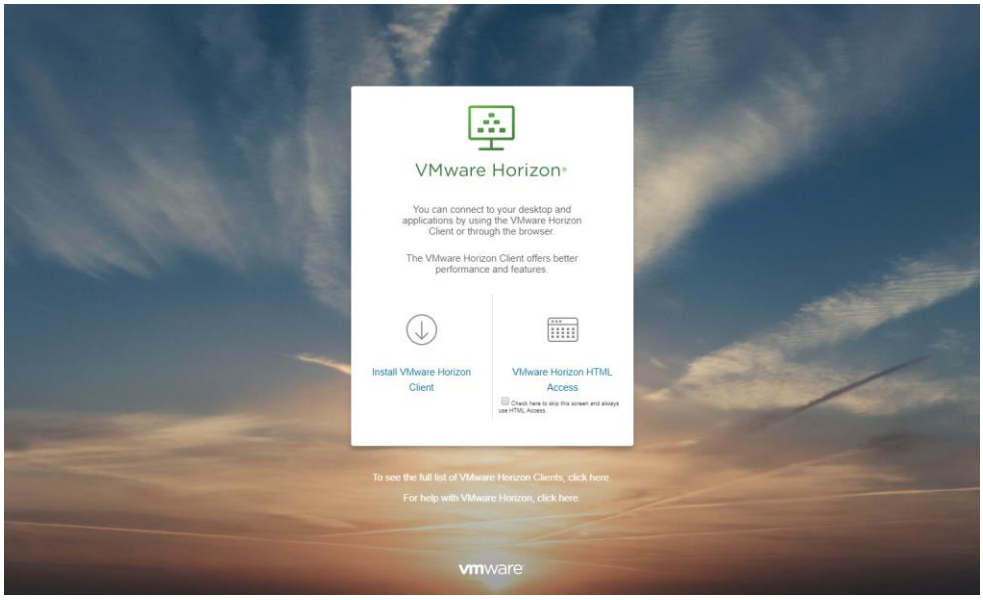
11. Verify your vCenter Server with View Composer is ready to go.



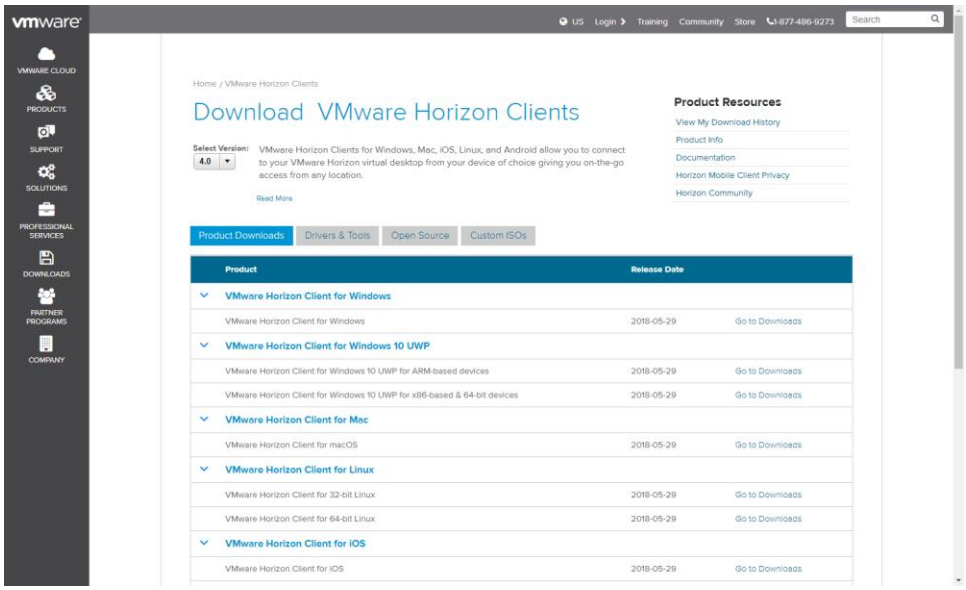
5.5 INSTALLING THE HORIZON CLIENT

Before connecting to a virtual desktop over a Blast/PCoIP connection, a VMware Horizon Client will need to be installed onto a desktop or device the virtual desktop will be accessed from.

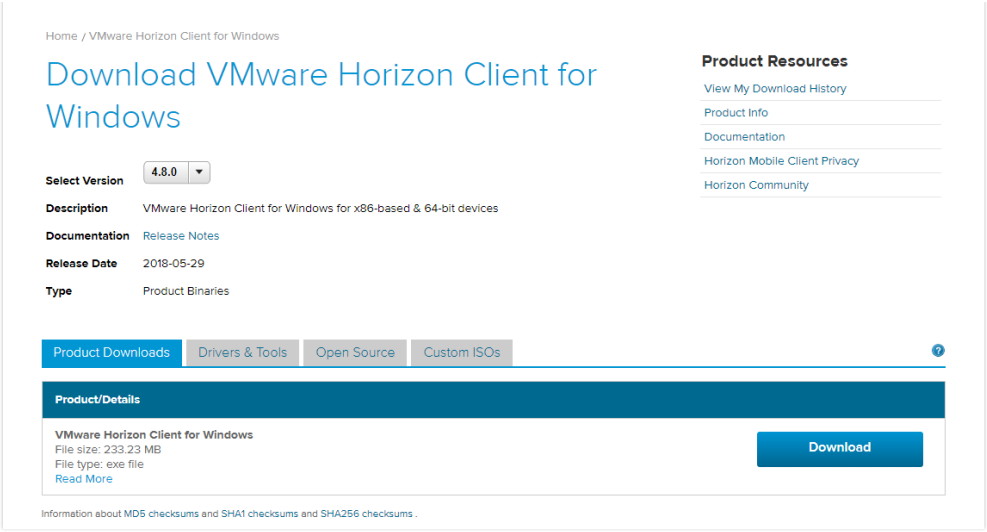
1. While logged into the physical device open an internet browser. Navigate to the Horizon View Connection Server URL installed earlier in this section. Click on Install VMware Horizon Client.



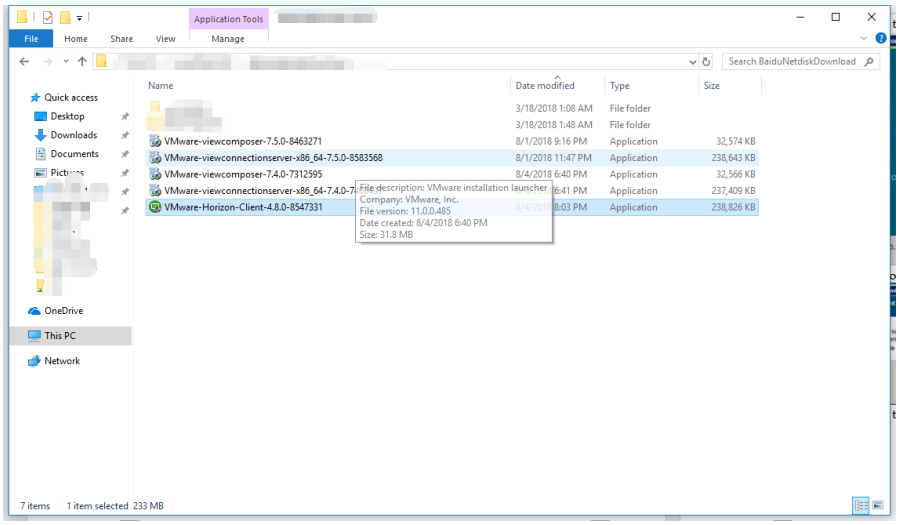
2. The **Download VMware Horizon Clients** page will open. Locate the download for the correct OS and click on **Go to Downloads** on the right-hand side.



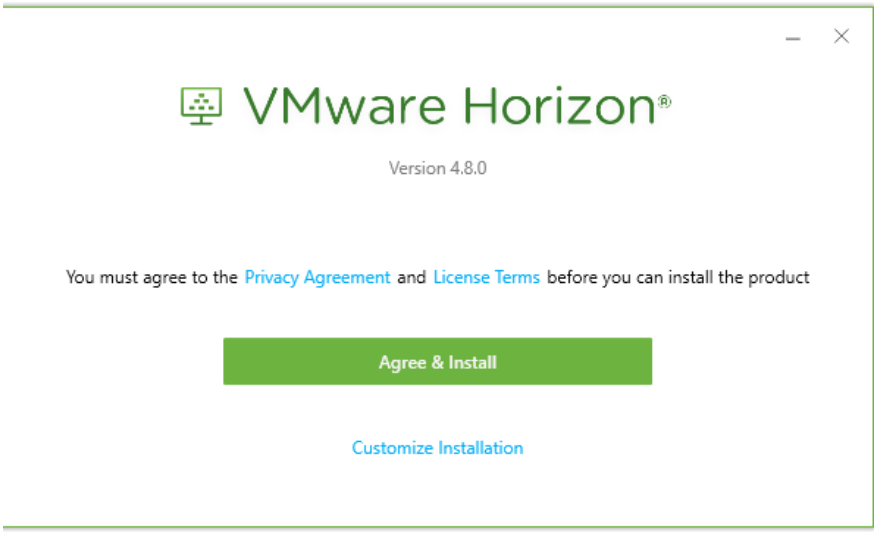
3. Click on **Download** to begin downloading.



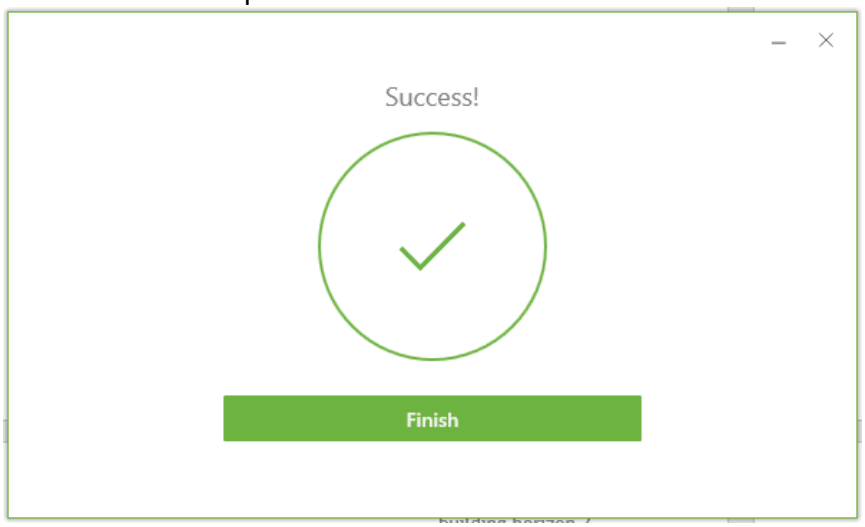
4. Locate the downloaded installer program and double click to begin installation.



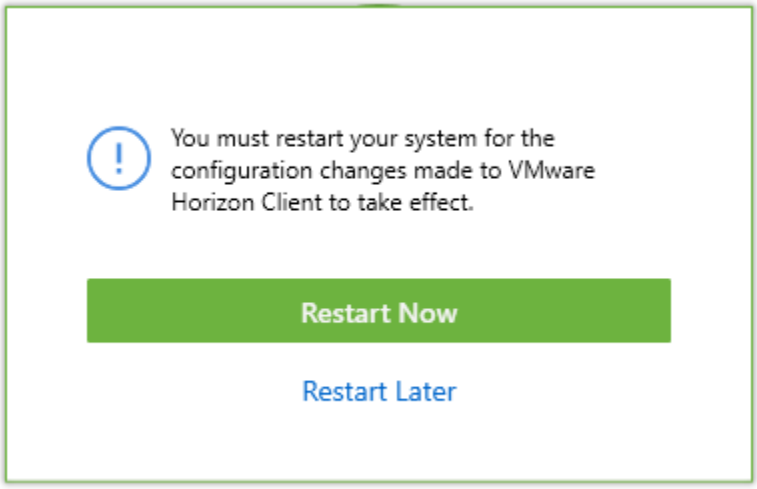
5. Click **Agree & Install** to continue installation.



6. Click Finish to complete installation.



7. Reboot Physical Desktop.



8. After reboot, log back in to see Horizon View Client installed.

Chapter 6.
NVIDIA VGPU MANAGER VIB INSTALLATION

This chapter discusses the following NVIDIA vGPU Manager **VIB** installation topics:

- Step-by-step guide to downloading and preparing the VIB file for installation on the host.
  - VIB file upload using either vSphere Web Client
  - MobaXterm (see Chapter 13, *Using MobaXterm*, on page 264)
  - WinSCP (see Chapter 14, *Using WINSCP*), on page 270)
- Installing the **VIB**.
- Updating the **VIB**.
- Uninstalling the **VIB**.

6.1 PREPARING THE VIB FOR INSTALL

Use the following procedure to prepare the **VIB** file:

1. Download the archive containing the **VIB** file.

Extract the contents of the archive to a folder. The file ending with **VIB** is the file that you must copy to the host data store for installation.

6.1.1 Uploading VIB in vSphere

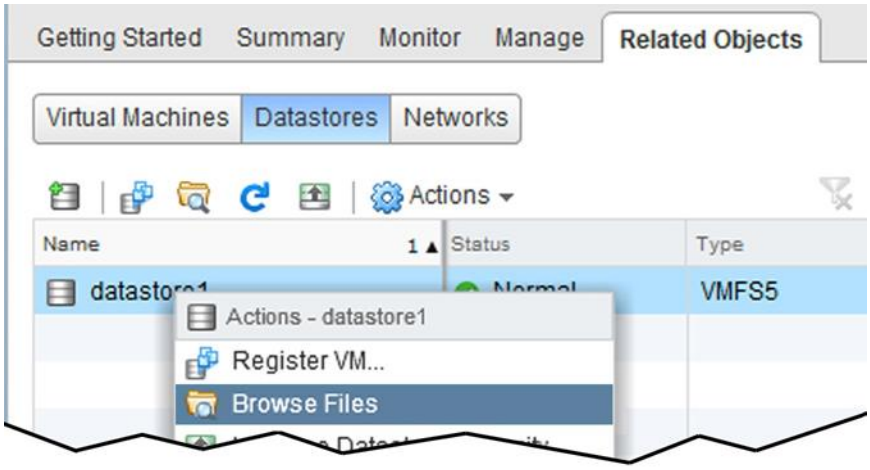
This section describes how to upload a **.VIB** file to the host using the vSphere Web Client. See Chapter 13, *Using MobaXterm*, on page 264 if you are using MobaXterm or Chapter 14, *Using WINSCP*, on page 270 if you are using WinSCP.

To upload the file to the data store using vSphere Web Client:

2. Click the **Related Objects** tab for the desired server

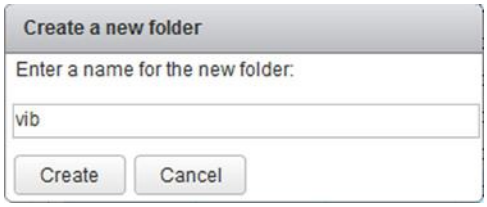
3. Select **Datastores**.

Either right click the data store and then select **Browse Files** or click the icon in the toolbar. The *Datastore Browser* window displays.



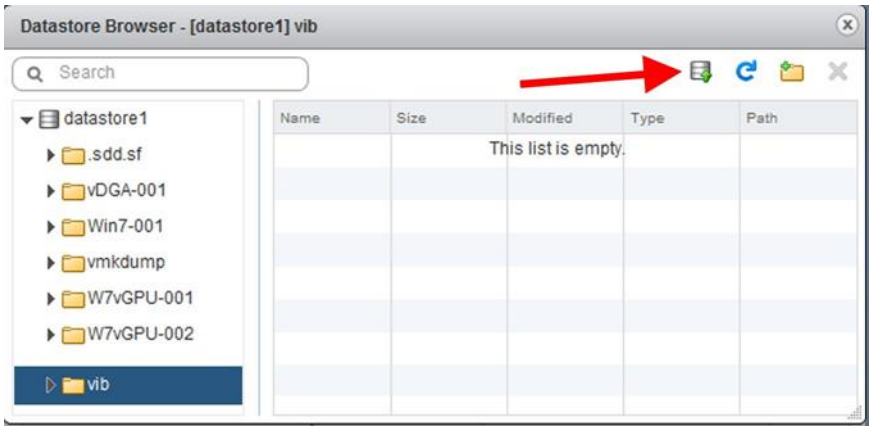
Click the **New Folder** icon.
The *Create a new folder* window displays.

Name the new folder **vib** and then click **Create**.



Select the **vib** folder in the *Datastore Browser* window.

Click the **Upload** icon.



The *Client Integration Access Control* window displays.

Select **Allow**.
The **.VIB** file is uploaded to the data store on the host.

If you do not click **Allow** before the timer runs out, then further attempts to upload a file will silently fail. If this happens, exit and restart vSphere Web Client., Repeat this procedure and be sure to click **Allow** before the timer runs out.

6.2 INSTALLING THE VIB

The NVIDIA Virtual GPU Manager runs on the ESXi host. It is provided in the following formats:

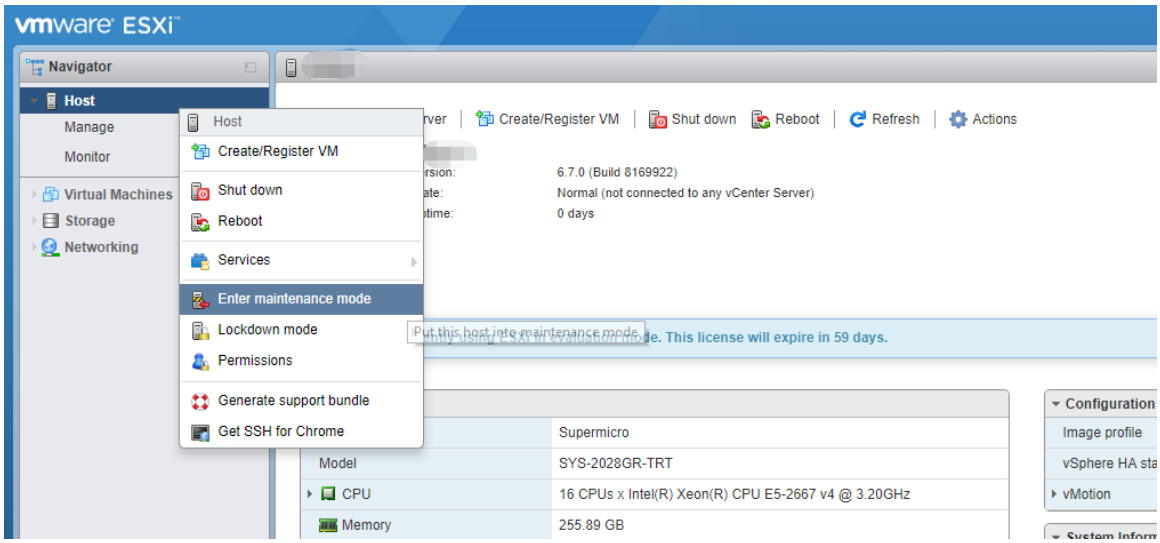
- As a VIB file, which must be copied to the ESXi host and then installed
- As an offline bundle that you can import manually as explained in Import Patches Manually in the VMware vSphere documentation

CAUTION: NVIDIA Virtual GPU Manager and Guest VM drivers must be matched from the same main driver branch. If you update vGPU Manager to a release from another driver branch, guest VMs will boot with vGPU disabled until their guest vGPU driver is updated to match the vGPU Manager version. Consult Virtual GPU Software for VMware vSphere Release Notes for further details..

To install the vGPU Manager VIB you need to access the ESXi host via the ESXi Shell or SSH. Refer to VMware's documentation on how to enable ESXi Shell or SSH for an ESXi host.

Before proceeding with the vGPU Manager installation make sure that all VMs are powered off and the ESXi host is placed in maintenance mode. Refer to VMware's documentation on how to place an ESXi host in maintenance mode.

Place the host into Maintenance mode by right-clicking it and then selecting **Maintenance Mode→Enter Maintenance Mode**.



Alternatively, you can place the host into Maintenance mode using the command prompt by entering

$ esxcli system maintenanceMode set -- enable=true

This command will not return a response. Making this change using the command prompt will not refresh the vSphere Web Client UI. Click the **Refresh** icon in the upper right corner of the vSphere Web Client window.

CAUTION: PLACING THE HOST INTO MAINTENANCE MODE DISABLES ANY VCENTER APPLIANCE RUNNING ON THIS HOST UNTIL YOU EXIT MAINTENANCE MODE AND THEN RESTART THAT VCENTER APPLIANCE.

Click **OK** to confirm your selection.

1. Use the esxcli command to install the vGPU Manager package:

```
[root@esxi:~] esxcli software vib install -v directory/NVIDIA-vGPU-

VMware_ESXi_6.0_Host_Driver_390.72-1OEM.600.0.0.2159203.vib

Installation Result    Message: Operation finished
successfully.

   Reboot Required: false

   VIBs Installed: NVIDIA-vGPU-

VMware_ESXi_6.0_Host_Driver_390.72-1OEM.600.0.0.2159203    VIBs
Removed:

   VIBs Skipped:
```

directory is the absolute path to the directory that contains the VIB file. You must specify the absolute path even if the VIB file is in the current working directory.

Reboot the ESXi host and remove it from maintenance mode.

Although the display states "**Reboot Required: false**", a reboot is necessary for the vib to load and xorg to start.

From the vSphere Web Client, exit **Maintenance Mode** by right-clicking the host and selecting **Exit Maintenance Mode.**

Alternatively, you may exit from Maintenance mode via the command prompt by entering:
$ esxcli system maintenanceMode set -- enable=false

This command will not return a response.

Making this change via the command prompt will not refresh the vSphere Web Client UI. Click the **Refresh** icon in the upper right corner of the vSphere Web Client window.

Reboot the host from the vSphere Web Client by right-clicking the host and then selecting **Reboot**.

You can reboot the host by entering the following at the command prompt:
$ reboot
This command will not return a response. The Reboot Host window displays.

When rebooting from the vSphere Web Client, enter a descriptive reason for the reboot in the **Log a reason for this reboot operation** field, and then click **OK** to proceed.

6.3 UPDATING THE VIB

Update the vGPU Manager VIB package if you want to install a new version of NVIDIA Virtual GPU Manager on a system where an existing version is already installed.

To update the vGPU Manager VIB you need to access the ESXi host via the ESXi Shell or SSH. Refer to VMware's documentation on how to enable ESXi Shell or SSH for an ESXi host.

Before proceeding with the vGPU Manager update, make sure that all VMs are powered off and the ESXi host is placed in maintenance mode. Refer to VMware's documentation on how to place an ESXi host in maintenance mode

1. Use the esxcli command to update the vGPU Manager package:

```
[root@esxi:~] esxcli software vib update -v directory/NVIDIA-vGPU-

VMware_ESXi_6.0_Host_Driver_390.72-1OEM.600.0.0.2159203.vib

Installation Result    Message: Operation finished
successfully.

  Reboot Required: false

  VIBs Installed: NVIDIA-vGPU-

VMware_ESXi_6.0_Host_Driver_390.72-1OEM.600.0.0.2159203

  VIBs Removed: NVIDIA-vGPU-

VMware_ESXi_6.0_Host_Driver_390.57-1OEM.600.0.0.2159203    VIBs
Skipped:
```

directory is the path to the directory that contains the VIB file.

2. Reboot the ESXi host and remove it from maintenance mode.

6.4 VERIFYING THE INSTALLATION OF THE VIB

After the ESXi host has rebooted, verify the installation of the NVIDIA vGPU software package for vSphere.

1. Verify that the NVIDIA vGPU software package installed and loaded correctly by checking for the NVIDIA kernel driver in the list of kernel loaded modules.

```
[root@esxi:~] vmkload_mod -l | grep nvidia nvidia                5
8420
```

2. If the NVIDIA driver is not listed in the output, check dmesg for any load-time errors reported by the driver.

3. Verify that the NVIDIA kernel driver can successfully communicate with the

   NVIDIA physical GPUs in your system by running the nvidia-smi command.

   The nvidia-smi command is described in more detail in NVIDIA System Management Interface nvidia-smi.

Running the nvidia-smi command should produce a listing of the GPUs in your platform.

```
[root@esxi:~] nvidia-smi

Fri Jul 20 17:56:22 2018

+-----------------------------------------------------+

| NVIDIA-SMI 390.72     Driver Version: 390.75        |
```

```
|-------------------------------+----------------------+----------------------+

| GPU  Name       Persistence-M| Bus-Id       Disp.A | Volatile Uncorr. ECC |

| Fan  Temp  Perf  Pwr:Usage/Cap|        Memory-Usage | GPU-Util  Compute M. |

|===============================+======================+======================|

|   0  Tesla M60         On   | 0000:85:00.0     Off |              Off |

| N/A   23C    P8    23W / 150W |    13MiB /  8191MiB |     0%     Default |

+-------------------------------+----------------------+----------------------+

|   1  Tesla M60         On   | 0000:86:00.0     Off |              Off |

| N/A   29C    P8    23W / 150W |    13MiB /  8191MiB |     0%     Default |

+-------------------------------+----------------------+----------------------+

|   2  Tesla P40         On   | 0000:87:00.0     Off |              Off |

| N/A   21C    P8    18W / 250W |    53MiB / 24575MiB |     0%     Default |

+-------------------------------+----------------------+----------------------+

+-----------------------------------------------------------------------------+

| Processes:                                                       GPU Memory |

|  GPU       PID  Type  Process name                             Usage      |

|=============================================================================|

|  No running processes found                                                 |
```

```
+-----------------------------------------------------------------------------+
```

If nvidia-smi fails to report the expected output for all the NVIDIA GPUs in your system, see Troubleshooting for troubleshooting steps.

SMI also allows GPU monitoring using the following command:

```
$ nvidia-smi -l
```

This command switch adds a loop, auto refreshing the display.

6.5 UNINSTALLING VIB

Use the following procedure to uninstall **VIB**:

4. Determine the name of the vGPU driver bundle

```
$ esxcli software vib list | grep -i nvidia
```

5. This command returns output similar to the following:

```
NVIDIA-VMware_ESXi_6.7_Host_Driver 390.72-1OEM.600.0.0.2159203          NVIDIA
VMwareAccepted    2018-07-20
```

6. Run the following command to uninstall the driver package:

```
$ esxcli software vib remove -n NVIDIA-VMware_ESXi_6.7_Host_Driver --maintenance-mode
```

The following message displays when installation is successful:

```
Removal Result
  Message: Operation finished successfully.
  Reboot Required: false
  VIBs Installed:
  VIBs Removed: NVIDIA_bootbank_NVIDIA-VMware_ESXi_6.7_Host_Driver_390.72-
1OEM.600.0.0.2159203
  VIBs Skipped:
```

Reboot the host to complete the uninstallation process.

6.6 6.0 ONLY: CONFIGURING SUSPEND AND RESUME FOR VMWARE VSPHERE

Since 6.1: Suspend-resume for VMs that are configured with vGPU is enabled by default and this task is not required.

NVIDIA vGPU software supports suspend and resume for VMs that are configured with vGPU.

For details about which VMware vSphere versions, NVIDIA GPUs, and guest OS releases support suspend and resume, see Virtual GPU Software for VMware vSphere Release Notes.

Before configuring suspend and resume for an ESXi host, ensure that the current NVIDIA Virtual GPU Manager for VMware vSphere package is installed on the host.

1. Set the registry key that is required to enable suspend and resume.

```
[root@esxi:~] esxcli system module parameters set -m nvidia -p
"NVreg_RegistryDwords=RMEnableVgpuMigration=1"
```

2. Reboot the ESXi host.

3. Confirm that suspend and resume are configured for the ESXi host.

```
[root@esxi:~] dmesg | grep NVRM
2018-03-08T05:05:21.084Z cpu51:2098073)NVRM: vmk_MemPoolCreate passed for 3162111 pages.
2018-03-08T05:05:21.336Z cpu51:2098073)NVRM: loading NVIDIA UNIX x86_64 Kernel Module
390.42  Sat Mar  3 02:52:47 PST 2018
2018-03-08T05:05:24.200Z cpu22:2098091)NVRM: nvidia_associate vmgfx0
2018-03-08T05:05:24.201Z cpu22:2098091)NVRM: nvidia_associate vmgfx1
2018-03-08T05:05:24.202Z cpu22:2098091)NVRM: nvidia_associate vmgfx2
2018-03-08T05:05:24.203Z cpu22:2098091)NVRM: nvidia_associate vmgfx3
2018-03-08T05:05:55.305Z cpu39:2100364)NVRM: Enabling vGPU live migration for GPU at
0000:3d:00.0
2018-03-08T05:05:55.983Z cpu39:2100364)NVRM: Enabling vGPU live migration for GPU at
0000:3e:00.0
2018-03-08T05:05:56.654Z cpu39:2100364)NVRM: Enabling vGPU live migration for GPU at
0000:af:00.0
2018-03-08T05:05:57.174Z cpu39:2100364)NVRM: GPU at 0000:af:00.0 has Software Scheduler
disabled.
2018-03-08T05:05:57.975Z cpu39:2100364)NVRM: Enabling vGPU live migration for GPU at
0000:d8:00.0
2018-03-08T05:05:58.494Z cpu39:2100364)NVRM: GPU at 0000:d8:00.0 has Software Scheduler
disabled.
```

If you want to unset the registry entry that is required to enable suspend and resume, set NVreg_RegistryDwords=RMEnableVgpuMigration=0. Then reboot the ESXi host.

6.7 CHANGING THE DEFAULT GRAPHICS TYPE IN VMWARE VSPHERE 6.5 AND LATER

The vGPU Manager VIBs for VMware vSphere 6.5 and later provide vSGA and vGPU functionality in a single VIB. After this VIB is installed, the default graphics type is Shared, which provides vSGA functionality. To enable vGPU support for VMs in VMware vSphere 6.5, you must change the default graphics type to Shared Direct. If you do not change the default graphics type, VMs to which a vGPU is assigned fail to start and the following error message is displayed:

```
The amount of graphics resource available in the parent resource pool is insufficient for the operation.
```

Note:

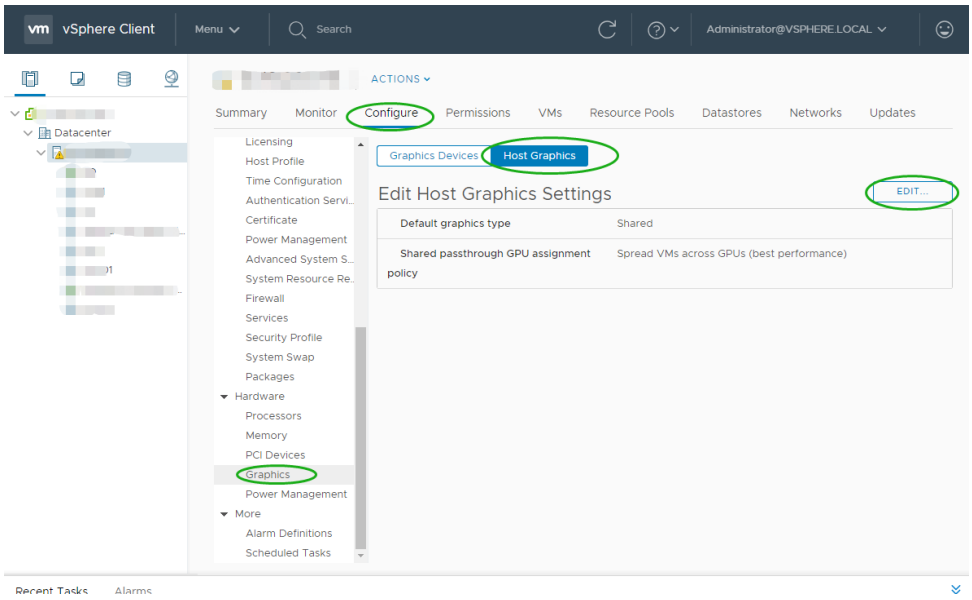
If you are using a supported version of VMware vSphere earlier than 6.5, or are configuring a VM to use vSGA, omit this task.

Change the default graphics type before configuring vGPU. Output from the VM console in the VMware vSphere Web Client is not available for VMs that are running vGPU.

Before changing the default graphics type, ensure that the ESXi host is running and that all VMs on the host are powered off.

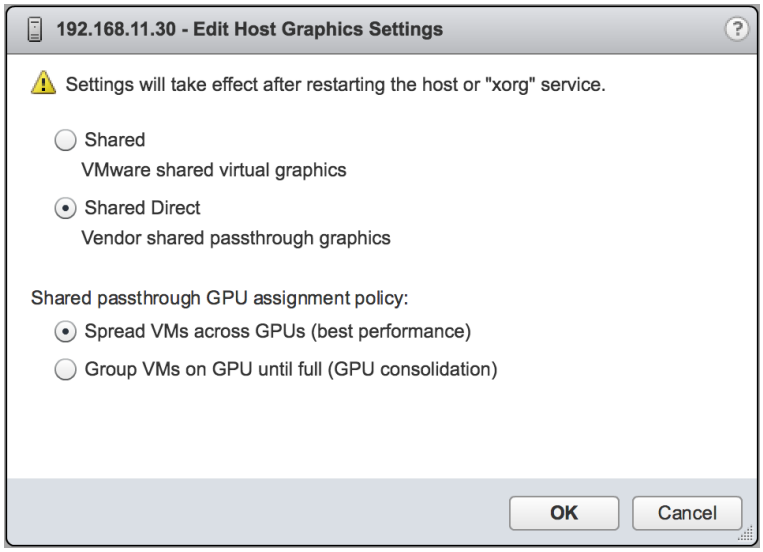
1. Log in to vCenter Server by using the vSphere Web Client.
2. In the navigation tree, select your ESXi host and click the **Configure** tab.
3. From the menu, choose **Graphics** and then click the **Host Graphics** tab.
4. On the **Host Graphics** tab, click **Edit**.

Figure 7. Shared default graphics type



5. In the Edit Host Graphics Settings dialog box that opens, select **Shared Direct** and click **OK**.

Figure 8. Host graphics settings for vGPU



Note: In this dialog box, you can also change the allocation scheme for vGPU-enabled VMs. For more information, see Modifying GPU Allocation Policy on VMware vSphere.

After you click OK, the default graphics type changes to Shared Direct.

6. Restart the ESXi host **or** stop and restart the Xorg service and nv-hostengine on the ESXi host.

To stop and restart the Xorg service and nv-hostengine, perform these steps:

a. Stop the Xorg service.

```
[root@esxi:~] /etc/init.d/xorg stop
```

b. Stop nv-hostengine.

```
[root@esxi:~] nv-hostengine -t
```

c. Wait for 1 second to allow nv-hostengine to stop.

d. Start nv-hostengine.

```
[root@esxi:~] nv-hostengine -d
```

e. Start the Xorg service.

```
[root@esxi:~] /etc/init.d/xorg start
```

After changing the default graphics type, configure vGPU as explained in Configuring a vSphere VM with Virtual GPU.

See also the following topics in the VMware vSphere documentation:

- Log in to vCenter Server by Using the vSphere Web Client
- Configuring Host Graphics

6.8 CHANGING VGPU SCHEDULING POLICY

GPUs based on the NVIDIA Maxwell™ graphic architecture implement a best effort vGPU scheduler that aims to balance performance across vGPUs. The best effort scheduler allows a vGPU to use GPU processing cycles that are not being used by other vGPUs. Under some circumstances, a VM running a graphics-intensive application may adversely affect the performance of graphics-light applications running in other VMs.

GPUs based on the NVIDIA Pascal™ architecture and the NVIDIA Volta architecture additionally support equal share and fixed share vGPU schedulers. These schedulers impose a limit on GPU processing cycles used by a vGPU, which prevents graphics-intensive applications running in one VM from affecting the performance of graphics-light applications running in other VMs. On GPUs based on the Pascal architecture, you can select the vGPU scheduler to use.

The best effort scheduler is the default scheduler for all supported GPU architectures.

The GPUs that are based on the Pascal architecture are the Tesla P4, Tesla P6, Tesla P40, and Tesla P100.

The GPUs that are based on the Volta architecture are the Tesla V100 SXM2, Tesla V100 PCIe, and Tesla V100 FHHL.

6.8.1 vGPU Scheduling Policies

In addition to the default best effort scheduler, GPUs based on the Pascal and Volta architectures support equal share and fixed share vGPU schedulers.

Equal Share Scheduler

The physical GPU is shared equally amongst the running vGPUs that reside on it. As vGPUs are added to or removed from a GPU, the share of the GPU's processing cycles allocated to each vGPU changes accordingly. As a result, the performance of a vGPU may increase as other vGPUs on the same GPU are stopped or decrease as other vGPUs are started on the same GPU.

Fixed Share Scheduler

Each vGPU is given a fixed share of the physical GPU's processing cycles, the amount of which depends on the vGPU type. As vGPUs are added to or removed from a GPU, the share of the GPU's processing cycles allocated to each vGPU remains constant. As a result, the performance of a vGPU remains unchanged as other vGPUs are stopped or started on the same GPU.

6.8.2 RmPVMRL Registry Key

The RmPVMRL registry key sets the scheduling policy for NVIDIA vGPUs.

Note: You can change the vGPU scheduling policy only on GPUs based on the Pascal and Volta architectures.

Type

Dword

Contents

Value	Meaning
0x00 (default)	Best Effort Scheduler
0x01	Equal Share Scheduler
0x11	Fixed Share Scheduler

Examples

This example sets the vGPU scheduler to Fixed Share Scheduler.

```
RmPVMRL=0x11
```

This example sets the vGPU scheduler to Equal Share Scheduler.

```
RmPVMRL=0x01
```

6.8.3 Changing the vGPU Scheduling Policy for All GPUs

Note: You can change the vGPU scheduling policy only on GPUs based on the Pascal and Volta architectures.

Perform this task in your hypervisor command shell.

1. Open a command shell as the root user on your hypervisor host machine. On all supported hypervisors, you can use secure shell (SSH) for this purpose. Individual hypervisors may provide additional means for logging in. For details, refer to the documentation for your hypervisor.

2. Set the RmPVMRL registry key to the value that sets the GPU scheduling policy that you want.

   On VMware vSphere, use the esxcli set command.

   ```
   # esxcli system module parameters set -m nvidia -p
   "NVreg_RegistryDwords=RmPVMRL=value"
   ```

3. ***value***
4. The value that sets the vGPU scheduling policy that you want, for example:
5. **0x01**
6. Sets the vGPU scheduling policy to Equal Share Scheduler.

7. **0x11**
8. Sets the vGPU scheduling policy to Fixed Share Scheduler.
9. For all supported values, see RmPVMRL Registry Key.
10. Reboot your hypervisor host machine.

6.8.4 Changing the vGPU Scheduling Policy for Select GPUs

Note: You can change the vGPU scheduling policy only on GPUs based on the Pascal and Volta architectures.

Perform this task in your hypervisor command shell.

1. Open a command shell as the root user on your hypervisor host machine. On all supported hypervisors, you can use secure shell (SSH) for this purpose. Individual hypervisors may provide additional means for logging in. For details, refer to the documentation for your hypervisor.

2. Use the lspci command to obtain the PCI domain and bus/device/function (BDF) of each GPU for which you want to change the scheduling behavior.

   On VMware vSphere, pipe the output of lspci to the grep command to display information only for NVIDIA GPUs.

   ```
   # lspci | grep NVIDIA
   ```

3. The NVIDIA GPUs listed in this example have the PCI domain 0000 and BDFs 85:00.0 and 86:00.0.

   ```
   0000:85:00.0 VGA compatible controller: NVIDIA Corporation GM204GL [Tesla M60] (rev a1)
   0000:86:00.0 VGA compatible controller: NVIDIA Corporation GM204GL [Tesla M60] (rev a1)
   ```

6. Use the module parameter NVreg_RegistryDwordsPerDevice to set the pci and RmPVMRL registry keys for each GPU.

   On VMware vSphere, use the esxcli set command.

   ```
   # esxcli system module parameters set -m nvidia \
   -p "NVreg_RegistryDwordsPerDevice=pci=pci-domain:pci-bdf;RmPVMRL=value\
   [;pci=pci-domain:pci-bdf;RmPVMRL=value...]"
   ```

7. For each GPU, provide the following information:
8. ***pci-domain***
9. The PCI domain of the GPU.
10. ***pci-bdf***
11. The PCI device BDF of the GPU.
12. ***value***
13. The value that sets the vGPU scheduling policy that you want, for example:
14. **0x01**
15. Sets the GPU scheduling policy to Equal Share Scheduler.
16. **0x11**
17. Sets the GPU scheduling policy to Fixed Share Scheduler.
18. For all supported values, see RmPVMRL Registry Key.
19. This example adds an entry to the /etc/modprobe.d/nvidia.conf file to change the scheduling behavior of two GPUs as follows:

- For the GPU at PCI domain 0000 and BDF 85:00.0, the vGPU scheduling policy is set to Equal Share Scheduler.
- For the GPU at PCI domain 0000 and BDF 86:00.0, the vGPU scheduling policy is set to Fixed Share Scheduler.

20. options nvidia NVreg_RegistryDwordsPerDevice=
    "pci=0000:85:00.0;RmPVMRL=0x01;pci=0000:86:00.0;RmPVMRL=0x11"
21. Reboot your hypervisor host machine.

6.8.5 Restoring Default vGPU Scheduler Settings

Perform this task in your hypervisor command shell.

1. Open a command shell as the root user on your hypervisor host machine. On all supported hypervisors, you can use secure shell (SSH) for this purpose. Individual hypervisors may provide additional means for logging in. For details, refer to the documentation for your hypervisor.

2. Unset the RmPVMRL registry key.
   On VMware vSphere, set the module parameter to an empty string.
   ```
   # esxcli system module parameters set -m nvidia -p "module-parameter="
   ```
   module-parameter
   The module parameter to set, which depends on whether the scheduling behavior was changed for all GPUs or select GPUs:

   - For all GPUs, set the NVreg_RegistryDwords module parameter.
   - For select GPUs, set the NVreg_RegistryDwordsPerDevice module parameter.
     For example, to restore default vGPU scheduler settings after they were changed for all GPUs, enter this command:
     ```
     # esxcli system module parameters set -m nvidia -p "NVreg_RegistryDwords="
     ```

3. Reboot your hypervisor host machine.

Chapter 7.
SELECTING CORRECT vGPU PROFILES

Choosing the right profile to maximize your stakeholders experience within the virtual instance is critical. Below, you will find guidance through the vGPU Manager and beyond to ensure your deployment is successful.

7.1 THE ROLE OF THE vGPU MANAGER

NVIDIA vGPU profiles assign custom amounts of dedicated graphics memory for each user. NVIDIA vGPU Manager assigns the correct amount of memory to meet the specific needs within the workflow for said user. Every virtual machine has dedicated graphics memory and must be assigned accordingly thus ensuring that it has the resources needed to handle the expected graphics load.

NVIDIA vGPU Manager allows up to eight users to share each physical GPU by assigning the graphics resources of the available GPUs to virtual machines using a balanced approach. Depending on the number of GPUs within each Tesla line card, there can be multiple user types assigned.

