NVIDIA vGPU Software for VMware vSphere Hypervisor

Deployment Guide

Document History

DU-09116-001_v002

Version	Date	Authors	Description of Change
01	2018	AJ	Initial Release
02	September 2020	AFS	Reflect changes to the products featured in this guide

Table of Contents

Chapter 1. Getting Started 6
1.1 Why NVIDIA vGPU? 6
1.2 NVIDIA vGPU Architecture 7
1.3 Supported NVIDIA GPUs 8
1.4 NVIDIA vGPU Software Licensed Products 9
1.5 Supported Graphics APIs 12
1.6 Before You Begin 13
1.6.1 Building an Evaluation Plan 13
1.6.2 Sizing Your Environment 13
1.6.3 Choosing Your Hardware 14
1.6.4 General Prerequisites 14
1.6.5 Preparation for Pre-Installation 16
1.6.6 Server BIOS Settings 16
Chapter 2. Installing VMware ESXi 17
2.1 Choosing an Install Method 17
2.2 Preparing USB Boot Media 17
2.3 Installing ESXi 19
2.4 Initial Host Configuration 23
Chapter 3. Installing VMware vCenter Server 27
3.1 Installing vCenter Server Appliance 27
3.1.1 About VCSA 27
3.1.2 vCenter Server Appliance (VCSA) Installation 28
3.2 Post Installation 39
3.2.1 Adding Licenses to Your vCenter Server 39
3.2.2 Adding a Host 42
3.2.3 Setting the NTP Service on All Hosts 45
3.2.4 Setting a vCenter Appliance to Auto-Start 46
3.2.5 Mounting an NFS ISO Datastore 48
Chapter 4. Building Horizon Composer Server 51
4.1 Installing Horizon 51
4.1.1 Registering the Horizon License 58
4.2 Registering vCenter Server 59
4.3 Installing Horizon Composer Server 60
4.4 Registering Horizon Composer Server 73
4.4.1 “Error while checking the administrator” message 78
Chapter 5. NVIDIA vGPU Manager VIB Installation 81

5.1 Preparing the .vib File for Install 81
5.1.1 Uploading the .vib File in vSphere Web Client 81
5.2 Installing vGPU Manager with the .vib File 83
5.3 Updating vGPU Manager with the .vib File 85
5.4 Verifying the Installation of vGPU Manager 85
5.5 Uninstalling vGPU Manager 86
5.6 Changing the Default Graphics Type in VMware vSphere 6.5 and Later 87
5.7 Changing the vGPU Scheduling Policy 89
5.7.1 vGPU Scheduling Policies 89
5.7.2 RmPVMRL Registry Key 90
5.7.3 Changing the vGPU Scheduling Policy for All GPUs 91
5.7.4 Changing the vGPU Scheduling Policy for Specific GPUs 91
5.7.5 Restoring Default vGPU Scheduling Policies 92
5.8 Disabling and Enabling ECC Memory 93
5.8.1 Disabling ECC Memory 94
5.8.2 Enabling ECC Memory 95

Chapter 6. Deploying the NVIDIA vGPU Software License Server 98

6.1 Platform Requirements 98
6.1.1 Hardware and Software Requirements 98
6.1.2 Platform Configuration Requirements 98
6.1.3 Network Ports and Management Interface 99
6.2 Installing the NVIDIA vGPU Software License Server on Windows 99
6.2.1 Installing the Java Runtime Environment on Windows 99
6.2.2 Installing the License Server Software on Windows 101
6.2.3 Obtaining the License Server's MAC Address 104
6.2.4 Managing your License Server and Getting your License Files 104
6.2.4.1 Creating a Licenser Server on the NVIDIA Licensing Portal 104
6.2.4.2 Downloading a License File 106
6.2.5 Installing a License 107

Chapter 7. Selecting the Correct vGPU Profiles 110

7.1 The Role of the vGPU Manager 110
7.2 Bringing Quadro to Virtualization 111
7.3 Matching Profiles to User Needs 111
7.4 Creative and Technical Professionals 111
7.5 Knowledge Worker Profiles 112

Chapter 8. Application Based VDI Hosting Sizing 113

8.1 How Many Users Can I Support on a Host? 113
8.2 An Example Application: Siemens NX 113
8.3 Application-Based Sizing 114
8.3.1 Recommended System Requirements 114
8.4 Testing Methodology 115

P4 with Quadro vDWS for Light and Medium Users 117
NVIDIA P40 with Quadro vDWS for Heavy Users 118
Sufficient System Memory for each Individual User 119
Flash Based Storage for Best Performance 119
Typical Networking Configuration for Quadro vDWS 119

Chapter 9. Creating Your First vGPU Virtual Desktop 120

9.1 Creating a Virtual Machine 120
9.2 Installing Microsoft Windows 125
9.3 Installing VMware Tools 129
9.4 Adding the Golden Master to the Domain 132
9.5 Installing Horizon Agent 135
9.6 Installing Horizon Direct Connection 138
9.7 Optimizing Windows 141
9.8 Additional Virtual Machine Settings 144
9.9 Enabling the NVIDIA vGPU 144
9.10 Installing the NVIDIA vGPU Driver: Microsoft Windows 147
9.11 Installing the NVIDIA vGPU Driver: Linux 152
9.11.1 Installing on Bare Metal 152
Prerequisites 152
To install the NVIDIA vGPU driver for Linux 152
9.12 Licensing NVIDIA vGPU Software 154
9.12.1 Licensing NVIDIA vGPU Software on Windows 154
9.12.2 Licensing NVIDIA vGPU Software on Linux 156
9.13 Finalizing the Installation 156

Chapter 10. Creating a Horizon vGPU Pool 160

10.1 Creating a Template 160
10.2 Creating a Customization Specification 161
10.3 Provisioning a Single Virtual Machine 166
10.4 Creating Full Clone Desktop Pools 169
10.5 Creating Linked Clone Desktop Pools 179
10.6 Enabling User Access to Desktop Pools 190

Chapter 11. VMware Horizon Client 192

11.1 Installing VMware Horizon Client 192
11.2 Configuring VMware Horizon Client Connection 195

Chapter 12. Troubleshooting 197

12.1 Forums 197
12.2 Filing a Bug Report 197

Appendix A. About This Document 199

A.1. Related Documentation 199
A.2. Support Contact Information 199

Chapter 1. Getting Started

NVIDIA vGPU software 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 alter the landscape of desktop virtualization and enable users to use these solutions with applications of all levels of complexity and graphics requirements. This chapter also describes the NVIDIA vGPU architecture, the NVIDIA GPUs recommended for virtualization, the three NVIDIA vGPU software licensed products for desktop virtualization, and key standards supported by NVIDIA virtual GPU technology.

1.1 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 provide the rich application support needed to be successful and ensure end user adoption. Dedicated GPU pass-through with vDGA provides 100% application compatibility, but is cost-effective only for the highest end use cases due to high cost and 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 was 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 that 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. This provides the 100% application compatibility of vDGA pass-through graphics, but with the

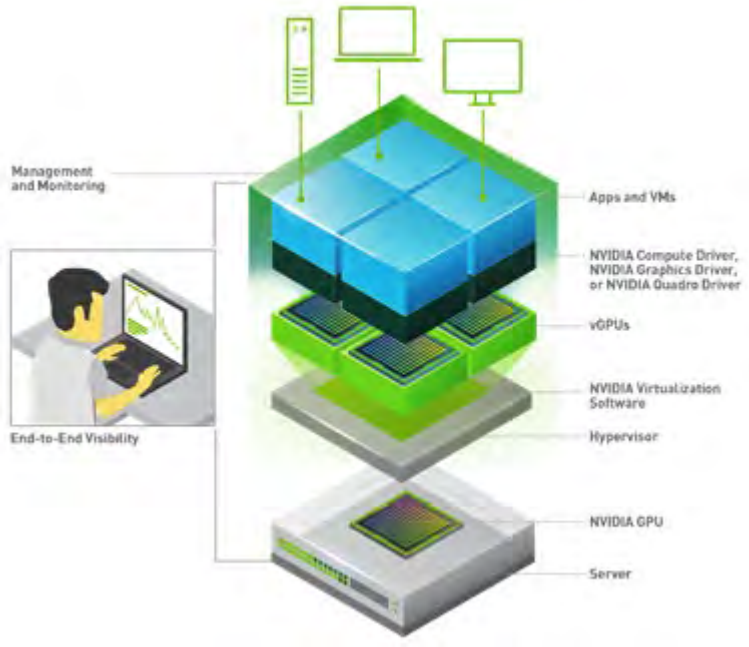
lower cost of multiple desktops sharing a single graphics card to provide a rich, yet more cost-effective user experience. With VMware Horizon, you can centralize, pool, and more easily manage traditionally complex and expensive distributed workstations and desktops. Now all your user groups can take advantage of the promise of virtualization.

1.2 NVIDIA vGPU Architecture

The high-level architecture of an NVIDIA virtual GPU enabled VDI environment is illustrated in Figure 1-1. Here, we have GPUs in the server, and the NVIDIA vGPU manager software (VIB) is installed on the host server. This software enables multiple VMs to share a single GPU, or if there are multiple GPUs in the server, they can be aggregated so that a single VM can access multiple GPUs. This GPU enabled environment provides not only unprecedented performance; it also enables support for more users on a server because work that was typically done by the CPU can be offloaded to the GPU. Physical NVIDIA GPUs can support multiple virtual GPUs (vGPUs) and 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. In the VM itself, vGPU drivers are installed which support the different license levels that are available.

Figure 1-1 NVIDIA vGPU Platform Solution Architecture

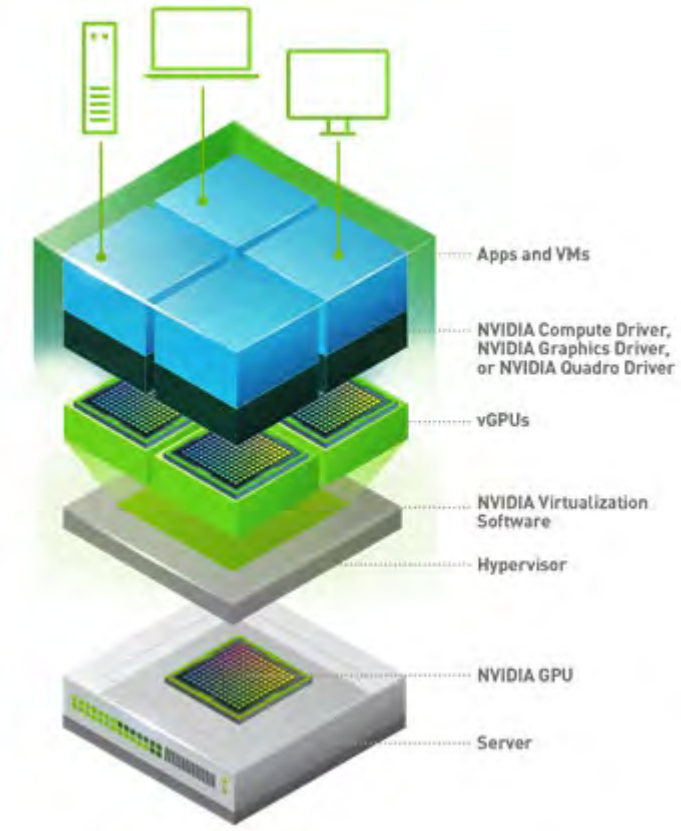


NVIDIA vGPUs are comparable to conventional GPUs in that they have a fixed amount of GPU memory and one or more virtual display outputs or *head.* Managed by the NVIDIA vGPU Manager installed in the hypervisor, the vGPU memory is allocated out of the physical GPU frame buffer at the time the vGPU is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

Note: These are virtual heads, meaning on GPUs there is no physical connection point for external physical displays.

All vGPUs that reside on a physical GPU share access to the GPU's engines, including the graphics (3D) and video decode and encode engines. Figure 1-1 shows the vGPU internal architecture. The VM's guest OS leverages direct access to the GPU for performance and critical fast paths. Noncritical performance management operations use a paravirtualized interface to the NVIDIA Virtual GPU Manager.

Figure 1-2 NVIDIA vGPU Internal Architecture



1.3 Supported NVIDIA GPUs

Table 1-1 lists the hardware specifications for NVIDIA GPUs that support NVIDIA vGPU software.