7.2 THE FULL LIST OF vGPU PROFILES

The profiles represent a very flexible deployment options of virtual GPUs, varying in size of frame buffer memory and number and resolution of display heads. The division of frame buffer is what defines the number of users possible per GPU with that specific profile, while the number of heads defines the number of displays supported. Max resolution is consistent

across all the profiles. The full list may also be found here: *https://docs.nvidia.com/grid/6.0/grid-vgpu-user-guide/index.html*

Table 7-1. NVIDIA vGPU Profiles

Tesla M60 – Physical GPUs per Board: 2							
Required SW Edition	**Virtual GPU Profile**	**Intended Use Case**	**Frame Buffer (Mbytes)**	**Virtual Display Heads**	**Max Resolution per Display Head**	**Maximum vGPUs per GPU**	**Maximum vGPU per Boards**
Quadro vDWS	M60-8Q	Designer	8192	4	4096x2160	1	2
Quadro vDWS	M60-4Q	Designer	4096	4	4096x2160	2	4
Quadro vDWS	M60-2Q	Designer	2048	4	4096x2160	4	8
Quadro vDWS	M60-1Q	Designer Power User	1024	2	4096x2160	8	16
Quadro vDWS	M60-0Q	Designer Power User	512	2	2560x1600	16	32
GRID Virtual PC	M60-1B	Power User	1024	4	2560x1600	8	16
GRID Virtual PC	M60-0B	Power User	512	2	2560x1600	16	32
GRID Virtual Apps	M60-8A	Virtual Application User	8192	1	1280x1024	1	2

GRID Virtual Apps	M60-4A	Virtual Application User	4096	1	1280x1024	2	4
GRID Virtual Apps	M60-2A	Virtual Application User	2048	1	1280x1024	4	8
GRID Virtual Apps	M60-1A	Virtual Application User	1024	1	1280x1024	8	16
Tesla M10 – Physical GPUs per Board: 4							
Required Software Edition	**Virtual GPU Profile**	**Intended Use Case**	**Frame Buffer (Mbytes)**	**Virtual Display Heads**	**Max Resolution per Display Head**	**Maximum vGPUs per GPU**	**Maximum vGPU per Boards**
Quadro vDWS	M10-8Q	Designer	8192	4	4096x2160	1	4
Quadro vDWS	M10-4Q	Designer	4096	4	4096x2160	2	8
Quadro vDWS	M10-2Q	Designer	2048	4	4096x2160	4	16
Quadro vDWS	M10-1Q	Designer Power User	1024	2	4096x2160	8	32
Quadro vDWS	M10-0Q	Designer Power User	512	2	2560x1600	16	64
GRID Virtual PC	M10-1B	Power User	1024	4	2560x1600	8	32

GRID Virtual PC	M10-0B	Power User	512	2	2560x1600	16	64
GRID Virtual Apps	M10-8A	Virtual Application User	8192	1	1280x1024	1	4
GRID Virtual Apps	M10-4A	Virtual Application User	4096	1	1280x1024	2	8
GRID Virtual Apps	M10-2A	Virtual Application User	2048	1	1280x1024	4	16
GRID Virtual Apps	M10-1A	Virtual Application User	1024	1	1280x1024	8	32
TESLA M6 – Physical GPUs per Board: 1							
Required License Edition	**Virtual GPU Profile**	**Intended Use Case**	**Frame Buffer (Mbytes)**	**Virtual Display Heads**	**Max Resolution per Display Head**	**Maximum vGPUs per GPU**	**Maximum vGPU per Boards**
Quadro vDWS	M6-8Q	Designer	8192	4	4096x2160	1	1
Quadro vDWS	M6-4Q	Designer	4096	4	4096x2160	2	2
Quadro vDWS	M6-2Q	Designer	2048	4	4096x2160	4	4

Quadro vDWS	M6-1Q	Designer Power User	1024	2	4096x2160	8	8
Quadro vDWS	M6-0Q	Designer Power User	512	2	2560x1600	16	16
GRID Virtual PC	M6-1B	Power User	1024	4	2560x1600	8	8
GRID Virtual PC	M6-0B	Power User	512	2	2560x1600	16	16
GRID Virtual Apps	M6-8A	Virtual Application User	8192	1	1280x1024	1	1
GRID Virtual Apps	M6-4A	Virtual Application User	4096	1	1280x1024	2	2
GRID Virtual Apps	M6-2A	Virtual Application User	2048	1	1280x1024	4	4
GRID Virtual Apps	M6-1A	Virtual Application User	1024	1	1280x1024	8	8
Tesla P100 PCIe 12GB – Physical GPUs per Board: 1							
Required Software Edition	**Virtual GPU Profile**	**Intended Use Case**	**Frame Buffer (Mbytes)**	**Virtual Display Heads**	**Max Resolution per Display Head**	**Maximum vGPUs per GPU**	**Maximum vGPU per Boards**

Quadro vDWS	P100C-12Q	Designer	12288	4	4096x2160	1	1
Quadro vDWS	P100C-6Q	Designer	6144	4	4096x2160	2	2
Quadro vDWS	P100C-4Q	Designer	4096	4	4096x2160	3	3
Quadro vDWS	P100C-2Q	Designer	2048	4	4096x2160	6	6
Quadro vDWS	P100C-1Q	Designer Power User	1024	2	4096x2160	12	12
GRID Virtual PC	P100C-2B	Power User	2048	2	4096x2160	6	6
GRID Virtual PC	P100C-2B4	Power user	2048	4	2560x1600	6	6
GRID Virtual PC	P100C-1B	Power User	1024	4	2560x1600	6	6
GRID Virtual Apps	P100C-12A	Virtual Application User	12288	1	1280x1024	1	1
GRID Virtual Apps	P100C-6A	Virtual Application User	6144	1	1280x1024	2	2
GRID Virtual Apps	P100-4A	Virtual Application User	4096	1	1280x1024	3	3

GRID Virtual Apps	P100-2A	Virtual Application User	2048	1	1280x1024	6	6
GRID Virtual Apps	P100C-1A	Virtual Application User	1024	1	1280x1024	12	12
Tesla P100 PCIe 16GB – Physical GPUs per Board: 1							
Required Software Edition	**Virtual GPU Profile**	**Intended Use Case**	**Frame Buffer (Mbytes)**	**Virtual Display Heads**	**Max Resolution per Display Head**	**Maximum vGPUs per GPU**	**Maximum vGPU per Boards**
Quadro vDWS	P100-16Q	Designer	16384	4	4096x2160	1	1
Quadro vDWS	P100-8Q	Designer	8192	4	4096x2160	2	2
Quadro vDWS	P100-4Q	Designer	4096	4	4096x2160	4	4
Quadro vDWS	P100-2Q	Designer	2048	4	4096x2160	8	8
Quadro vDWS	P100-1Q	Designer Power User	1024	2	4096x2160	16	16
GRID Virtual PC	P100-2B	Power User	2048	2	4096x2160	8	8
GRID Virtual PC	P100-2B4	Power User	2048	4	2560x1600	8	8

GRID Virtual PC	P100-1B	Power User	1024	4	2560x1600	16	16
GRID Virtual Apps	P100-16A	Virtual Application User	16384	1	1280x1024	1	1
GRID Virtual Apps	P100-8A	Virtual Application User	8192	1	1280x1024	2	2
GRID Virtual Apps	P100-4A	Virtual Application User	4096	1	1280x1024	4	4
GRID Virtual Apps	P100-2A	Virtual Application User	2048	1	1280x1024	8	8
GRID Virtual Apps	P100-1A	Virtual Application User	1024	1	1280x1024	16	16
Tesla P100 SXM2 – Physical GPUs per Board: 1							
Required Software Edition	**Virtual GPU Profile**	**Intended Use Case**	**Frame Buffer (Mbytes)**	**Virtual Display Heads**	**Max Resolution per Display Head**	**Maximum vGPUs per GPU**	**Maximum vGPU per Boards**
Quadro vDWS	P100X-16Q	Designer	16384	4	4096x2160	1	1
Quadro vDWS	P100X-8Q	Designer	8192	4	4096x2160	2	2

Quadro vDWS	P100X-4Q	Designer	4096	4	4096x2160	4	4
Quadro vDWS	P100X-2Q	Designer	2048	4	4096x2160	8	8
Quadro vDWS	P100X-1Q	Designer Power User	1024	2	4096x2160	16	16
GRID Virtual PC	P100X-2B	Power User	2048	2	4096x2160	8	8
GRID Virtual PC	P100X-2B4	Power User	2048	4	2560x1600	8	8
GRID Virtual PC	P100X-1B	Power User	1024	4	2560x1600	16	16
GRID Virtual Apps	P100X-16A	Virtual Application User	16384	1	1280x1024	1	1
GRID Virtual Apps	P100X-8A	Virtual Application User	8192	1	1280x1024	2	2
GRID Virtual Apps	P100X-4A	Virtual Application User	4096	1	1280x1024	4	4
GRID Virtual Apps	P100X-2A	Virtual Application User	2048	1	1280x1024	8	8

GRID Virtual Apps	P100X-1A	Virtual Application User	1024	1	1280x1024	16	16
Tesla P40 – Physical GPUs per Board: 1							
Required Software Edition	**Virtual GPU Profile**	**Intended Use Case**	**Frame Buffer (Mbytes)**	**Virtual Display Heads**	**Max Resolution per Display Head**	**Maximum vGPUs per GPU**	**Maximum vGPU per Boards**
Quadro vDWS	P40-24Q	Designer	24576	4	4096x2160	1	1
Quadro vDWS	P40-12Q	Designer	12288	4	4096x2160	2	2
Quadro vDWS	P40-8Q	Designer	8192	4	4096x2160	4	4
Quadro vDWS	P40-6Q	Designer	6144	4	4096x2160	4	4
Quadro vDWS	P40-4Q	Designer	4096	4	4096x2160	6	6
Quadro vDWS	P40-3Q	Designer	3072	4	4096x2160	8	8
Quadro vDWS	P40-2Q	Designer	2048	4	4096x2160	12	12
Quadro vDWS	P40-1Q	Designer Power User	1024	2	4096x2160	24	24

GRID Virtual PC	P24-2B	Power User	2048	2	4096x2160	12	12
GRID Virtual PC	P40-2B4	Power User	2048	4	2560x1600	12	12
GRID Virtual PC	P40-1B	Power User	1024	4	2560x1600	24	24
GRID Virtual Apps	P40-24A	Virtual Application User	24576	1	1280x1024	1	1
GRID Virtual Apps	P40-12A	Virtual Application User	12288	1	1280x1024	2	2
GRID Virtual Apps	P40-8A	Virtual Application User	8192	1	1280x1024	3	3
GRID Virtual Apps	P40-6A	Virtual Application User	6144	1	1280x1024	4	4
GRID Virtual Apps	P40-4A	Virtual Application User	4096	1	1280x1024	6	6
GRID Virtual Apps	P40-3A	Virtual Application User	3072	1	1280x1024	8	8
GRID Virtual Apps	P40-2A	Virtual Application User	2048	1	1280x1024	12	12

GRID Virtual Apps	P40-1A	Virtual Application User	1024	1	1280x1024	24	24
Tesla P6 – Physical GPUs per Board: 1							
Required Software Edition	**Virtual GPU Profile**	**Intended Use Case**	**Frame Buffer (Mbytes)**	**Virtual Display Heads**	**Max Resolution per Display Head**	**Maximum vGPUs per GPU**	**Maximum vGPU per Boards**
Quadro vDWS	P6-16Q	Designer	16384	4	4096x2160	1	1
Quadro vDWS	P6-8Q	Designer	8192	4	4096x2160	2	2
Quadro vDWS	P6-4Q	Designer	4096	4	4096x2160	4	4
Quadro vDWS	P6-2Q	Designer	2048	4	4096x2160	8	8
Quadro vDWS	P6-1Q	Designer Power User	1024	2	4096x2160	16	16
GRID Virtual PC	P6-2B	Power User	2048	2	4096x2160	8	8
GRID Virtual PC	P6-2B4	Power User	2048	4	2560x1600	8	8
GRID Virtual PC	P6-1B	Power User	1024	4	2560x1600	16	16

GRID Virtual Apps	P6-16A	Virtual Application User	16384	1	1280x1024	1	1
GRID Virtual Apps	P6-8A	Virtual Application User	8192	1	1280x1024	2	2
GRID Virtual Apps	P6-4A	Virtual Application User	4096	1	1280x1024	4	4
GRID Virtual Apps	P6-2A	Virtual Application User	2048	1	1280x1024	8	8
GRID Virtual Apps	P6-1A	Virtual Application User	1024	1	1280x1024	16	16
Tesla P4 – Physical GPUs per Board: 1							
Required Software Edition	**Virtual GPU Profile**	**Intended Use Case**	**Frame Buffer (Mbytes)**	**Virtual Display Heads**	**Max Resolution per Display Head**	**Maximum vGPUs per GPU**	**Maximum vGPU per Boards**
Quadro vDWS	P4-8Q	Designer	8192	4	4096x2160	1	1
Quadro vDWS	P4-4Q	Designer	4096	4	4096x2160	2	2
Quadro vDWS	P4-2Q	Designer	2048	4	4096x2160	4	4

Quadro vDWS	P4-1Q	Designer Power User	1024	2	4096x2160	8	8
GRID Virtual PC	P4-2B	Power User	2048	2	4096x2160	4	4
GRID Virtual PC	P4-2B4	Power User	2048	4	2560x1600	4	4
GRID Virtual PC	P4-1B	Power User	1024	4	2560x1600	8	8
GRID Virtual Apps	P4-8A	Virtual Application User	8192	1	1280x1024	1	1
GRID Virtual Apps	P4-4A	Virtual Application User	4096	1	1280x1024	2	2
GRID Virtual Apps	P4-2A	Virtual Application User	2048	1	1280x1024	4	4
GRID Virtual Apps	P4-1A	Virtual Application User	1024	1	1280x1024	8	8
Tesla V100 SXM2 16GB – Physical GPUs per Board: 1							
Required Software Edition	**Virtual GPU Profile**	**Intended Use Case**	**Frame Buffer (Mbytes)**	**Virtual Display Heads**	**Max Resolution per Display Head**	**Maximum vGPUs per GPU**	**Maximum vGPU per Boards**

Quadro vDWS	V100X-16Q	Designer	16384	4	4096x2160	1	1
Quadro vDWS	V100X-8Q	Designer	8192	4	4096x2160	2	2
Quadro vDWS	V100X-4Q	Designer	4096	4	4096x2160	4	4
Quadro vDWS	V100X-2Q	Designer	2048	4	4096x2160	8	8
Quadro vDWS	V100X-1Q	Designer Power User	1024	2	4096x2160	16	16
GRID Virtual PC	V100X-2B	Power User	2048	2	4096x2160	8	8
GRID Virtual PC	V100X-2B4	Power User	2048	4	2560x1600	8	8
GRID Virtual PC	V100X-1B	Power User	1024	4	2560x1600	16	16
GRID Virtual Apps	V100X-16A	Virtual Application User	16384	1	1280x1024	1	1
GRID Virtual Apps	V100X-8A	Virtual Application User	8192	1	1280x1024	2	2
GRID Virtual Apps	V100X-4A	Virtual Application User	4096	1	1280x1024	4	4

GRID Virtual Apps	V100X-2A	Virtual Application User	2048	1	1280x1024	8	8
GRID Virtual Apps	V100X-1A	Virtual Application User	1024	1	1280x1024	16	16
Tesla V100 SXM2 32GB – Physical GPUs per Board: 1							
Required Software Edition	**Virtual GPU Profile**	**Intended Use Case**	**Frame Buffer (Mbytes)**	**Virtual Display Heads**	**Max Resolution per Display Head**	**Maximum vGPUs per GPU**	**Maximum vGPU per Boards**
Quadro vDWS	V100DX-32Q	Designer	32768	4	4096x2160	1	1
Quadro vDWS	V100DX-16Q	Designer	16384	4	4096x2160	2	2
Quadro vDWS	V100DX-8Q	Designer	8192	4	4096x2160	4	4
Quadro vDWS	V100DX-4Q	Designer	4096	4	4096x2160	8	8
Quadro vDWS	V100DX-2Q	Designer	2048	4	4096x2160	16	16
Quadro vDWS	V100DX-1Q	Designer Power User	1024	2	4096x2160	32	32
GRID Virtual PC	V100DX-2B	Power User	2048	2	4096x2160	16	16

GRID Virtual PC	V100DX-2B4	Power User	2048	4	2560x1600	16	16
GRID Virtual PC	V100DX-1B	Power User	1024	4	2560x1600	32	32
GRID Virtual Apps	V100DX-32A	Virtual Application User	32768	1	1280x1024	1	1
GRID Virtual Apps	V100DX-16A	Virtual Application User	16384	1	1280x1024	2	2
GRID Virtual Apps	V100DX-8A	Virtual Application User	8192	1	1280x1024	4	4
GRID Virtual Apps	V100DX-4A	Virtual Application User	4096	1	1280x1024	8	8
GRID Virtual Apps	V100DX-2A	Virtual Application User	2048	1	1280x1024	16	16
GRID Virtual Apps	V100DX-1A	Virtual Application User	1024	1	1280x1024	32	32
Tesla V100 PCIe 16GB – Physical GPUs per Board: 1							
Required Software Edition	**Virtual GPU Profile**	**Intended Use Case**	**Frame Buffer (Mbytes)**	**Virtual Display Heads**	**Max Resolution per Display Head**	**Maximum vGPUs per GPU**	**Maximum vGPU per Boards**

Quadro vDWS	V100-16Q	Designer	16384	4	4096x2160	1	1
Quadro vDWS	V100-8Q	Designer	8192	4	4096x2160	2	2
Quadro vDWS	V100-4Q	Designer	4096	4	4096x2160	4	4
Quadro vDWS	V100-2Q	Designer	2048	4	4096x2160	8	8
Quadro vDWS	V100-1Q	Designer Power User	1024	2	4096x2160	16	16
GRID Virtual PC	V100-2B	Power User	2048	2	4096x2160	8	8
GRID Virtual PC	V100-2B4	Power User	2048	4	2560x1600	8	8
GRID Virtual PC	V100-1B	Power User	1024	4	2560x1600	16	16
GRID Virtual Apps	V100-16A	Virtual Application User	16384	1	1280x1024	1	1
GRID Virtual Apps	V100-8A	Virtual Application User	8192	1	1280x1024	2	2
GRID Virtual Apps	V100-4A	Virtual Application User	4096	1	1280x1024	4	4

GRID Virtual Apps	V100-2A	Virtual Application User	2048	1	1280x1024	8	8
GRID Virtual Apps	V100-1A	Virtual Application User	1024	1	1280x1024	16	16
Tesla V100 PCIe 32GB – Physical GPUs per Board: 1							
Required Software Edition	**Virtual GPU Profile**	**Intended Use Case**	**Frame Buffer (Mbytes)**	**Virtual Display Heads**	**Max Resolution per Display Head**	**Maximum vGPUs per GPU**	**Maximum vGPU per Boards**
Quadro vDWS	V100D-32Q	Designer	32768	4	4096x2160	1	1
Quadro vDWS	V100D-16Q	Designer	16384	4	4096x2160	2	2
Quadro vDWS	V100D-8Q	Designer	8192	4	4096x2160	4	4
Quadro vDWS	V100D-4Q	Designer	4096	4	4096x2160	8	8
Quadro vDWS	V100D-2Q	Designer	2048	4	4096x2160	16	16
Quadro vDWS	V100D-1Q	Designer Power User	1024	2	4096x2160	32	32
GRID Virtual PC	V100D-2B	Power User	2048	2	4096x2160	16	16

GRID Virtual PC	V100D-2B4	Power User	2048	4	2560x1600	16	16
GRID Virtual PC	V100D-1B	Power User	1024	4	2560x1600	32	32
GRID Virtual Apps	V100D-32A	Virtual Application User	32768	1	1280x1024	1	1
GRID Virtual Apps	V100D-16A	Virtual Application User	16384	1	1280x1024	2	2
GRID Virtual Apps	V100D-8A	Virtual Application User	8192	1	1280x1024	4	4
GRID Virtual Apps	V100D-4A	Virtual Application User	4096	1	1280x1024	8	8
GRID Virtual Apps	V100D-2A	Virtual Application User	2048	1	1280x1024	16	16
GRID Virtual Apps	V100D-1A	Virtual Application User	1024	1	1280x1024	32	32
Tesla V100 FHHL 16GB – Physical GPUs per Board: 1							
Required Software Edition	**Virtual GPU Profile**	**Intended Use Case**	**Frame Buffer (Mbytes)**	**Virtual Display Heads**	**Max Resolution per Display Head**	**Maximum vGPUs per GPU**	**Maximum vGPU per Boards**

Quadro vDWS	V100L-16Q	Designer	16384	4	4096x2160	1	1
Quadro vDWS	V100L-8Q	Designer	8192	4	4096x2160	2	2
Quadro vDWS	V100L-4Q	Designer	4096	4	4096x2160	4	4
Quadro vDWS	V100L-2Q	Designer	2048	4	4096x2160	8	8
Quadro vDWS	V100L-1Q	Designer Power User	1024	2	4096x2160	16	16
GRID Virtual PC	V100L-2B	Power User	2048	2	4096x2160	8	8
GRID Virtual PC	V100L-2B4	Power User	2048	4	2560x1600	8	8
GRID Virtual PC	V100L-1B	Power User	1024	4	2560x1600	16	16
GRID Virtual Apps	V100L-16A	Virtual Application User	16384	1	1280x1024	1	1
GRID Virtual Apps	V100L-8A	Virtual Application User	8192	1	1280x1024	2	2
GRID Virtual Apps	V100L-4A	Virtual Application User	4096	1	1280x1024	4	4

GRID Virtual Apps	V100L-2A	Virtual Application User	2048	1	1280x1024	8	8
GRID Virtual Apps	V100L-1A	Virtual Application User	1024	1	1280x1024	16	16

7.3 BRINGING QUADRO TO VIRTUALIZATION

Like the NVIDIA Quadro line cards for professional graphics applications, the vGPU profiles ending in **Q** undergo the same rigorous application certification process as those Quadro-class processors. As a result, you can expect 100% compatibility and performance with your applications, and ISVs requiring Quadro can certify against the NVIDIA vGPU profiles as well.

7.4 MATCHING PROFILES TO USER NEEDS

As stated earlier in the deployment guide, you need to define your user needs and match them to available NVIDIA vGPU profiles.

If we take the three example SOLIDWORKS use cases and their computing needs defined earlier, we can now match the video RAM and display needs to require vGPU profiles. See below:

- Entry Level Engineer / Design Reviewer
  - 8 GB RAM
  - 4 vCPUs (2.4 GHz)
  - 2-4 GB video RAM
  - Single 1920x1080 display
- Mid-Level Engineer
  - 16 GB RAM
  - 4-8 vCPUs (2.6 GHz)

- 4-8 GB video RAM
- Two 1920x1080 displays and/or One 2560x1600

- Advanced Engineer
  - 32-64 GB RAM
  - 8-12 vCPUs (3.2 GHz)
  - 8-16 GB video RAM
  - Two 2560x1600 displays and/or One 3840 x 2160

7.5 CREATIVE AND TECHNICAL PROFESSIONALS

Creative and Technical Professionals with the most demanding of workflows previously required a physical computing device like a mobile or desktop workstation. This meant that the benefits of virtualization, such as security, ease of management, mobility and collaboration could not be achieved. With the NVIDIA Quadro Virtual Data Center Workstation (Quadro vDWS) software, coupled with Tesla data center GPUs, the benefits of virtualization can be extended to these users.

The key to picking the proper profile for the creative and technical professionals is compatibility and performance. Quadro vDWS software supports Quadro drivers allowing users to benefit from the acceleration and stability that Quadro brings to professional applications used by the most demanding customers today.

It's critical to understand what applications your users will require on their virtual machines. It's necessary to balance frame buffer and resource needs to manage the density and performance for your deployment. Professional applications have a range of workflows from design, to rendering, to video editing all with varying degrees of resource demands that must be estimated to properly ensure the best experience.

Start your profile selection by looking to your primary application's requirements. Professional application software vendors are certified to run with NVIDIA Quadro to ensure the performance is tuned for maximum efficiency. Each ISV usually has a dedicated page on proper GPU hardware to use, so using those recommendations, select the appropriate vGPU profile to meet your end user's needs. To understand more about the graphics requirements of your applications, consult your application vendor.

7.6 KNOWLEDGE WORKER PROFILES

Today's digital workplace is changing due to the increased graphics requirements for Windows 10 and other productivity applications. The new digital worker expects a graphics-rich experience with immersive visual quality and processing speed. To satisfy these expectations, software developers are creating rich, graphically intensive user experiences for physical desktops in the enterprise. In a virtual desktop (VDI) environment without a GPU to offload these additional processing requirements, the end user sees slower performance, reduced feature sets, and applications that simply won't launch.

Applications that are accelerated by graphics have doubled over the past 5 years according to a study by Lakeside SysTrack Community data. Below are some examples and the percentage of GPU demands based on the transition from a Windows 7 environment moving to a Windows 8 environment.

- Google Chrome – 36% increase
- Mozilla Firefoz – 59% increase
- Microsoft Outlook – 85% increase
- Microsoft Powerpoint – 64% increase
- Microsoft Excel – 53% increase
- Skype – 409% increase

While professional applications like Dassault Systèmes SOLIDWORKS and CATIA and others are logically required to have a GPU for graphics acceleration, it's also clear that common business applications now need GPU resources to run efficiently.

Chapter 8.
APPLICATION BASED VDI HOST SIZING

8.1 HOW MANY USERS CAN I GET ON A HOST?

Using one application, Siemens NX, a typical dialogue with a customer often sounds like this…

Customer: How many users can I get on a host?

NVIDIA: What is their primary application?

Customer: Siemens NX.

NVIDIA: Are they primarily 3D or 2D data users?

Customer: 3D mostly.

NVIDIA: Would you describe them as light, medium, or heavy users?

Customer: Medium to heavy.

NVIDIA: Power users to designers then.

Customer: I need performance AND scalability numbers that I can use to justify the project…

8.2 AN EXAMPLE APPLICATION: SIEMENS NX

Siemens NX™ is a popular industrywide solution for integrated 3D computer-aided design, manufacturing and engineering analysis (CAD/CAM/CAE) that helps companies realize the value of the digital twin, which is now available in a cost-effective private cloud environment. Siemens worked closely with NVIDIA Corporation to certify the deployment of NX in the private cloud using a virtual desktop infrastructure (VDI) with NVIDIA virtual GPU (vGPU) technology. Siemens NX users are split into the following categories with a mapping of potential NVIDIA vGPU profiles associated with each:

	Light User	Medium User	Heavy User
Siemens NX Usage Classification	Light 2D to Medium 2D	Heavy 2D to Light 3D	Medium 3D to Heavy 3D
Guaranteed Performance	12 Users Per Server P4-4Q 4vCPU 8GB RAM	6 Users Per Server P4-8Q 8vCPU 8GB RAM	3 Users Per Server P40-24Q 12vCPU 8GB RAM2
Typical Deployment	16-24 Users Per Server P4-2Q 4vCPU 8-16GB RAM	12-18 Users Per Server P4-2Q / P4-4Q 8vCPU 16-32GB RAM	6-9 Users Per Server P40-8Q / P40-12Q 12vCPU >96GB RAM

You'll notice that there are two rows of configurations. This is because it's CRITICAL to ensure that the customer understands and deploys against either a dedicated level of performance to the user or against a typical deployment which may have a reduced performance at CERTAIN TIMES but not always. If a guaranteed level of performance is critical, then map accordingly.

8.3 APPLICATION BASED SIZING

This section contains an overview of the testing, the methodology, and the results that support the findings in this deployment guide. We also detail the lab environment used to test the contents of this guide.

First, we look at the specifications of a physical workstation build.

8.3.1 Recommended System Requirements

CPU speed	Minimum: Hyperthreaded dual core* Recommended: Quad core* Optimal: 2x hyperthreaded hexa-core*
Platform	x64 with SSE2 extensions
Memory/RAM	Minimum: 4 GB Recommended: 16 GB Optimal: 32 GB
Display properties	24-bit color depth
GPU	Good: NVIDIA Quadro P1000 Better: NVIDIA Quadro P2000 Best: NVIDIA Quadro P5000
Disk space	Recommended 1TB Storage Hard Drive

For the test we key on recommended specifications when feasible. The goal is to test both performance and scalability, to maintain the flexibility and manageability advantages of virtualization without sacrificing the performance end users expect from NVIDIA powered graphics.

8.4 TESTING METHODOLOGY

The Professional Visualization Performance Engineering Team at NVIDIA uses a custom designed benchmark engine to conduct vGPU testing at scale. This engine automates the testing process from provisioning virtual machines, establishing remote connections, executing SPECviewperf and analyzing the results across all virtual machines. Dedicated performance scores mentioned in this deployment guide are based on SPECviewperf 12.1.1 which was run in parallel on all virtual machines and scores are averaged across three runs. Below you will find the VM Configuration used:

VM CONFIGURATION	
Operating System	Windows 10 (1703)
vCPUs	8
vMemory	16GB
Hard Disk	100GB
vGPU Driver Version	NVIDIA Virtual GPU Software 6.0 (391.03)
vGPU Software Edition	Quadro vDWS
vSYNC	Default
Frame Rate Limiter	Default (Off)
VDA Version	7.2
Direct Connect Version	7.2
Number of Screens	1
Screen Resolution	1920 x 1080
HYPERVISOR CONFIGURATION	

Hypervisor	VMware vSphere 6.5.0
Remote Stack	VMware Horizon 7 with PCoIP
Remote Stack Version	7.2
VM Version	Vmx-13
VM Tools	10279
GPU Allocation Policy	Depth-First
vGPU Manager Version	NVIDIA Virtual GPU 6.0 Software (390.42)

Proof of Concepts (POC) are more likely to be successful when end users are classified into different groups as outlined above as it allows standardization for optimal performance while keeping management overhead minimal. Customers often segment their end users into three different user groups for each application and bundle similar user types per host. The below table outlines the most common user groups for Siemens NX along with recommended configurations.

Light Users	Guaranteed Performance Level	Users per Host	vCPUs	vGPU Profile	vMemory	CPUs	GPUs	Memory	Storage Type	Networking
Guaranteed Performance	Quadro P1000	12	8	P4-4Q	8GB	2x Intel Xeon Gold 6154	6x Tesla P4	128GB	Flash Based	10Gbit
Typical Customer Deployment	-	16-24	4	P4-2Q	12-16GB	2x Intel Xeon Gold 6154	6x Tesla P4	384GB	Flash Based	10Gbit

Medium Users	Guaranteed Performance Level	Users per Host	vCPUs	vGPU Profile	vMemory	CPUs	GPUs	Memory	Storage Type	Networking
Guaranteed Performance	Quadro P2000	6	8	P4-8Q	16GB	2x Intel Xeon Gold 6154	6x Tesla P4	256GB	Flash Based	10Gbit
Typical Customer Deployment	-	12-18	4	P4-4Q	16-32GB	2x Intel Xeon Gold 6154	6x Tesla P4	384-512GB	Flash Based	10Gbit

Heavy Users	Guaranteed Performance Level	Users per Host	vCPUs	vGPU Profile	vMemory	CPUs	GPUs	Memory	Storage Type	Networking
Guaranteed Performance	Quadro P5000	3	12	P40-24Q	8GB	2x Intel Xeon Gold 6154	3x Tesla P40	128GB	Flash Based	10Gbit
Typical Customer Deployment	-	6-9	8-12	P40-8Q, P40-12Q	>96GB	2x Intel Xeon Gold 6154	3x Tesla P40	512GB-1TB	Flash Based	10Gbit

NVIDIA recommends configuring your dual socket server of choice using the Intel Xeon Gold 6154 CPU. The high frequency of 3.0 GHz alongside 18 cores of the CPU are well suited for optimal performance for each end user while supporting the highest user scale on the server which makes it also a cost-effective solution for Siemens NX.

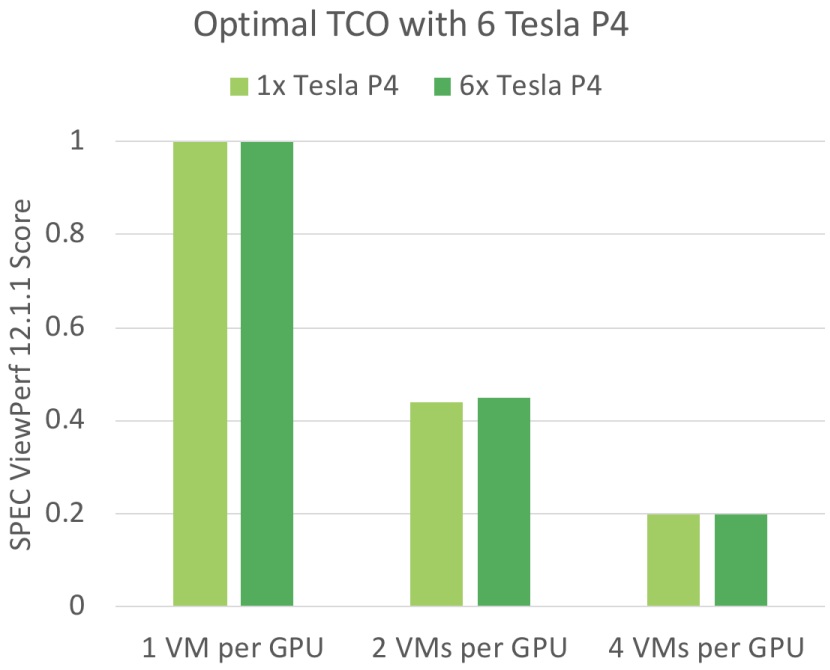
Based on SPECviewperf 12.1.1, the below section outlines what typical customers usually configure based on various needs.

Tesla P4 with Quadro vDWS for Light and Medium Users

Quadro vDWS in combination with Tesla P4 is NVIDIA's recommended virtualization solution for Siemens NX.

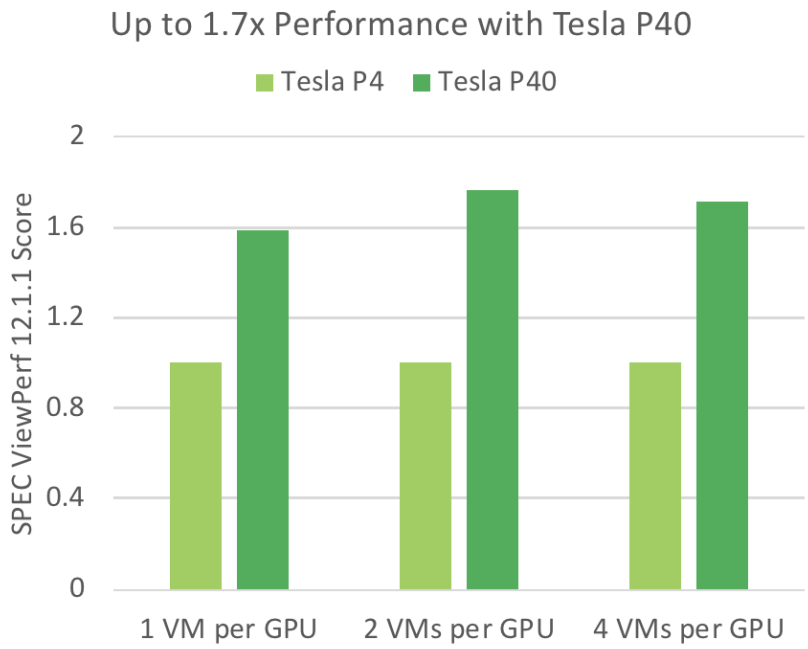
The form factor of the Tesla P4 (single width, half height and powered through the PCIe bus) allows the use of up to six Tesla P4s in a server with two Intel Xeon Gold 6154 CPUs. SPECviewperf 12.1.1 tests show that six Tesla P4 GPUs alongside two Intel Xeon Gold 6154 CPUs is a well-balanced configuration for Siemens NX.

The below bar graph demonstrates that there are enough CPU resources available to host six Tesla P4s in one server running SPECviewperf 12.1.1 Siemens NX on 24 virtual machines. No performance decrease running 24 virtual machines (4 per GPU) in comparison to 4 (4 per GPU) demonstrates a great balance between CPU and GPU resources.



Tesla P40 with Quadro vDWS for Heavy Users

NVIDIA recommends the Tesla P40 in combination with Quadro vDWS for heavy users that require the additional performance of a Tesla P40 over a Tesla P4. The Tesla P40 is the optimal choice for heavy users due to the additional performance and framebuffer while the Tesla P4 is recommended for light and medium users due to the ability of hosting up to six Tesla P4s in a two-socket, 2Userver.



Sufficient System Memory for each Individual User

While the industry standard benchmark SPECviewperf 12.1.1 performs optimally with 8 GB of system memory for each virtual machine, Siemens NX customers typically assign 16 - 32 GB of system memory to medium users for optimal performance. System memory requirements don't change with the transition to Quadro vDWS, therefore the same amount of system memory that is used in a physical workstation should be assigned to the Quadro vDWS accelerated virtual machine.

Flash Based Storage for Best Performance

NVIDIA recommends the use of flash-based storage (SSDs, etc.) for optimal performance. Flash based storage is the common choice with physical workstation and Siemens NX users expect similar performance in virtual environments.

A typical configuration for non-persistent virtual machines is to use direct attached storage (DAS) of the server in a RAID 5 or RAID 10 configuration. For persistent virtual machines, a high performing all-flash storage solution is the preferred option.

Typical Networking Configuration for Quadro VDWS

There is no typical network configuration since it varies based on multiple factors like choice of hypervisor, persistent/non-persistent virtual machines, choice of storage solution and many more but most customers are using 10 Gbit networking for optimal performance.

Chapter 9. CREATING YOUR FIRST vGPU VIRTUAL DESKTOP

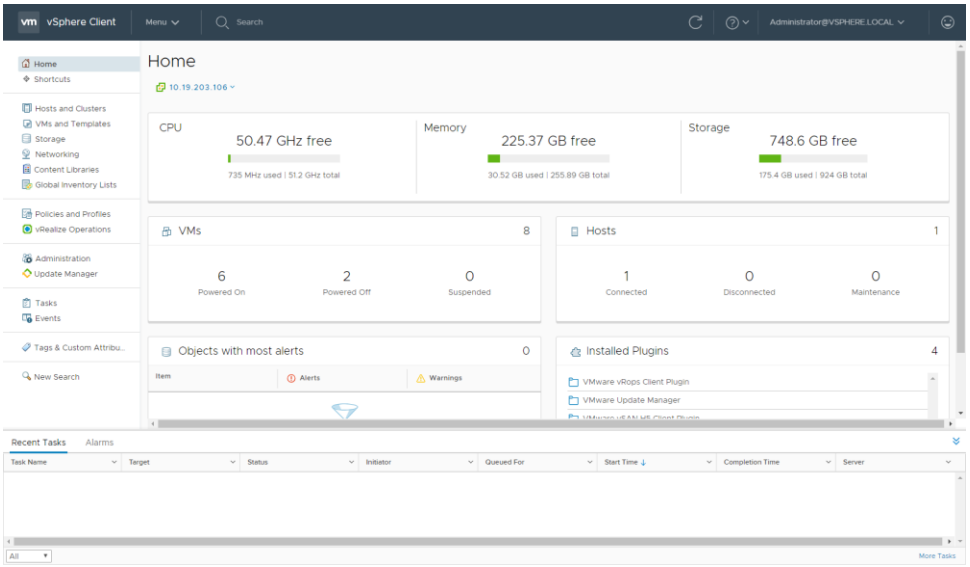
This chapter describes how to:

- Create and configure a virtual machine in vSphere
- Install Windows and VMware Tools on the VM
- Customize Windows settings
- Install Horizon Agent and Horizon Direct Connection on the VM
- Adjust some additional VM settings and enable VM console access
- Enable the NVIDIA vGPU and finalizing the installation

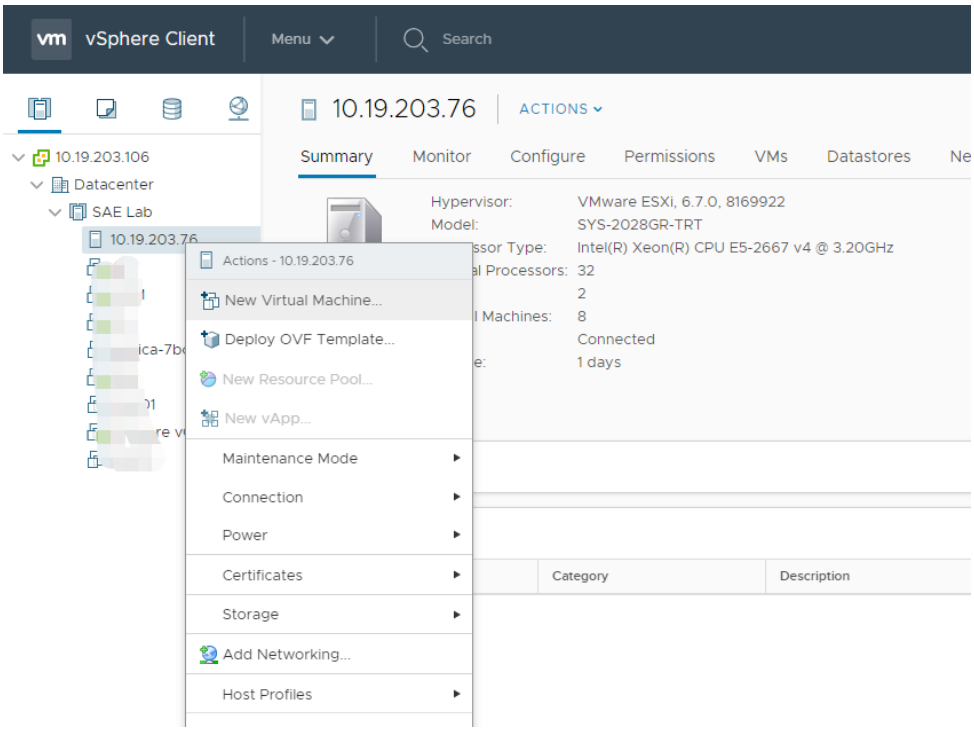
9.1 CREATING A VIRTUAL MACHINE

These instructions are to assist in making a VM from scratch that will support NVIDIA vGPU. Later the VM will be used as a gold master image. Use the following procedure to configure a vGPU for a single guest desktop:

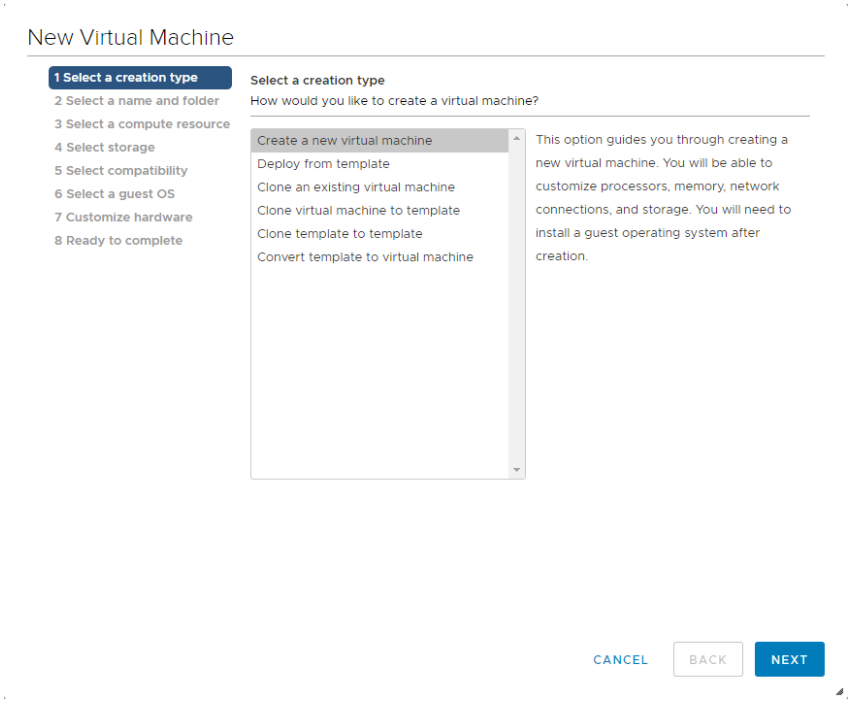
1. Browse to the host or cluster using the *vSphere Web Client*.



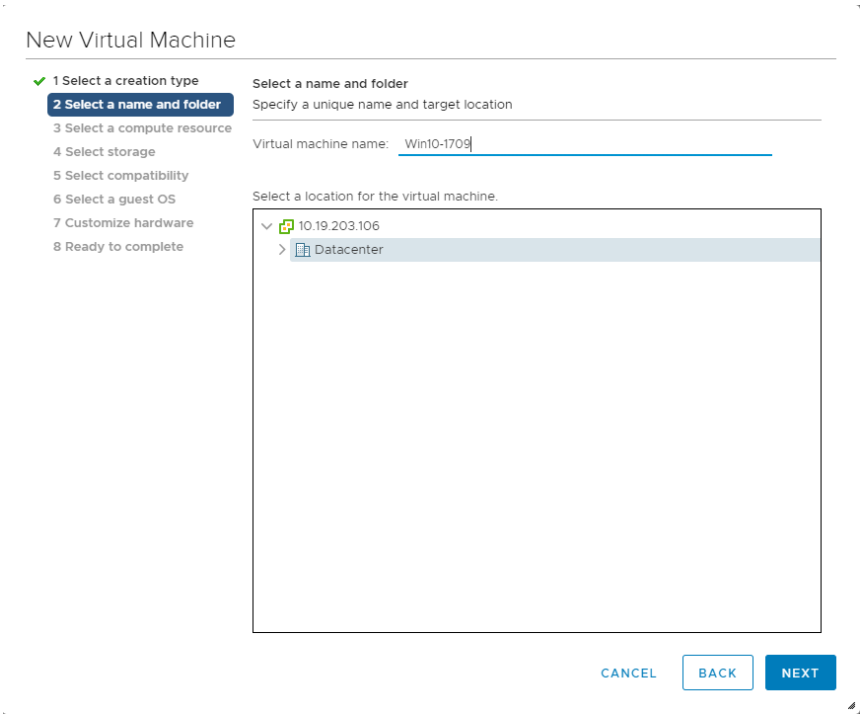
2. Right-click the desired host or cluster and select **New Virtual Machine**.
The *New Virtual Machine* wizard begins.



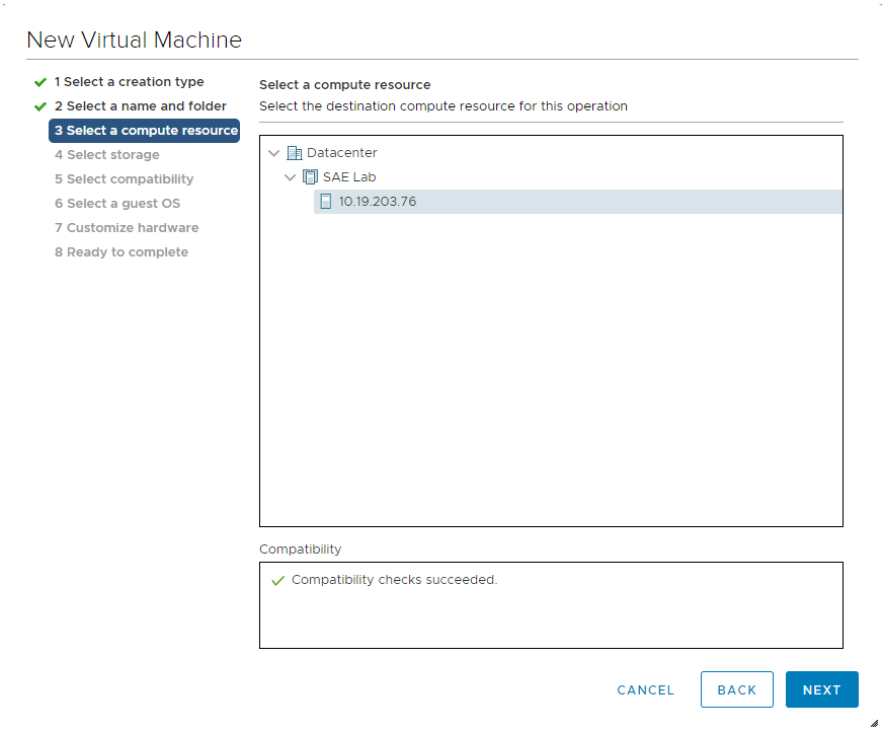
Select **Create a new virtual machine** and click **Next.**



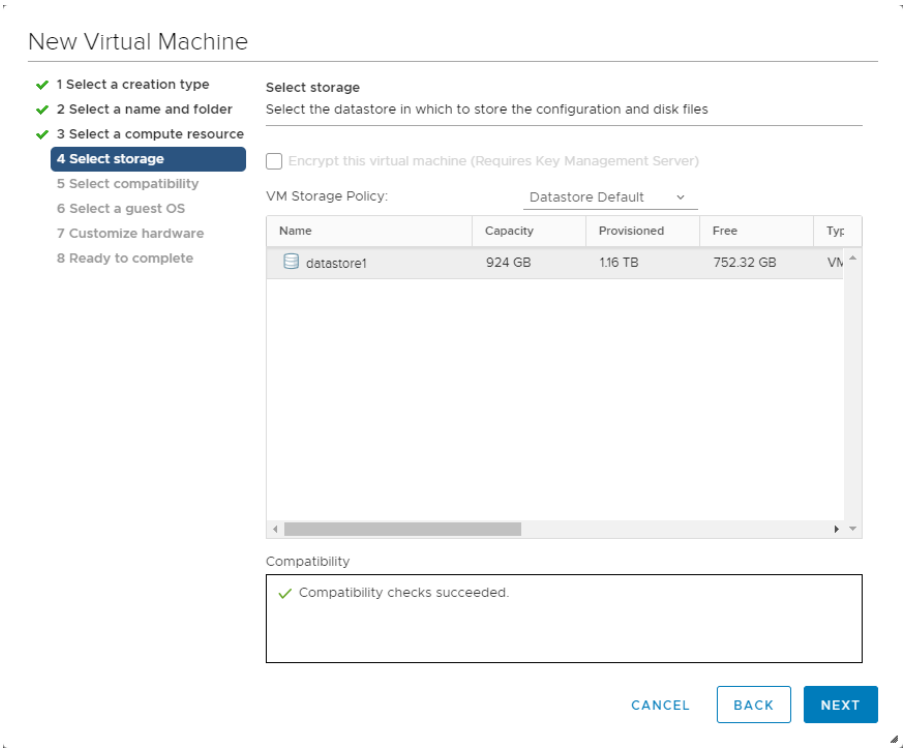
Enter a name for the virtual machine. Choose the location to host the virtual machine using the **Select a location for the virtual machine** section. Click Next to continue.



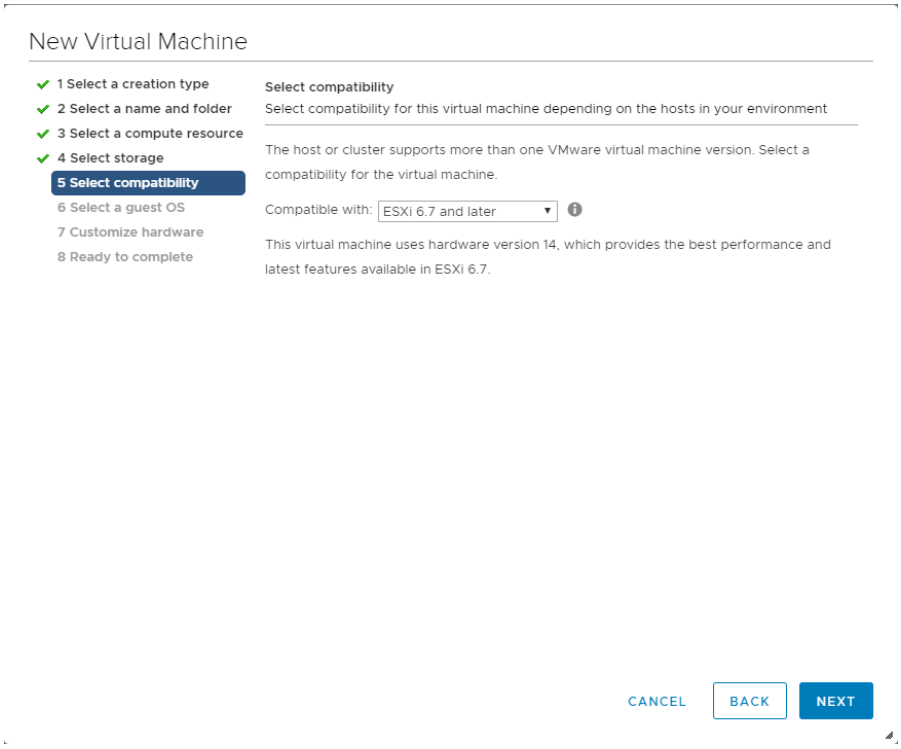
Select a compute resource to run the VM. Click Next to continue. NOTE: This compute resource should include an NVIDIA vGPU enabled Tesla card installed and be correctly configured.