Table 1-1 NVIDIA GPU Hardware Specifications

	V100S	RTX 8000 Passive	RX 6000 Passive	M10	T4	P6

GPUs / Board (Architecture)	1 (Volta)	1 (Turing)	1 (Turing)	4 (Maxwell)	1 (Turing)	1 (Pascal)
CUDA Cores	5,120	4,608	4,608	2,562 (640 per GPU)	2,560	2,048
Memory Size	32 GB HBM2	48 GB GDDR6	24 GB GDDR5	32 GB GDDR5 (8GB per GPU)	16 GB GDDR6	16 GB GDDR5
vGPU Profiles	1 GB, 2 GB, 4 GB, 8 GB, 16 GB, 32 GB	1 GB, 2 GB, 3 GB, 4 GB, 6 GB, 8 GB, 12 GB, 24 GB, 48 GB	1 GB, 2 GB, 3 GB, 4 GB, 6 GB, 8 GB, 12 GB, 24 GB	0.5 GB, 1 GB, 2 GB, 4GB, 8 GB	0.5 GB, 1 GB, 2 GB, 4GB, 8 GB, 16 GB	0.5 GB, 1 GB, 2 GB, 4GB, 8 GB, 16 GB
Form Factor	PCIe 3.0 Dual Slot (Rack servers)	PCIe 3.0 Dual Slot (Rack servers)	PCIe 3.0 Dual Slot (Rack servers)	PCIe 3.0 Dual Slot (Rack servers)	PCIe 3.0 Single Slot (Rack servers)	MXM (Blade servers)
Power	250W	295W	295W	225W	75W	90W
Thermal	Passive	Active	Active	Passive	Passive	Bare Board

NVIDIA virtual GPU software is supported with NVIDIA GRID® GPUs. Determine the GRID GPU best suited for your environment, depending on whether you are optimizing for performance or density, and whether the GPUs will be installed in rack servers or blade servers. You can download datasheets from the links below.

- Highest performance – The following NVIDIA GPUs are targeted at users with demanding workflows, compressed design cycles, and complex simulations:
  - NVIDIA V100S
  - NVIDIA RTX™ 8000
  - NVIDIA RTX 6000
- Highest density:
  - NVIDIA T4
  - NVIDIA M10
- Blade server optimized:
  - NVIDIA P6

1.4 NVIDIA vGPU Software Licensed Products

NVIDIA virtual GPU software divides GRID GPU resources so the GPU can be shared across multiple virtual machines running any application.

The portfolio of NVIDIA virtual GPU software products for desktop virtualization is as follows:

- 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 need an NVIDIA GPU and software license that addresses your specific use case.

Table 1-2 lists the virtual GPU software products and recommended NVIDIA GPUs for each product.

Table 1-2 Virtual GPU Software Products

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 Support6	Y		Y
	Bare Metal Support7	Y		Y
	NVIDIA Graphics Driver	Y[4]	Y	Y
	NVIDIA Quadro Driver	Y		
	Guaranteed Quality of Service Scheduling8	Y	Y	Y
Display	Maximum Active Displays	Four 4K	Four QHD / Two 4K[18]	One[9]
	Maximum Display Resolution	4096×2160	4096×2160	1280×1024
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 include uses for double precision compute workloads (i.e. Running 3D models and design workflows, intensive CAE simulations)	V100S[15]		
	Mid to high-end professional graphics users, includes use for single precision compute workloads (i.e. Rendering & creating complex designs)	NVIDIA RTX 8000 Passive NVIDIA RTX 6000 Passive		
	Entry to mid-level professional graphics users, including deep learning inference workloads and Pascal specific GPU architecture features	T4		
	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	P6	P6	P6

[1] Assumes cost of subscription, NVIDIA GRID software, and hardware, with 3-year amortization of hardware of two 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 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 1280×1024 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 2560×1600 displays on 2B profile.
[18] Supports up to four 2560×1600 displays on 1B profile. Support for two 4K displays starts with NVIDIA virtual GPU software release 6.0, and support for four 2560×1600 displays on 2GB profile starts with NVIDIA virtual GPU software release 6.2
[19] 32Q profile available with 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, an NVIDIA 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 V100S, RTX 8000, RTX 6000 (passive and active cooling), and M10 are full height, full length, double width PCI-e cards, and are passively cooled (no onboard fan), requiring certified servers designed to power and cool them properly. NVIDIA T4 is a single slot 75W GPU, meaning you can accommodate two NVIDIA T4s in the same physical space you could fit a single double width card. NVIDIA P6 and T4 are designed to fit into blade server form factors. Refer to the web page How to Buy NVIDIA Virtual GPU Solutions for a list of certified server platforms for NVIDIA 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 must have auxiliary power cables connected to it (except NVIDIA T4); see the chart below for specific pin counts. Most industry standard servers require an enablement kit for properly mounting NVIDIA GPU cards. Check with your server OEM of choice for more specific requirements.

1.5 Supported Graphics APIs

This version of NVIDIA vGPU software includes support for:

- Full DirectX 12, Direct2D, and DirectX Video Acceleration (DXVA)
- OpenGL 4.6
- NVIDIA vGPU SDK (remote graphics acceleration)

- Vulkan 1.1

For information about the virtual GPUs that support OpenCL and NVIDIA® CUDA® applications, see NVIDIA CUDA Toolkit and OpenCL Support on NVIDIA vGPU Software in *NVIDIA Virtual GPU Software User Guide*.

Note: Only OpenCL and CUDA applications *without* Unified Memory are supported on NVIDIA virtual GPUs. Unified Memory and CUDA tools are *not* supported on NVIDIA virtual GPUs.

1.6 Before You Begin

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

1.6.1 Building an Evaluation Plan

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

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

1.6.2 Sizing 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 2560×1600 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 2560×1600 displays and/or One 3840×2160
- 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 2560×1600 displays and/or two 3840×2160

Be sure to include enough RAM, CPUs, and GPUs to meet your sizing requirements.

1.6.3 Choosing Your Hardware

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

- NVIDIA certified servers with NVIDIA cards (see the web page NVIDIA vGPU Partners for a list of certified NVIDIA servers. It is helpful to cross check this list with the VMware HCL to ensure compatibility for your deployment. The following specifications are recommended:
  - 2.4 GHz CPU or faster (Intel Xeon Gold 6200 Family Product)
  - High-speed RAM
  - Fast networking

  If you are using local storage, input/output operations per second (IOPS) plays a major role in performance. If you are using VMware for a Virtual SAN, see the VMware Virtual SAN requirements website for more details.
- High performance storage with Virtual SAN or high IOPS storage system
- High performance end points for testing access

1.6.4 General Prerequisites

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

Review the Getting Started page of the VMware Horizon documentation. It provides a roadmap for implementing Horizon as a server with Citrix clients and has a link to a lists of VMware education courses and other resources.

- One or more NVIDIA certified servers with NVIDIA cards. For a list of certified servers, see the NVIDIA vGPU Partners page Find a Virtualization Partner to Get Started. Cross-check this list against the VMware Compatibility Guide to ensure compatibility with your environment.

  Choose servers that meet the following requirements:

  - 2.4 GHz CPU or faster (Intel Xeon Gold 6200 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.
- Free 60-day evaluation for VMware vSphere and vSphere with Operations Management can be downloaded from Product Evaluation Center for VMware vSphere 7.0.

  The appropriate NVIDIA GPU for your use case. See Table 1-1 for a better understanding of which GPU suits your deployment requirements. For additional guidance, contact your NVIDIA and VMware reps.