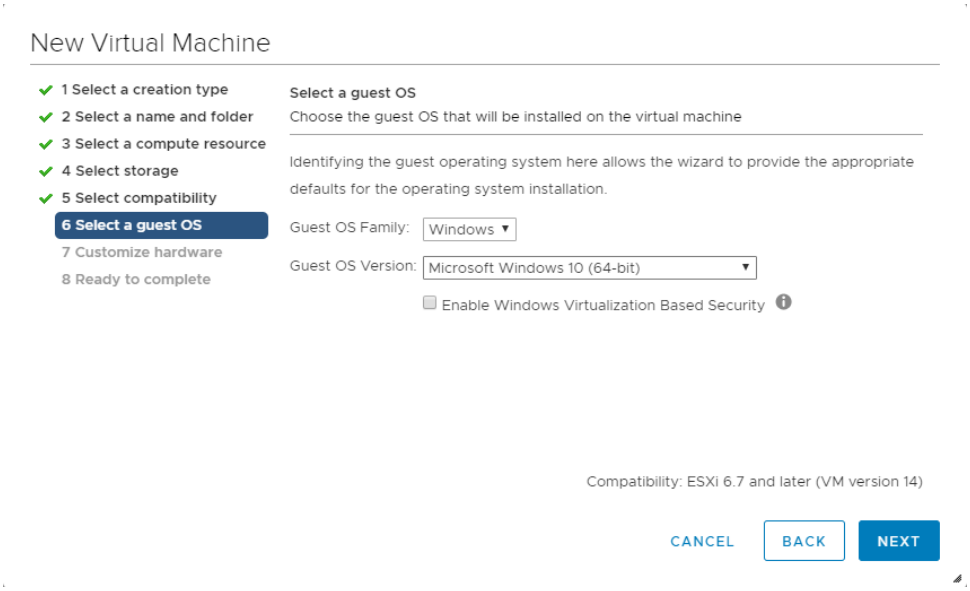
Select the datastore to host the virtual machine. Click **Next** to continue.



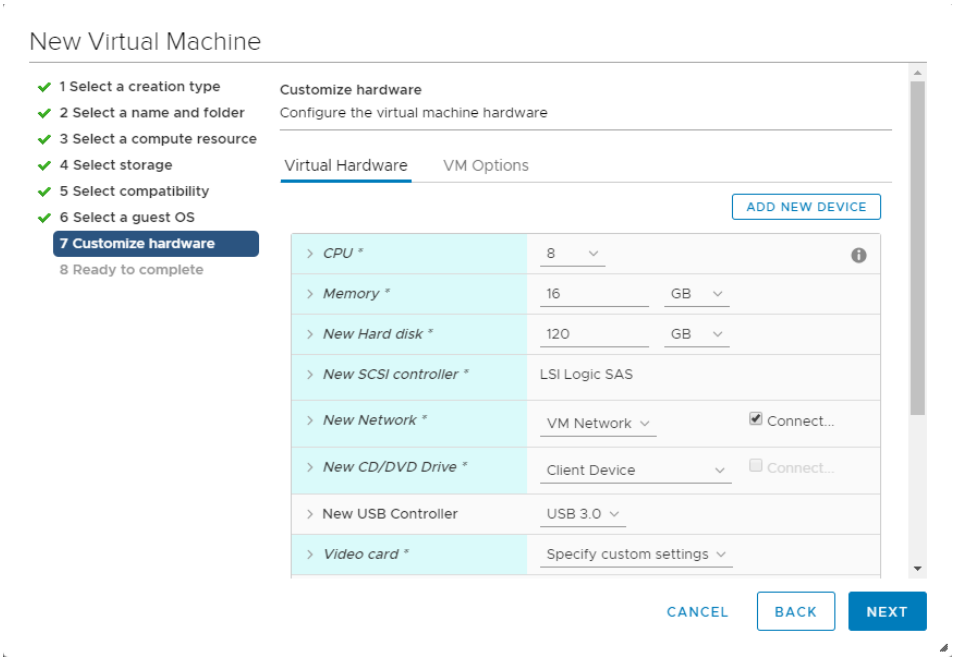
Select compatibility for the virtual machine. This allows VMs to run on different versions of vSphere. To run vGPU select ESXi 6.0 and later. Click **Next** to continue.



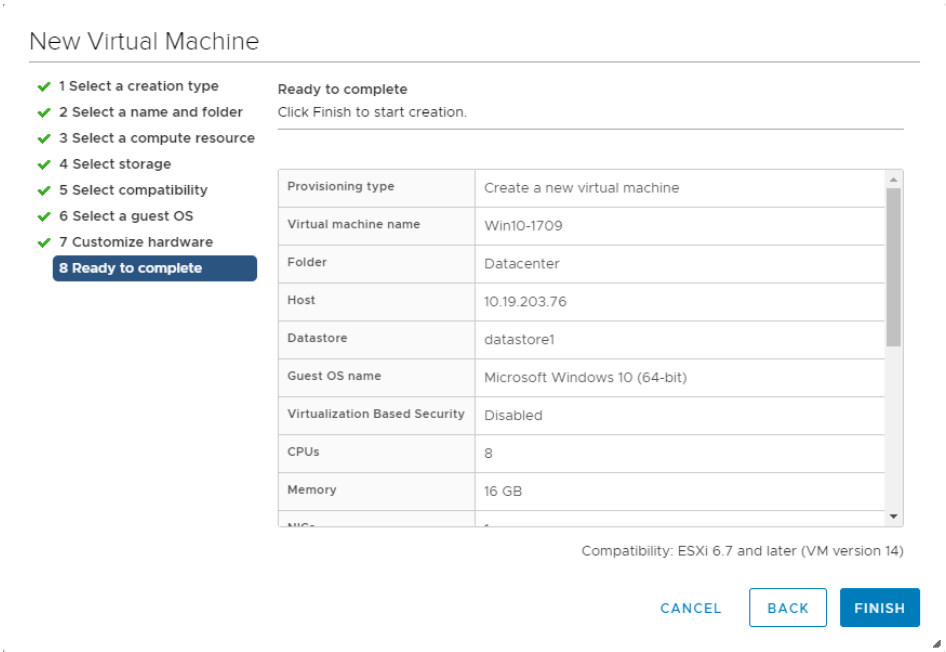
Select the appropriate Windows OS from the **Guest OS Family** and **Guest OS Version** pull-down menus. Click Next to continue.



Customize hardware is next. Set the virtual hardware based on your desktop workload requirements. Click Next to continue.



Review the New Virtual Machine configuration prior to completion. Click **Finish** when ready.

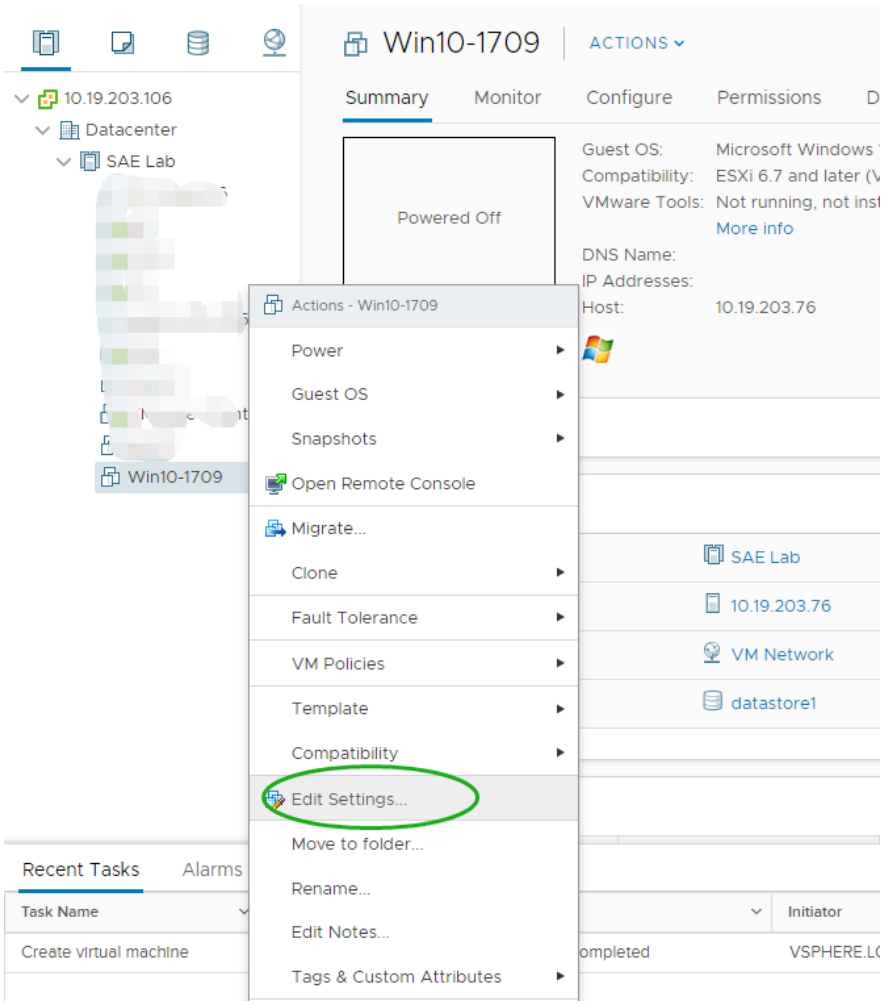


The new virtual machine container has now been created.

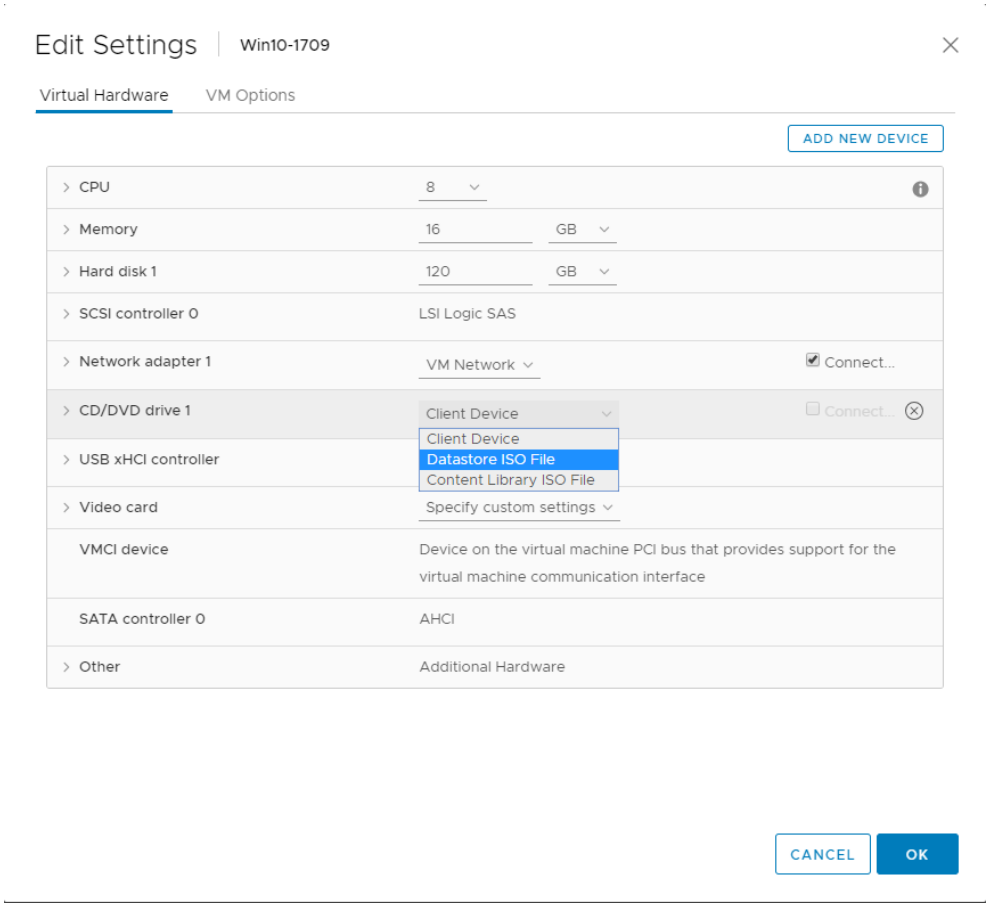
9.2 INSTALLING WINDOWS

Use the following procedure to install Windows on the virtual machine:

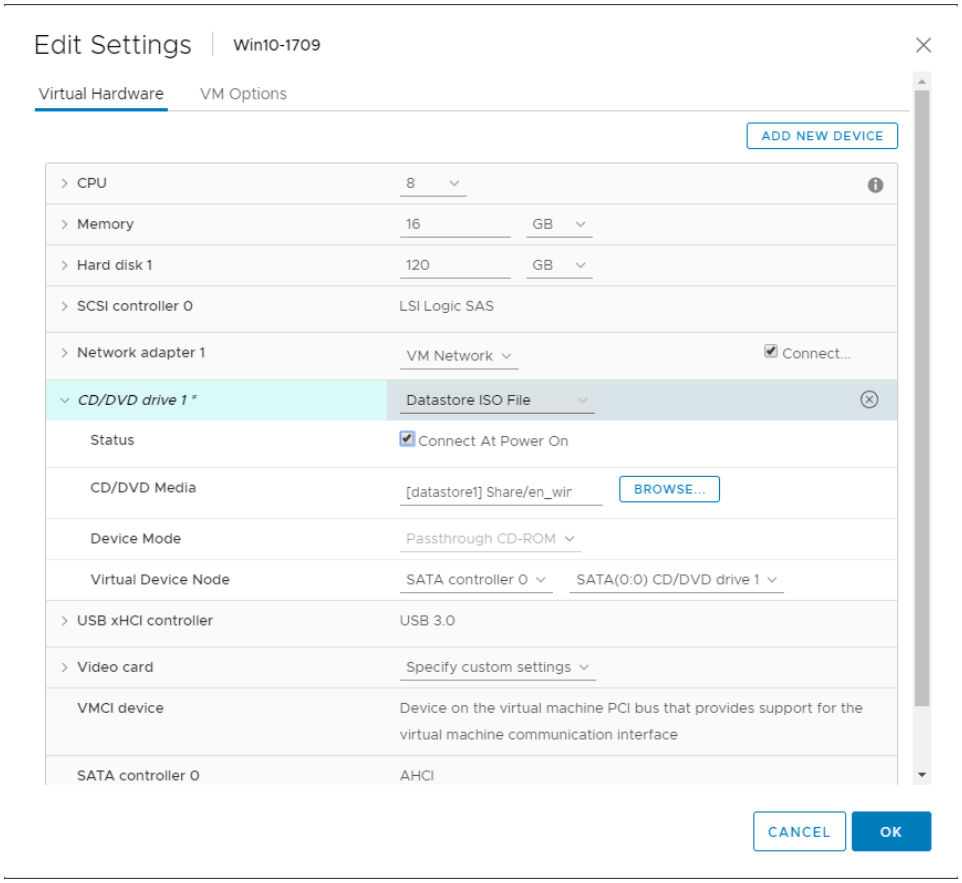
1. Select the virtual machine, right click on the virtual machine and select Edit Settings.



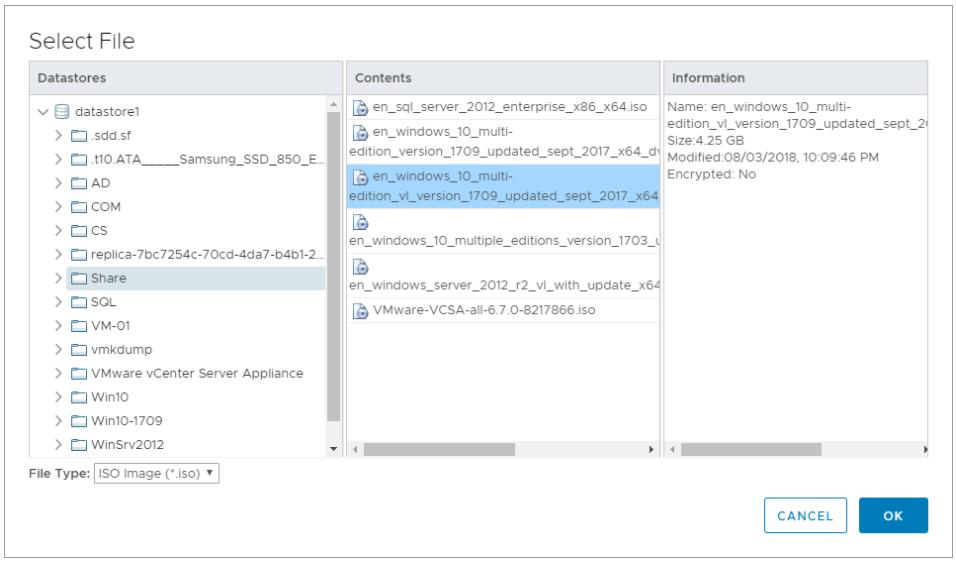
Locate the CD/DVD entry under the Virtual Hardware tab. Select the arrow drop down to reveal data sources for the CD/DVD media. (In this example a Datastore ISO file will be used.) Check the **Connect** checkbox for **CD/DVD drive 1**. This will connect the ISO file to the VMs virtual CD/DVD drive.



Toggle the carrot next to the CD/DVD drive 1 icon to reveal the details of the virtual device. For Status check the **Connect At Power On** checkbox. This will connect the ISO file to the VM's virtual CD/DVD drive during boot up. Next Click on the Browse button for the CD/DVD Media.



Navigate to and select the OS ISO file to be used for installation. Click OK to select the ISO.



Right-click the virtual machine, and then select **Power>Power On** to start the virtual machine and boot from the .ISO to install the operating system.

The virtual machine boots from the selected .ISO.

If you are creating a new virtual machine and using the vSphere Web Client's VM console functionality, then the mouse may not be usable within the virtual machine until after the both the operating system and VMware tools have been installed.

Perform a Custom (fresh) installation of Windows 10 on the virtual machine.
During installation, Windows reboots the VM several times.

Disconnect the .ISO from the VM when Windows is done installing.

Go through the initial Windows setup wizard to name the computer, create a local account, set the time-zone, choose update installation policy, etc.

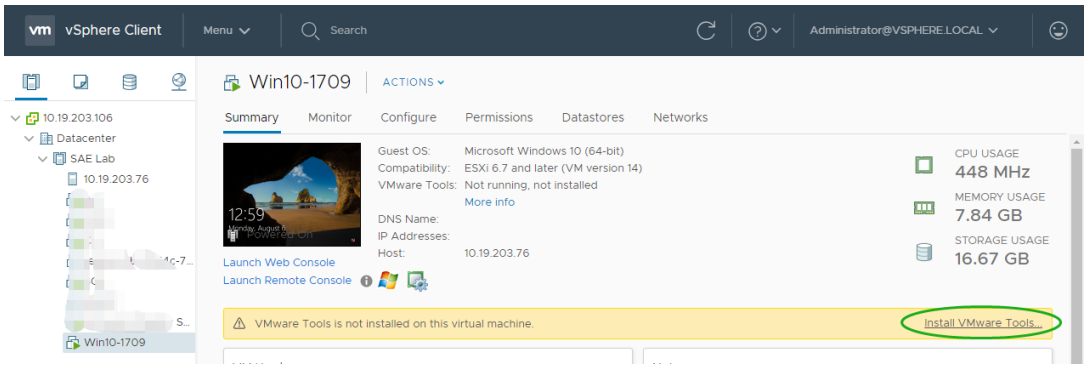
Windows 10 is now installed on the virtual machine.

9.3 INSTALLING VMWARE TOOLS ON THE VM

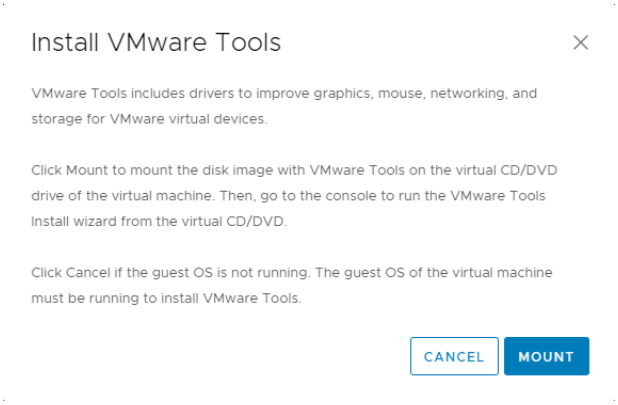
After Windows completes the initial installation and configuration process, the next step is to install VMware Tools on the virtual machine.

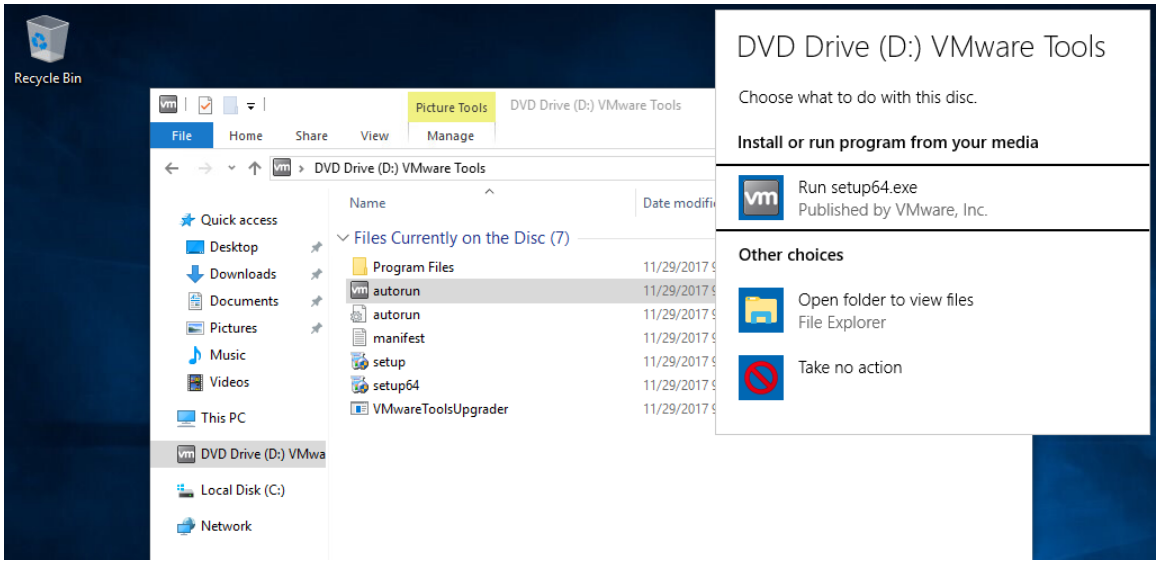
1. Select the **Summary** tab from the virtual machine console.

Click the **Install VMware Tools** link in the yellow bar.
The *Install VMware Tools* window displays.

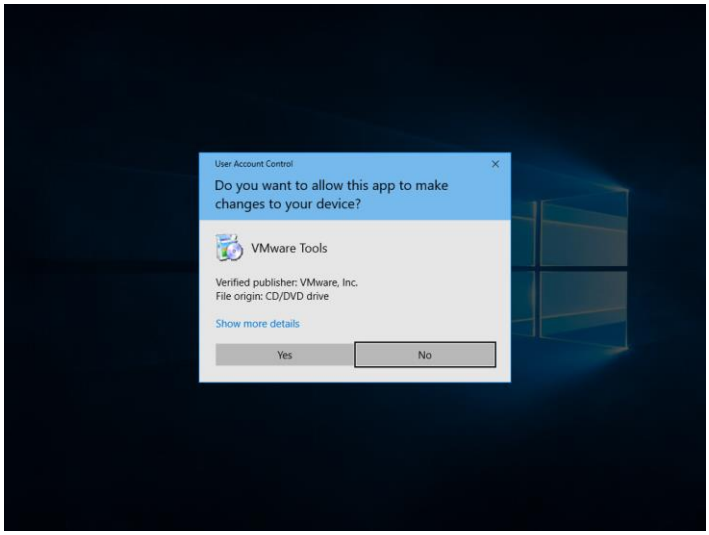


Click **Install VMware Tools**.
Back in the virtual machine console, Windows 10 detects the CD image and the *AutoPlay* window should open. If not, browse to the virtual CD-ROM in the virtual machine and access it manually.



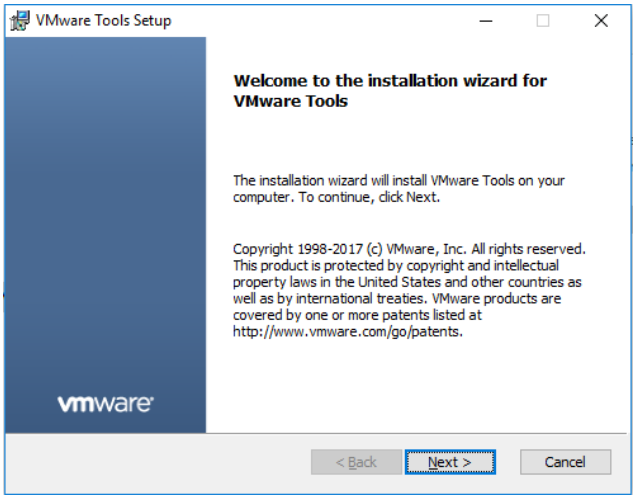


Click **Run setup.exe or setup64.exe**.
The *User Account Control* popup may display.

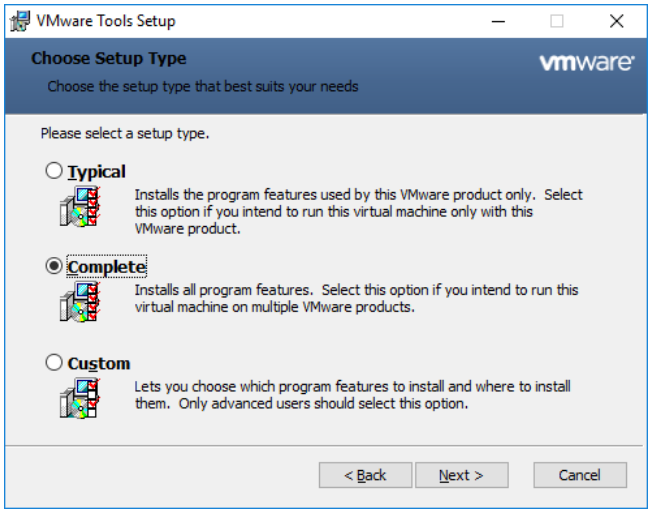


If UAC prompts, then click **Yes**.

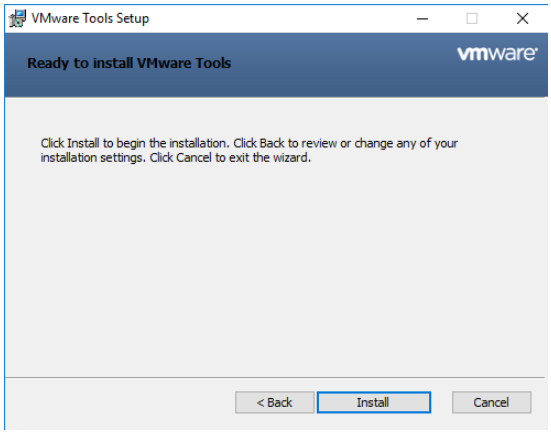
The installer begins, click **Next.**



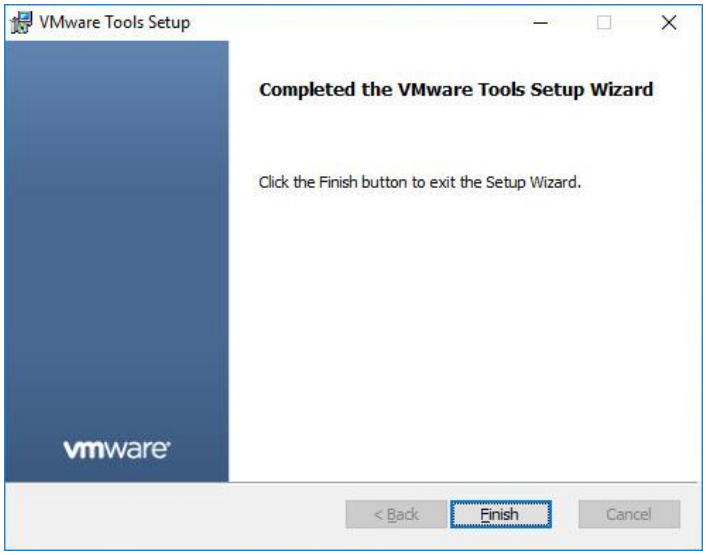
When prompted, select **Complete** installation, click **Next**, and accept all defaults.



When prompted, select **Install** to begin installation:



Click **Finish** in the VMware Tools installer and reboot the virtual machine when prompted. This reboot is critical to ensure the tools are now the loaded drivers.



VMware Tools is now installed on the virtual machine

9.4 ADDING THE GOLDEN MASTER TO THE DOMAIN

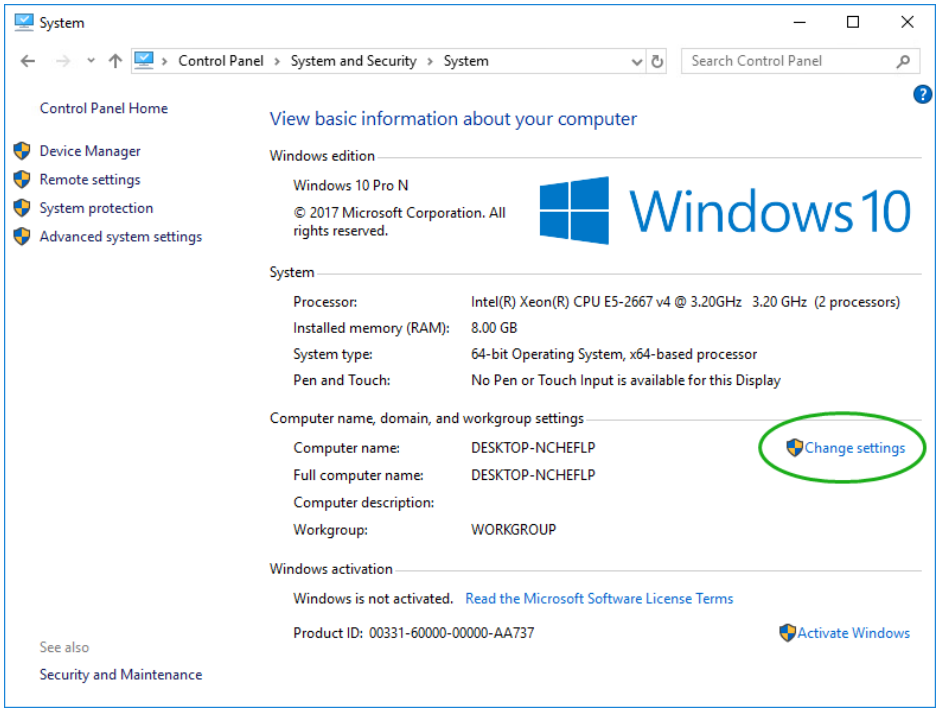
By joining the VM to the Windows Active Directory domain you are then able to manage it as you would any physical desktop in the domain.

Customize Windows on the virtual machine as follows:

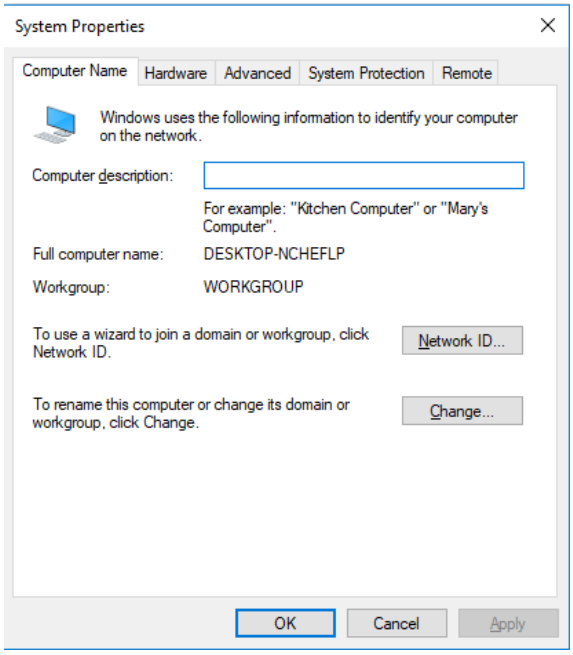
- Join the domain
- Add appropriate Domain groups to Local Administrators

Adding a VM to the domain:

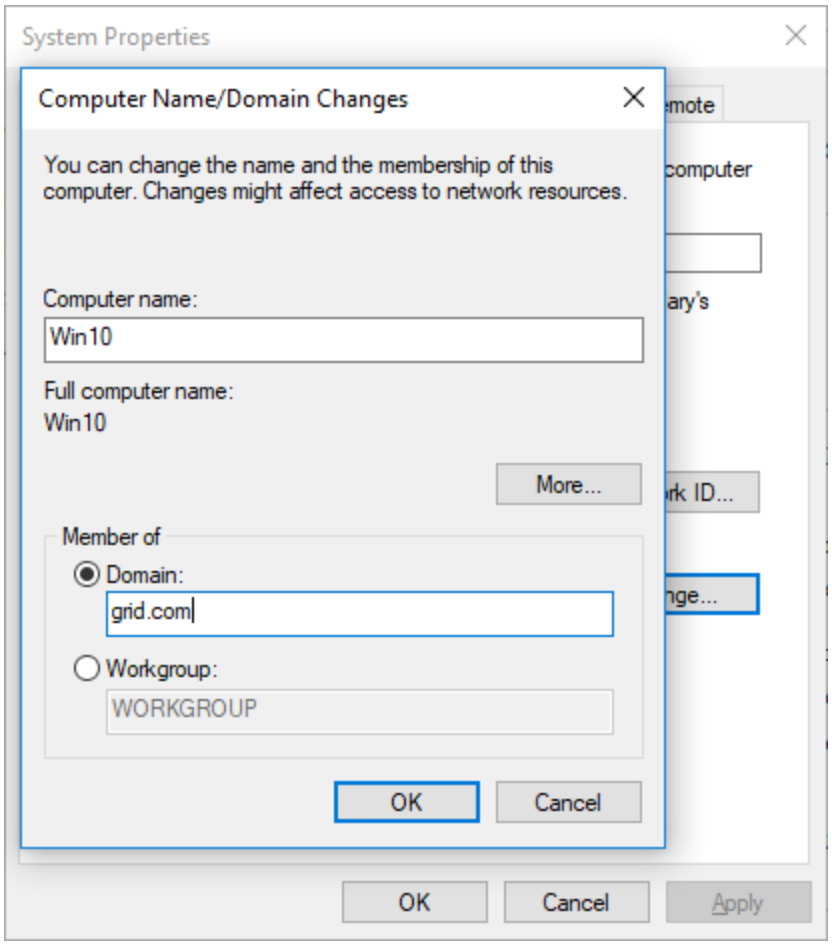
1. On the VM, go to **Control Panel**, **System and Security**, **System**



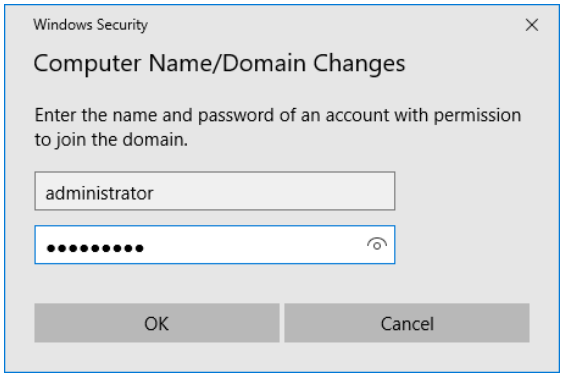
2. This brings up the System Properties window, on the Computer Name tab click Change:



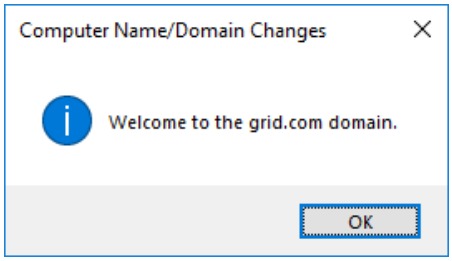
3. On the Computer Name/Domain Changes window, enter in an appropriate Computer name, then Domain name, and click OK. Our chosen naming is shown below, use what is appropriate for your POC/trial.



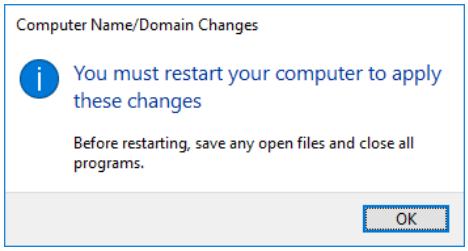
4. A security window pops up, fill in your specific domain administrator credentials and click OK:



5. On successful authentication you will see the following welcome pop-up showing your VM is now on the domain (the domain name should reflect your domain information):

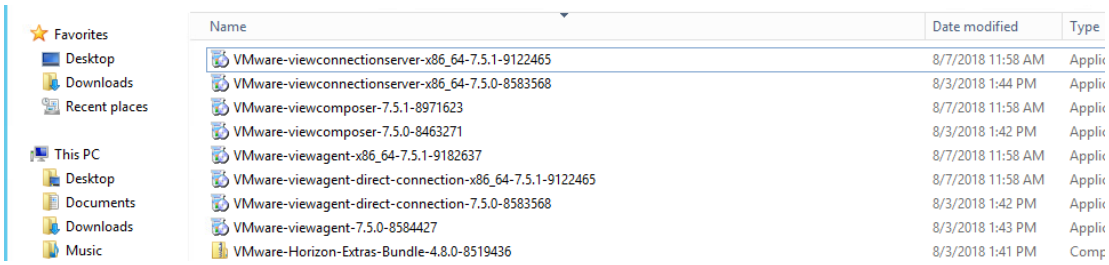


6. Click OK and the VM needs to reboot to complete the process, click OK again and the VM reboots immediately.



9.5 INSTALLING HORIZON AGENT

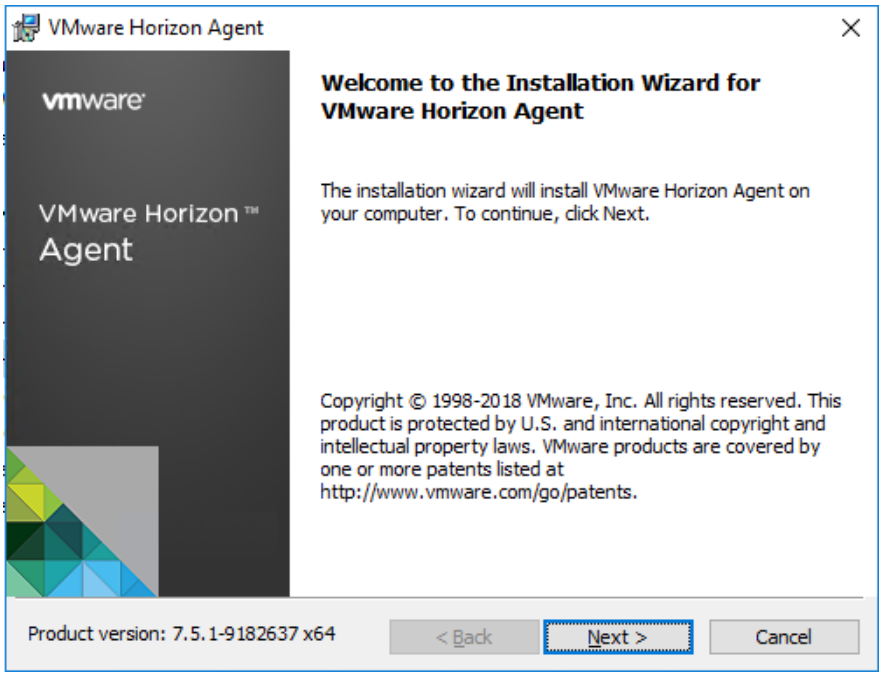
You need to install the correct version of the Horizon Agent for your virtual machine:



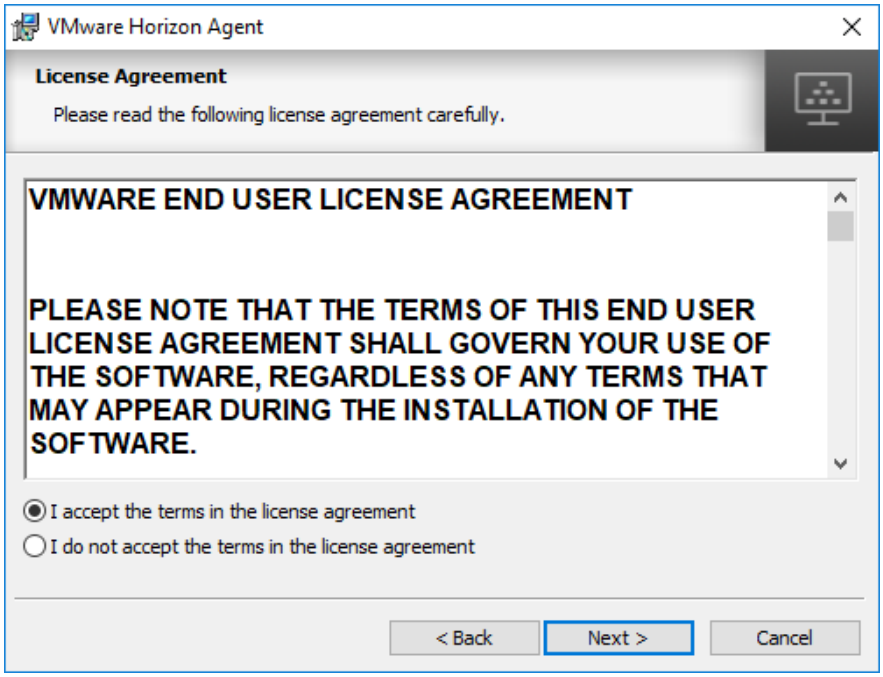
- For 32-bit virtual machines: VMware-viewagent-7.5.1-9182637.exe
- For 64-bit virtual machines: VMware-viewagent-x86_64-7.5.1-9182637.exe

Use the following procedure to install Horizon Agent on the virtual machine:

1. Launch the installer.
   The *Welcome* window displays.

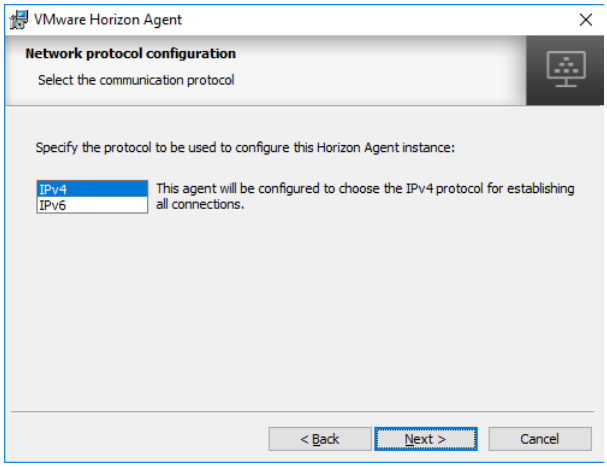


Click **Next** to proceed.
The *License Agreement* window displays.

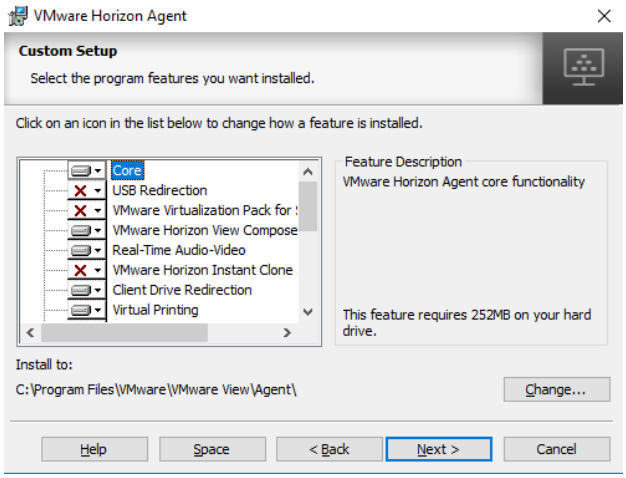


Check the **I accept the terms** in the license agreement radio button, click **Next** to proceed.

Network protocol configuration window appears, choose the appropriate network protocol, typically this is IPv4, and click **Next** to proceed:

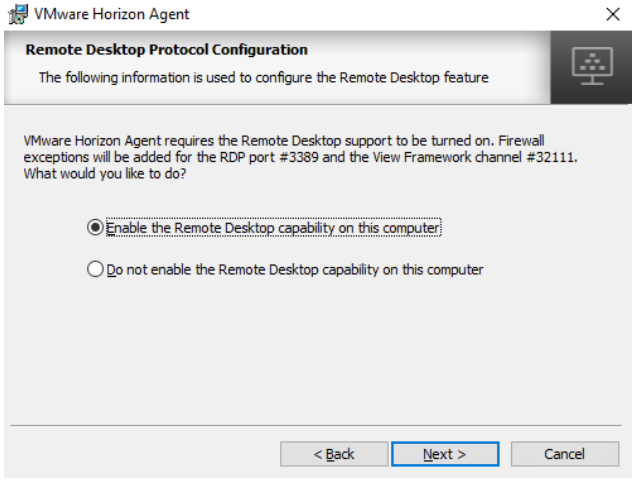


The *Custom Setup* window displays.

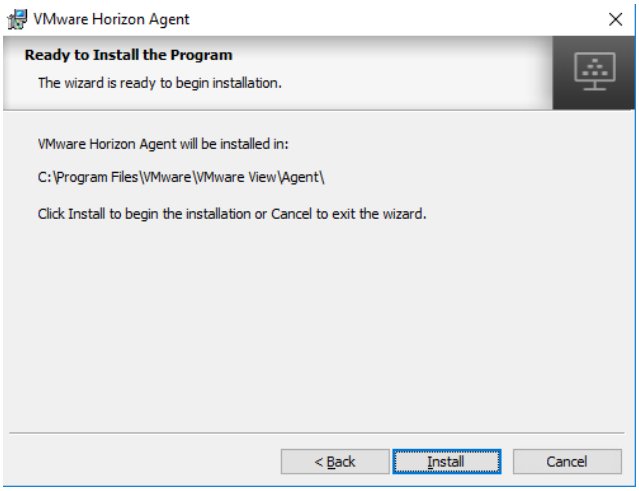


Accept all defaults values by clicking **Next**.

Enable remote desktop capability if prompted to do so.

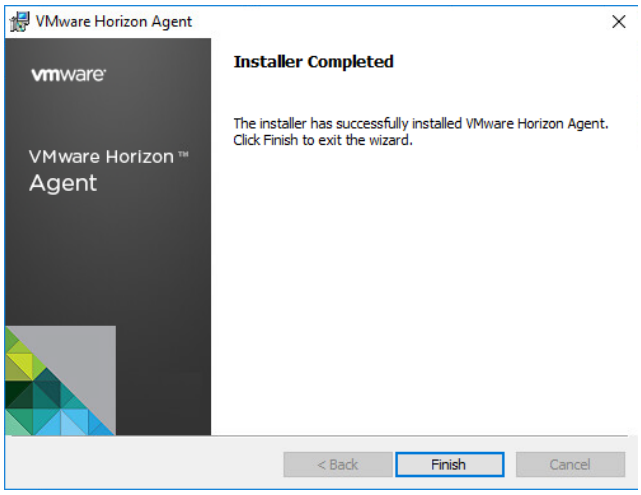


The *Ready to install the program* window displays.