- VMware vSphere 6.7 build
  - vSphere and vCenter Server are available only from the VMware website. See the link above.
  - You may deploy vCenter Server 6 on a Microsoft 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 the web page Welcome to My VMware.
- NVIDIA vGPU software:
  - NVIDIA vGPU manager vSphere Installation Bundle (VIB)
  - NVIDIA WDDM guest driver (64-bit versions)

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

As of this document's publication date (see Document History), the current NVIDIA vGPU software builds are as follows:

- `NVIDIA-VMware-450.55-1OEM.670.0.0.8169922.x86_64.vib`
- `NVD.NVIDIA_bootbank_NVIDIA-VMware_450.55-1OEM.670.0.0.8169922-offline_bundle.zip`
- `451.48_grid_server2012R2_64bit_international.exe`
- `451.48_grid_win10_server2016_server2019_64bit_international.exe`
- `NVIDIA-Linux-x86_64-450.51.05-grid.run`

- Microsoft software:
  - Windows Server 2016 (recommended) or Server 2012R2 (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.
  - Putty (SSH) and WinSPC (SPC), available from www.putty.org and www.winscp.net, respectively.
- Licenses:
  - From the VMware Horizon website:
    - VMware vSphere 6.7
    - VMware Horizon 7
  - Microsoft (volume licenses recommended)
- Testing and Benchmarking tools:
  - NVIDIA System Management interface (NSMI)
  - OPTIONAL: VMware vRealize Operations for Horizon (V4H)
  - OPTIONAL: Lakeside Systrack 7 with GPU monitoring
  - User satisfaction survey
- A server system on which to install View Connection Server 6:
  - Runs Microsoft Windows Server 2016

- Must be joined to a domain
- Must be assigned a static IP address

1.6.5 Preparation for Pre-Installation

Before you install NVIDIA vGPU software:

1. Determine how vSphere will run on the physical hosts. Consider booting from a thumb drive, as this is an Early Access build.
2. Download and install any of the following for SSH and SCP:
   - An SSH tool (such as PuTTY) and an SCP tool (such as WinSCP)
   - MobaXterm, which handles both SSH and SCP functions
3. Set the BIOS on the physical hosts as appropriate. See 1.6.6, *Server BIOS* Settings on page 16.
4. Install Virtual Clone Drive or a similar tool to mount and unmount ISO images easily within the virtual machines.

1.6.6 Server BIOS Settings

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

- Hyperthreading – Enabled
- Power Setting or System Profile– High Performance
- CPU Performance (if applicable) – Enterprise or High Throughput
- Memory Mapped I/O above 4-GB - Enabled (if applicable)

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

Chapter 2. Installing VMware ESXi

This chapter covers the following ESXi installation topics:

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

> This deployment guide assumes you are building an environment as a proof of concept, not as a production deployment. Consequently, it recommends settings and other choices that make the process faster and easier. Before you build your production environment, consult the corresponding guides for each technology and make choices appropriate for your needs.

2.1 Choosing an Install Method

With the ability to install from and onto an SD card or a USB flash drive, ESXi offers more flexibility than a local hard drive install. See the vSphere documentation regarding best practices for logs when booting from a USB flash drive or similar device. This lab uses Supermicro's IPMI and virtual media to boot from an ISO file and install on local storage.

2.2 Preparing USB Boot Media

Reference VMware KB: "Installing ESXi on a supported USB flash drive or SD flash card (2004784)"

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

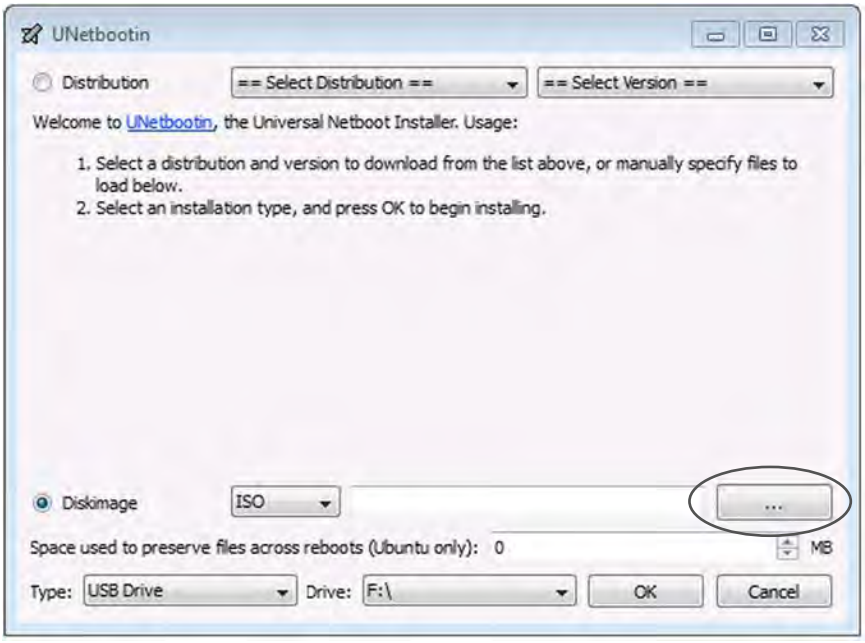
Use the following procedure to prepare a USB flash drive for booting:

5. Download `UNetbootin` from the GitHub UNetbootin page.

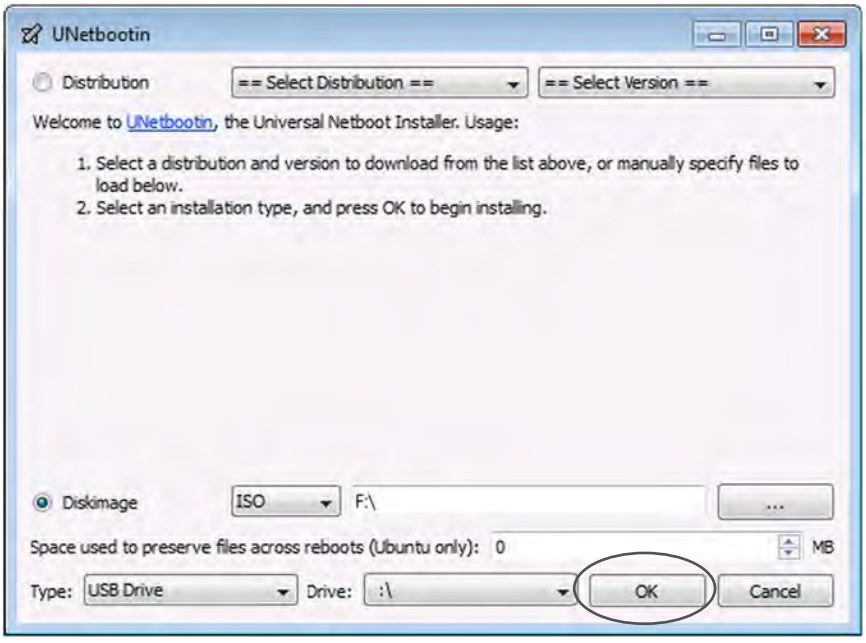
   The Windows version of the application does not include an installer; however, the OSX version is packaged in a `.DMG` file that you must mount. You must also copy the application to the `Applications` folder before launching.