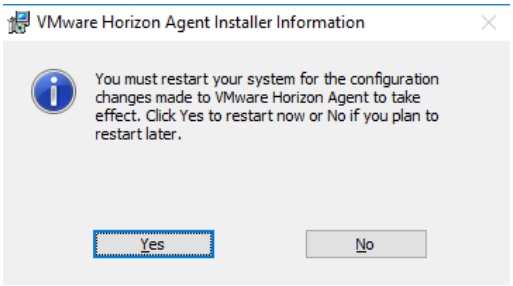
Click **Install** to proceed

Horizon Agent installs on the virtual machine and the *Installer Completed* window displays when the installation process concludes.



Click **Finish** to exit the installer.

Reboot the virtual machine when prompted to do so.



VMware Horizon Agent is now installed on the virtual machine

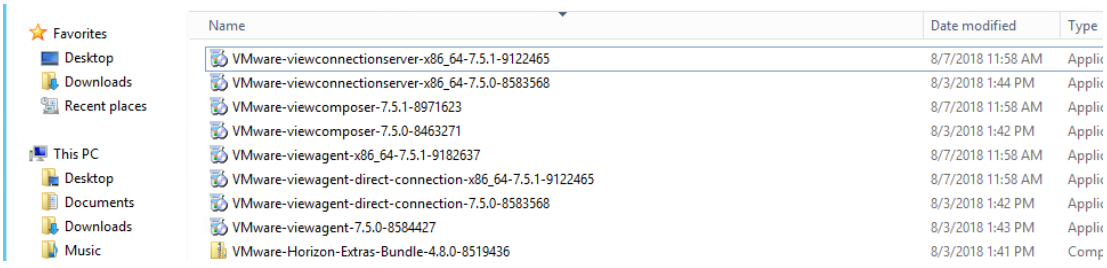
9.6 INSTALLING HORIZON DIRECT CONNECTION

For debugging purposes while configuring VM's with vGPU, install the correct version of Horizon Direct Connection agent for your virtual machine, it is provided with the Horizon 7.5 code:

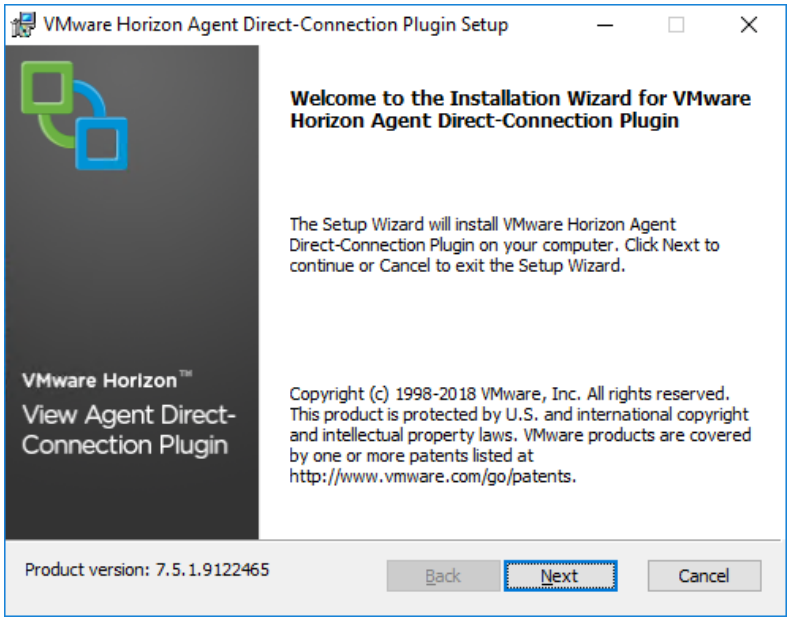
- For 32-bit virtual machines: VMware-viewagent-direct-connection-7.5.1-9122465.exe
- For 64-bit virtual machines: VMware-viewagent-direct-connection-x86_64-7.5.1-9122465.exe

Use the following procedure to install Horizon Direct Connection on the virtual machine:

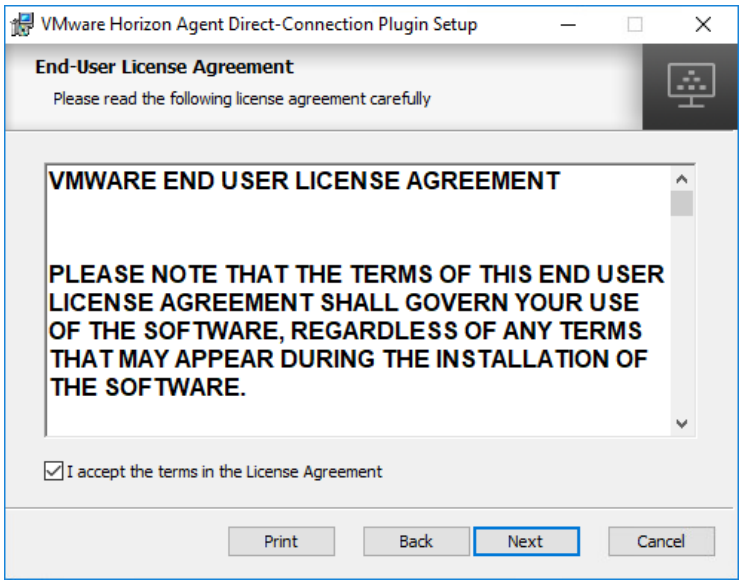
1. Launch the installer.



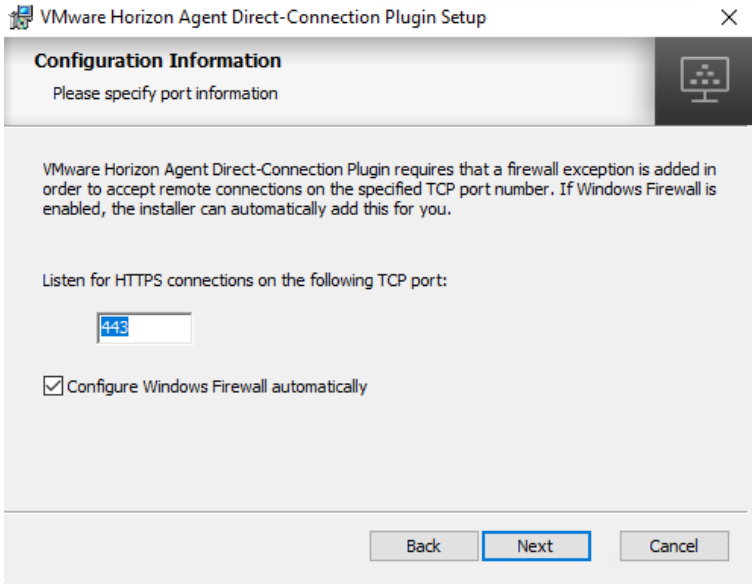
2. Click **Next** to proceed.



Check the **I accept the terms in the license agreement** radio button. Click Next to continue.

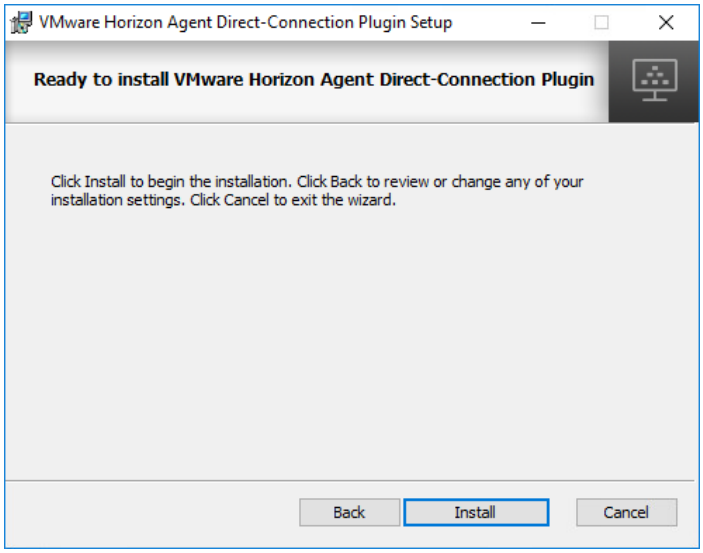


Enter port number 443 in the **Listen for HTTPS connections** field. Check the **Configure Windows Firewall automatically** checkbox. Click **Next** to proceed.

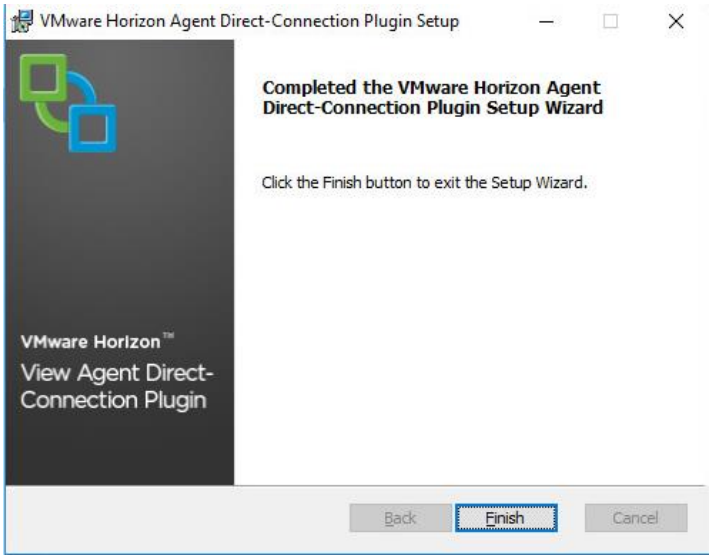


The default HTTPS port is 443; however, you may use a different port if desired.

The *Ready to install the program* window displays. Click **Install** to proceed.



Click **Finish** to exit the installer, and then reboot the virtual machine when prompted to do so.



Test **Direct Connection** using latest VMware Horizon Client by adding a server, entering the IP address of the virtual machine, and logging on to the virtual machine with the appropriate credentials when prompted.

Horizon Direct Connection is now installed on the virtual machine

9.7 OPTIMIZING WINDOWS

The VMware OS Optimization Tool helps optimize Windows 7/8/2008/2012/10 systems for use with VMware Horizon View. The optimization tool includes customizable templates to enable or disable Windows system services and features, per VMware recommendations and best practices, across multiple systems. Since most Windows system services are enabled by default, the optimization tool can be used to easily disable unnecessary services and features to improve performance.

https://labs.vmware.com/flings/vmware-os-optimization-tool

You can perform the following actions using the VMware OS Optimization Tool:

- Local Analyze/Optimize
- Remote Analyze
- Optimization History and Rollback
- Managing Templates

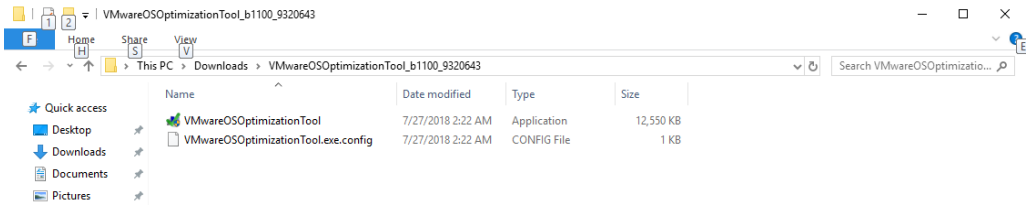
Review the *VMware Horizon Optimization Guide for Windows* for OS-specific tweaks. This document is available from

https://www.vmware.com/content/dam/digitalmarketing/vmware/en/pdf/techpaper/vmware-view-virtual-desktops-windows-optimization.pdf

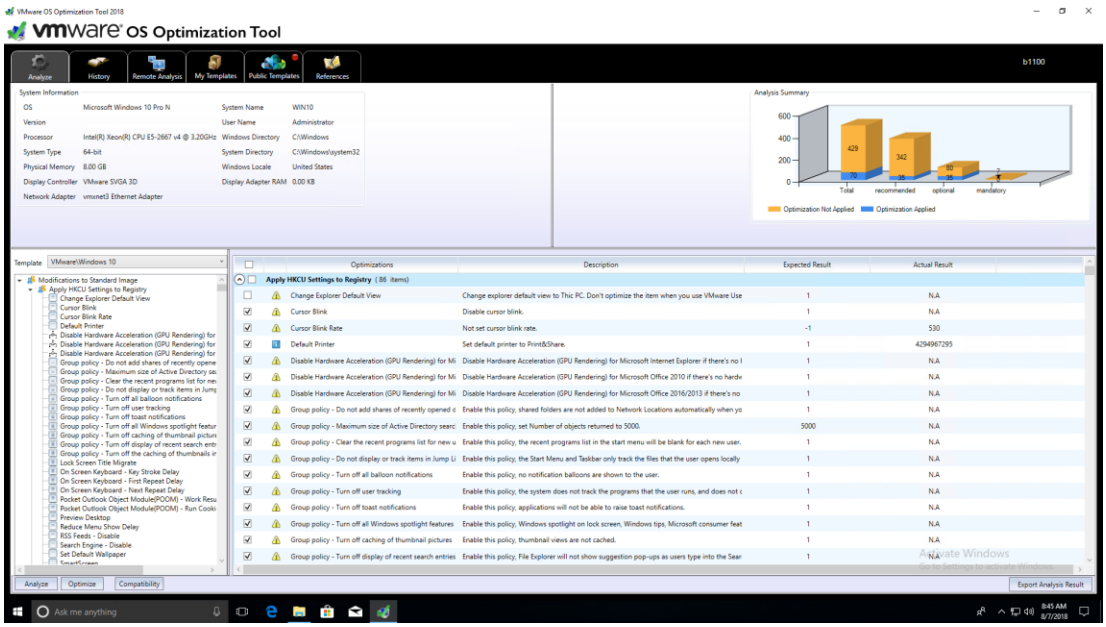
Do not run the batch files recommended in this document without understanding the changes they make, as they may disable more functionality than anticipated.

To install the optimizer:

1. Download and then browse to the folder containing the VMware OS Optimizer Tool, double click on *VMwareOSOptimizerTool.exe* to run it.

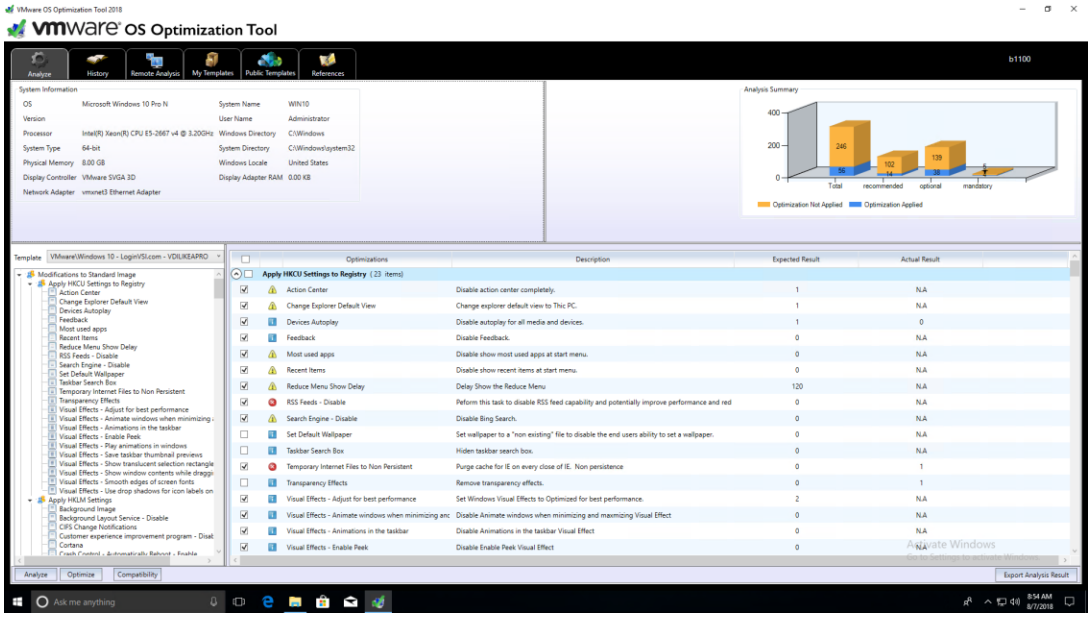


The tool opens and displays the Analyze tab with System Information, click the **Analyze** button at the bottom left to begin:

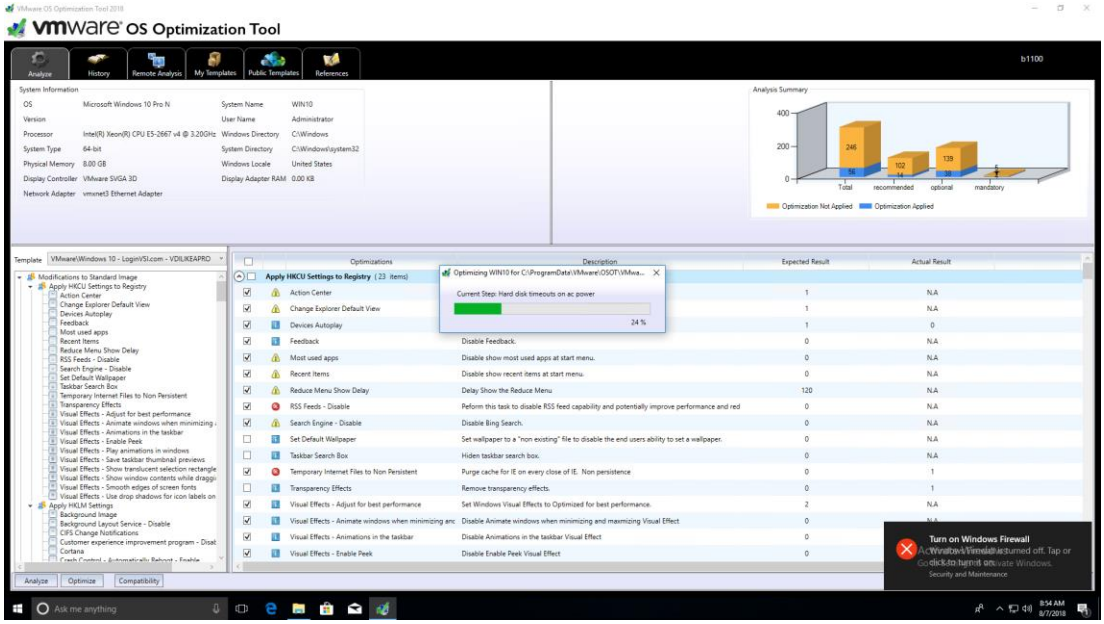


Choose Template you like, here we chose Win10 – LoginVSI, you can also create your own template

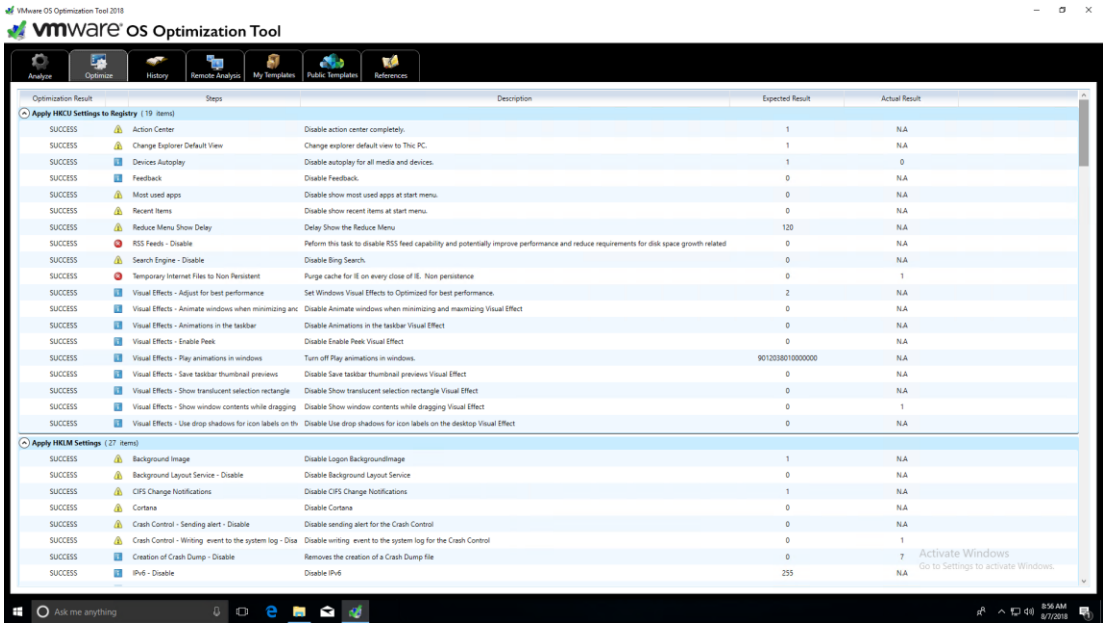
Upon completion the main window will show the results, and there is now an Optimize button at the bottom left:



Scroll through and select or deselect any of the optimizations depending on your needs, we chose the defaults. Click **Optimize** and the tool will run the selected scripts to optimize the workstation:



The results page shows once the process is complete:

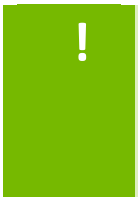


9.8 ADDITIONAL VIRTUAL MACHINE SETTINGS

1. Perform the following additional tasks on the virtual machine as required:

To evaluate browser-based HTML5 applications, consider using newer browsers that utilize hardware acceleration within virtual desktop environments.

- Turn Off Windows Firewall for all network types.



CAUTION: THESE INSTRUCTIONS ASSUME THAT THE VM IS BEING USED AS A PROOF-OF-CONCEPT ONLY AND THAT DISABLING THE FIREWALL WILL THEREFORE POSE ONLY A MINIMAL SECURITY BREACH. ALWAYS FOLLOW YOUR ESTABLISHED SECURITY PROCEDURES AND BEST PRACTICES WHEN SETTING UP SECURITY FOR A PRODUCTION MACHINE OR ANY ENVIRONMENT THAT CAN BE ACCESSED FROM OUTSIDE YOUR NETWORK.

Shut down the virtual machine once this is completed

Close the remote console; this will not be functional when vGPU is configured.

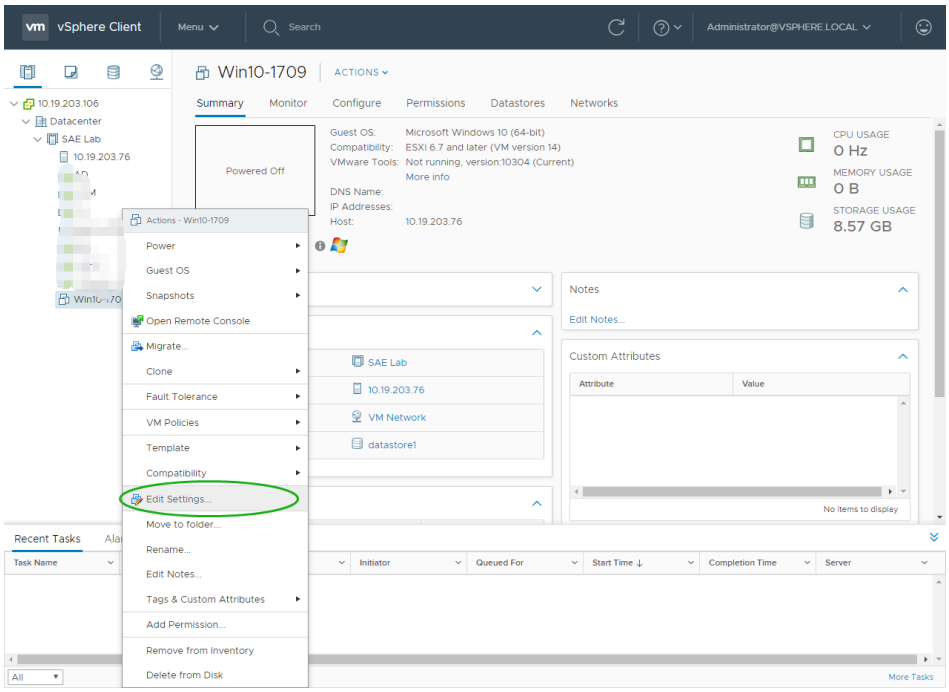
Take a snapshot of the virtual machine to preserve your work. Label this snapshot pre-vGPU, and revert to it if you encounter any problems going forward, such as driver issues

9.9 ENABLING THE NVIDIA VGPU

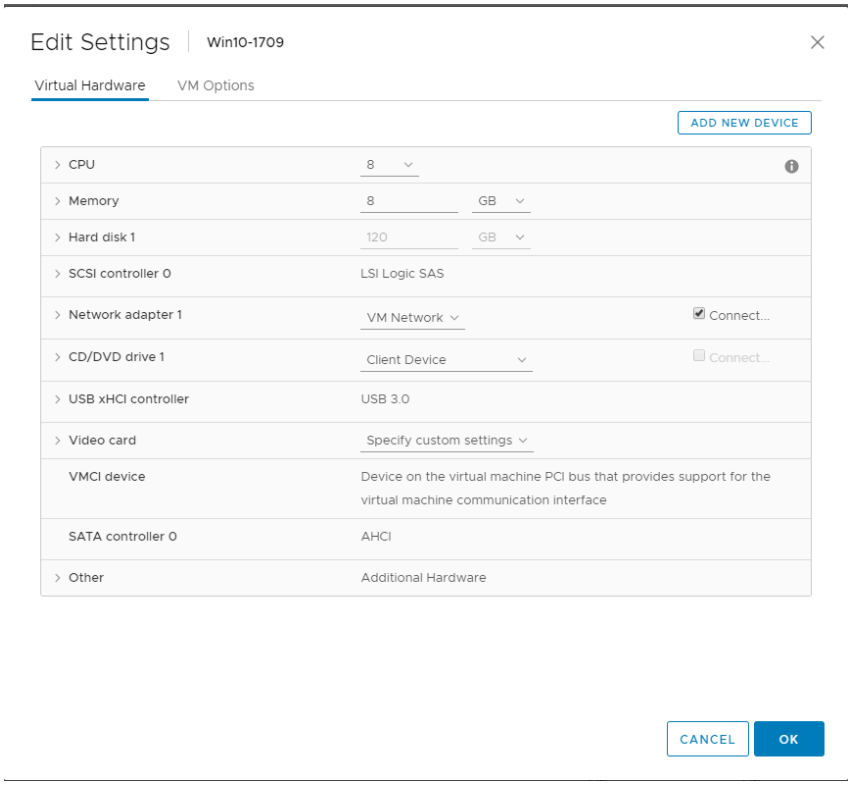
Use the following procedure to enable vGPU support for your virtual machine (you must edit the virtual machine settings):

1. Power down the virtual machine.

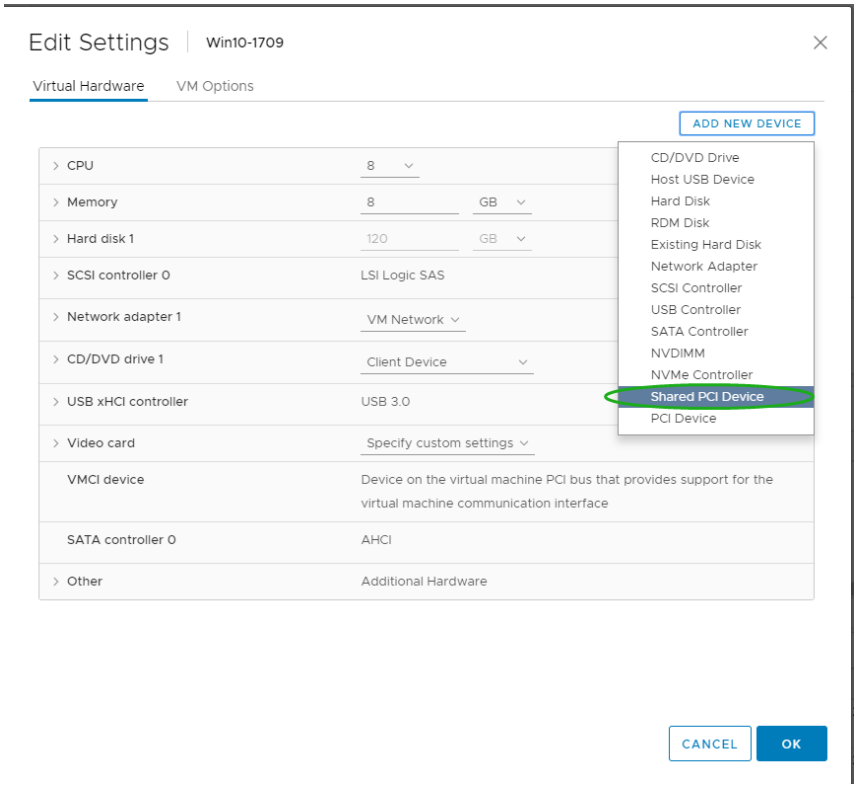
Click on the VM in the Navigator window. Right click the VM and Edit Settings.



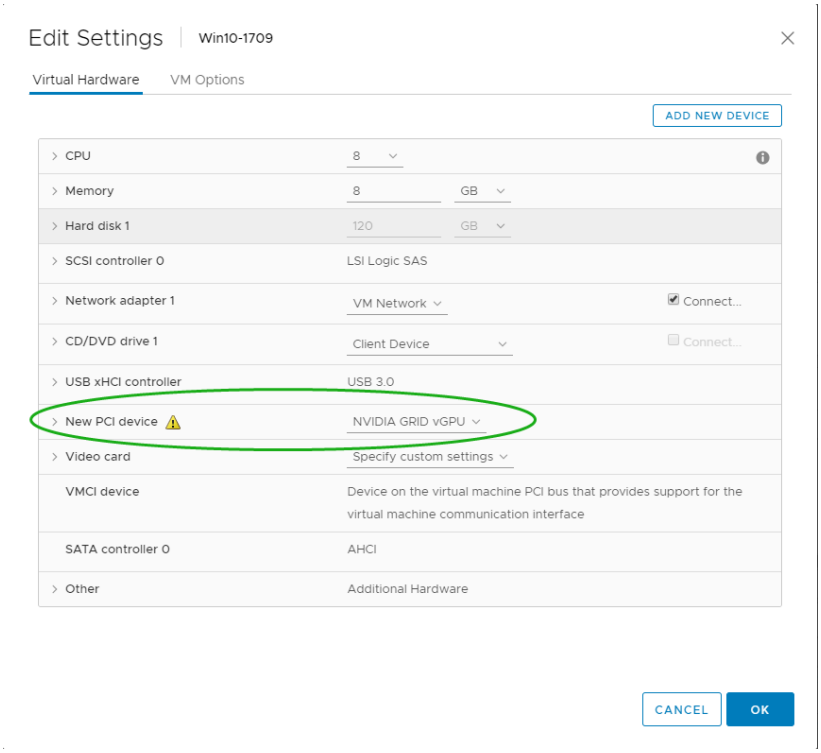
The Edit Settings dialog appears.



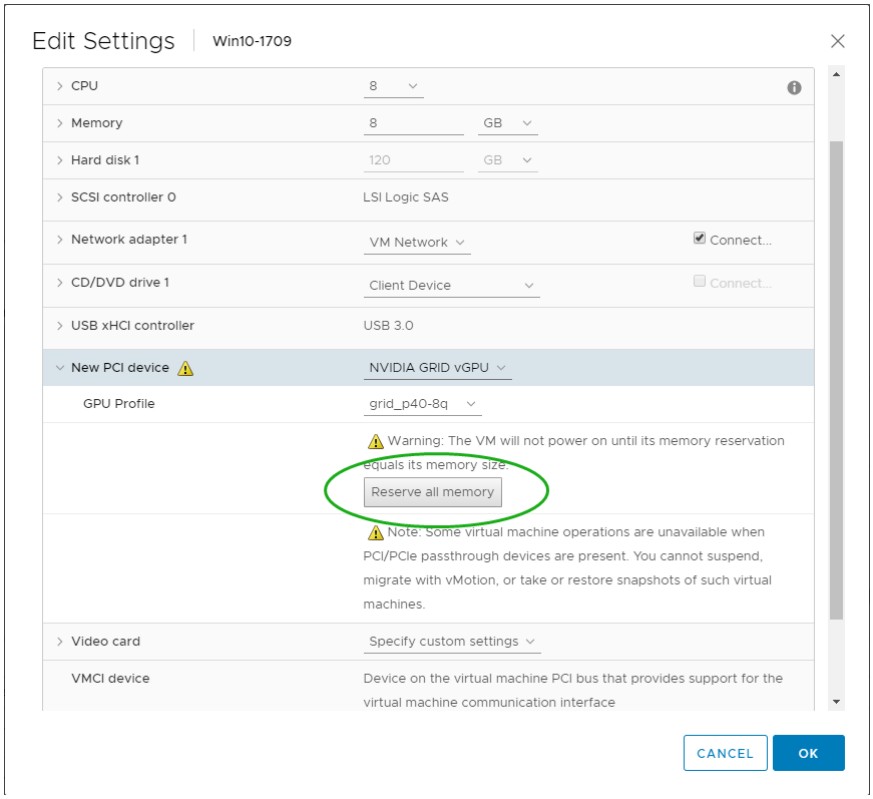
Click on the New Device bar and select Shared PCI device.



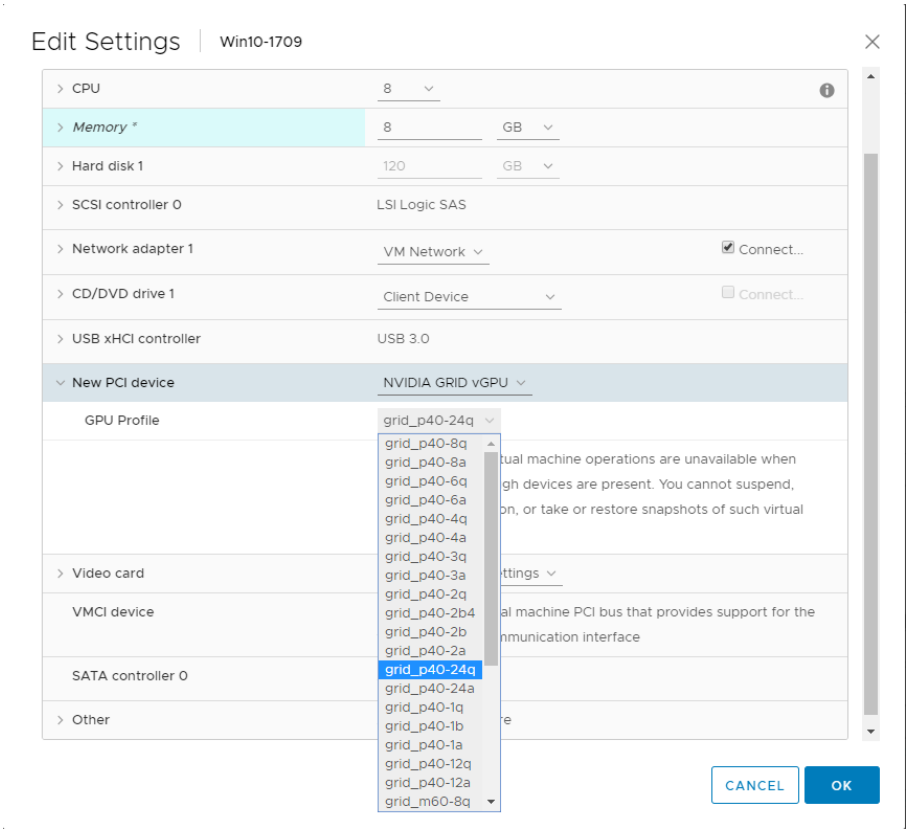
Click on Add to continue



The new PCI device will show the NVIDIA GRID vGPU device has been added.



Use the GPU Profile selection bar to configure the GPU. **Click** Reserve all memory!



Click **OK** to complete the configuration.

9.10 SWITCHING THE MODE ON A TESLA M60 OR M6

Tesla M60 and M6 GPUs support compute mode and graphics mode. NVIDIA vGPU requires GPUs that support both modes to operate in graphics mode.

Note: Only Tesla M60 and M6 GPUs require and support mode switching. Other GPUs that support NVIDIA vGPU do not require or support mode switching.

Recent Tesla M60 GPUs and M6 GPUs are supplied in graphics mode. However, your GPU might be in compute mode if it is an older Tesla M60 GPU or M6 GPU, or if its mode has previously been changed.

If your GPU supports both modes but is in compute mode, you must use the gpumodeswitch tool to change the mode of the GPU to graphics mode. If you are unsure which mode your GPU is in, use the gpumodeswitch tool to find out the mode.

For more information, see gpumodeswitch User Guide.

9.11 DISABLING ECC MEMORY

Tesla M60, Tesla M6, and GPUs based on the Pascal GPU architecture, for example Tesla P100 or Tesla P4, support error correcting code (ECC) memory for improved data integrity. Tesla M60 and M6 GPUs in graphics mode are supplied with ECC memory disabled by default, but it may subsequently be enabled using nvidia-smi. GPUs based on the Pascal GPU architecture are supplied with ECC memory enabled.

However, NVIDIA vGPU does not support ECC memory. If ECC memory is enabled, NVIDIA vGPU fails to start. Therefore, you must ensure that ECC memory is disabled on all GPUs if you are using NVIDIA vGPU.

Before you begin, ensure that NVIDIA Virtual GPU Manager is installed on your hypervisor.

Use nvidia-smi to list the status of all GPUs, and check for ECC noted as enabled on GPUs.

```
# nvidia-smi -q

==============NVSMI LOG==============

Timestamp                       : Tue Dec 19 18:36:45 2017
Driver Version                  : 384.99

Attached GPUs                   : 1
GPU 0000:02:00.0

[...]

    Ecc Mode
        Current                 : Enabled
        Pending                 : Enabled

[...]
```

1. Change the ECC status to off on each GPU for which ECC is enabled.

   - If you want to change the ECC status to off for all GPUs on your host machine, run this command:

     ```
     # nvidia-smi -e 0
     ```

   - If you want to change the ECC status to off for a specific GPU, run this command:

     ```
     # nvidia-smi -i id -e 0
     ```

     id is the index of the GPU as reported by nvidia-smi.

     This example disables ECC for the GPU with index 0000:02:00.0.

     ```
     # nvidia-smi -i 0000:02:00.0 -e 0
     ```

2. Reboot the host.

3. Confirm that ECC is now disabled for the GPU.

```
# nvidia—smi —q

==============NVSMI LOG==============

Timestamp                           : Tue Dec 19 18:37:53 2017
Driver Version                      : 384.99

Attached GPUs                       : 1
GPU 0000:02:00.0
[...]

    Ecc Mode
        Current                     : Disabled
        Pending                     : Disabled

[...]
```

If you later need to enable ECC on your GPUs, run one of the following commands:

- If you want to change the ECC status to on for all GPUs on your host machine, run this command:

  ```
  # nvidia-smi -e 1
  ```

- If you want to change the ECC status to on for a specific GPU, run this command:

  ```
  # nvidia-smi -i id -e 1
  ```

id is the index of the GPU as reported by nvidia-smi.

This example enables ECC for the GPU with index 0000:02:00.0.

```
# nvidia-smi -i 0000:02:00.0 -e 1
```

After changing the ECC status to on, reboot the host.

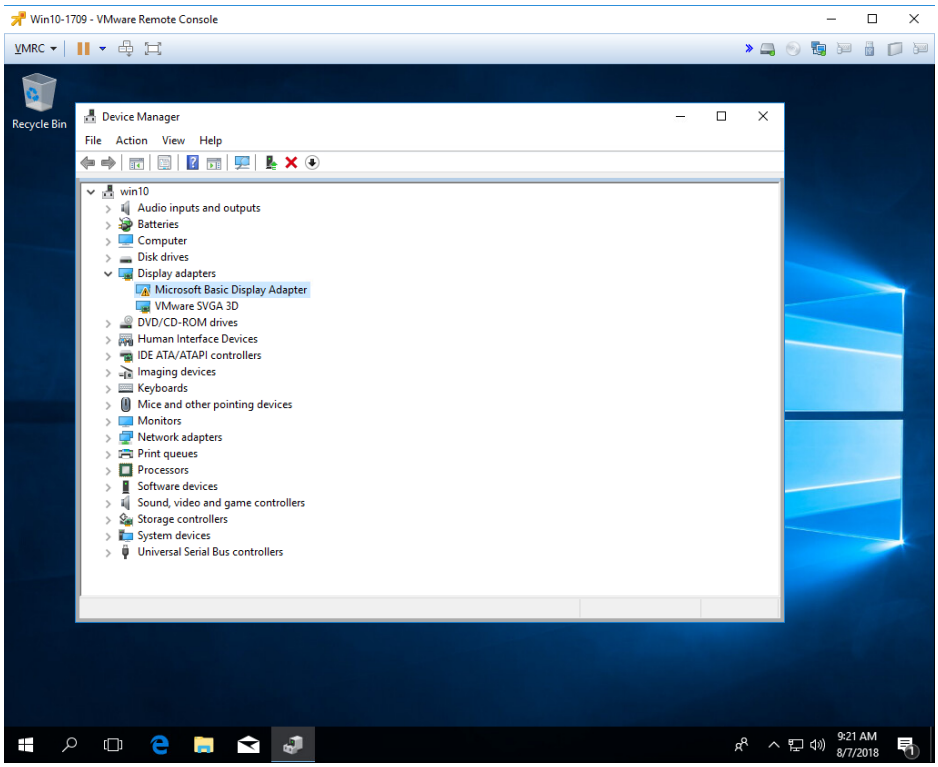
9.12 INSTALLING NVIDIA DRIVER IN WINDOWS VIRTUAL DESKTOP

1. Start the virtual machine, and then connect to it using either VMware Remote Console through the vSphere Web Client or VMware Horizon Client (via Direct Connection). *(When connected, a popup warning requesting that you restart the computer to apply changes will display the first time it is booted after enabling the NVIDIA GRID vGPU.)*

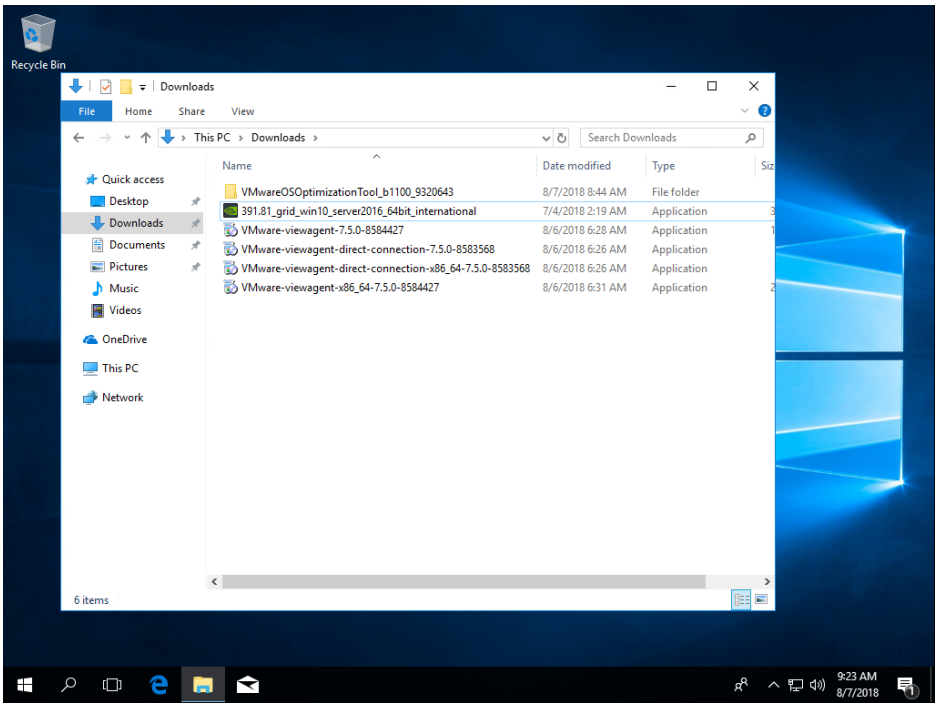
Click **Restart Later** to continue booting the virtual machine.

> **CAUTION:** DO NOT RESBOOT THE VIRTUAL MACHINE IF YOU HAVE OLDER NVIDIA DRIVERS INSTALLED. DOING THIS WILL CAUSE A BLUE SCREEN.

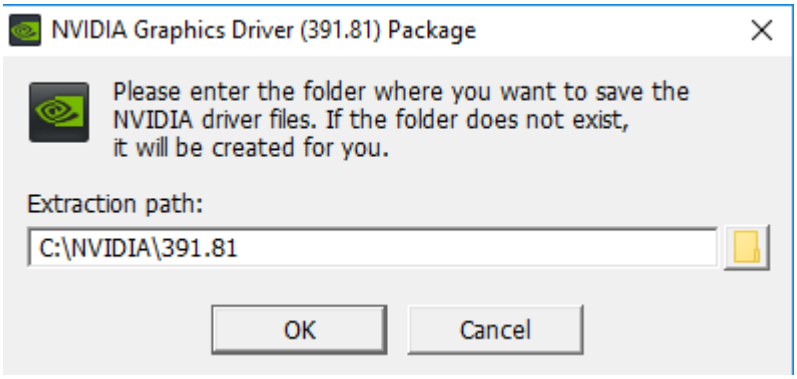
Log into Windows and open the Device Manager.
The **Standard VGA Graphics Adapter** displays in the **Display adapters** section of the Device Manager with an exclamation point by it to indicate a driver problem. This is normal.



Locate the NVIDIA driver and double-click the **Setup** icon to launch it. (Recommendation: Have the software on a share volume that can be mounted by the VM for quick access.).



Click OK to continue install.

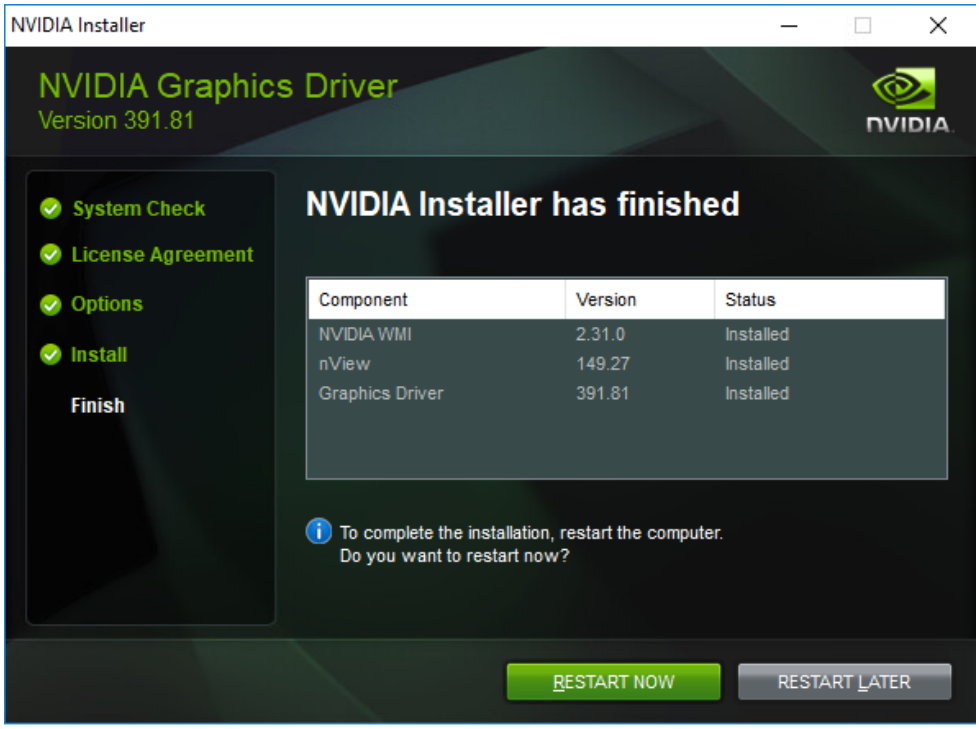


The NVIDIA software license agreement window displays. Click the **AGREE AND CONTINUE** button to proceed.