   Alternatively, you can use YUMI, which allows booting from multiple installation images on one USB device, and lets you load the entire installation into RAM. You can download YUMI from Pendrivelinux.com.

6. Start the application, select `Diskimage`, and click the `...` button to browse for the installation `.ISO` file:



7. Browse to the location that contains the installation `.ISO` file, then select Open.
8. Select the mounted USB flash drive on which to perform the installation, then click OK. The copying process begins, and YUMI displays a series of progress bars.

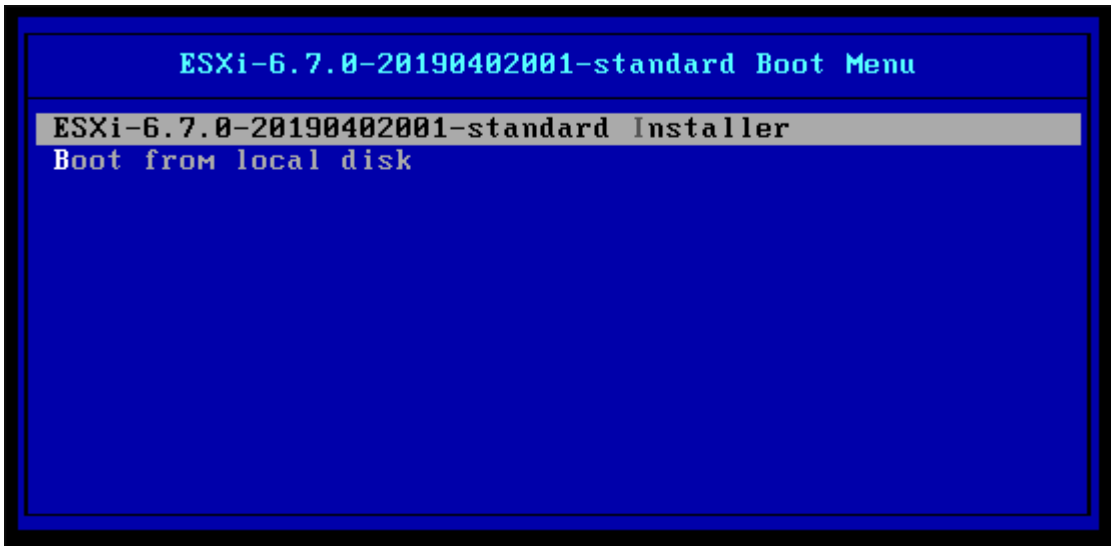


9. When the copying process is complete, click Exit, then remove the USB flash drive.
10. To install from this USB flash drive, insert it into the host using either an internal USB port or on an external USB port, then set that as the primary boot source or select from the boot menu on power up.

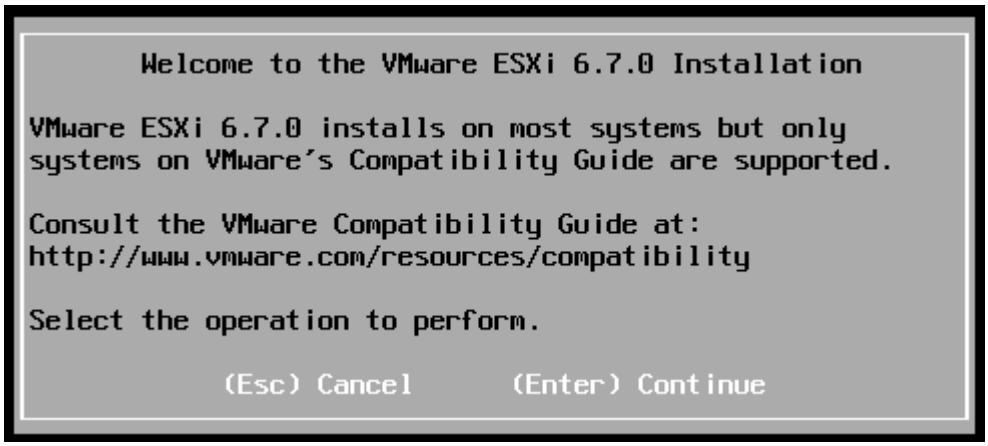
2.3 Installing ESXi

Use the following procedure to install ESXi regardless of boot source. Select the boot medium that holds the vSphere ISO from your host's boot menu.

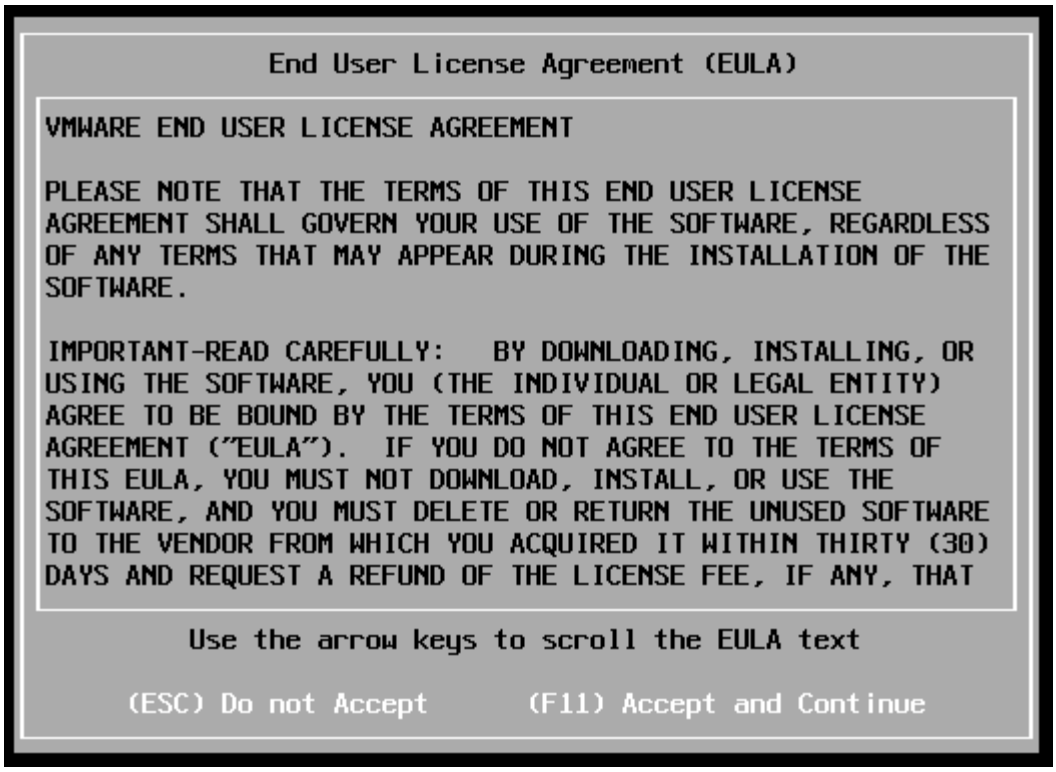
1. Apply power to start the host. It displays the following menu when it starts up:



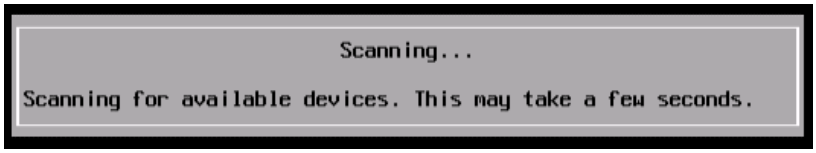
2. Select the installer with the arrow keys, then press Enter to begin booting the ESXi 6.7 installer.
3. The installer displays a compatibility warning:



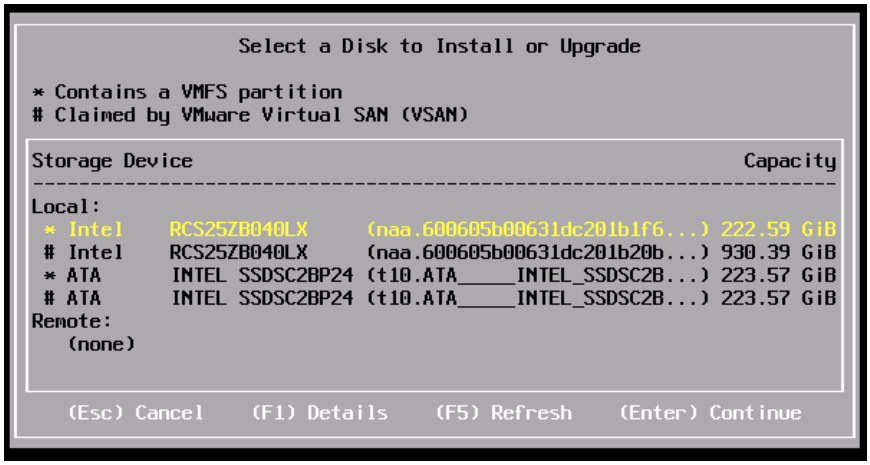
4. Press Enter to continue. The installer displays the End User License Agreement (EULA):



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



When the installer is done scanning the host it displays all drives available for install:

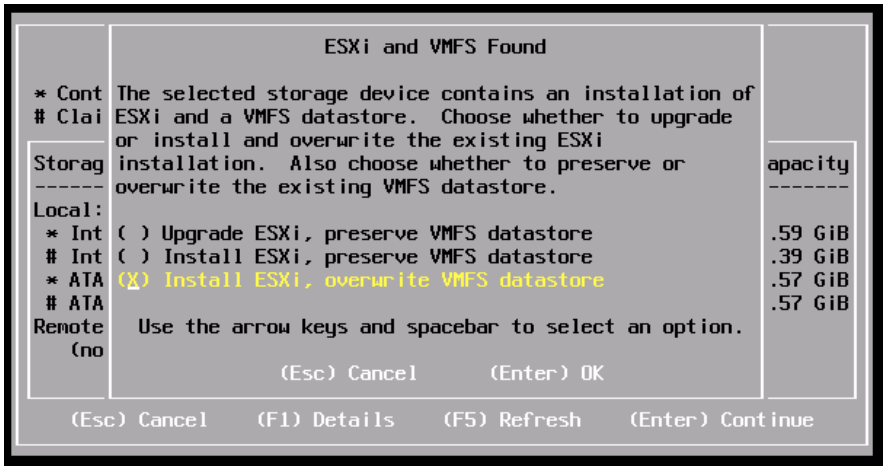


6. Use the arrow keys to select the drive on which you want to install ESXi, then press Enter to continue.

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

7. The installer scans the drive you selected to determine its suitability for install:

8. The installer displays the Confirm Disk Selection window:



9. Press Enter to accept your selection and continue.

For this EA2 release, Upgrade ESXi is not a supported selection.

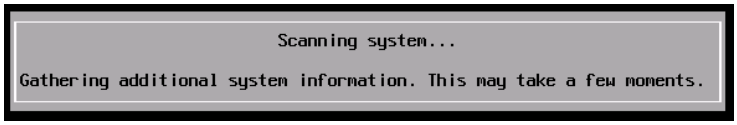
10. The installer displays the Please select a keyboard layout window:

11. Select a keyboard layout with the arrow keys, then press Enter**.** The installer displays the Enter a root password window:

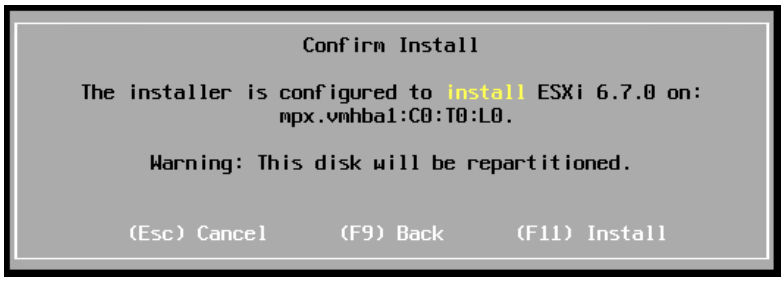
12. Enter a root password in the Root password field.

> CAUTION: To give you host maximum protection from unauthorized access, use a root password that contains at least eight (8) characters, including a mix of lower case and capital letters, digits, and special characters.

13. Confirm the password in the Confirm password field, then press Enter to continue. The installer rescans the system:



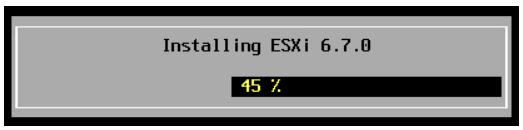
14. When the installer finishes rescanning, it displays the Confirm Install window:



Press F11 to proceed with the installation.