The ***Installation Options*** window displays. Check the **Custom (Advanced)** radio button, then click **Next**. The ***Custom installation*** options window appears.

Check the **Perform a clean installation** checkbox, and then click **Next**.

A window displays when the NVIDIA Graphics Driver installation is complete.



Reboot the VM to complete the install.

After restarting, the mouse cursor may not track properly using VNC or vSphere console. If so, use View Client to Direct Connect.

9.13 INSTALLING NVIDIA DRIVER IN LINUX VIRTUAL DESKTOP

- Installation in a VM: After you create a Linux VM on the hypervisor and boot the VM, install the NVIDIA vGPU software display driver in the VM to fully enable GPU operation.

- Installation on bare metal: When the physical host is booted before the NVIDIA vGPU software display driver is installed, the vesa Xorg driver starts the X server. If a primary display device is connected to the host, use the device to access the desktop. Otherwise, use secure shell (SSH) to log in to the host from a remote host. If the Nouveau driver for NVIDIA graphics cards is present, disable it before installing the NVIDIA vGPU software display driver.

  Installation of the NVIDIA vGPU software display driver for Linux requires:

  - Compiler toolchain

  - Kernel headers

1. Copy the NVIDIA vGPU software Linux driver package, for example NVIDIA-Linux_x86_64-390.75-grid.run, to the guest VM or physical host where you are installing the driver.

2. Before attempting to run the driver installer, exit the X server and terminate all OpenGL applications.

   - On Red Hat Enterprise Linux and CentOS systems, exit the X server by transitioning to runlevel 3:

     ```
     [nvidia@localhost ~]$ sudo init 3
     ```

   - On Ubuntu platforms, do the following:

     a. Use **CTRL**-**ALT**-**F1** to switch to a console login prompt.

     b. Log in and shut down the display manager:

     ```
     [nvidia@localhost ~]$ sudo service lightdm stop
     ```

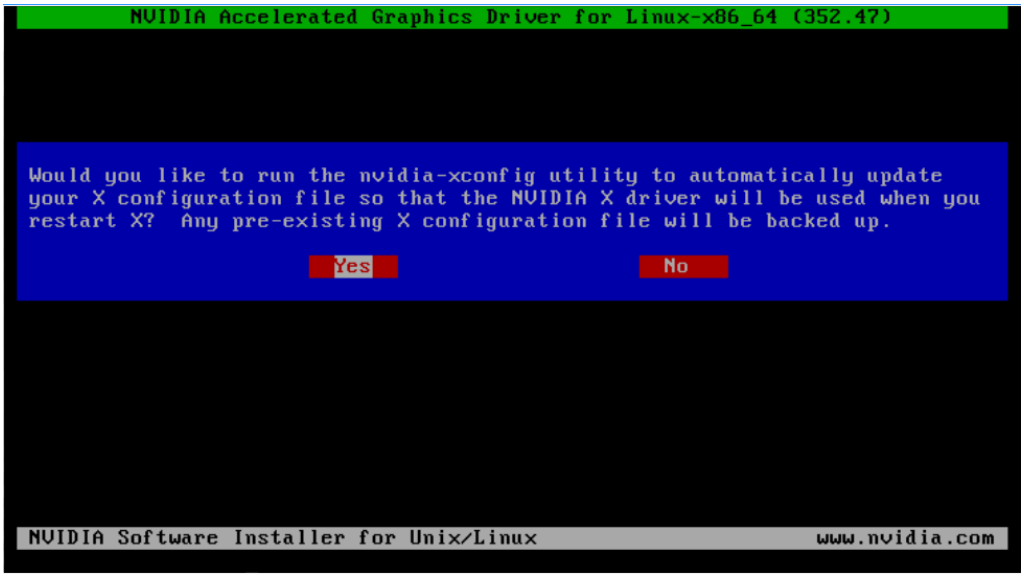
3. From a console shell, run the driver installer as the root user.

```
sudo sh ./ NVIDIA-Linux_x86_64-352.47-grid.run
```

In some instances, the installer may fail to detect the installed kernel headers and sources. In this situation, re-run the installer, specifying the kernel source path with the --kernel-source-path option:

```
sudo sh ./ NVIDIA-Linux_x86_64-352.47-grid.run \
–kernel-source-path=/usr/src/kernels/3.10.0-229.11.1.el7.x86_64
```

4. When prompted, accept the option to update the X configuration file (xorg.conf).



5. Once installation has completed, select **OK** to exit the installer.

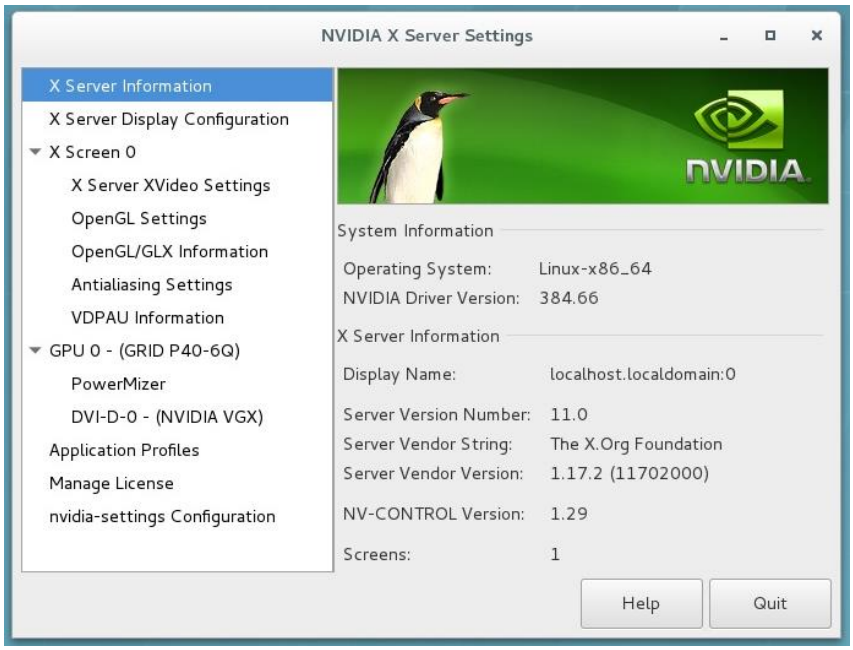
6. Verify that the NVIDIA driver is operational.

    a. Reboot the system and log in.

    b. Run nvidia-settings.

```
[nvidia@localhost ~]$ nvidia-settings
```

The NVIDIA X Server Settings dialog box opens to show that the NVIDIA driver is operational.



- Installation in a VM: After you install the NVIDIA vGPU software display driver, you can license any NVIDIA vGPU software licensed products that you are using. For instructions, refer to Virtual GPU Client Licensing User Guide.

- Installation on bare metal: After you install the NVIDIA vGPU software display driver, complete the bare-metal deployment as explained in Bare-Metal Deployment.

9.14 LICENSING NVIDIA VGPU

NVIDIA vGPU is a licensed product. When booted on a supported GPU, a vGPU runs at reduced capability until a license is acquired.

The performance of an unlicensed vGPU is restricted as follows:

- Frame rate is capped at 3 frames per second.

- GPU resource allocations are limited, which will prevent some applications from running correctly.

- On vGPUs that support CUDA, CUDA is disabled.

- 6.0, 6.1 only: Screen resolution is limited to no higher than 1280×1024.

These restrictions are removed when a license is acquired.

After you license NVIDIA vGPU, the VM that is set up to use NVIDIA vGPU is capable of running the full range of DirectX and OpenGL graphics applications.

If licensing is configured, the virtual machine (VM) obtains a license from the license server when a vGPU is booted on these GPUs. The VM retains the license until it is shut down. It then releases the license back to the license server. Licensing settings persist across reboots and need only be modified if the license server address changes, or the VM is switched to running GPU pass through.

Note: For complete information about configuring and using NVIDIA vGPU software licensed features, including vGPU, refer to Virtual GPU Client Licensing User Guide.

9.14.1 Licensing NVIDIA vGPU on Windows

1. Open NVIDIA Control Panel:

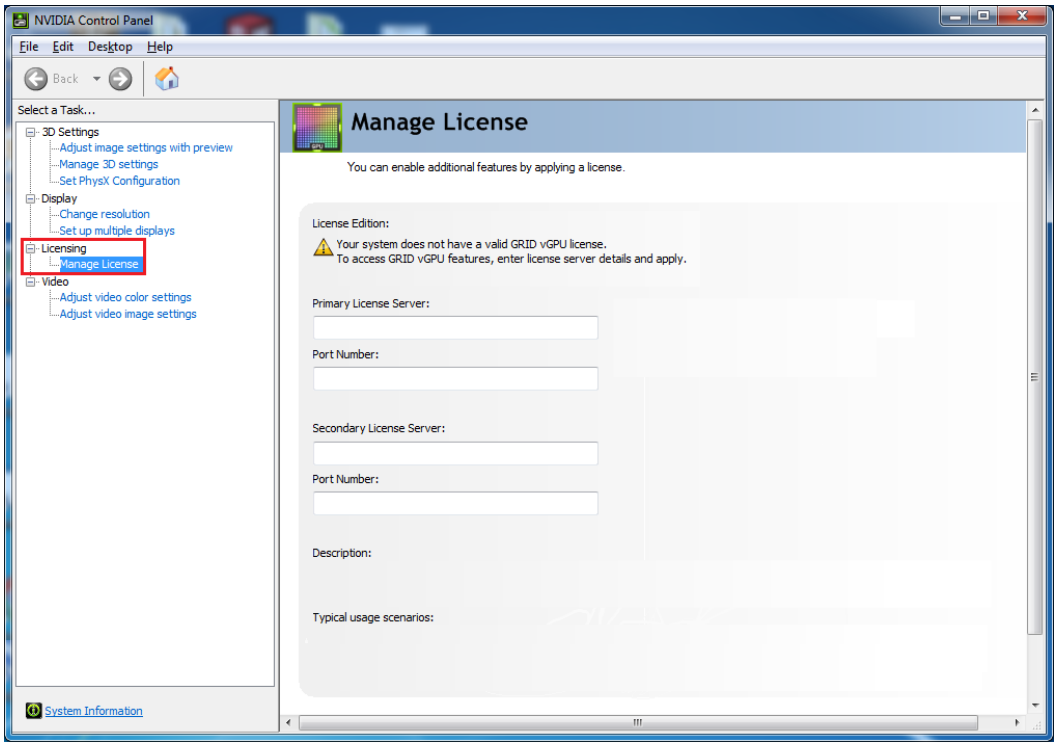
    - Right-click on the Windows desktop and select **NVIDIA Control Panel** from the menu.

    - Open Windows Control Panel and double-click the **NVIDIA Control Panel** icon.

2. In NVIDIA Control Panel, select the **Manage License** task in the **Licensing** section of the navigation pane.

    Note: If the **Licensing** section and **Manage License** task are not displayed in NVIDIA Control Panel, the system has been configured to hide licensing controls in NVIDIA Control Panel. For information about registry settings, see Virtual GPU Client Licensing User Guide.

    The Manage License task pane shows that NVIDIA vGPU is currently unlicensed.



3. In the **Primary License Server** field, enter the address of your primaryNVIDIA vGPU software License Server. The address can be a fully-qualified domain name such as gridlicense1.example.com, or an IP address such as 10.31.20.45. If you have only one license server configured, enter its address in this field.
4. Leave the **Port Number** field under the **Primary License Server** field unset. The port defaults to 7070, which is the default port number used by NVIDIA vGPU software License Server.
5. In the **Secondary License Server** field, enter the address of your secondary NVIDIA vGPU software License Server. If you have only one license server configured, leave this field unset. The address can be a fully-qualified domain name such as gridlicense2.example.com, or an IP address such as 10.31.20.46.
6. Leave the **Port Number** field under the **Secondary License Server** field unset. The port defaults to 7070, which is the default port number used by NVIDIA vGPU software License Server.
7. Click **Apply** to assign the settings. The system requests the appropriate license for the current vGPU from the configured license server.

The vGPU within the VM should now exhibit full frame rate, resolution, and display output capabilities. The VM is now capable of running the full range of DirectX and OpenGL graphics applications.

If the system fails to obtain a license, see Virtual GPU Client Licensing User Guide for guidance on troubleshooting.

9.14.2 Licensing NVIDIA vGPU on Linux

1. Start NVIDIA X Server Settings by using the method for launching applications provided by your Linux distribution. For example, on Ubuntu Desktop, open the Dash, search for NVIDIA X Server Settings, and click the **NVIDIA X Server Settings** icon.

2. In the NVIDIA X Server Settings window that opens, click **Manage GRID License**. The License Edition section of the NVIDIA X Server Settings window shows that NVIDIA vGPU is currently unlicensed.

3. In the **Primary Server** field, enter the address of your primary NVIDIA vGPU software License Server. The address can be a fully-qualified domain name such as gridlicense1.example.com, or an IP address such as 10.31.20.45. If you have only one license server configured, enter its address in this field.
4. Leave the **Port Number** field under the **Primary Server** field unset. The port defaults to 7070, which is the default port number used by NVIDIA vGPU software License Server.
5. In the **Secondary Server** field, enter the address of your secondary NVIDIA vGPU software License Server. If you have only one license server configured, leave this field unset. The address can be a fully-qualified domain name such as gridlicense2.example.com, or an IP address such as 10.31.20.46.
6. Leave the **Port Number** field under the **Secondary Server** field unset. The port defaults to 7070, which is the default port number used by NVIDIA vGPU software License Server.
7. Click **Apply** to assign the settings. The system requests the appropriate license for the current vGPU from the configured license server.

The vGPU within the VM should now exhibit full frame rate, resolution, and display output capabilities. The VM is now capable of running the full range of DirectX and OpenGL graphics applications.

If the system fails to obtain a license, see Virtual GPU Client Licensing User Guide for guidance on troubleshooting.

9.15 FINALIZING THE INSTALLATION

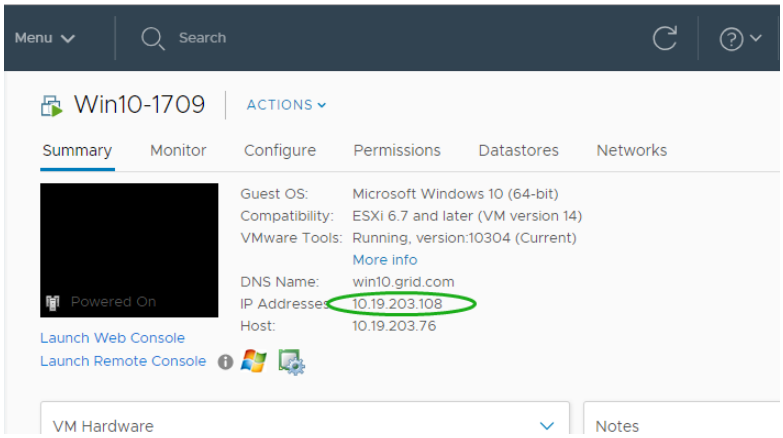
The final phase of the NVIDIA vGPU configuration is to connect to the virtual machine, verify settings, and then experience the power of fully virtualized GPU support in your favorite testing, benchmarking, multimedia, or 2D/3D/animation applications.

Use the following procedure to finalize the installation.

Record in vSphere Web Client, the IP address assigned to the virtual machine by selecting the virtual machine, and then selecting the Summary tab.

1. Select the virtual machine in the ***vSphere Web Client*** and select the **Summary** tab.

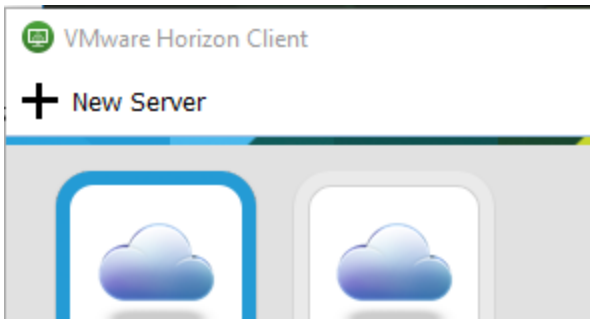
Find the IP address assigned to the virtual machine by selecting the VM in the Navigator window and clicking on the Summary tab. The VM's IP Address will appear in the window.



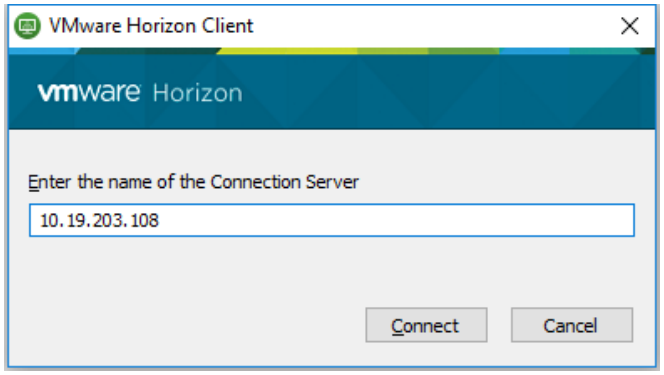
The IP address information will only be displayed if the virtual machine has VMware Tools running and has successfully negotiated an IP address from a DHCP server or when using a static address.

Start the **VMware Horizon Client** from the desktop to connect from.

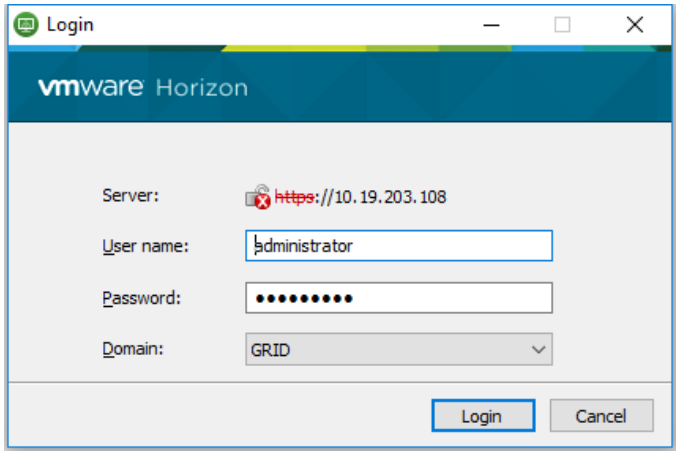
2. Click on **Add Server** to register a new virtual machine connection.



3. Enter the IP address of the VDI desktop to connect to in the **Hostname** field and click **Continue**. *(You are connecting directly to a desktop, don't enter the Horizon View Connection Broker address at this time.)*



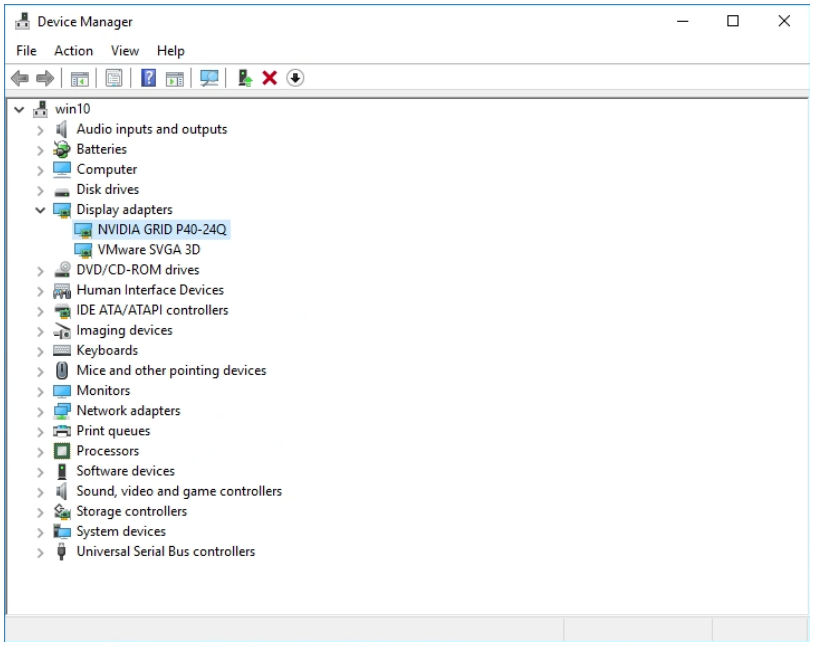
Enter the local user name and password (or the domain user and password if the virtual machine is a domain member) and click **Continue**.



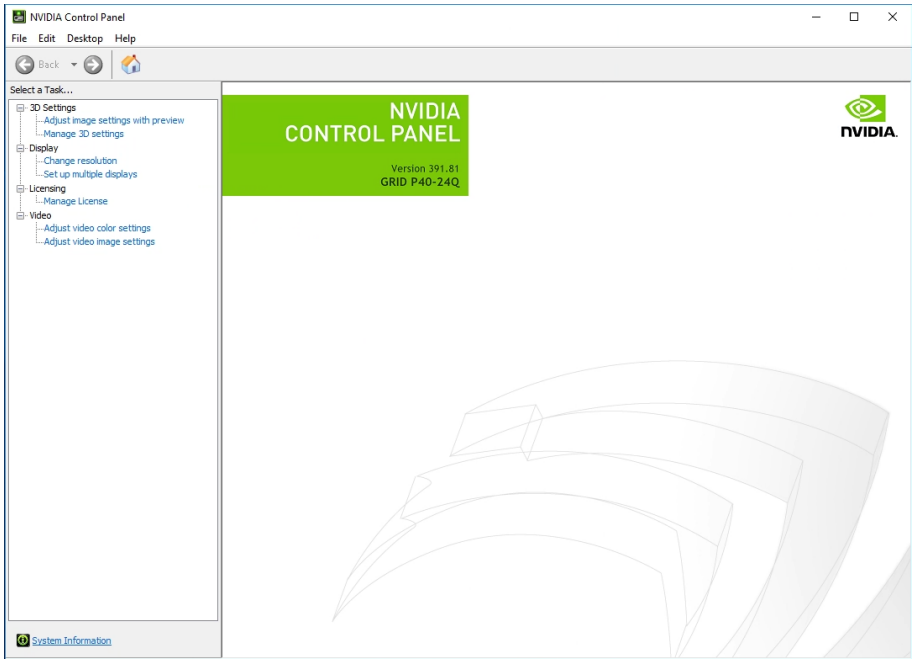
Log into the virtual machine.

Open **Device Manager** and confirm that the display adapter is now a vGPU display.

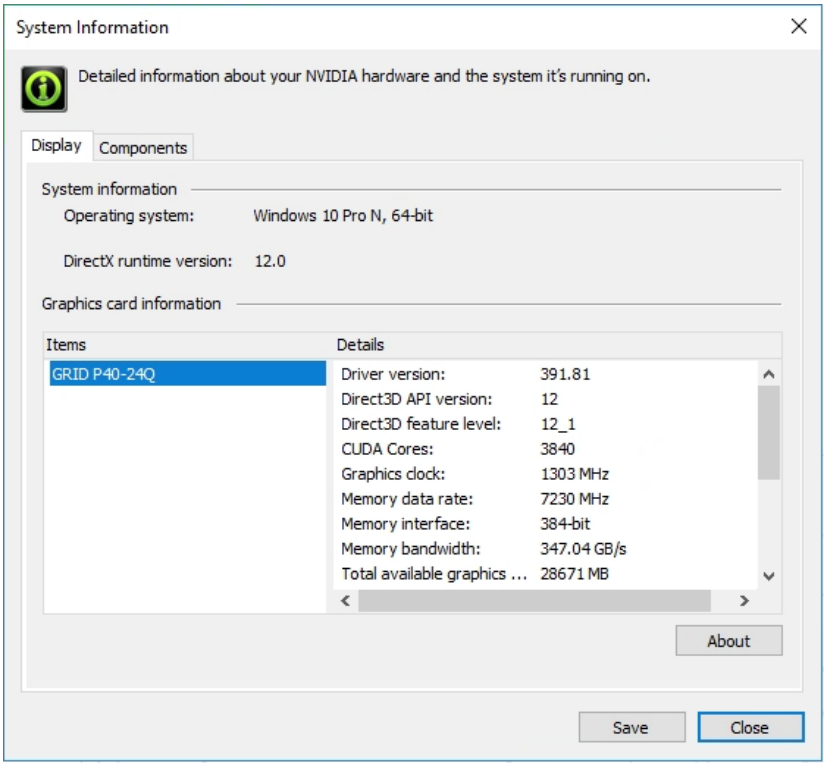
Open Display adapters in Device Manager:



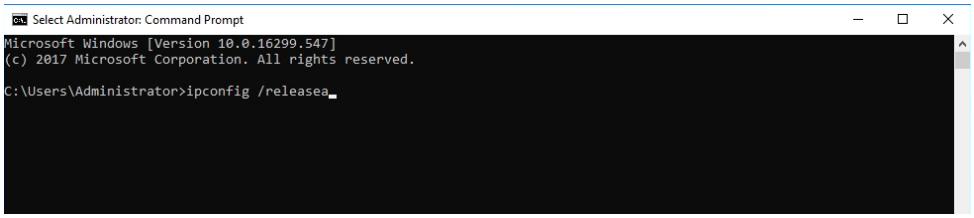
Right click the desktop and select **NVIDIA Control Panel.**



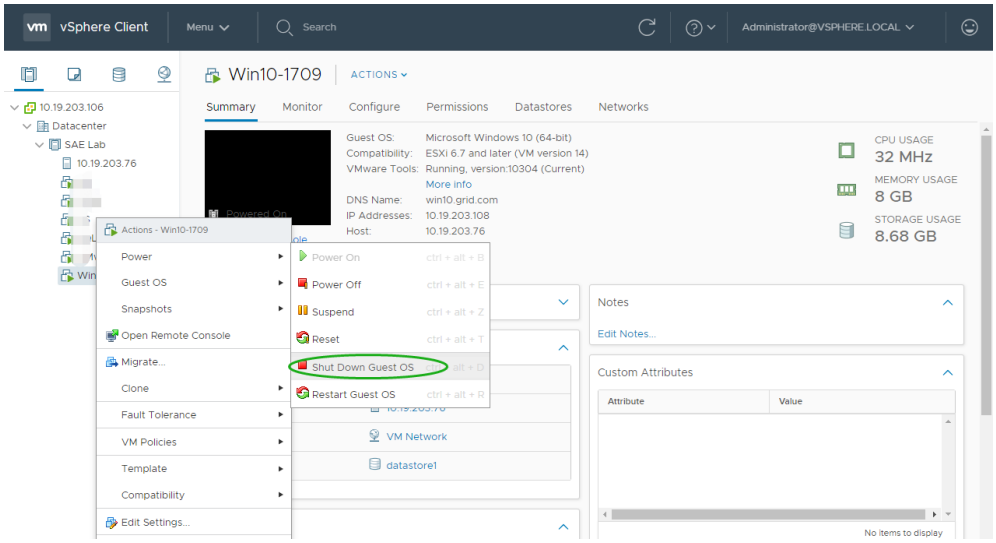
Select **System Information** on the bottom left. Confirm the DirectX, graphics card, and driver versions and select **Close** when done.



If you plan to use this VM as your Goldmaster Image, release the DHCP address before logging out and shutting down. Open a command prompt and enter: ipconfig /release.



You will immediately lose connectivity to the VM. Use the vSphere Web Client to shutdown the Guest OS.



A vGPU enabled virtual machine is now confirmed. Next step is to build and deploy a Horizon View vGPU Desktop Pool.

Chapter 10. CREATING A HORIZON VGPU POOL

This chapter describes the following:

- Creating a template from an existing virtual machine.
- Provisioning a single vGPU-enabled virtual machine from a template
- Creating a Horizon desktop pool
- Provisioning vGPU-enabled virtual machines from a template

To create a pool of virtual machines, you must first convert an existing virtual machine into a template, which then allows you to either create a single virtual machine or use it to dynamically create virtual machines on demand.

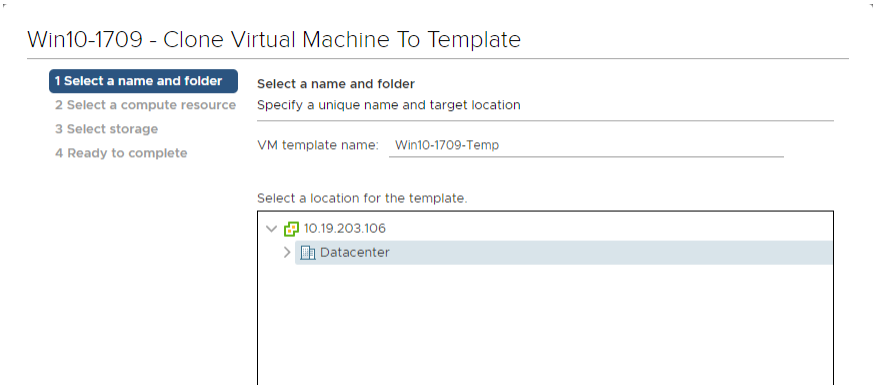
A complete demonstration of these features is beyond the scope of this document; however, converting a virtual machine to a template and then deploying virtual machines from that template is a fundamental operation that can reduce evaluation time.

10.1 CREATING A TEMPLATE

Use the following procedure to create a template from an existing virtual machine:

1. Right-click a virtual machine and then select **Clone→Clone to Template...**

The **Clone Virtual Machine To Template** popup displays with the **Select a name and folder** tab selected.



Enter a name for the template

Select either a datacenter or virtual machine folder to create the template

Select **Next** to display the **Select a compute resource** tab.

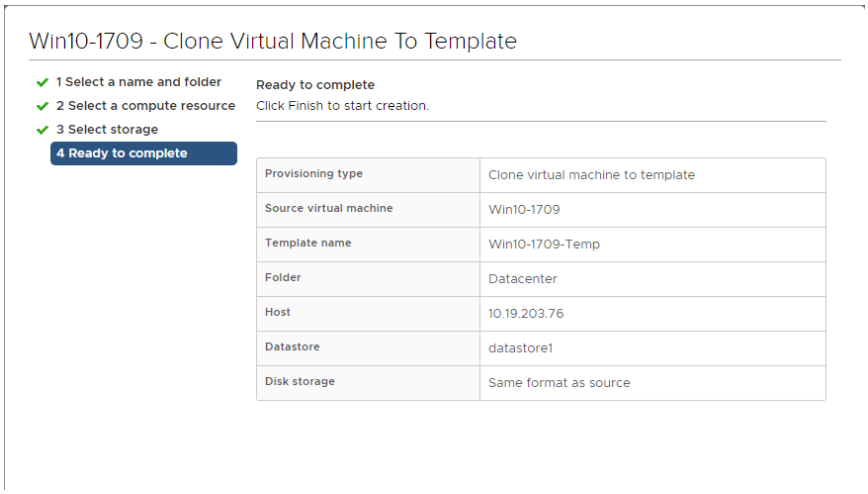
Select the host or cluster to store the template

Select **Next** to display the **Select storage** tab.

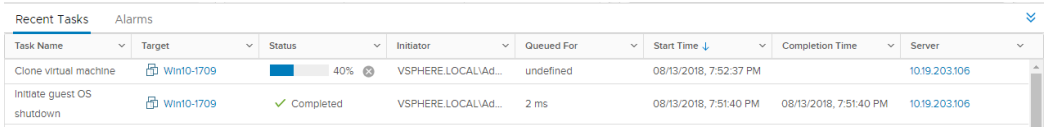
Select the storage type and target data store.

Select **Next** to display the **Ready to complete** tab.

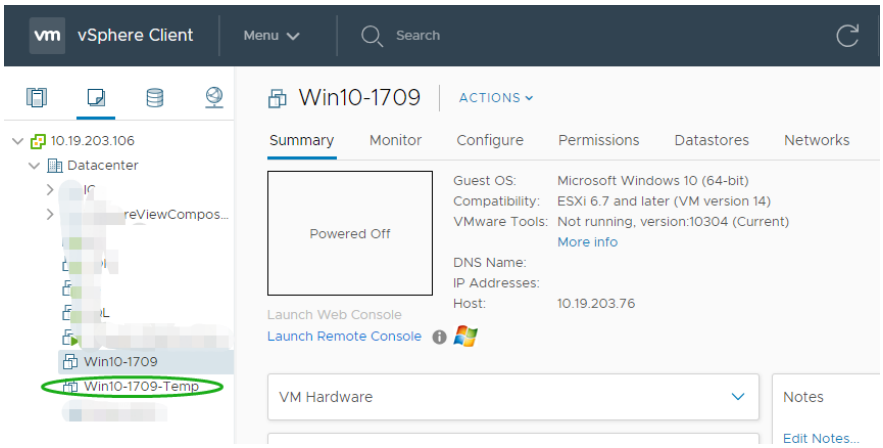
Review the settings and click **Finish** to start cloning the virtual machine to a template.



The vSphere Web Client **Recent Tasks** view displays the progress of the cloning operation.



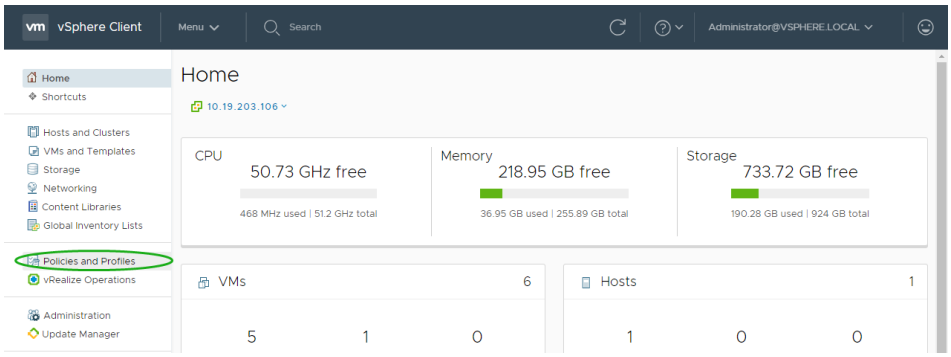
Upon completion, the template displays within the target datacenter and/or host.



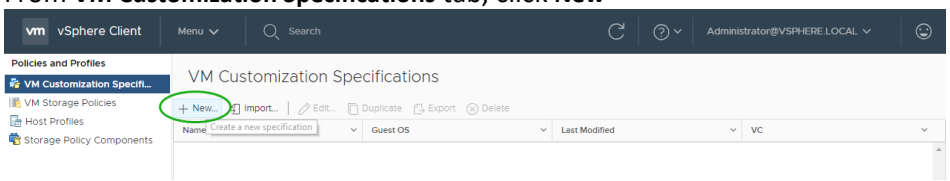
10.2 CREATING A CUSTOMIZATION SPECIFICATION FILE

Before provision a virtual machine from a template, we need to create a **VM Customization Specifications.** Use the following procedure to create a **VM Customization Specifications:**

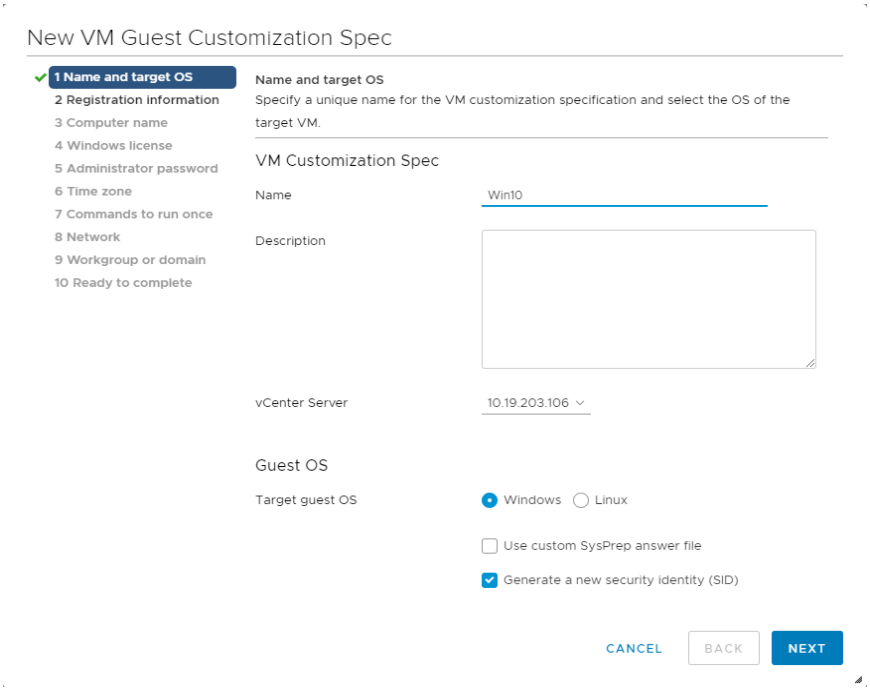
1. Click the **Polices and Profiles** from **home** view



2. From **VM Customization Specifications** tab, click **New**



The **New VM Guest Customization Spec** window display pops up.

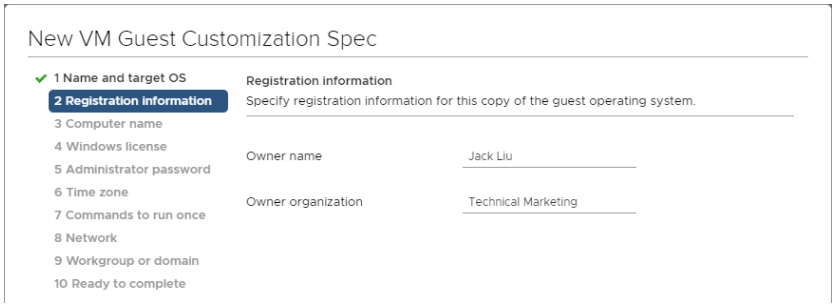


3. Proceed as follows:
   - Check **Windows** as Target guest OS.
   - Check the **Generate New Security ID (SID)** checkbox.

- If your organization currently utilizes an answer file, then you can load that file by checking the **Use custom SysPrep answer file** checkbox.
- If your organization does not use this functionality, then leave the **Use custom SysPrep answer file** checkbox cleared; enter a name for the new specification in the **Customization Spec Name** field and click **Next** to proceed.

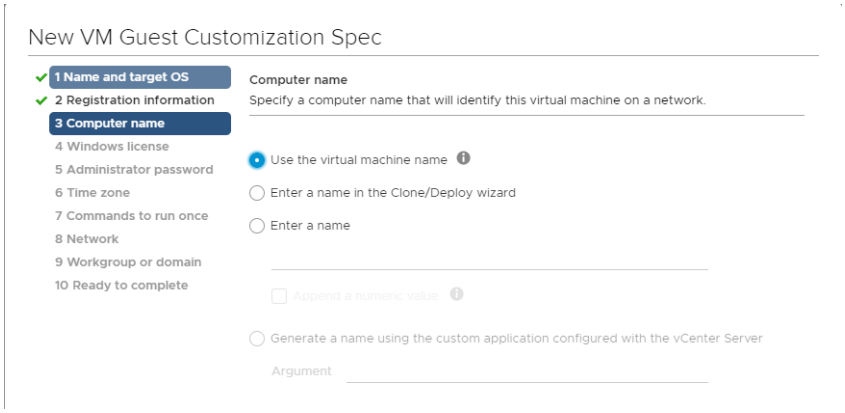
The **Set Registration Information** tab displays.

4. Enter the user and organization names in the **Name** and **Organization** fields, respectively.

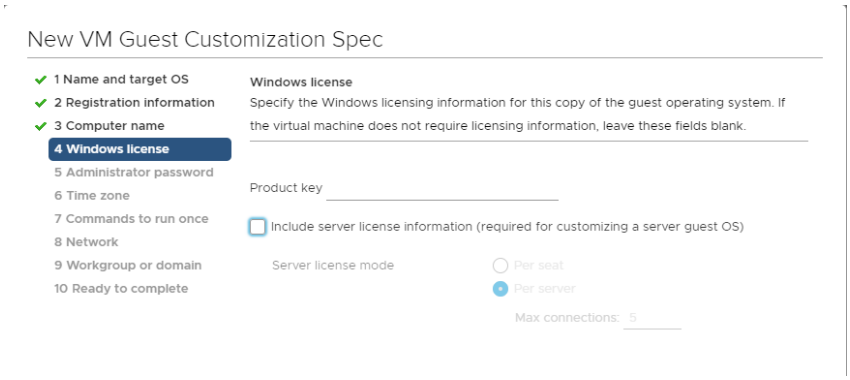


Select **Next** to display the **Set Computer Name** tab.

5. Enter a computer name.

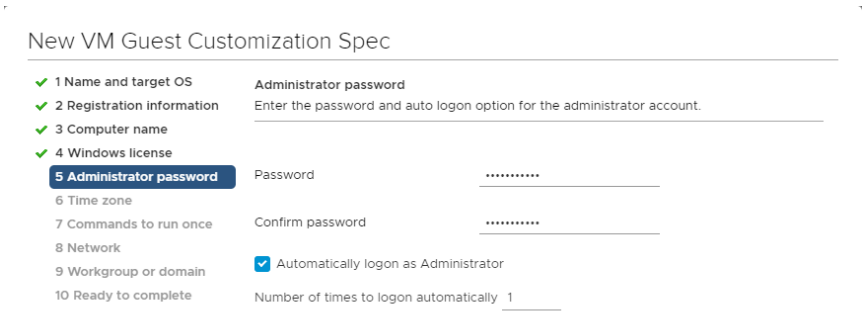


6. Select **Next** to display the **Enter Windows License** tab.



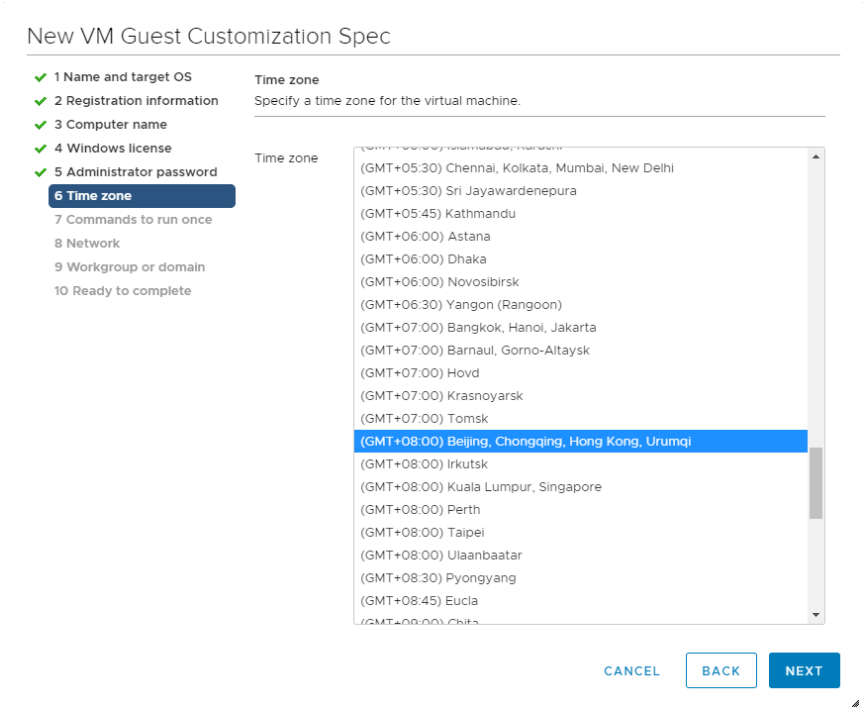
Enter your Windows product key information or leave this field blank

7. Select **Next** to display **Set Administrator Password**.



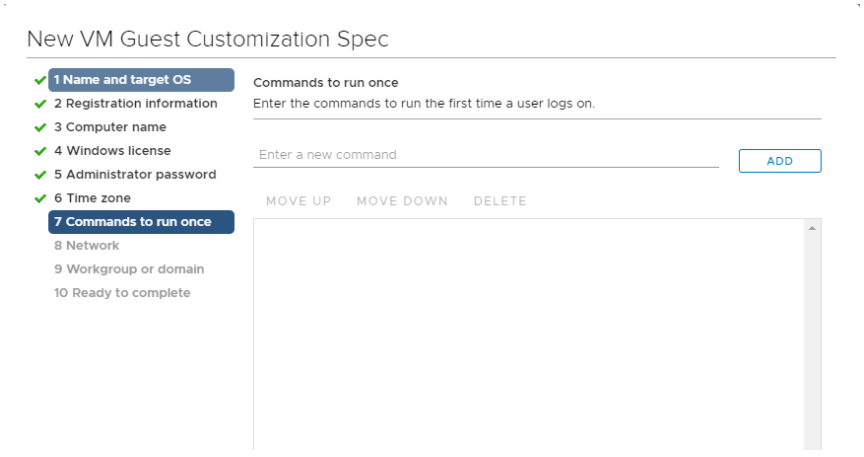
Enter and confirm the administrator password (case sensitive) in the **Password** and **Confirm Password** fields,

8. Select **Next** to display the **Time Zone** tab.



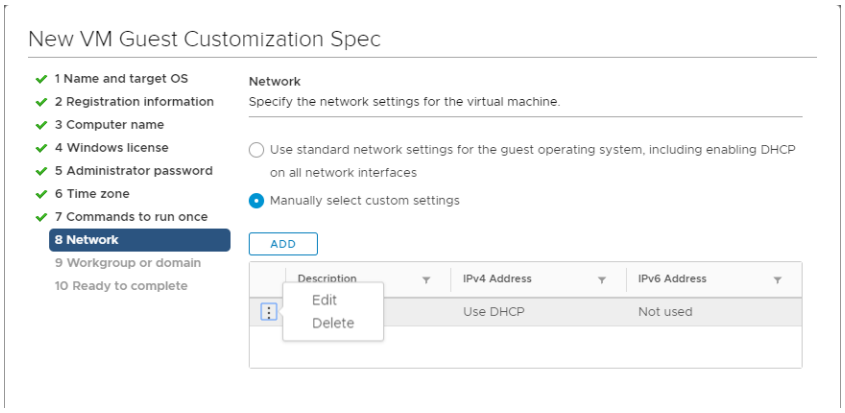
Use the **Time Zone** pull-down menu to select your time zone.

9. Select **Next** to display the **Run Once** tab.



Enter any one-time commands in this tab if needed.

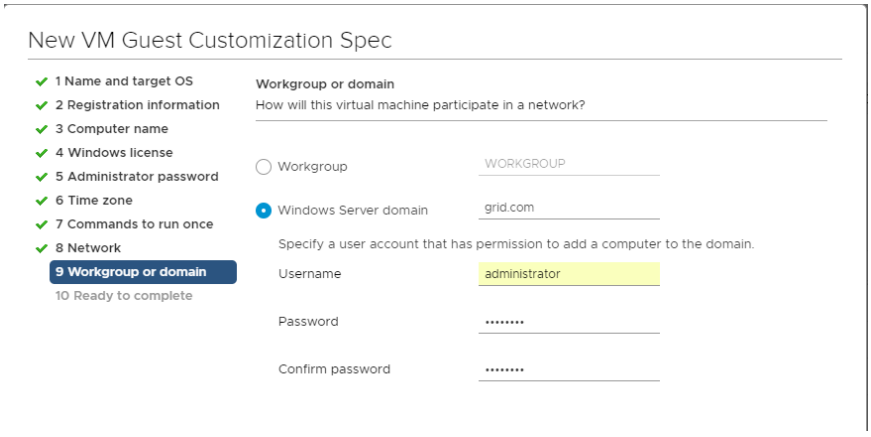
10. Select **Next** to display the Configure Network Tab.



Check the appropriate radio button to either:

- **Use standard network settings…**
  This option automatically selects network settings
- **Use manual network settings**
  For this option you enter the network description, IPv4 address, and/or IPv6 address in the respective fields.

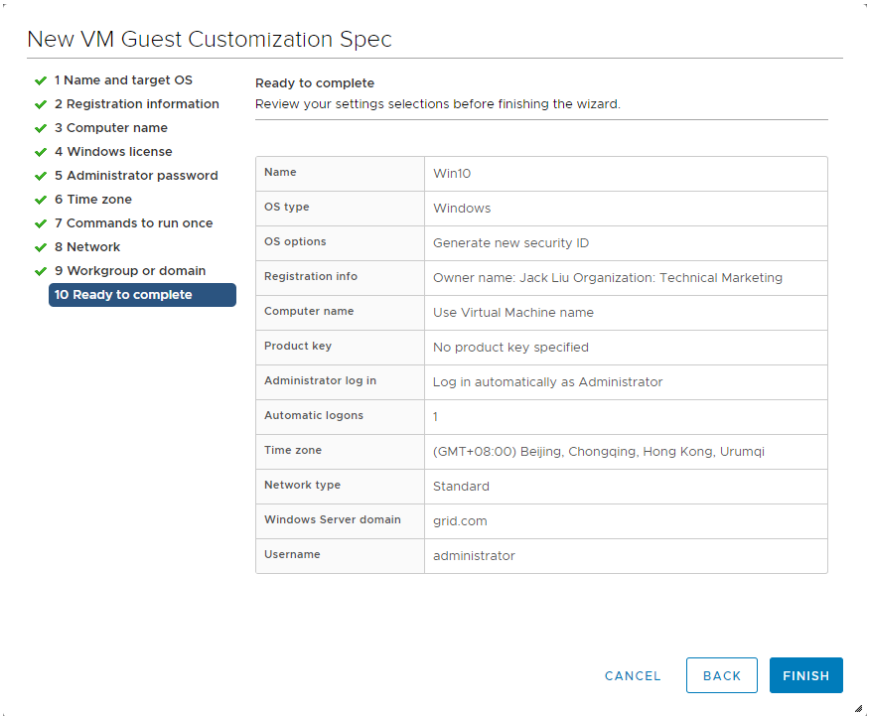
11. Click **Next** to display the **Set Workgroup or Domain** tab.



Check the appropriate radio button to select either a workgroup or domain.

Enter the information workgroup or domain for the virtual machine to join in the appropriate fields

12. Select **Next** to display the **Ready to complete** tab.



Review your settings.

Select **Finish** to view the new customization specification from the template.

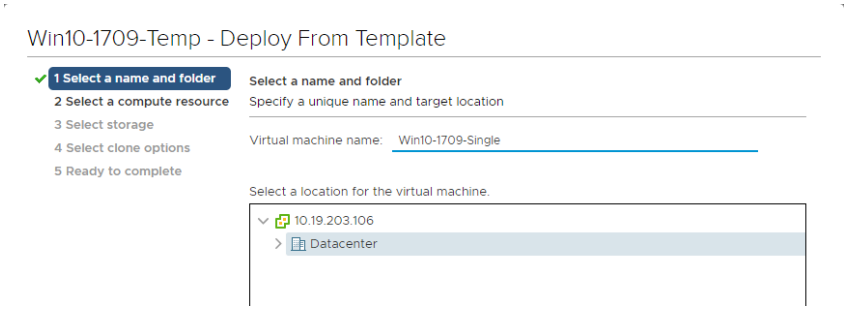
You do not need to create a new virtual machine guest customization specification every time you clone a virtual machine from a template.

10.3 PROVISIONING A SINGLE VIRTUAL MACHINE

Use the following procedure to provision a single vGPU-enabled virtual machine from a template:

1. Right-click the template and select **New VM from this Template...**

The ***Deploy From Template*** window displays with the **Select a name and folder** tab selected.



Enter a name for the new virtual machine.

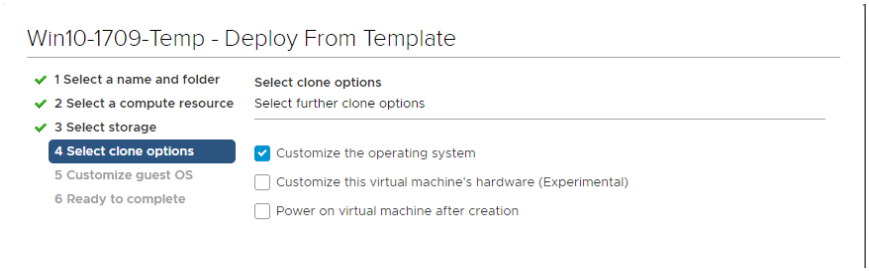
Select **Next** to display the **Select a compute resource** tab.

Select the cluster or host

Select **Next** to display the **Select storage** tab.

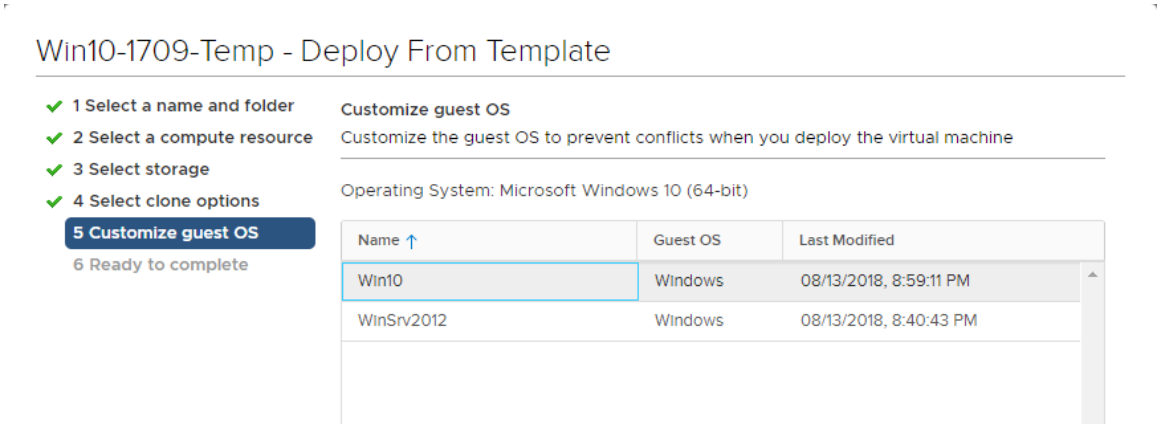
Select the target storage type and data store.

Select **Next** to display the **Select clone options** tab.



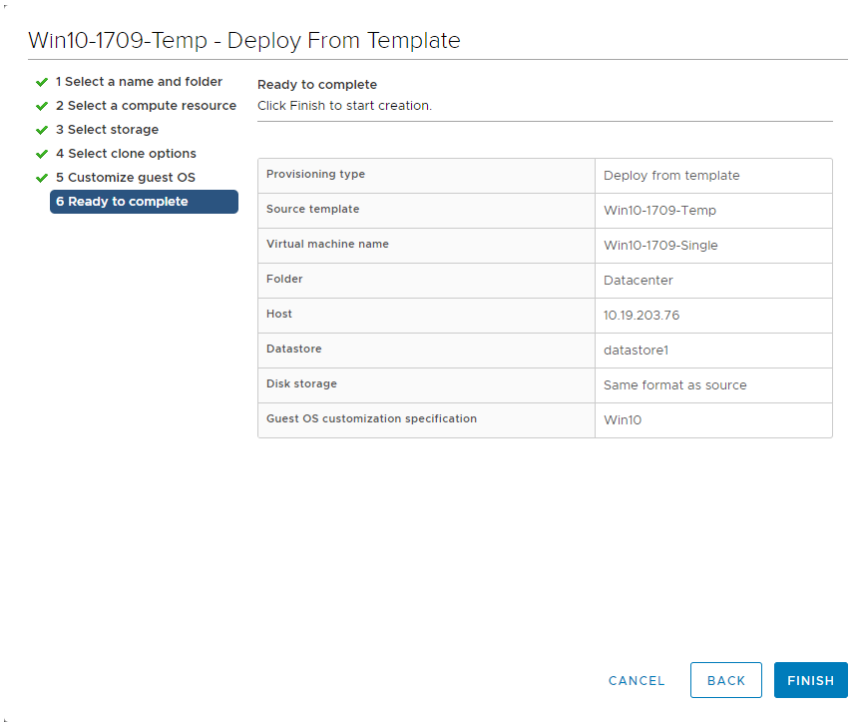
Check the **Customize the operating system** checkbox.

Select **Next** to display the **Customize guest OS** tab.



Select the **template** created on previous section.

The **Ready to complete** tab of the ***Deploy From Template*** window displays with the new customization specification listed.



Review the settings and select **Finish** to begin creating the virtual machine from the template.

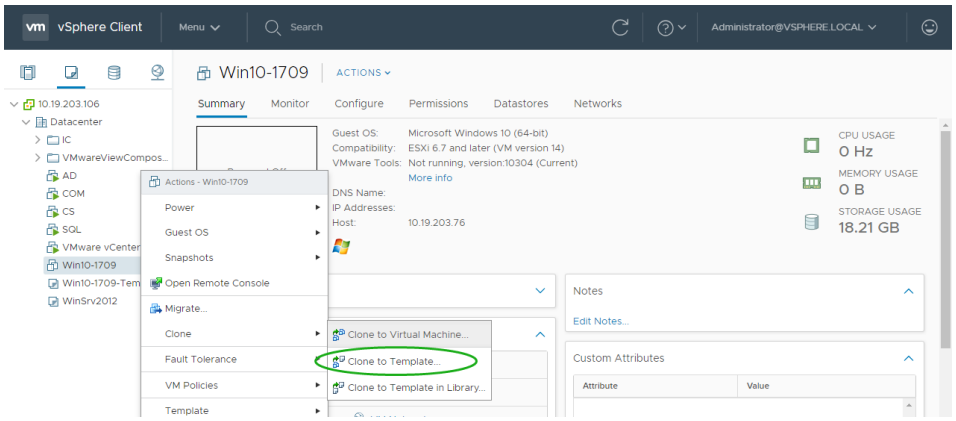
The cloned virtual machine is now visible in the datacenter and on the host that it was cloned to. Adding the new virtual machine to a Horizon View Desktop Pool and then entitling users and/or groups is performed as described in section 10.6, Entitling Users Access to Desktop Pools.

Horizon View can leverage templates to automatically create virtual machines on demand for time and resource saving functionality. Refer to the *Horizon View* documentation for instructions on using this functionality.

The next sections describe how to install desktop pools and then grant entitlements to users and groups to use that pool. Desktop pools can be created from a Goldmaster Image with either Full Clones or with Linked Clones with the help of Horizon View Composer. In the following sections both use cases will be explained

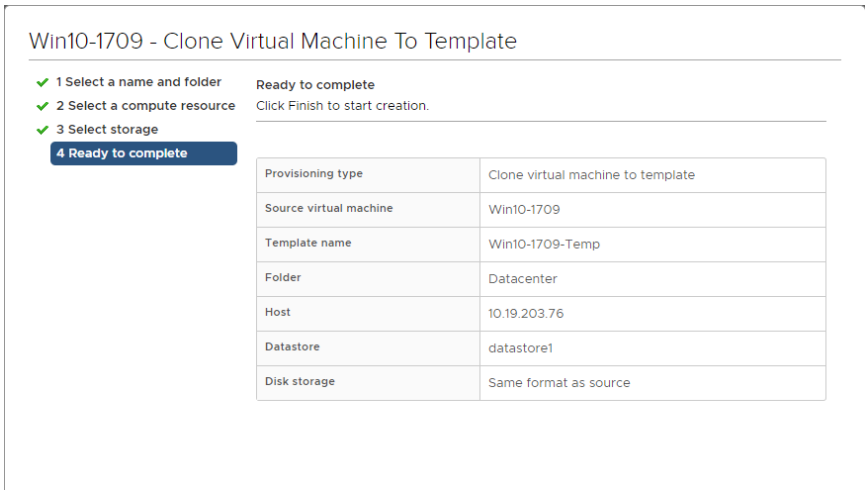
10.4 CREATING FULL CLONE DESKTOP POOLS

1. Full clone desktop pools require a Template of the Goldmaster image. Select the reference VM in the Navigator and right click. Select Clone, Clone to Template. *(A template can also be created by Converting a VM to a Template use the Template option.)*

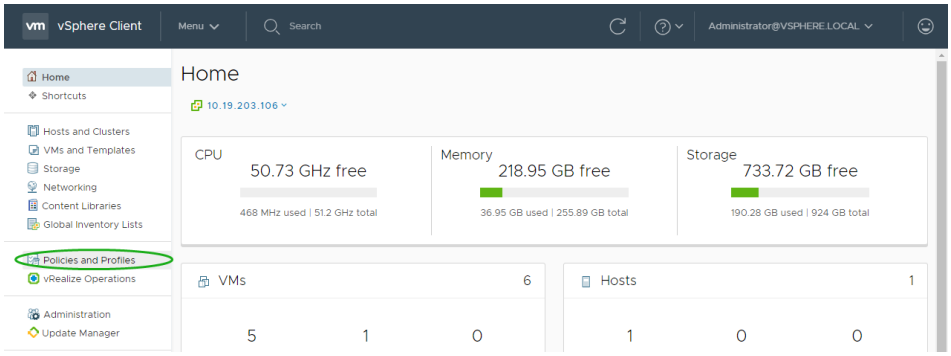


2. The **Clone Virtual Machine To Template** wizard walks you through the steps, described in section 10.1.

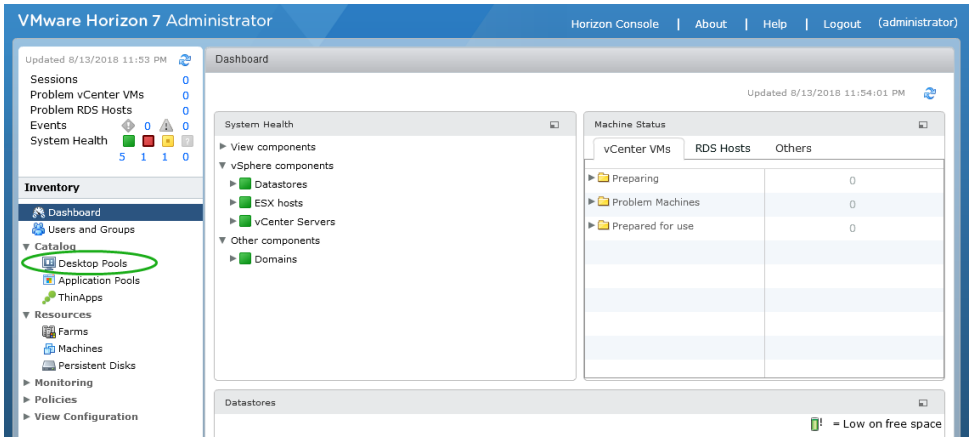
3. Finish to create the Template.



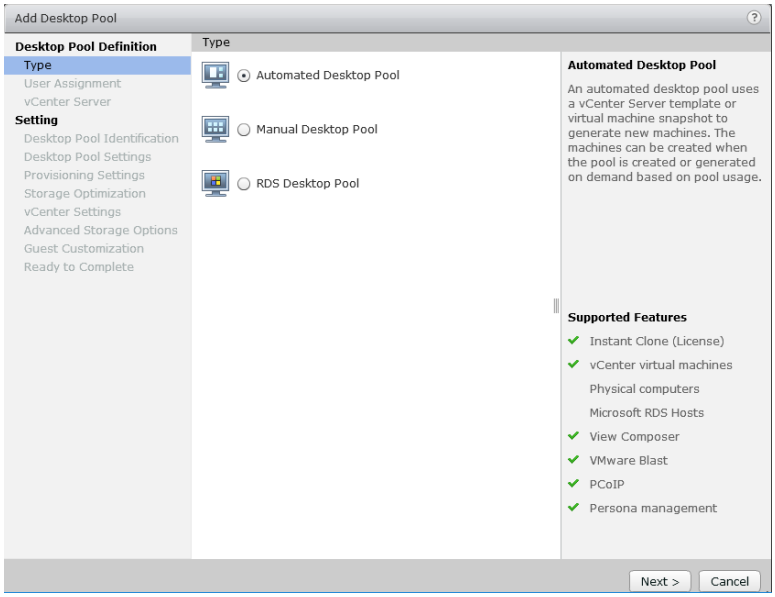
4. Create a Customization Specification file to deploy a full clone pool. The **New VM Guest Customization Spec** Wizard walks you through the steps, described in section 10.2.



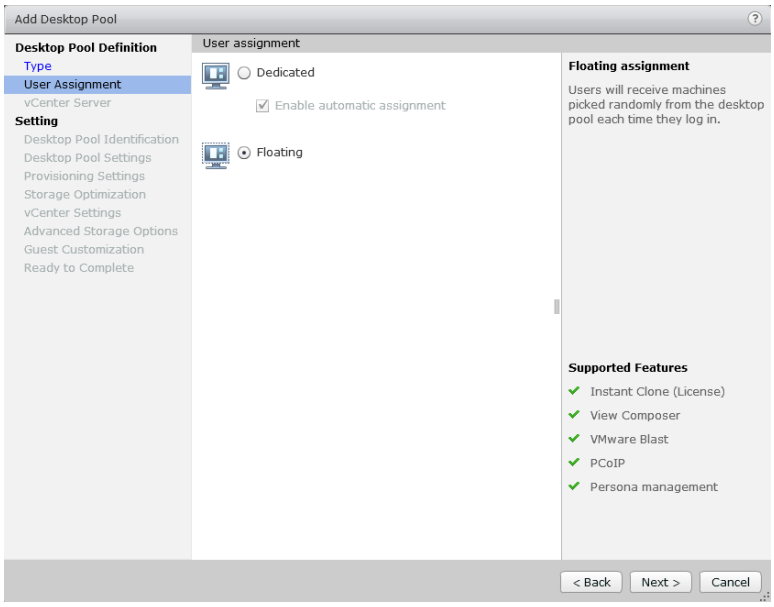
5. Login to the Horizon View Manager. Select **Desktop Pools** in the Inventory window and Click on Add.



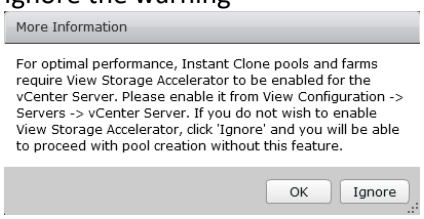
6. Select **Automated Pool**



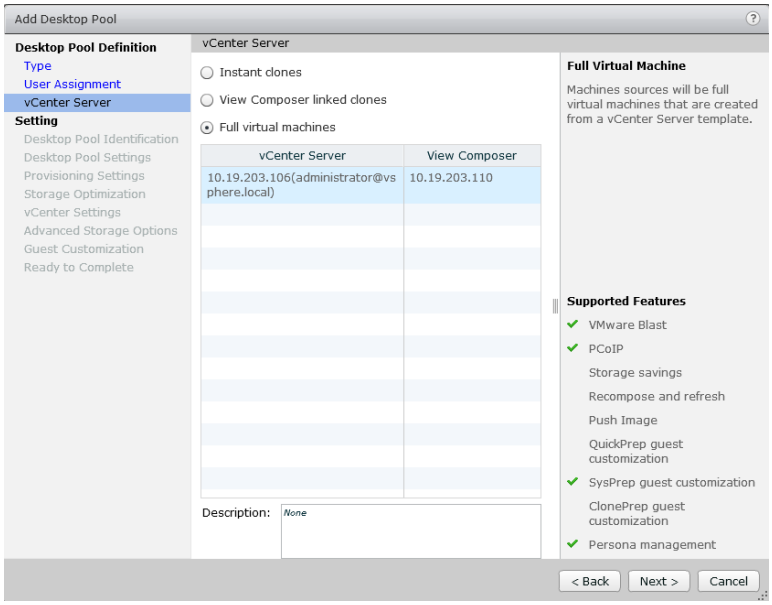
7. Select **Floating** User Assignment



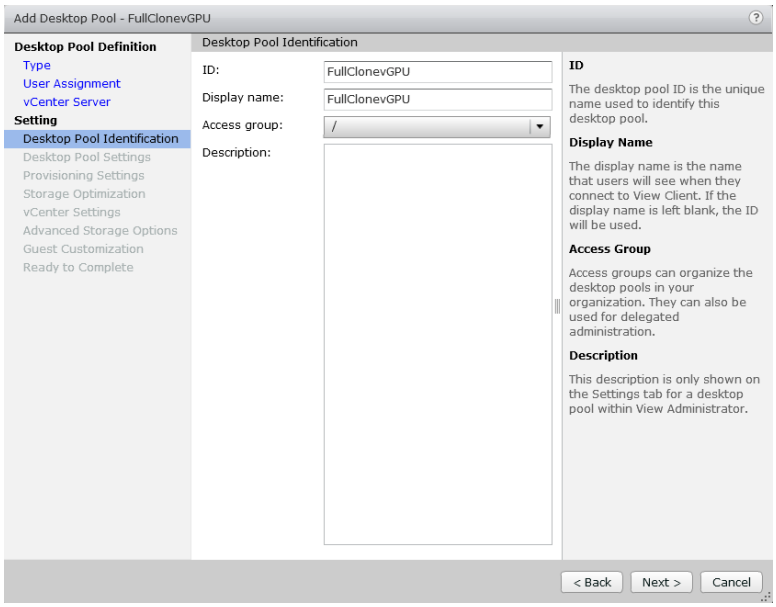
Ignore the warning



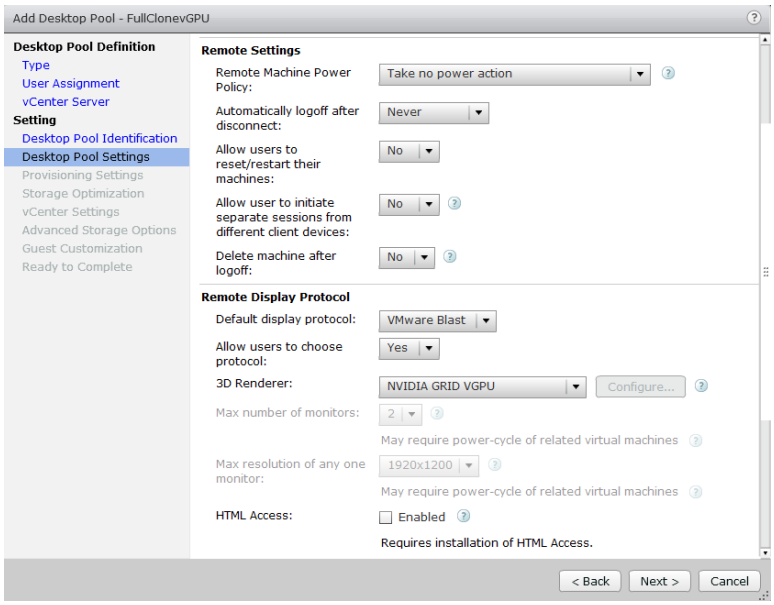
8. Select Full Virtual Machines.



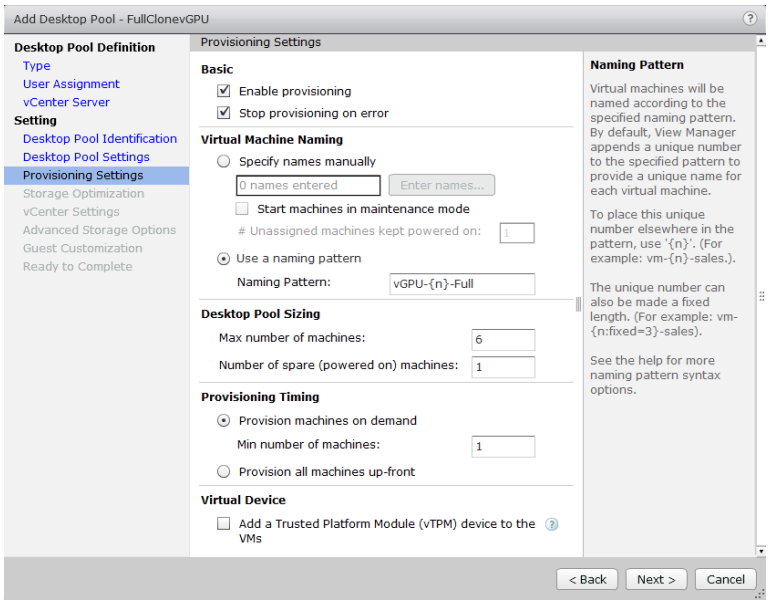
9. Name the Desktop Pool.



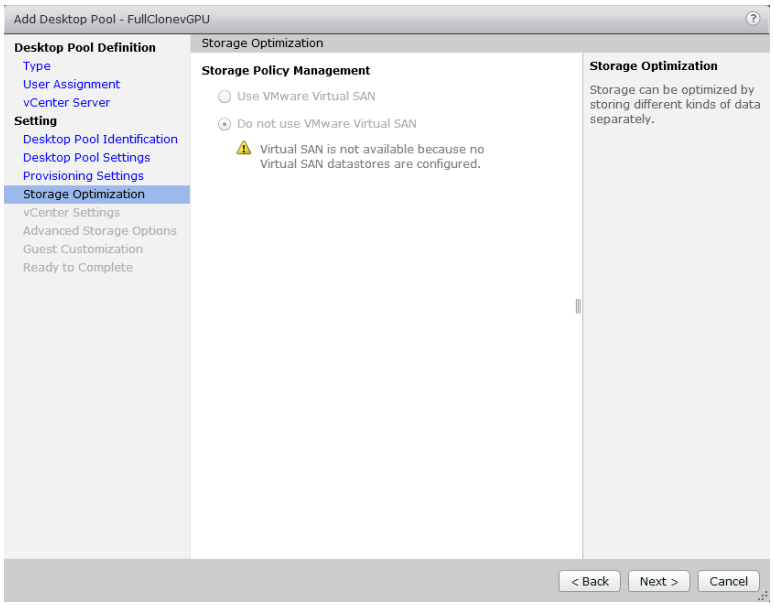
10. In Desktop Pool Settings, Set Default display protocol to Blast, allow users to decide to Yes, and 3D Renderer to NVIDIA GRID VGPU. *(HTML access is supported in this version if it is desired.)*



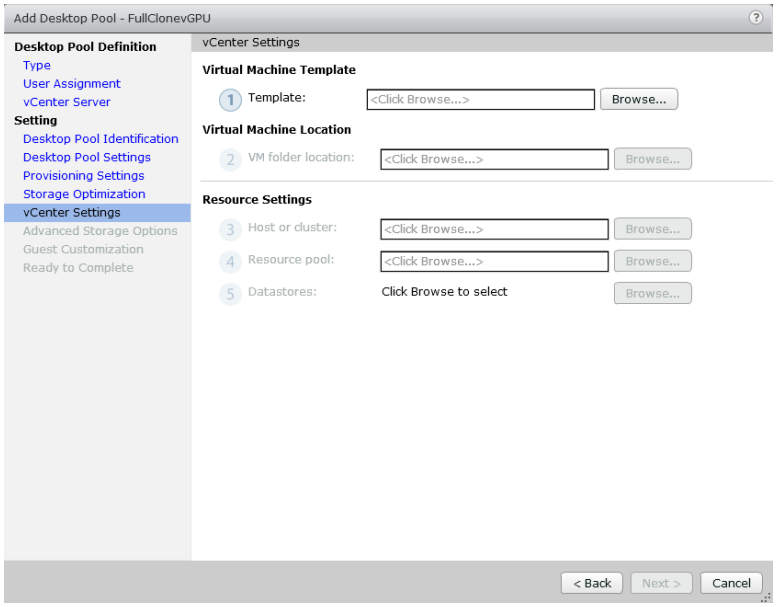
11. In Provisioning Settings, Click on **Select a naming pattern**, enter a naming pattern that will help differentiate the VMs in this pool from other pools you create. Choose **Desktop Pools Sizing** and **Provisioning Timing** to values that make sense for your infrastructure.



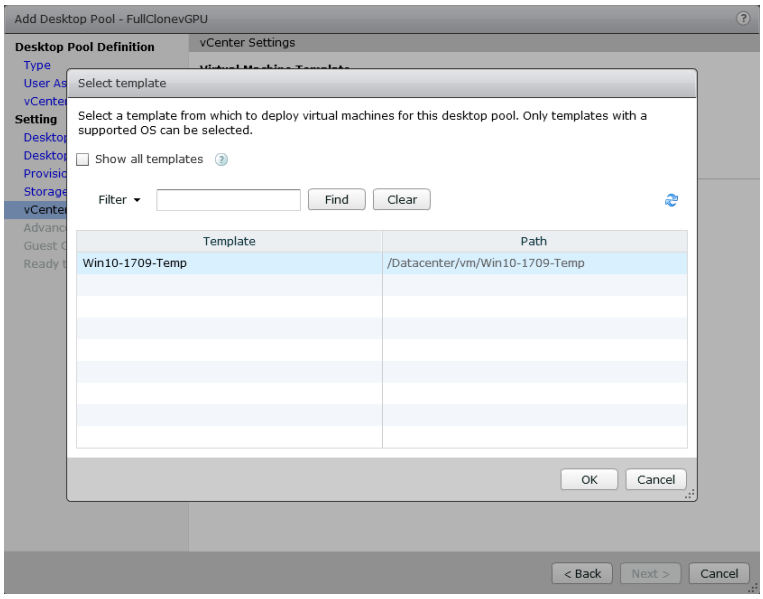
12. Storage Policy Management options available when Virtual SAN is configured.



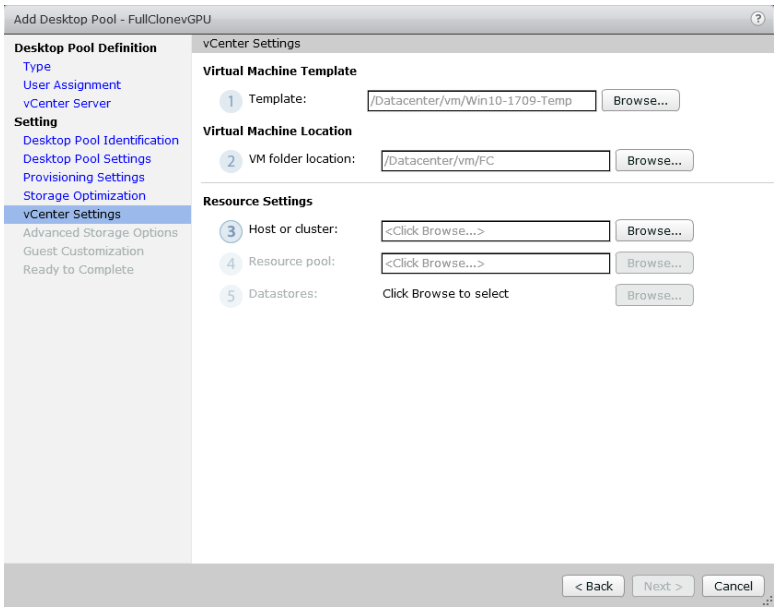
13. Select Browse to locate Template file.



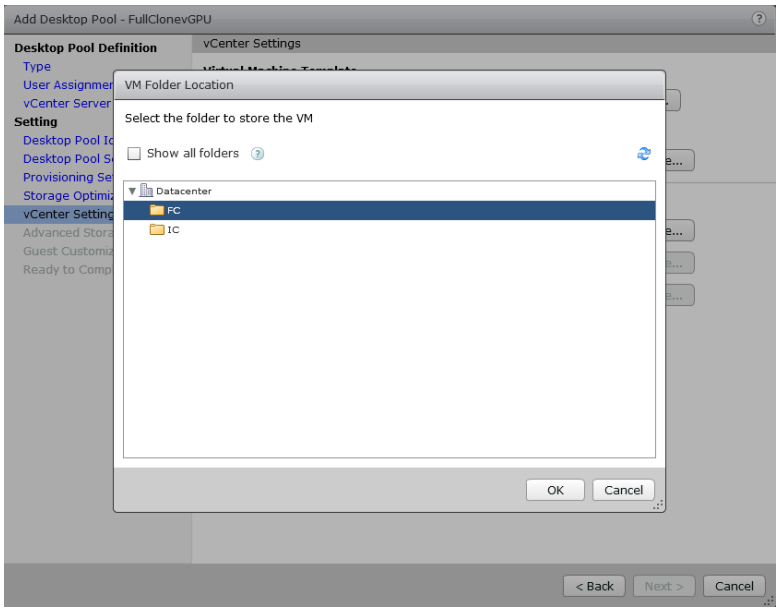
14. Select the Template created earlier for the Goldmaster Image. Click Next to continue.



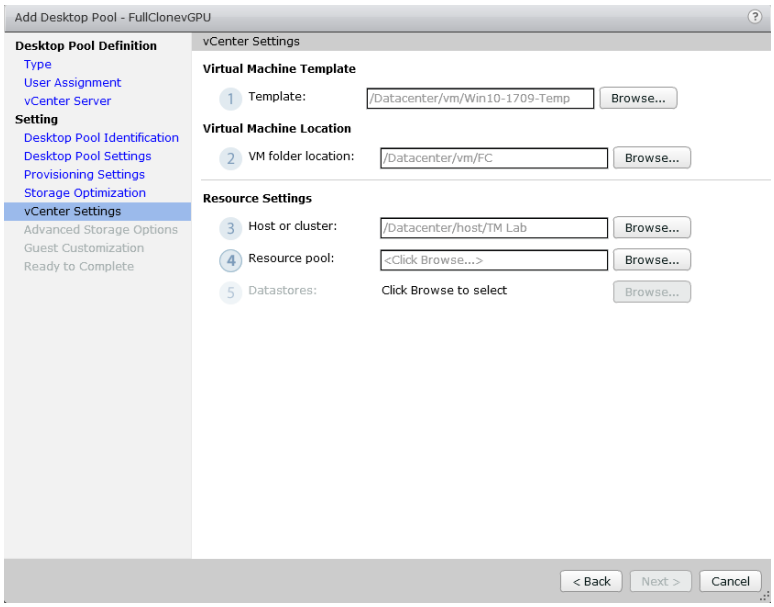
15. Browse for VM folder location. Click Next to continue.



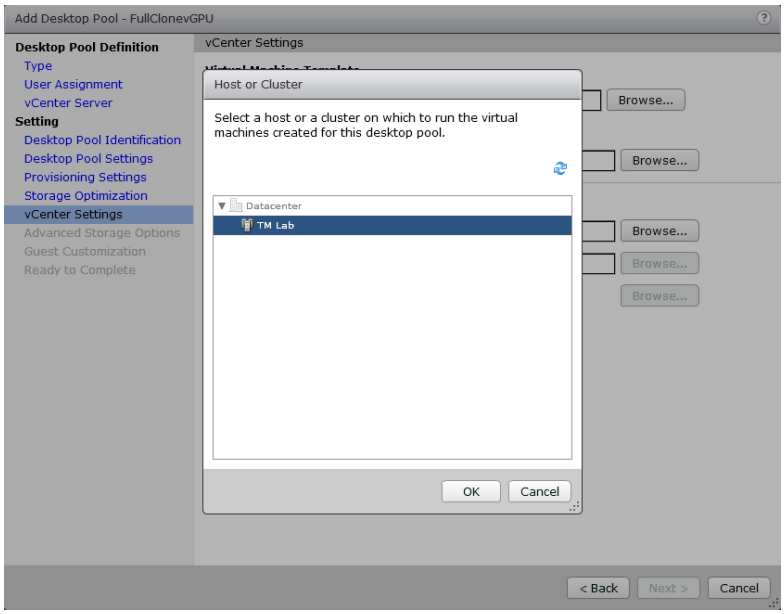
16. Select a folder to store the VMs



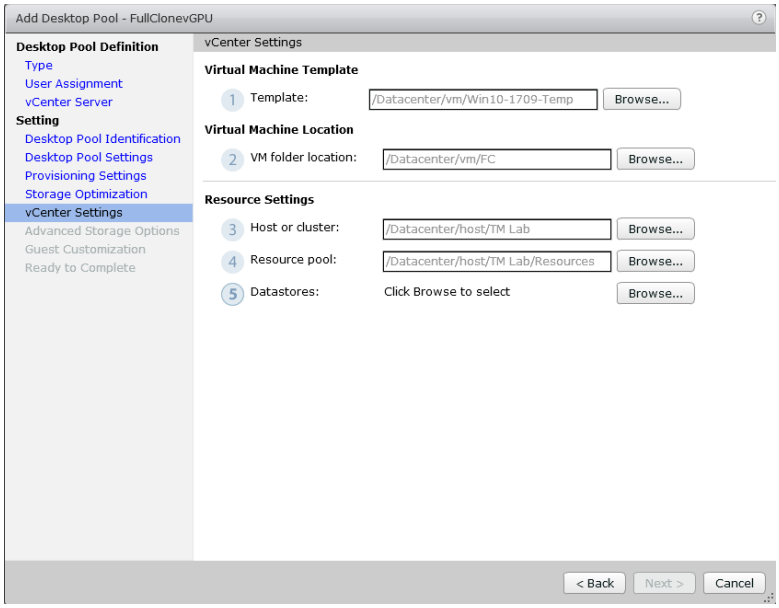
17. Browse for Host or Cluster.



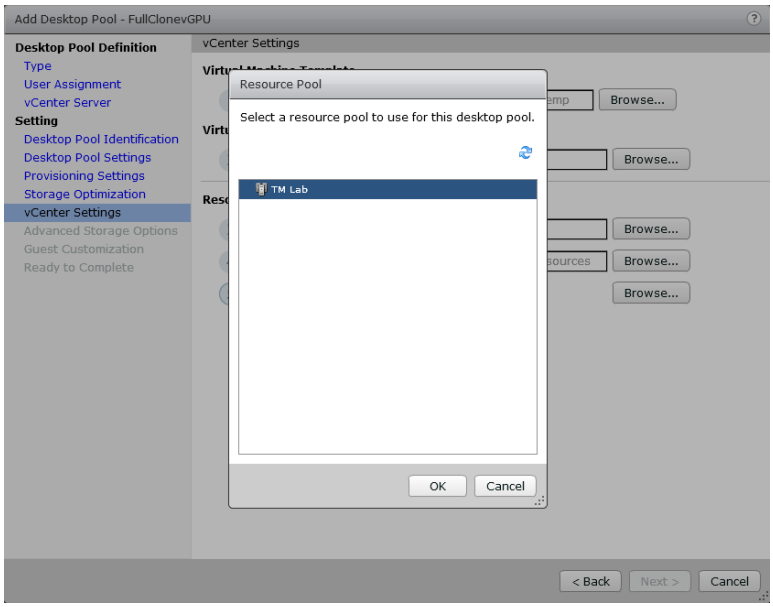
18. Select a Host or Cluster.



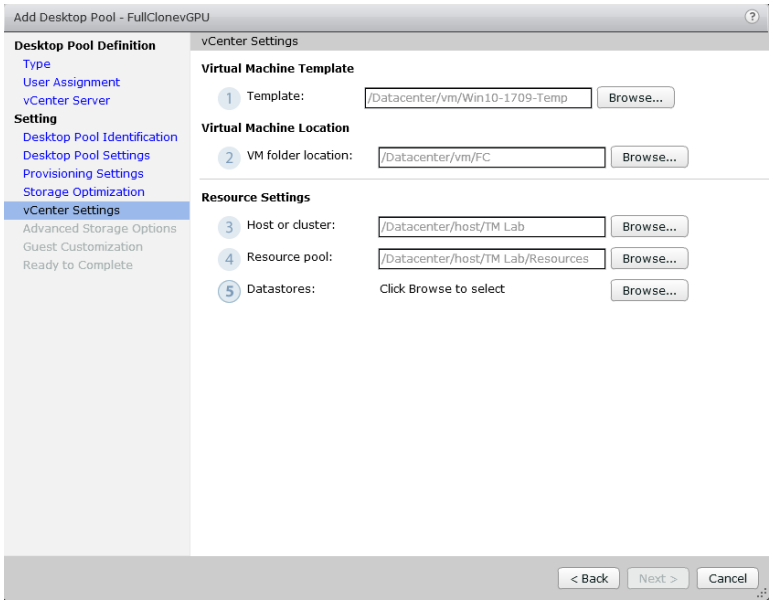
19. Browse for a Resource Pool.



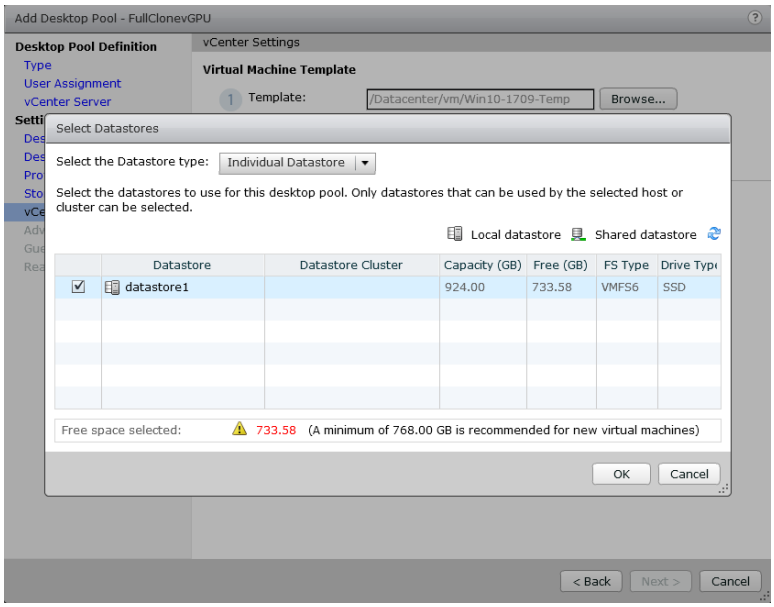
20. Select a Resource Pool.



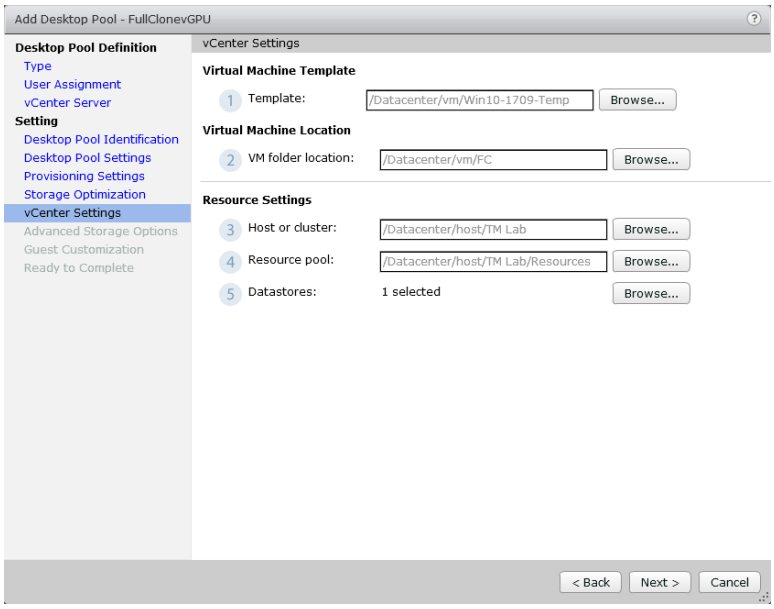
21. Browse for a Datastore for the Desktop Pool.



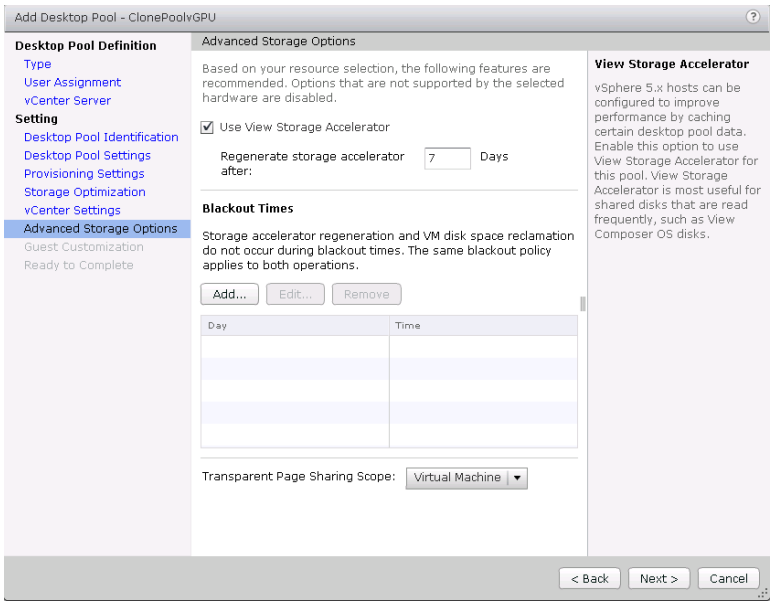
22. Select Datastores.



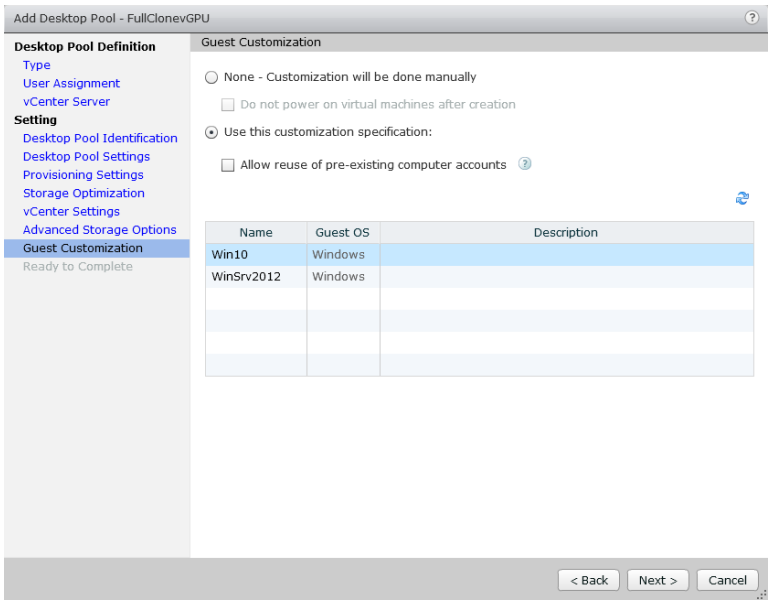
23. Review Settings and click Next to continue.



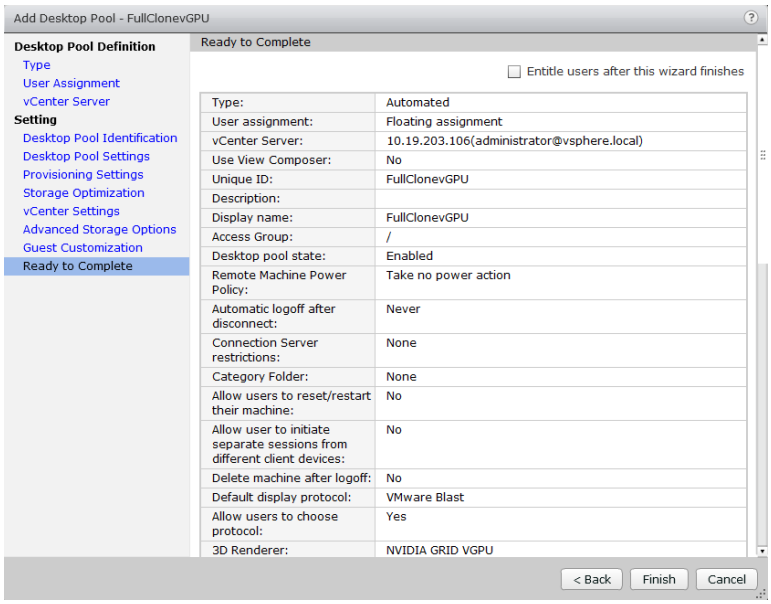
24. In advanced settings, click View Storage Accelerator if available. Choose blackout times for non-refresh of View Storage Accelerator cache to minimize impact on production users.