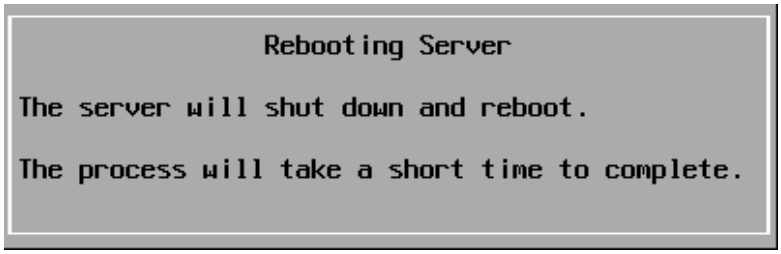
> CAUTION: The installer will repartition the selected disk. All data on the selected disk will be destroyed.

15. The ESXi 6.7 installation proceeds:



16. The installer displays the Installation Complete window when the installation process is completed:

17. Make sure that your installation medium has been ejected and your system is set to the boot disk. Then press Enter to reboot the system.



The installation is now complete.

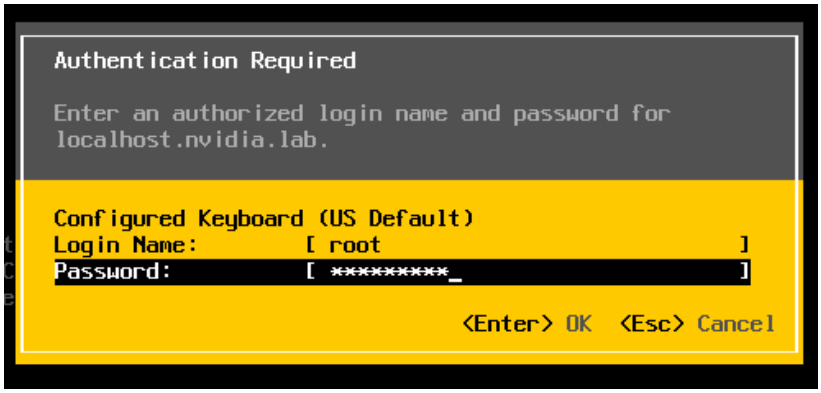
2.4 Initial Host Configuration

ESXi 6.7 displays a countdown timer when you first boot it. You can wait for the countdown to expire or press Enter to proceed with booting. ESXi 6.7 displays a series of notifications during the boot process.

When the boot process completes, VMware ESXi 6.7 displays this screen:

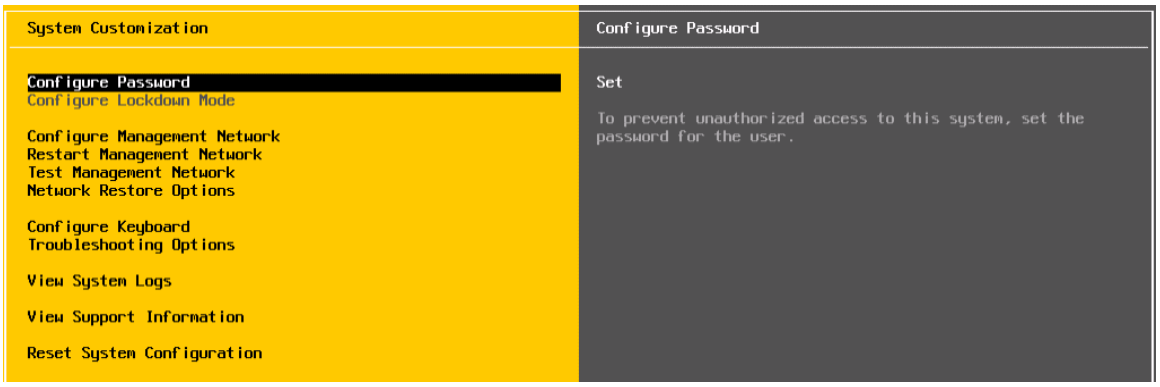
To configure the host:

1. Press F2. ESXi displays the Authentication Required window:



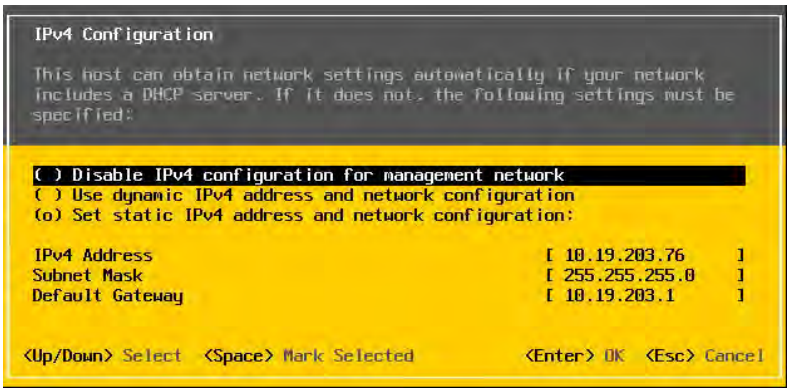
2. Enter the root account credentials that you created during the installation process, then press Enter.

ESXi displays the System Customization screen:



3. Select Configure Management Network with the down-arrow key, then press Enter. ESXi displays the Network Adapters window:

4. Select the adapter to use as the default management network with the arrow keys, then press Enter. ESXi displays the IPv4 Configuration window:



5. Select Set static IPv4 address and network configuration with the up and down arrow keys, then enter the IPv4 address, subnet mask, and default gateway in their respective fields. Then press Enter to apply the new management network settings.
6. ESXi displays the Confirm Management Network dialog. Press the Y key to confirm your selection.
7. ESXi displays the DNS Configuration window. Add the primary and (if available) secondary DNS server address(es) in their respective fields.
8. Enter this vSphere host's host name in the Hostname field, then press Enter.

9. Select Test Management Network on the main ESXi 6.7 screen to open the Test Management Network window.
10. Perform the following tests:
    - Ping the default gateway.
    - Ping the DNS server.
    - Resolve a known address.
11. Return to the main ESXi 6.7 screen when you have completed testing, and then select Troubleshooting Options. ESXi displays the Troubleshooting Mode Options window:

12. Enable the ESXi Shell by selecting Enable ESXi Shell and pressing Enter. (Successive keypresses toggle vSphere Shell between enabled and disabled.) The window on the right displays the shell's status.

13. Enable SSH by selecting Enable SSH and pressing Enter. (Successive keypresses toggle SSH between enabled and disabled.) The window on the right displays the SSH status.

This completes the ESXi 6.7 installation. The ESXi 6.7 host is accessible by the IP that was set in the Management Network in this section.