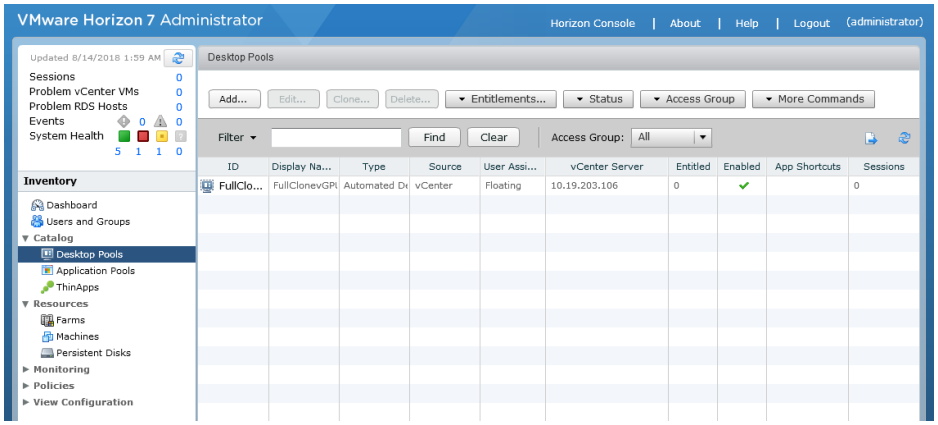
25. Select Use this customization specification to simplify deployment. Click Next to continue.



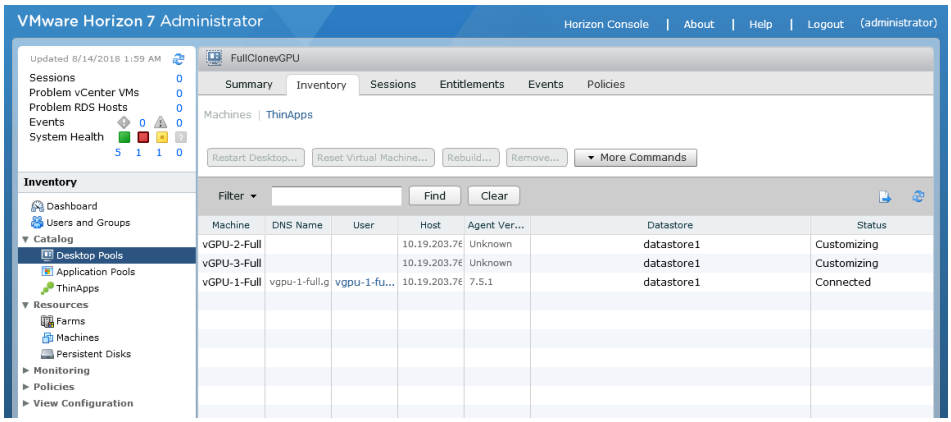
26. Review the selections, Click Finish to deploy the Desktop Pool.



27. Monitor Pool for status.



28. Look under Status on the far right column for “Connected”. This means VDI desktops are ready for access.

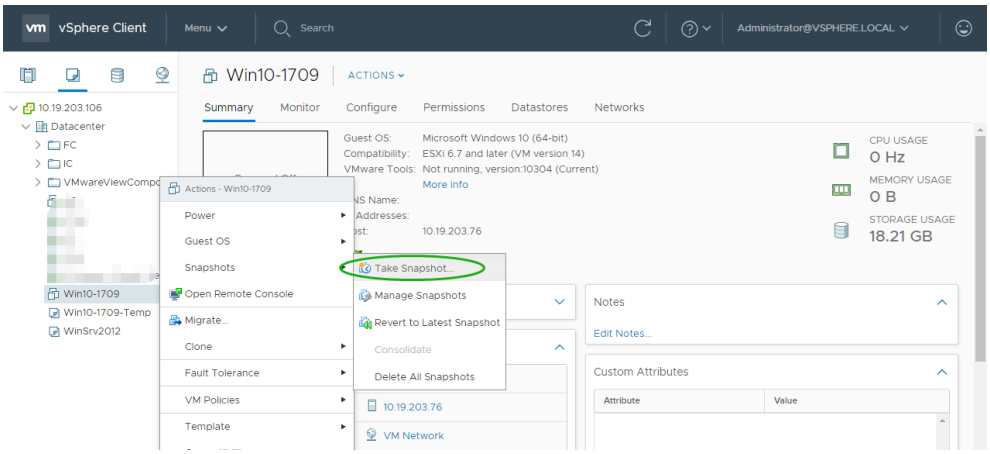


10.5 CREATING LINKED CLONE DESKTOP POOLS

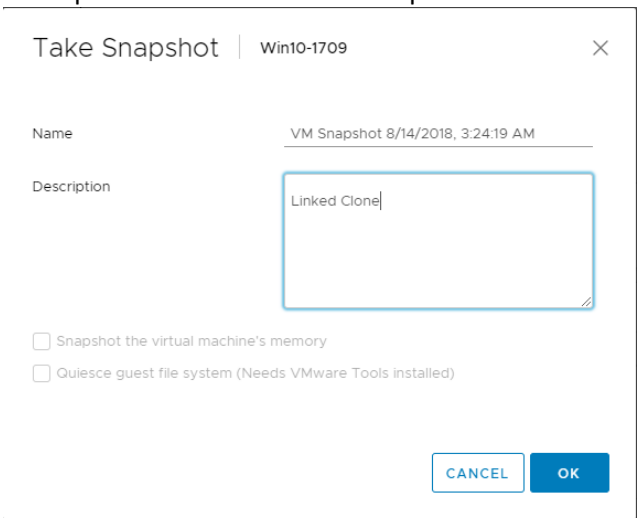
Creating a linked pool of VDI desktops is very similar to the full clone process. A linked pool leverages the View Composer to deploy multiple desktops off of one Goldmaster Image. Linked clones consume less diskspace, are easlily managed, upgraded and deployed with little impact on the enduser.

Similar to Clone Pools, we select a VM as Goldmaster Image. However instead of using a template of the Goldmaster VM, we take a snapshot of it. *(Note: Run ipconfig /release in a command prompt prior to shutting down the Goldmaster VM to take the snapshot.)*

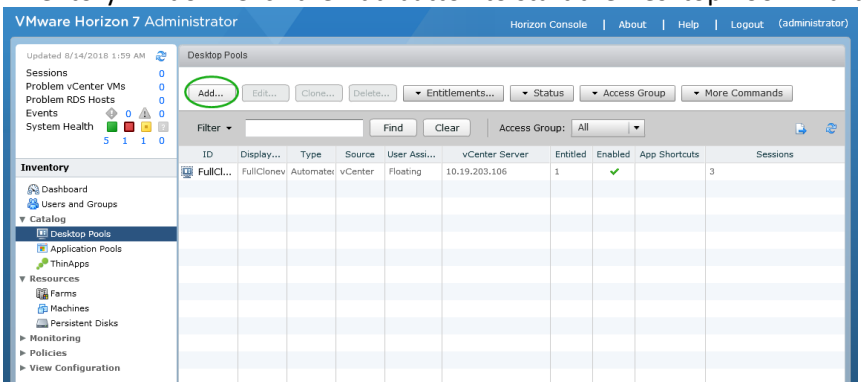
1. In the vSphere Web Client right click the Goldmaster VM. Navigate to Snapshots, Select Take Snapshot.



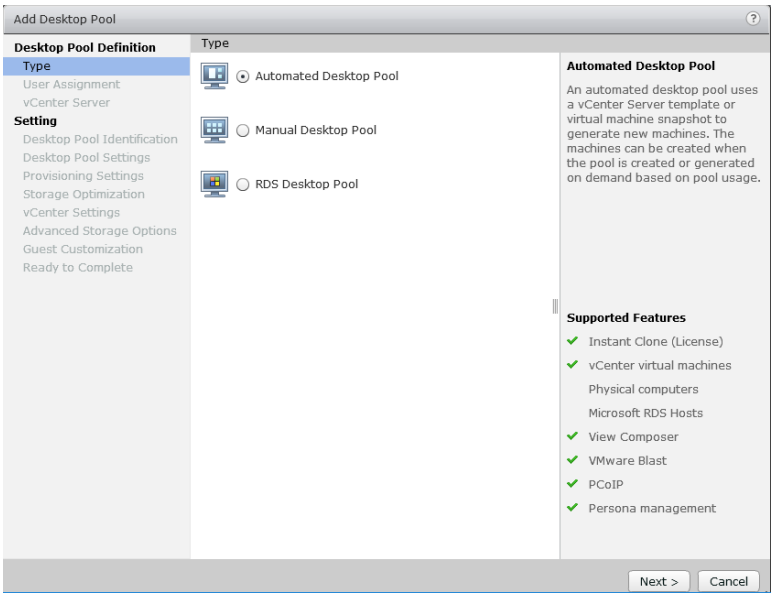
2. Name the Snapshot to conform to your VM naming standards with a meaningful description. Click OK to create snapshot.



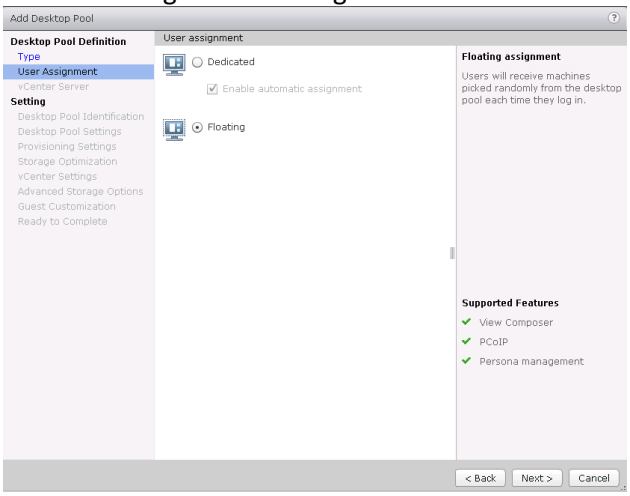
3. In the VMware Horizon View Management Console, Select Desktop Pools in the Inventory window. Click the Add button to start the Desktop Pool wizard.



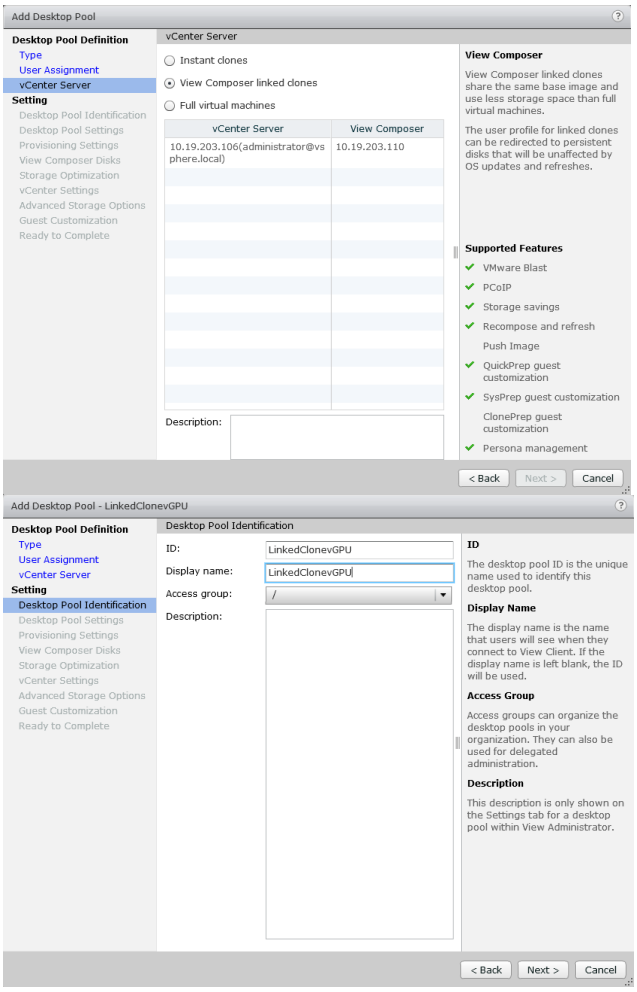
4. Select Automated Desktop Pool. Click on Next to continue.



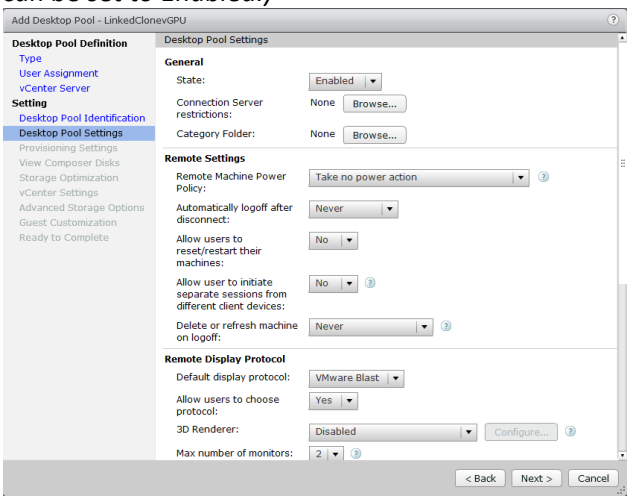
5. Select Floating for User Assignment. Click on Next to continue.



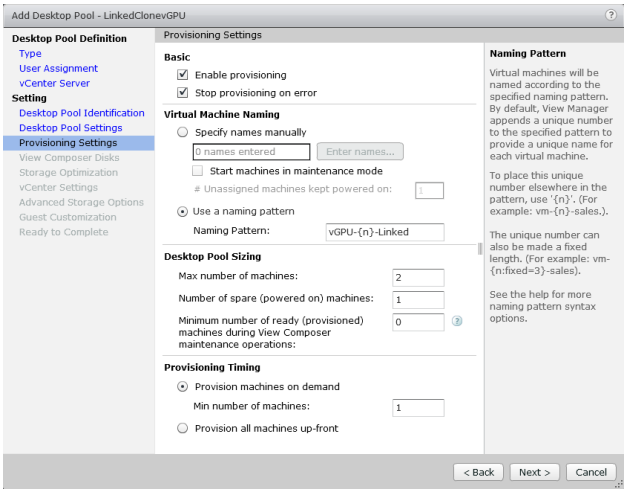
6. Select View Composer linked clones. Verify the vSphere Server and View Composer. Click on Next to continue.



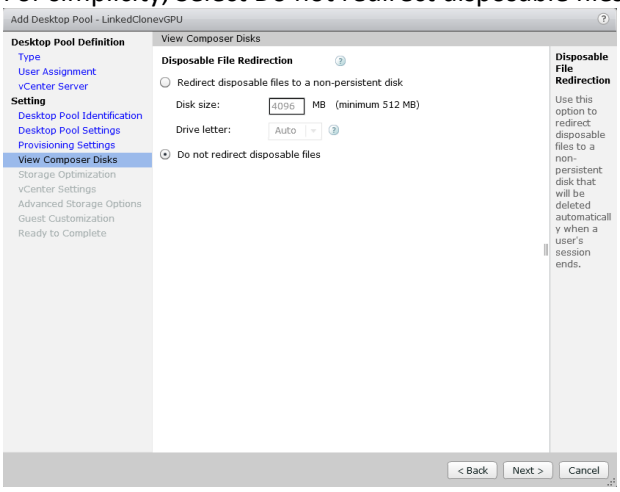
7. Set the Remote Display Protocol to Blast, Set Allow users to choose to Yes, Set 3D Renderer to NVIDIA GRID VGPU. *(In this version, HTML Access is fully supported and can be set to Enabled.)*



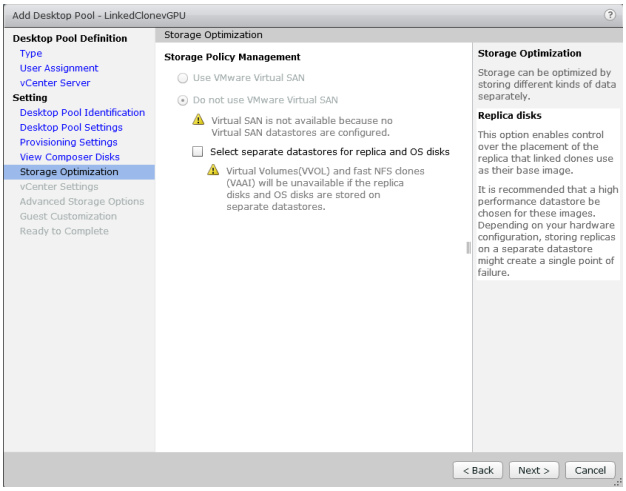
8. Configure the Provisions Settings to conform with your infrastructure and testing needs.



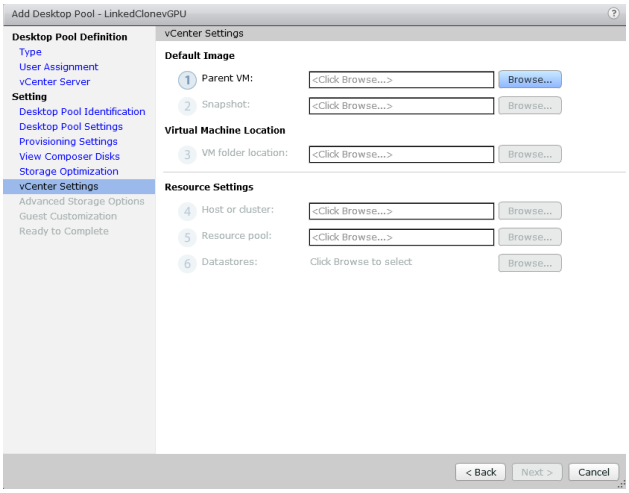
9. For simplicity, Select Do not redirect disposable files. Click on Next to continue.



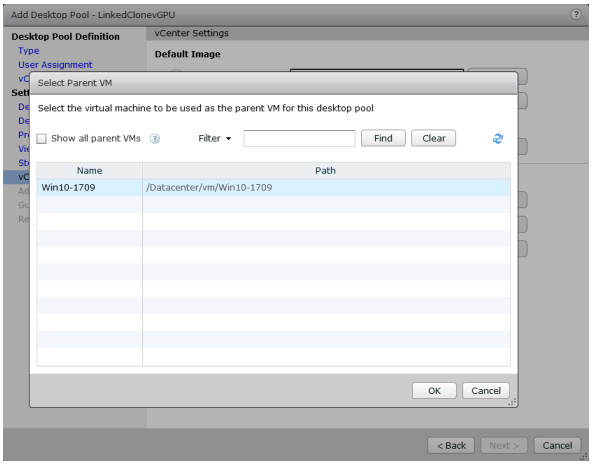
10. Click on Next to continue.



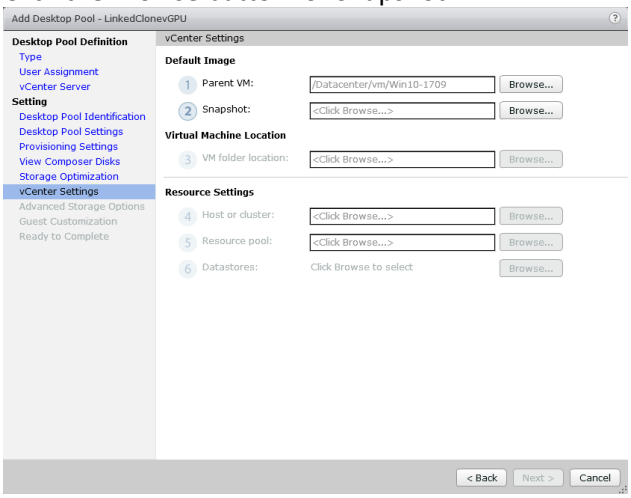
11. Click on the Browse button for the Parent VM



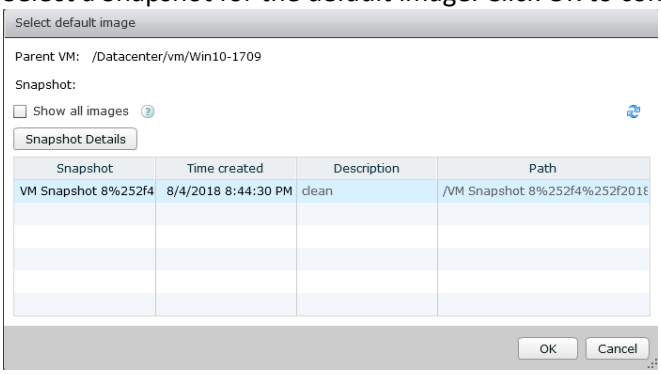
12. Select the Goldmaster VM. Use the Filter option help narrow the search. Click OK to continue.



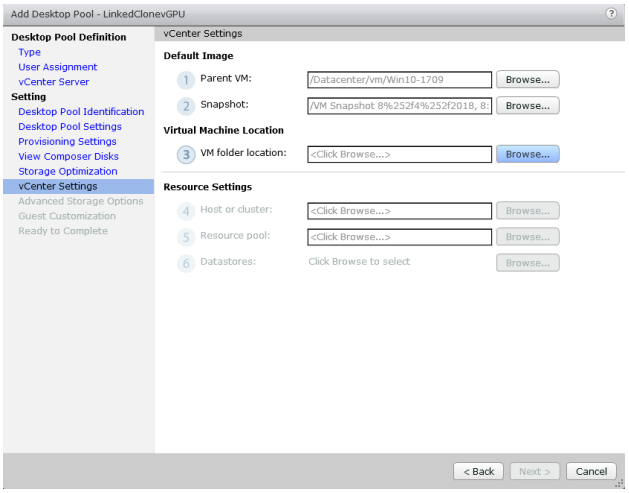
13. Click the Browse button for Snapshot



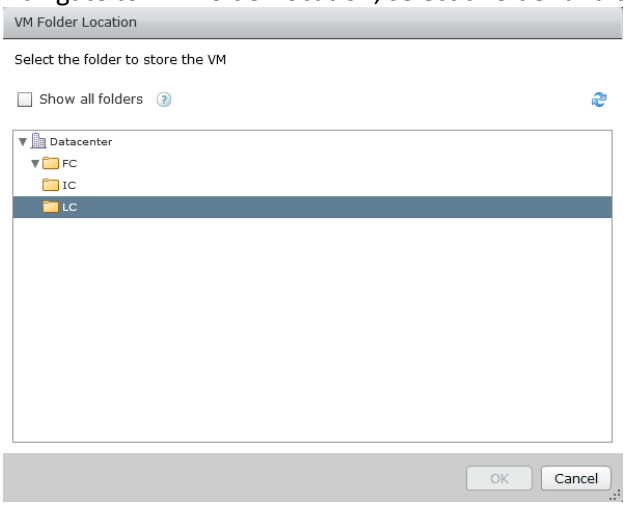
14. Select a Snapshot for the default image. Click OK to continue.



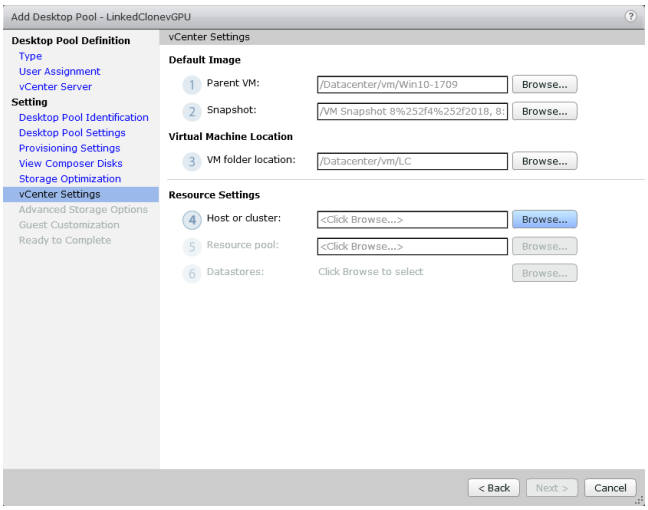
15. Click Browse for the VM folder location.



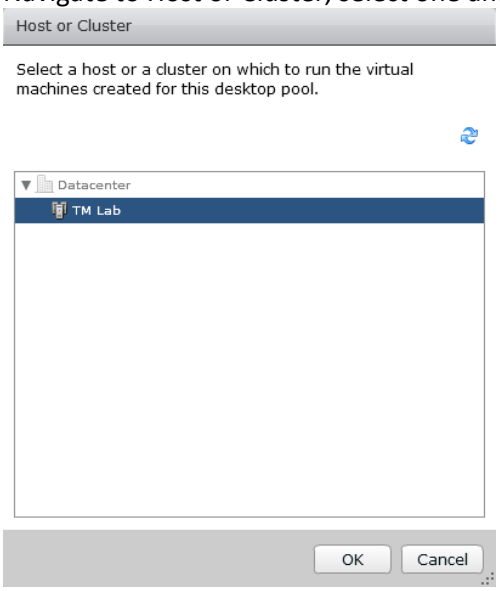
16. Navigate to VM Folder location, select a folder and click OK to continue.



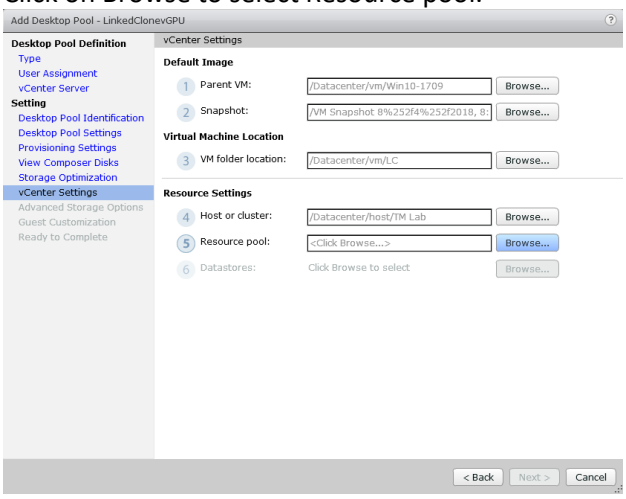
17. Click on Browse to Select a Host or Cluster



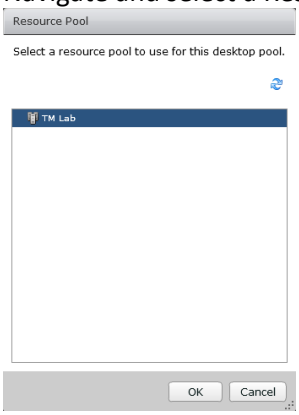
18. Navigate to Host or Cluster, select one and click OK to continue.



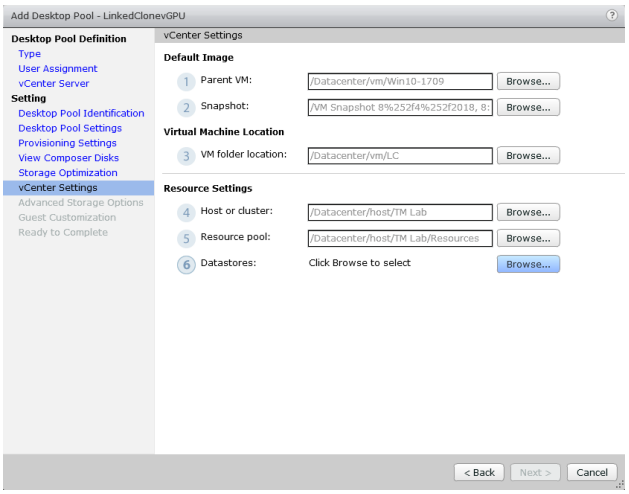
19. Click on Browse to select Resource pool.



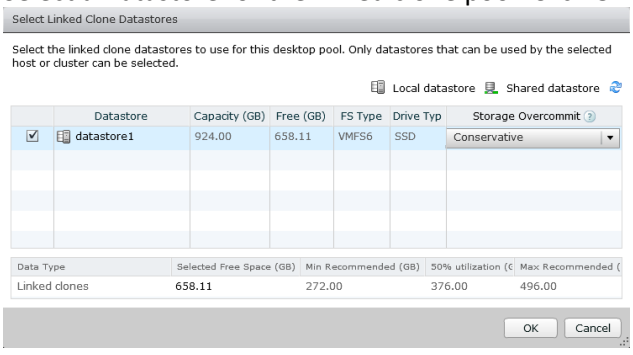
20. Navigate and select a Resource Pool. Click OK to continue.



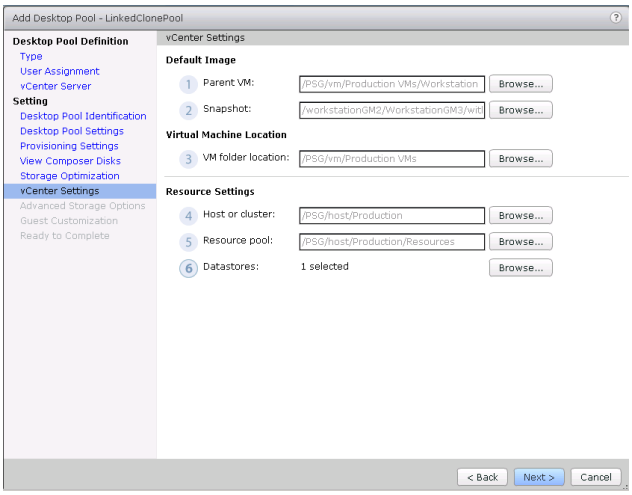
21. Click Browse to select the Datastore for the linked clone pool.



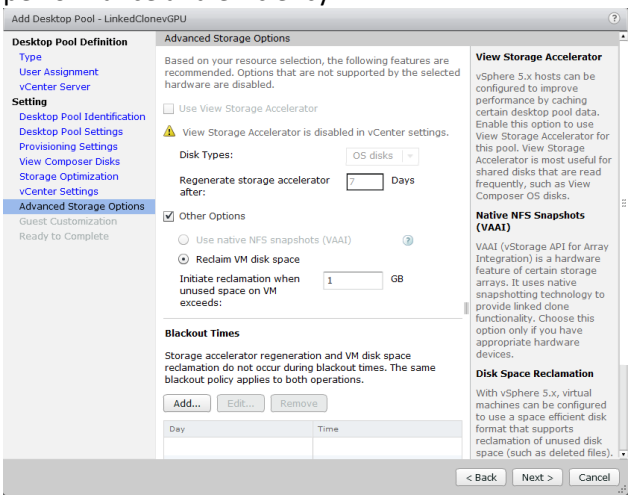
22. Select a Datastore for the linked clone pool. Click OK to continue.



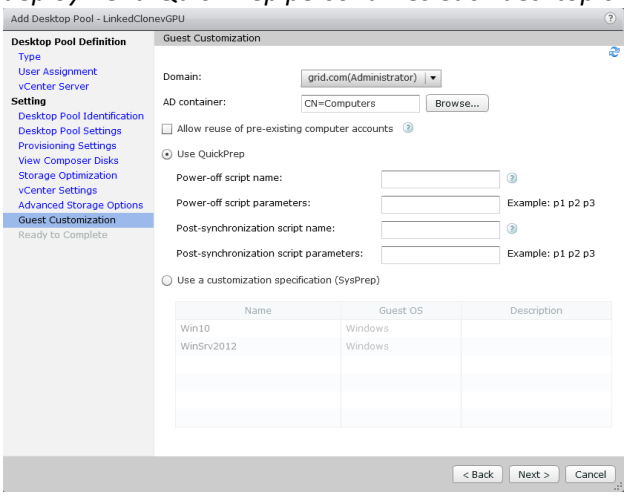
23. Click Next to continue.



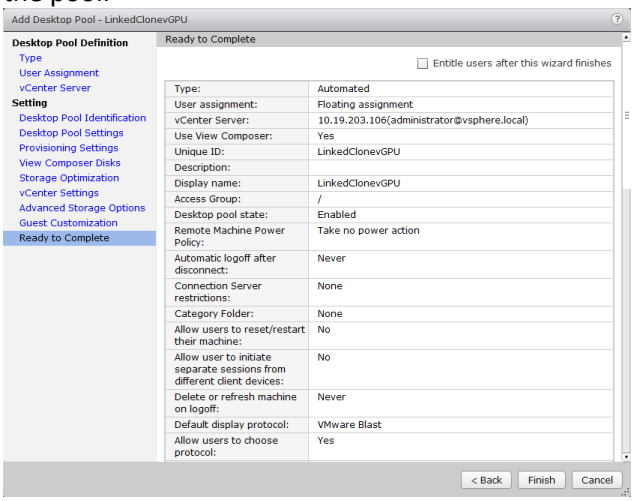
24. Enable View Storage Accelerator and Reclaim VM disk space if available to improve performance and efficiency.



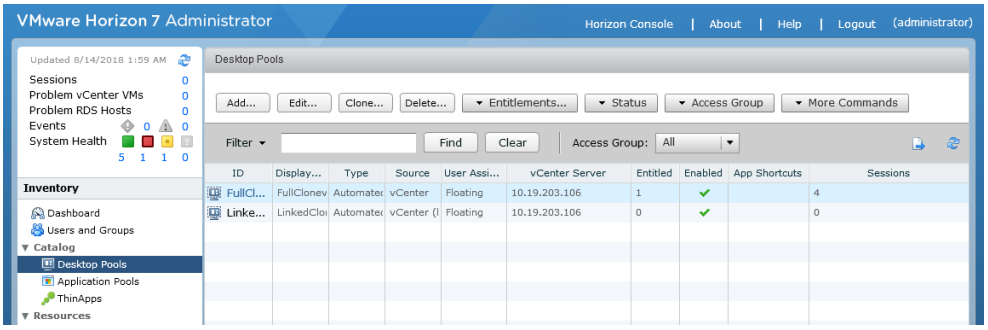
25. Select use QuickPrep for Guest Customization. Click Next to continue. *(QuickPrep is a VMware system tool executed by View Composer during a linked-clone desktop deployment. QuickPrep personalizes each desktop created from the Master Image.)*



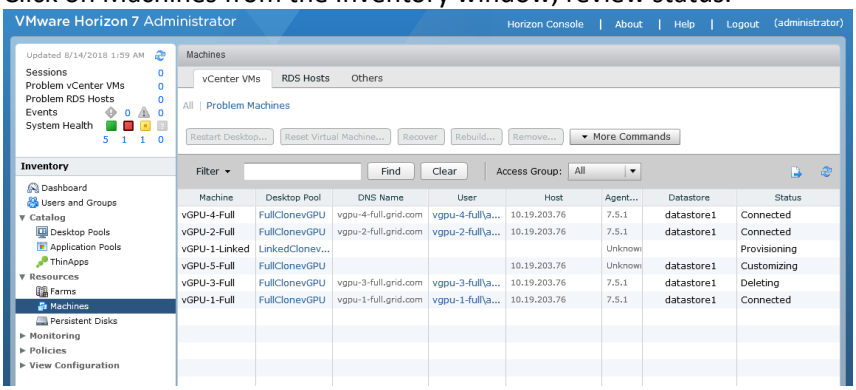
26. Review Add Desktop Pool settings. Click Back to fix any errors. Click finish to create the pool.



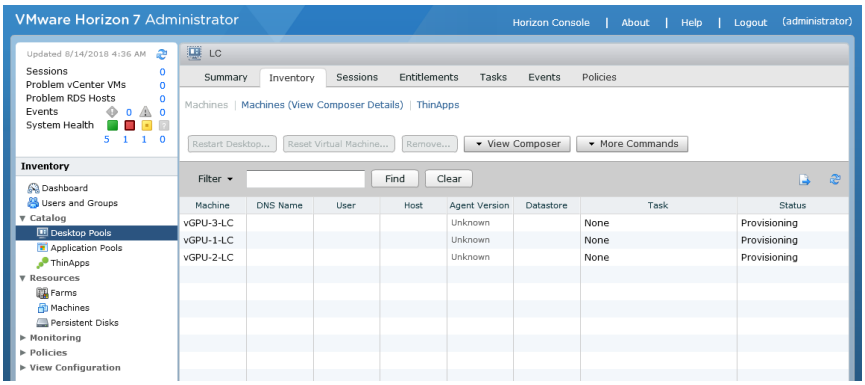
27. Monitor Linked Pool creation status. Refresh window using the recycle icon on left hand side.



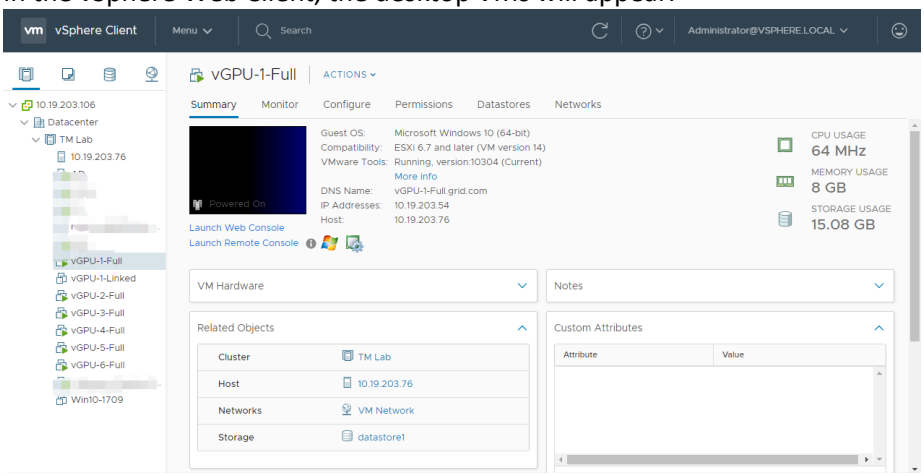
28. Click on Machines from the Inventory window, review status.



29. When status shows Connected, then VDI desktops are ready.

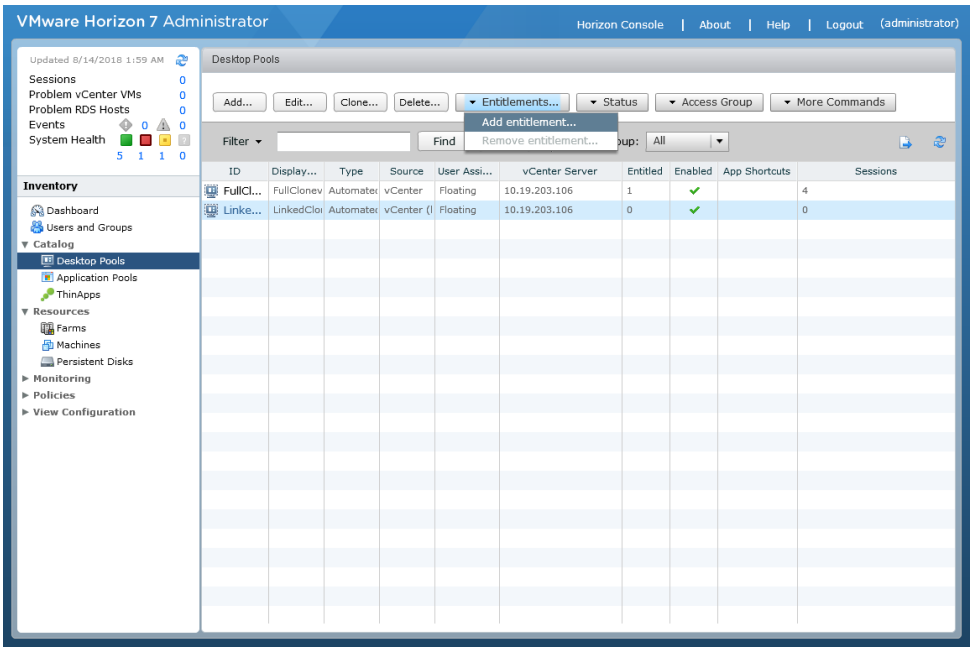


30. In the vSphere Web Client, the desktop VMs will appear.

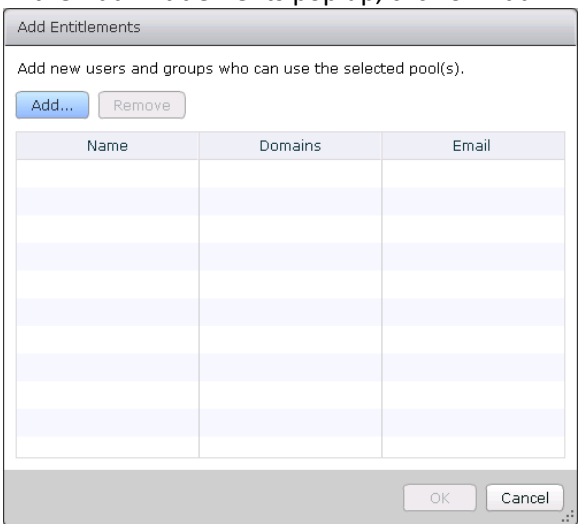


10.6 ENTITLING USERS ACCESS TO DESKTOP POOLS

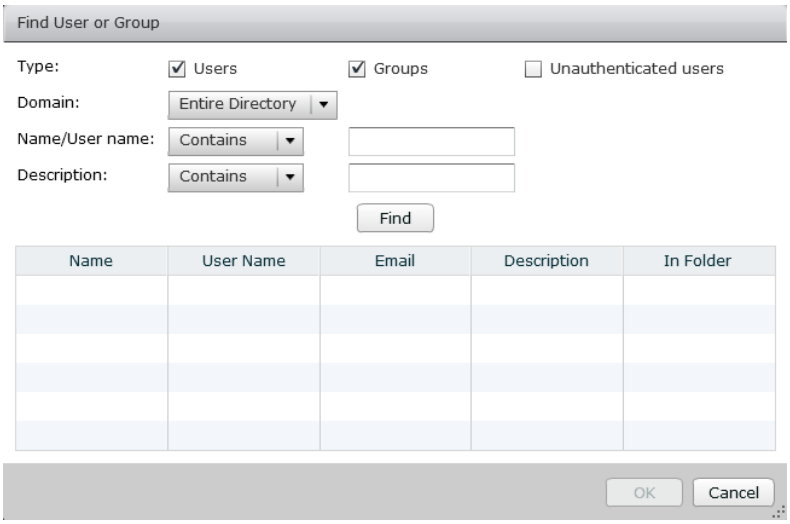
1. Entitle Users to access desktops in the pool. Select Add entitlement from the Entitlements button.



2. In the Add Entitlements pop up, click on Add.



3. Type in a string to search for users or groups. Highlight Users or Groups to Entitle in the search results below and click OK to add them.

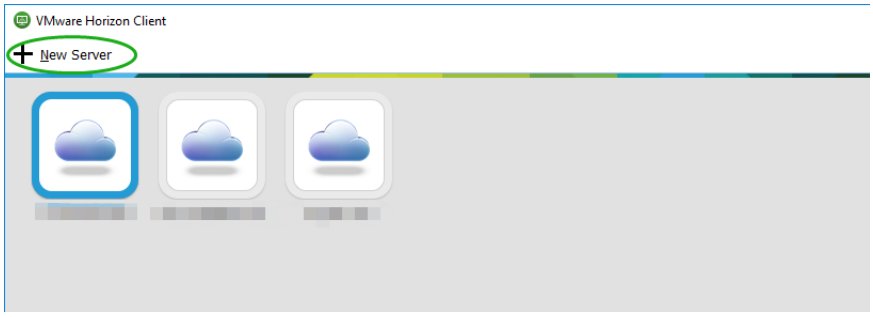


10.7 CONNECTING VMWARE HORIZON CLIENT TO THE VIEW CONNECTION SERVER

Use the following procedure to connect a VMware Horizon client to the View connection server:

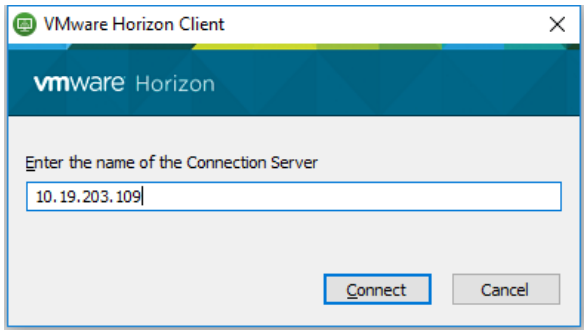
1. Launch the VMware Horizon client

Select the **New Server** (plus sign) icon.

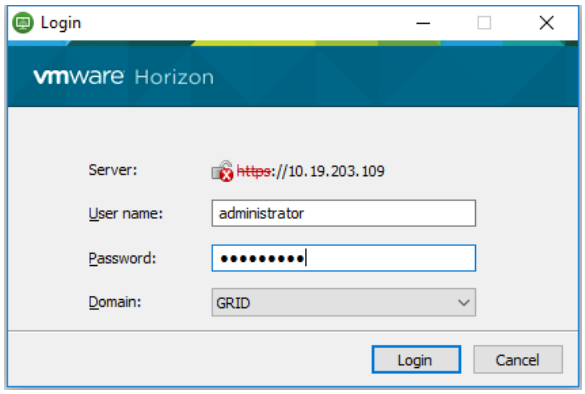


Enter the server name or IP address in the **Enter the name of the Connection Server** field.

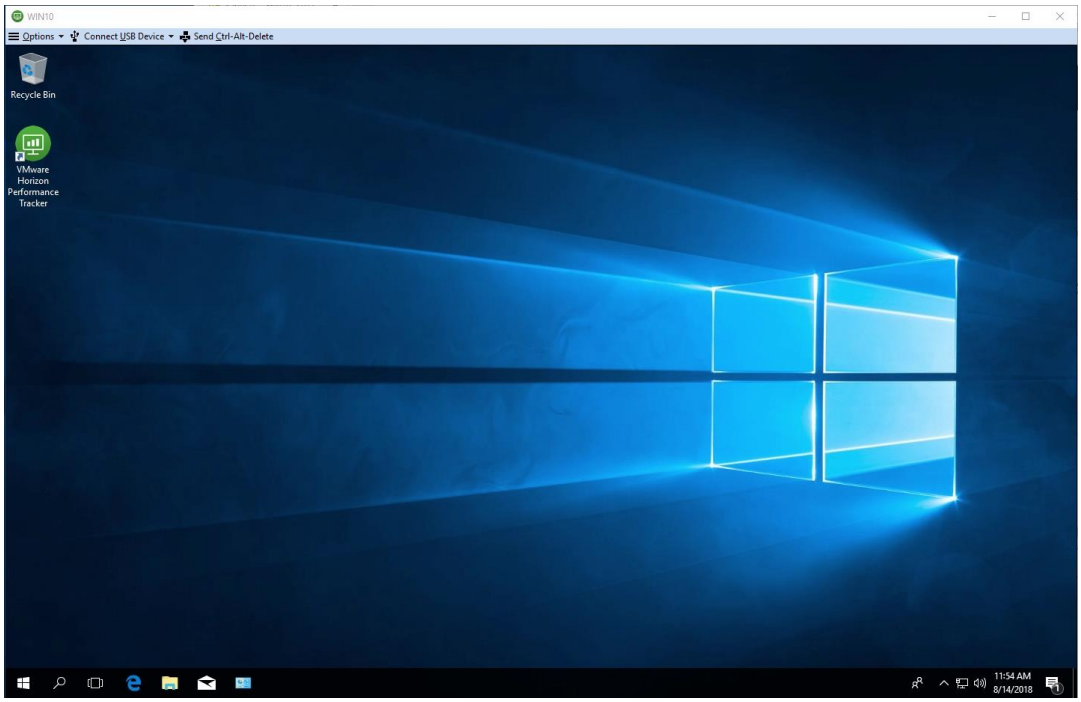
Select **Connect** to display the ***Login*** popup.



Enter the username and password of a user who is a member of the domain and who has been granted a desktop pool entitlement. Select **Login** to display a list of available vGPU-enabled desktop pool(s). Double-click the desktop you want to connect to.



Success, you are now logged in.


WIN10
Options
Connect USB Device
Send Ctrl-Alt-Delete
Recycle Bin
VMware Horizon Performance Tracker
11:54 AM
8/14/2018

Chapter 11. PROOF OF CONCEPT, MONITORING, & BENCHMARKING

This chapter describes the definition and deployment reasons for a Proof of Concept (POC) and benchmarking your NVIDIA vGPU installation, including:

- Proof of Concept
- NVIDIA System Management Interface (NSMI)
- NVIDIA Management Library (NVML)
- Testing and simulation applications and tools

11.1 PROOF OF CONCEPT

It's critical before purchasing and deploying to drive a proof of concept within your unique environment first. NVIDIA recommends identifying users within your organization to two groups: Those that require a Dedicated Quality of Service and those that require Good Quality of Service and Highest User Density. The two are split by the following definitions:

- DEDICATED QUALITY OF SERVICE:

  The equal and fixed share GPU scheduling policies provide equal GPU performance across all vGPUs sharing the same physical GPU (please refer to Chapter 7 for further details per Tesla GPU). Dedicated performance simplifies the POC since it allows the use of commonly used benchmarks for physical workstations, like SPECViewPerf, to compare the performance with the end users' current workstation (physical or virtual).

- GOOD QUALITY OF SERVICE AND HGHEST USER DENSITY:

The most successful POCs reaching the highest user densities with the lowest total cost of ownerships are performed when actual end users use their Quadro vDWS virtual machines in production while objective measurements and subjective feedback from the end user is gathered. Benchmarks like SPECViewPerf should only be used to determine temporary performance impacts that can occur when many other virtual machines are executing GPU intensive tasks like zooming and panning in parallel. Benchmarks typically stimulate these peak workloads which mimic the times where highest performance across all virtual machines are required. They do not consider the times when the system is not fully loaded which hypervisors and the best effort GPU scheduling policy leverage for higher user densities at similar performance.

11.2 TESTING VIRTUAL VS PHYSICAL DESKTOPS

Benchmarking and testing within an open virtual environment vs. on a closed desktop environment are different as there are efficiencies and administration needs that are also addressed along the way. A desk side solution is a single solution that can be optimized for a single use however, a virtual solution can not only be optimized for that same workload but also can be reconfigured for and optimized for other workloads. Please keep these ideas in mind while mapping out your environment and running the appropriate tests.

Addressing the challenges in today's virtualized environments require visibility into all phases of the virtual workspace lifecycle – from design and assessment to operations and support. Each phase has its own unique challenges:

- Design and Assessment: IT architects need to be able to right-size their VDI environment for the best end-user experience and ROI.
- Operations: IT administrators need to efficiently deploy VDI with the best performance and minimal downtime, and proactively manage and monitor their environment.
- Support: IT help desk and admins need to provide timely VDI user support and issue resolution.

11.3 NVIDIA SYSTEM MANAGEMENT INTERFACE

The NVIDIA System Management Interface (NSMI) is a command line utility based on the NVIDIA Management Library (NVML). The NVML helps you manage and monitor NVIDIA GPU devices by allowing you to query and (if you have the appropriate privileges) modify the GPU device state. GUI-based tool utilizes NVML and Windows Management Instrumentation (WMI)

to provide great visibility into the GPU. Information can be reported in either XML or human-readable plain text using either standard or file output.

At the host level, data can be accessed through NVML and nvidia-smi. NVML exposes the NVIDIA virtual GPU management SDK metrics to third-party vendors, while nvidia-smi is a management interface/command line tool that allows software programs or management and monitoring tools to interact with GPUs on the server.

Refer to the *NSMI* documentation and see Chapter 12, Troubleshooting, on page 261 for more information.

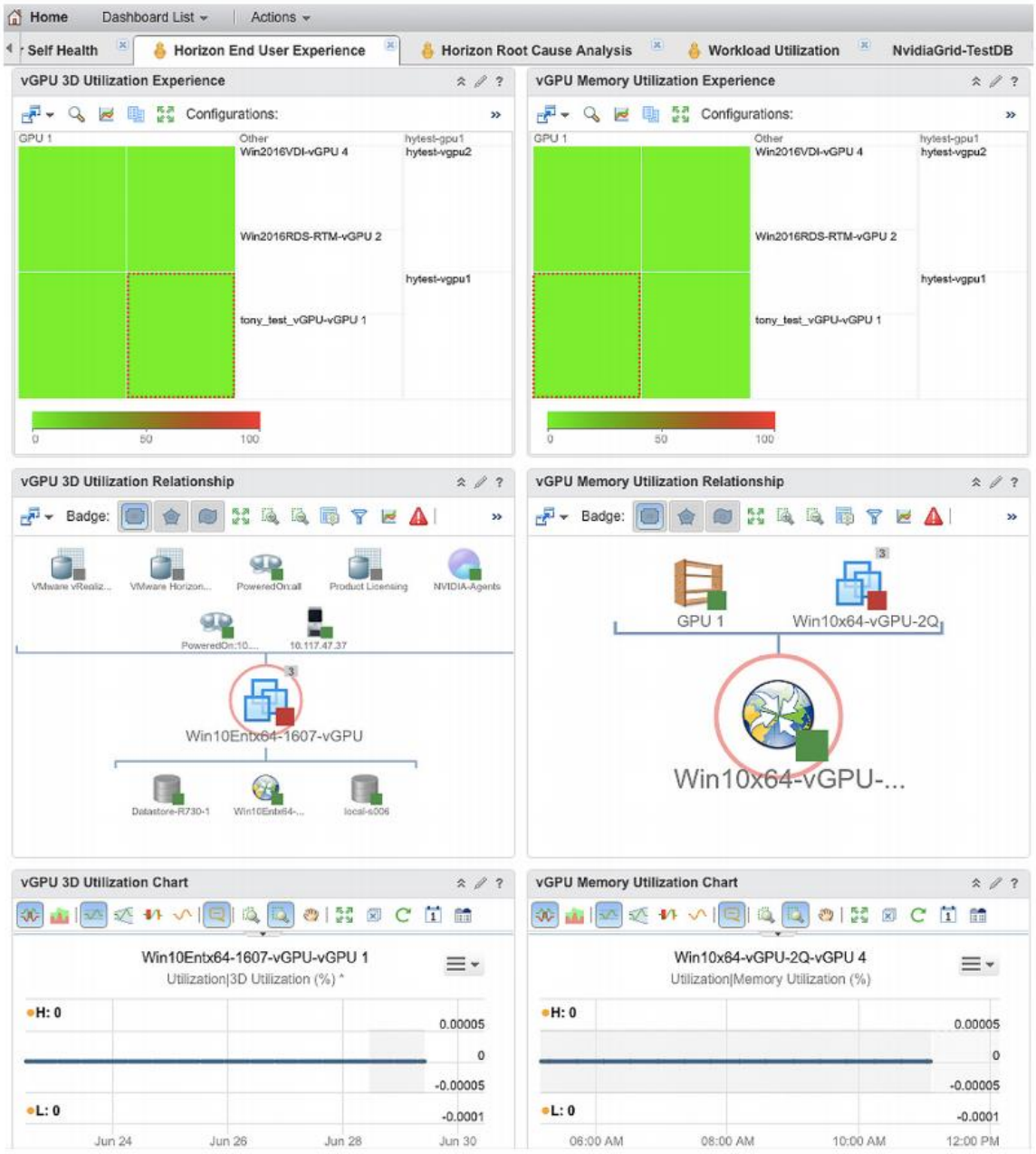
NSMI ships with the graphics drivers for the following operating systems:

- 64-bit Windows Server 2008R2
- 64-bit Windows 7
- Linux

11.3.1 NVIDIA VIRTUAL GPU in the VMware Ecosystem

NVIDIA vGPU and GPU integration with VMware comes in different forms. First, GPU information is reflected in the VMware Horizon admin-focused dashboards in V4H (vRealize Operations for Horizon). Second, NVIDIA is building GPU-focused dashboards that will address both system admin as well as help desk use cases.

Migration with VMware suspense-and-resume technology is also supported on NVIDIA GPU-enabled virtual environments. The screen capture below outlines NVIDIA GPU insights captured within the vROps for Horizon:

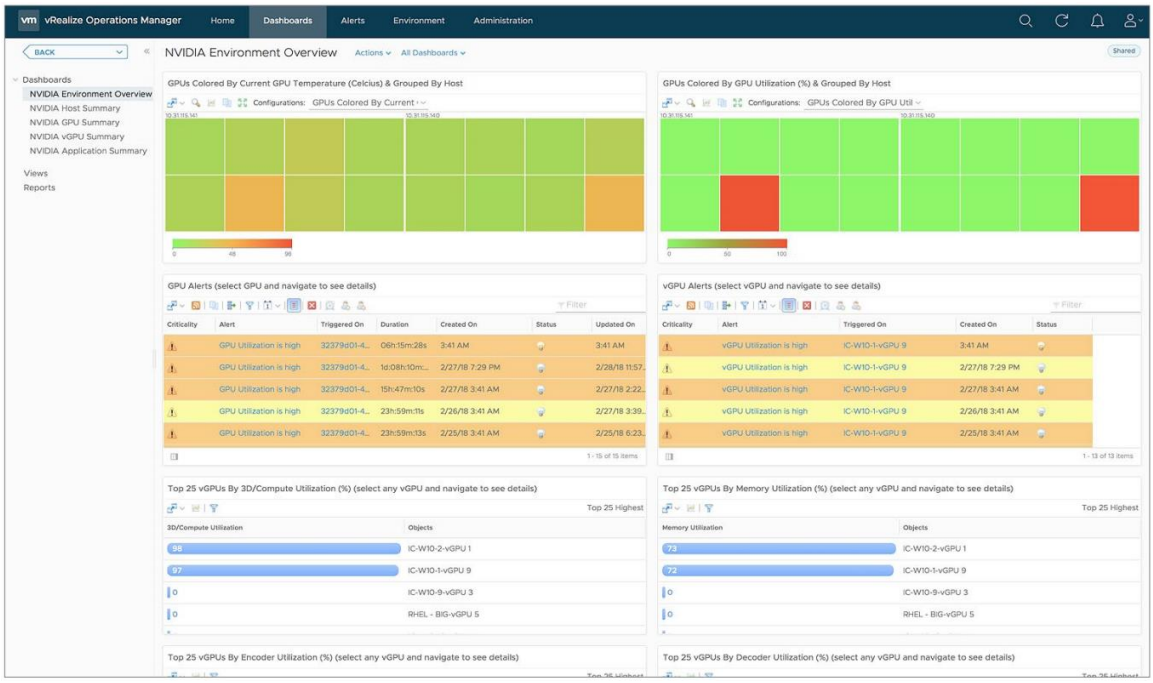


Heat Map, Relationship and Topology view, and User-Desktop Centric View

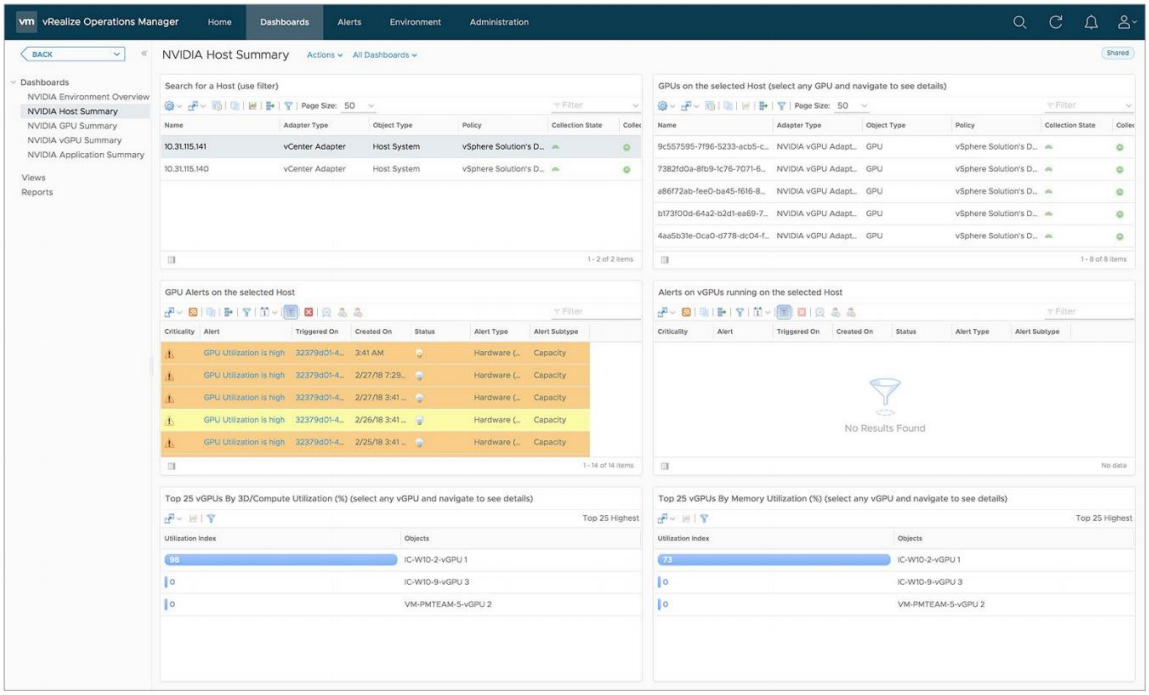
VMware Horizon customers who do not wish to use vRealize Operations for Horizon can leverage the NVIDIA virtual GPU Management SDK free of charge.

The NVIDIA vGPU management SDK provides dashboards with end-to-end GPU insights in vROps for the most comprehensive view of the virtualized infrastructure, supporting architects, admin, and help desk use cases:

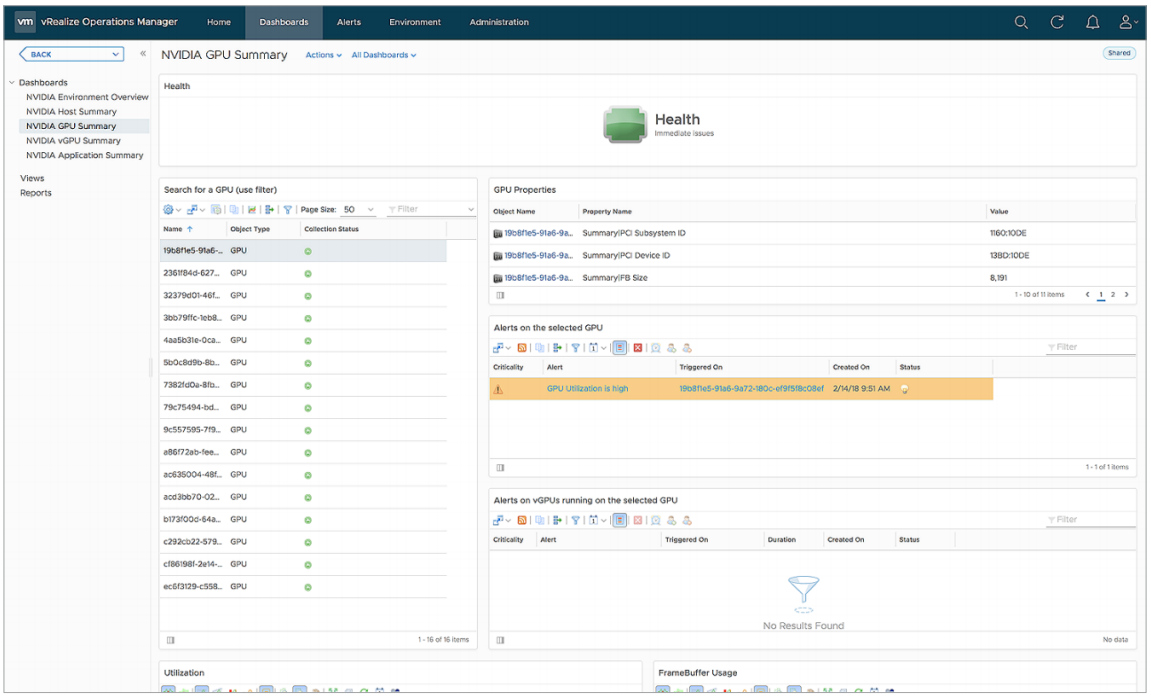
- ENVIRONMENT VIEW for complete visibility of a cluster, with real-time GPU, memory, temperature, alerts, and top 25 GPUs and VMs for a cluster
- HOST VIEW for individual host drill down, with real-time view of a host including GPU, Encode, Decode, memory, temperature, alerts, and top 25 GPUs and VMs on that host
- GPU VIEW for individual GPU drilldown, with real-time GPU, Encode, Decode, FB utilization, alerts, and top 10 apps per VM
- vGPU VIEW gives insights on each GPU, with real-time GPU, Encode, Decode, FB utilization, alerts, and top 10 apps per VM
- APPLICATION VIEW for the most granular visibility into GPU, with real-time GPU, Encode, Decode, FB utilization per app



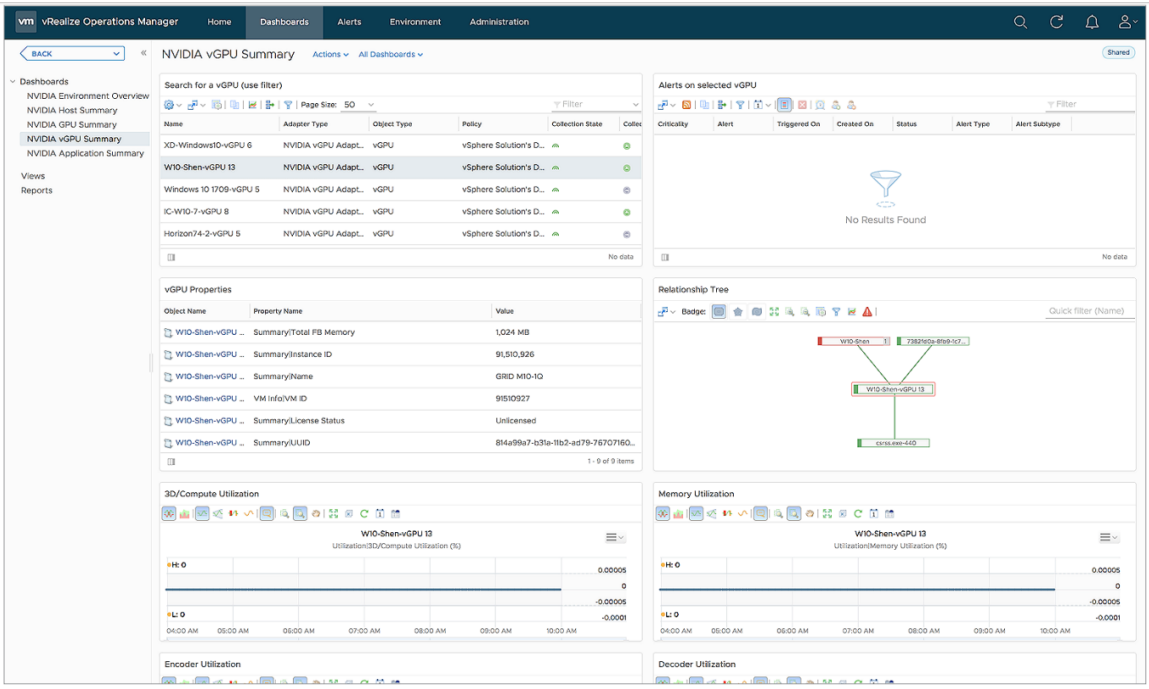
Environment View



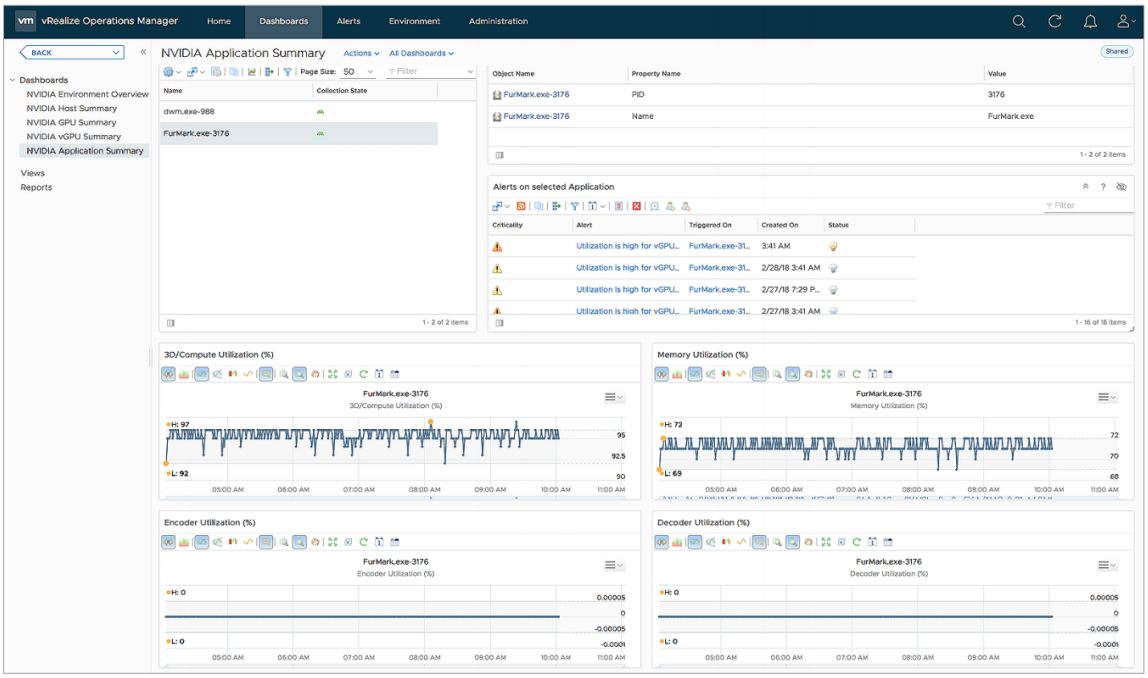
Host View



GPU View



vGPU View



Application View

Feature	NVIDIA Virtual GPU Management Pack for vROps	vROps for Horizon with NVIDIA Virtual GPU
HOST MONITORING		
3D	X	
Encode / Decode	X	
Memory Usage & Utilization	X	
GUEST MONITORING		
3D	X	X
Encode / Decode	X	X
Memory Usage & Utilization	X	X (only utilization)
APPLICATION MONITORING		
3D	X	X
Encode / Decode	X	X

Memory Usage & Utilization	X	X

11.4 LINKS TO APPS AND TOOLS

This section contains sample links to some applications and tools that may assist in your deployment and knowledge.

11.4.1 Browsers, Applications, & Tools

- Internet Explorer

  https://www.microsoft.com/en-us/download/internet-explorer.aspx
- Google Chrome

  https://www.google.com/chrome/browser/
- Mozilla Firefox

  https://www.mozilla.org/firefox
- NVIDIA: www.nvidia.com
- NVIDIA Management and Monitoring

  https://www.nvidia.com/en-us/design-visualization/solutions/it-management/
- NVIDIA Virtual GPU Software Management SDK

  https://developer.nvidia.com/nvidia-grid-software-management-sdk
- Autodesk Maya

  https://www.autodesk.com/products/maya/overview
- Autodesk 3DSMax

  https://www.autodesk.com/products/3ds-max/overview
- Autodesk Revit

  https://www.autodesk.com/products/revit/overview
- Autodesk AutoCAD

  https://www.autodesk.com/products/autocad/overview
- Dassault Systèmes SOLIDWORKS

  https://www.solidworks.com/
- Dassault Systèmes CATIA

https://www.3ds.com/products-services/catia/

- PTC Creo

  https://www.ptc.com/en/products/cad/creo

- Siemens NX

  https://www.plm.automation.siemens.com/global/en/products/nx/

- Adobe Creative Cloud

  https://www.adobe.com/creativecloud.html

Chapter 12. TROUBLESHOOTING

This section includes links and examples of where to explore and post in order to find solutions or assistance.

Before troubleshooting or filing a bug report, review the release notes for information about known issues with the current release, and potential workarounds.

12.1 FORUMS

NVIDIA forums are a very inclusive source of solutions to many problems that may be faced when deploying a virtualized environment. Please search on the NVIDIA forums located here first:

https://gridforums.nvidia.com/

You may also wish to look through the NVIDIA Enterprise Services Knowledgebase to further find support articles and links here:

https://nvidia-esp.custhelp.com/app/answers/list/autologout/1

Keep in mind that not all issues within your deployment may be answered in the NVIDIA vGPU forums. You may also have to reference forums from the hardware supplier, the hypervisor and application themselves. Some examples of key forums to look through are here:

VMware Forums: https://communities.vmware.com/welcome

Citrix Forums: https://discussions.citrix.com/

HPE ProLiant Server Forums: https://community.hpe.com/t5/ProLiant/ct-p/proliant

Dell Server Forums: https://www.dell.com/community/Servers/ct-p/ESServers

Lenovo Server Forums: https://forums.lenovo.com/t5/Datacenter-Systems/ct-p/sv_eg

Autodesk Knowledge Network: https://knowledge.autodesk.com/

Adobe Forums: https://forums.adobe.com/welcome

Dassault Systèmes User Groups: https://www.3ds.com/support/users-communities/

12.2 FILING A BUG REPORT

When filing a bug or requesting support assistance, it is critical to include information about the environment, so that the technical staff that can help you resolve the issue. NVIDIA includes the nvidia-bug-report.sh script within the. vib installation package to collect and package this critical information. The script collects the following information:

- VMware version
- X.Org log and configuration
- PCI information
- CPU information
- GPU information
- **esxcfg** information for PLX devices
- **esxcfg** information for GPU devices
- VIB information
- NVRM messages from vmkernel.log
- System **dmesg** output
- Which virtual machines have vGPU or vSGA configured
- NSMI output

When running this script:

- You may specify the output location for the bug report using either the -o or –output switch followed by the output file name. If you do not specify an output directory, the script will write the bug report to the current directory.
- If you do not specify a file name, the script will use the default name nvidia-bug-report.log.gz.

- If the selected directory already contains a bug report file, then the script will change the name of that existing report file to nvidia-bug-report.log.old.gz before generating a new nvidia-bug-report.log.gz file.

To collect a bug report, issue the command:

```
$ nvidia-bug-report.sh
```

The system displays the following message during the collection process:

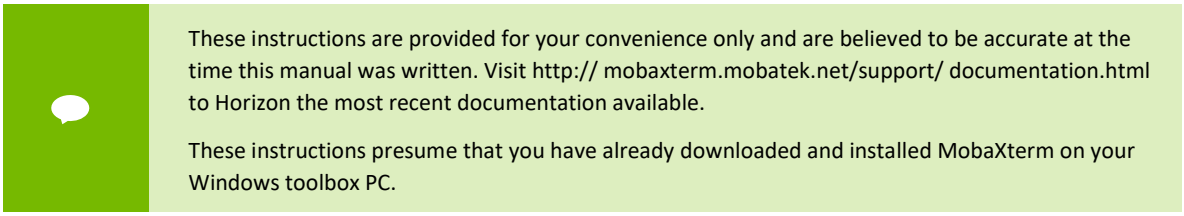
```
nvidia-bug-report.sh will now collect information about your system and create the file 'nvidia-bug-report.log.gz' in the current directory. It may take several seconds to run. In some cases, it may hang trying to capture data generated dynamically by the vSphere kernel and/or the NVIDIA kernel module. While the bug report log file will be incomplete if this happens, it may still contain enough data to diagnose your problem.
```

Be sure to include the **nvidia-bug-report.log.gz log** file when reporting problems to NVIDIA.

Chapter 13. USING MOBAXTERM

This section describes how to use the personal version of MobaXterm to:

- Connect to the host server
- Edit text files using the internal text editor
- Upload a .VIB file to a data store



These instructions are provided for your convenience only and are believed to be accurate at the time this manual was written. Visit http:// mobaxterm.mobatek.net/support/ documentation.html to Horizon the most recent documentation available.

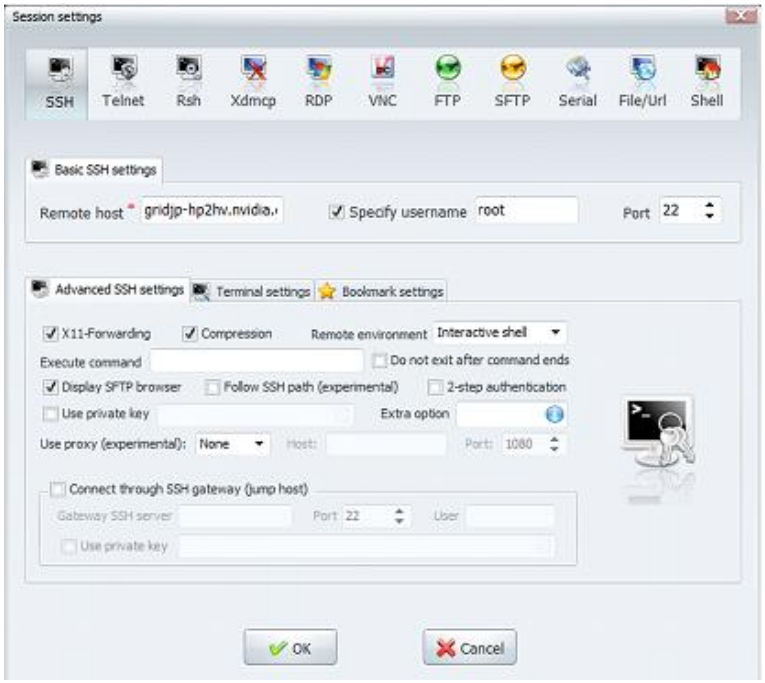
These instructions presume that you have already downloaded and installed MobaXterm on your Windows toolbox PC.

13.1 CONNECTING TO THE HOST SERVER

Use the following procedure to connect to the host server:

1. Launch MobaXterm.

2. Select **Click New Session** in the **Sessions** tab to open the ***Session Settings*** window.



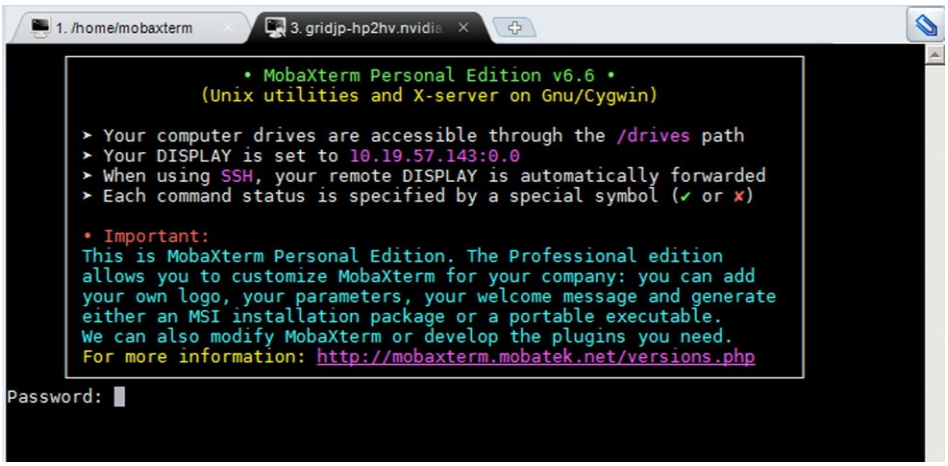
Select the **SSH** icon at the top left of the window.

Enter either the IP address or hostname of the remote host in the **Remote host** field.

Check the **Specify user name** checkbox

Enter root in the field.

Select **OK** to open a new SSH connection to the remote host.
The connection displays in a new MobaXterm tab with a Password prompt.



Enter your root password for the host in the connection tab.
You are now logged into the host server.

13.2 USING THE MOBAXTERM TEXT EDITOR

Once the connection has been established, the left side of the MobaXterm interface displays the file system of the remote host in the left-hand side.

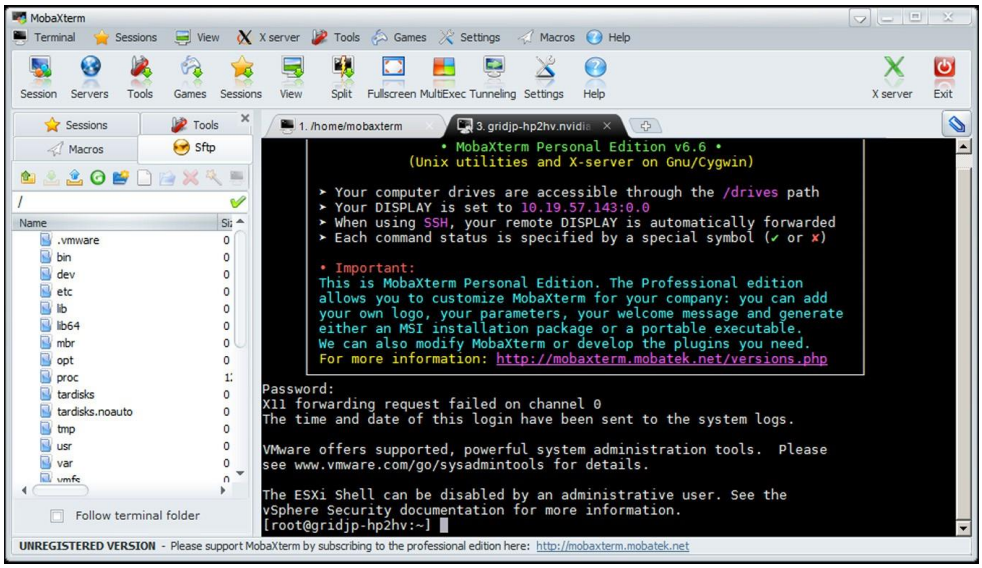


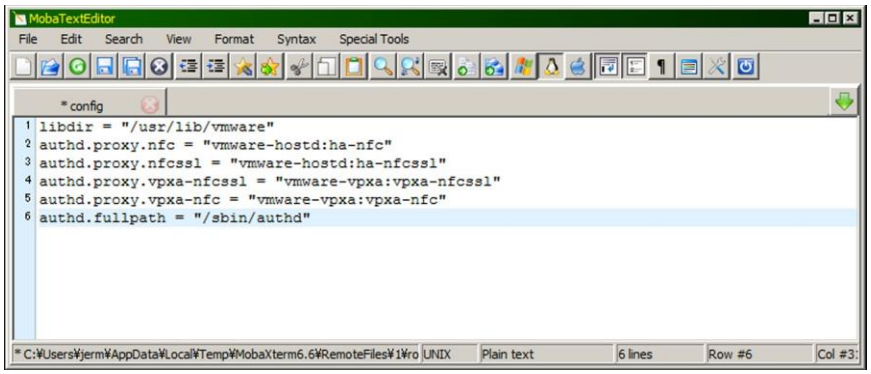
Figure 13-1. Remote Host File System

Use the following procedure to edit a file on the remote host using the built-in editor included with MobaXterm:

1. Either navigate to or enter the path to the file you are going to edit.

The following example will edit the **config** file located in **/etc/VMware** on the remote host.

Right click the **config** file and select **Open with default text editor**.
The ***MobaTextEdtior*** window opens and displays the contents of the file.



Add the line **vGPU.consolidation = true** to the **config** file.

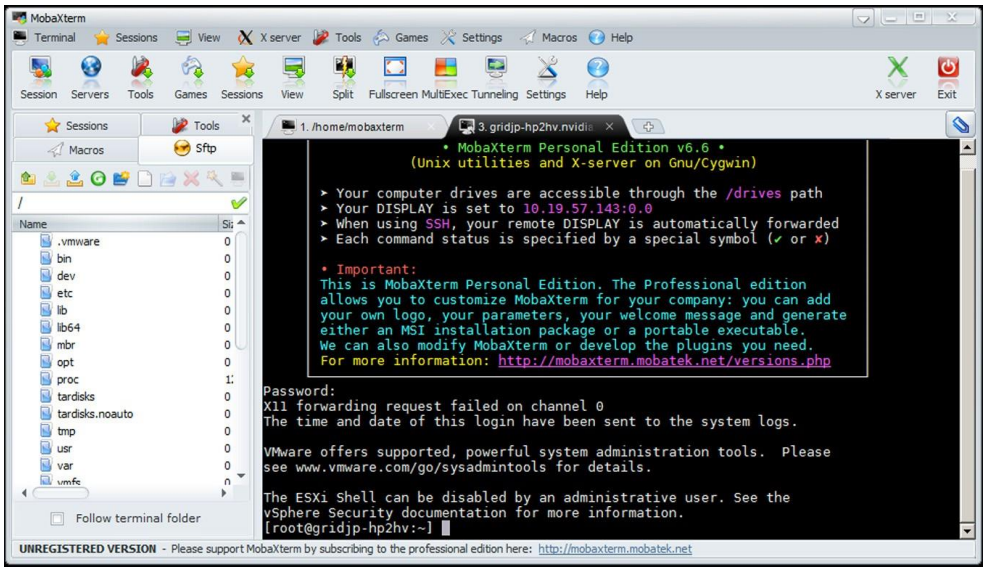
Select **File**→**Save** to save your changes.

13.3 UPLOADING A .VIB FILE

Use the following procedure to upload a .VIB file to the host using MobaXterm:

1. Use MobaXterm to establish an SSH connection to the vSphere host

2. Select the **Sftp** tab to display the host file tree.



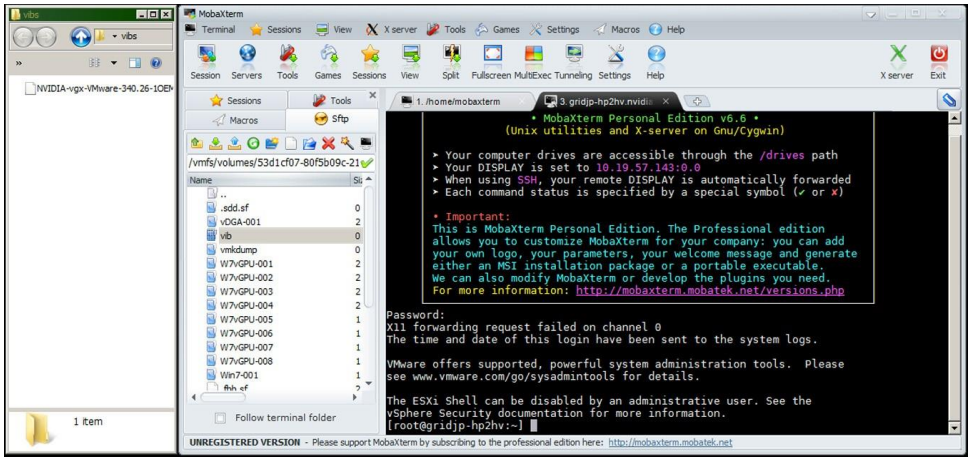
Navigate to the **/vmfs/volumes/** folder.

Select the name of your data store.

The SFTP browser does not automatically handle the link when attempting to use the symbolically-linked data store name and displays a dialog showing the symbolic link target. Click **OK**.

Select the link target from the dialog in the **Sftp** tab to continue navigating to the data store. You are now be able to access all the folders and configuration files on the VM.

Copy the .VIB file to the data store by dragging and dropping it from your local machine into the target location.



This example shows the target folder already created. If needed, you can click the **New Folder** icon in the **Sftp** tab to create a new folder for the .VIB file.

Chapter 14. USING WINSCP

This section describes how to upload a .VIB file using WinSCP. Before doing this:

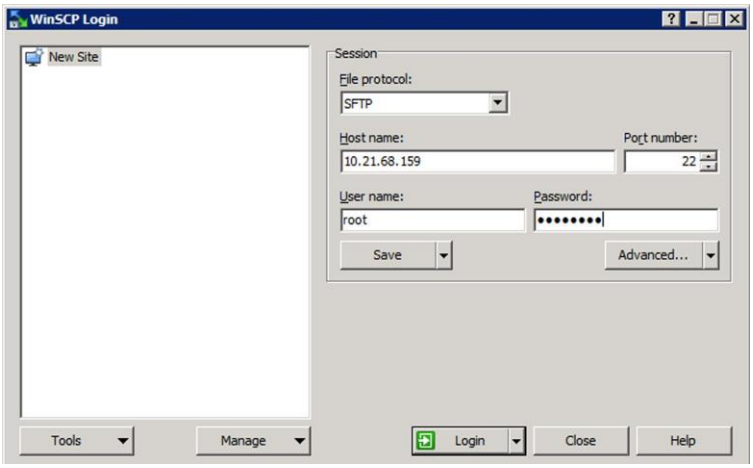
- Ensure that SSH is enabled on vSphere (see Chapter 3, *Installing* on page 25).
- Download and install WinSCP on a Windows PC that has network connectivity to your vSphere host.

Use the following procedure to upload a .VIB file using WinSCP:

1. Start WinSCP to display the Login screen.

Enter the connection details required to connect to the vSphere host.

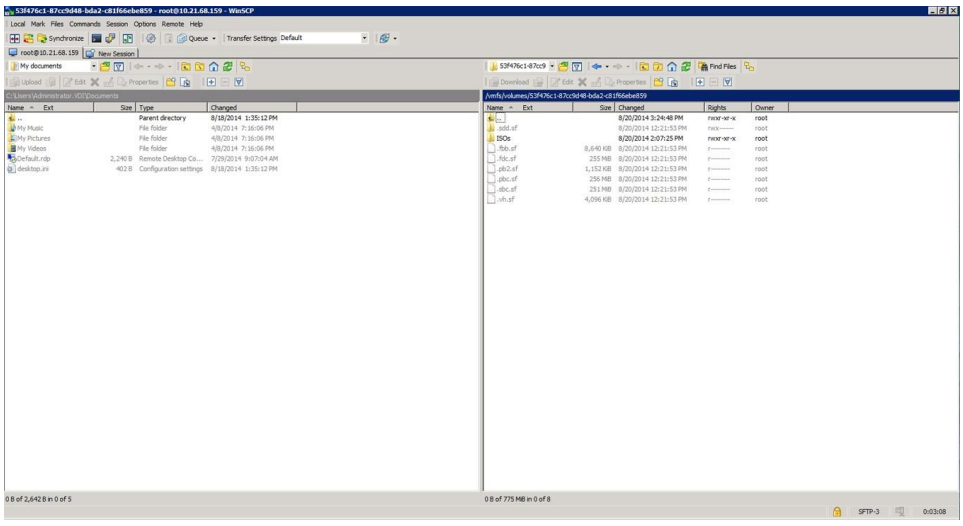
Select **Login** to display the ***WinSCP Login*** window.



If this is your first time connecting to this server, a warning dialog will appear asking you to confirm the connection.

Navigate to the local folder containing the .VIB file that you want to upload to the vSphere host once the connection is established. Use the left pane of the WinSCP interface.

Navigate to a data store you want to upload the .VIB file to using the right pane of the WinSCP interface.



Right-click the .VIB file and select **Upload** to upload the file to the host.

Notice

The information provided in this specification is believed to be accurate and reliable as of the date provided. However, NVIDIA Corporation (“NVIDIA”) does not give any representations or warranties, expressed or implied, as to the accuracy or completeness of such information. NVIDIA shall have no liability for the consequences or use of such information or for any infringement of patents or other rights of third parties that may result from its use. This publication supersedes and replaces all other specifications for the product that may have been previously supplied.

NVIDIA reserves the right to make corrections, modifications, enhancements, improvements, and other changes to this specification, at any time and/or to discontinue any product or service without notice. Customer should obtain the latest relevant specification before placing orders and should verify that such information is current and complete.

NVIDIA products are sold subject to the NVIDIA standard terms and conditions of sale supplied at the time of order acknowledgement, unless otherwise agreed in an individual sales agreement signed by authorized representatives of NVIDIA and customer. NVIDIA hereby expressly objects to applying any customer general terms and conditions with regard to the purchase of the NVIDIA product referenced in this specification.

NVIDIA products are not designed, authorized or warranted to be suitable for use in medical, military, aircraft, space or life support equipment, nor in applications where failure or malfunction of the NVIDIA product can reasonably be expected to result in personal injury, death or property or environmental damage. NVIDIA accepts no liability for inclusion and/or use of NVIDIA products in such equipment or applications and therefore such inclusion and/or use is at customer’s own risk.

NVIDIA makes no representation or warranty that products based on these specifications will be suitable for any specified use without further testing or modification. Testing of all parameters of each product is not necessarily performed by NVIDIA. It is customer’s sole responsibility to ensure the product is suitable and fit for the application planned by customer and to do the necessary testing for the application in order to avoid a default of the application or the product. Weaknesses in customer’s product designs may affect the quality and reliability of the NVIDIA product and may result in additional or different conditions and/or requirements beyond those contained in this specification. NVIDIA does not accept any liability related to any default, damage, costs or problem which may be based on or attributable to: (i) the use of the NVIDIA product in any manner that is contrary to this specification, or (ii) customer product designs.

No license, either expressed or implied, is granted under any NVIDIA patent right, copyright, or other NVIDIA intellectual property right under this specification. Information published by NVIDIA regarding third-party products or services does not constitute a license from NVIDIA to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property rights of the third party, or a license from NVIDIA under the patents or other intellectual property rights of NVIDIA. Reproduction of information in this specification is permissible only if reproduction is approved by NVIDIA in writing, is reproduced without alteration, and is accompanied by all associated conditions, limitations, and notices.

ALL NVIDIA DESIGN SPECIFICATIONS, REFERENCE BOARDS, FILES, DRAWINGS, DIAGNOSTICS, LISTS, AND OTHER DOCUMENTS (TOGETHER AND SEPARATELY, “MATERIALS”) ARE BEING PROVIDED “AS IS.” NVIDIA MAKES NO WARRANTIES, EXPRESSED, IMPLIED, STATUTORY, OR OTHERWISE WITH RESPECT TO THE MATERIALS, AND EXPRESSLY DISCLAIMS ALL IMPLIED WARRANTIES OF NONINFRINGEMENT, MERCHANTABILITY, AND FITNESS FOR A PARTICULAR PURPOSE. Notwithstanding any damages that customer might incur for any reason whatsoever, NVIDIA’s aggregate and cumulative liability towards customer for the products described herein shall be limited in accordance with the NVIDIA terms and conditions of sale for the product.

VESA DisplayPort

DisplayPort and DisplayPort Compliance Logo, DisplayPort Compliance Logo for Dual-mode Sources, and DisplayPort Compliance Logo for Active Cables are trademarks owned by the Video Electronics Standards Association in the United States and other countries.

HDMI

HDMI, the HDMI logo, and High-Definition Multimedia Interface are trademarks or registered trademarks of HDMI Licensing LLC.

ROVI Compliance Statement

NVIDIA Products that support Rovi Corporation’s Revision 7.1.L1 Anti-Copy Process (ACP) encoding technology can only be sold or distributed to buyers with a valid and existing authorization from ROVI to purchase and incorporate the device into buyer’s products.

This device is protected by U.S. patent numbers 6,516,132; 5,583,936; 6,836,549; 7,050,698; and 7,492,896 and other intellectual property rights. The use of ROVI Corporation's copy protection technology in the device must be authorized by ROVI Corporation and is intended for home and other limited pay-per-view uses only, unless otherwise authorized in writing by ROVI Corporation. Reverse engineering or disassembly is prohibited.

OpenCL

OpenCL is a trademark of Apple Inc. used under license to the Khronos Group Inc.

Trademarks

NVIDIA, the NVIDIA logo, NVIDIA GRID, and NVIDIA GRID vGPU™ are trademarks and/or registered trademarks of NVIDIA Corporation in the U.S. and other countries. Other company and product names may be trademarks of the respective companies with which they are associated.

Copyright

© 2018 NVIDIA Corporation. All rights reserved.