Chapter 3. Installing VMware vCenter Server

This chapter covers installing VMware vCenter Server, including:

- Installing VCenter Server Appliance
- Adding Licenses to Your vCenter Server
- Adding a Host
- Setting the NTP Service on a Host
- Setting a vCenter Appliance to Auto-Start
- Mounting an NFS ISO Data Store

Review the prerequisites in General Prerequisites in section 1.6.4 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.

3.1 Installing vCenter Server Appliance

3.1.1 About VCSA

The VCSA is a preconfigured virtual appliance built on Project Photon OS that allows you to manage multiple ESXI 6.7 host and perform configuration changes from a single pane of glass. Since the OS was developed by VMware, it offers better performance and boot times than the previous Linux-based appliance. Furthermore, it uses an embedded vPostgres database, giving VMware full control of the software stack, and resulting in significant optimization for vSphere environments, and quicker release of security patches and bug fixes.

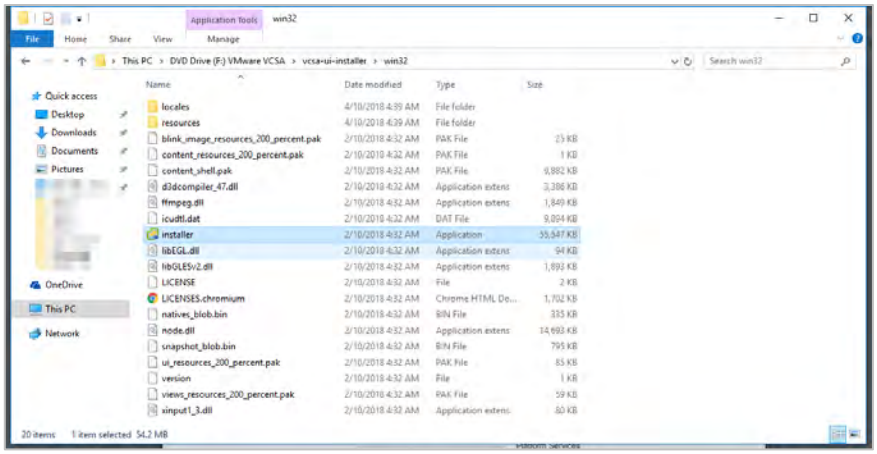
The VCSA scales up to 2000 hosts and 35,000 virtual machines. A couple of releases ago the VCSA reached feature parity with its Microsoft Windows counterpart, and is now the preferred deployment method for vCenter Server. Features such as Update Manager are bundled into the VCSA, as are file-based backup 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 must be deployed to an ESXi host running version 5.5 or above. However, all hosts you intend to connect to VCSA should be running ESXi 6.0 or above. Hosts running v5.5 and earlier cannot be managed by the VCSA and do not have a direct upgrade path.
  - You must check compatibility of any third-party products and plugins that may be used for backup, virus protection, monitoring, etc., as they may need upgrading for ESXi compatibility.
  - To check version compatibility with other VMware products, see the Product Interoperability Matrix.
- Architectural Considerations:
  - When you implement a new vSphere environment, you must plan its topology in accordance with the VMware vCenter Server and Platform Services Controller Deployment Types.
  - Most deployments include vCenter Server and Platform Service Controller in one appliance, following the embedded deployment model, which is used in this guide.
- 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 in v6.5.
  - Storage requirements for the smallest environments start at 250 GB and increase depending on your specific database requirements. See the document Storage Requirements for further details.
  - If the PSC is deployed as a separate appliance it requires two CPUs, 4 GB of RAM, and 60 GB of disk storage.
  - Environments with ESXi host(s) that have more than 512 LUNs and 2048 paths should be sized large or x-large.
  - The ESXi host on which you deploy the VCSA must not be in lockdown or Maintenance Mode.
  - All vSphere components must be configured to use an NTP server. The installation may fail or the vCenter Server Appliance vpxd service may be unable to start if the clocks are not synchronized.
  - FQDN resolution must be enabled when you deploy vCenter Server.
  - Required Ports for vCenter Server and Platform Services Controller.
  - vSphere VMware Configuration Maximums.

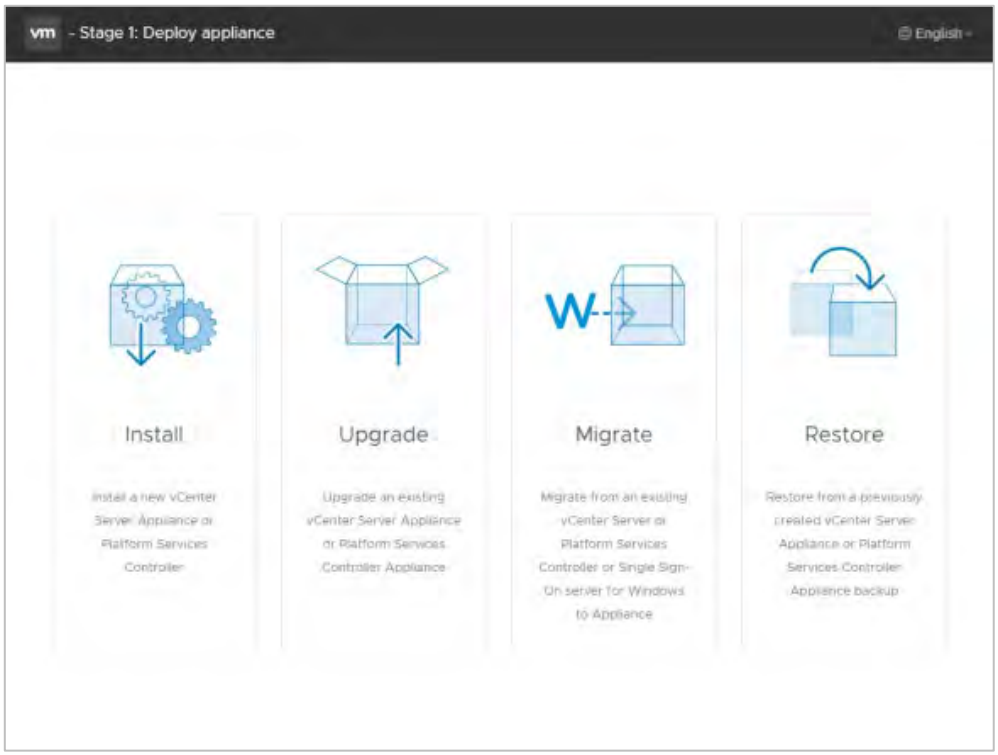
3.1.2 vCenter Server Appliance (VCSA) Installation

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

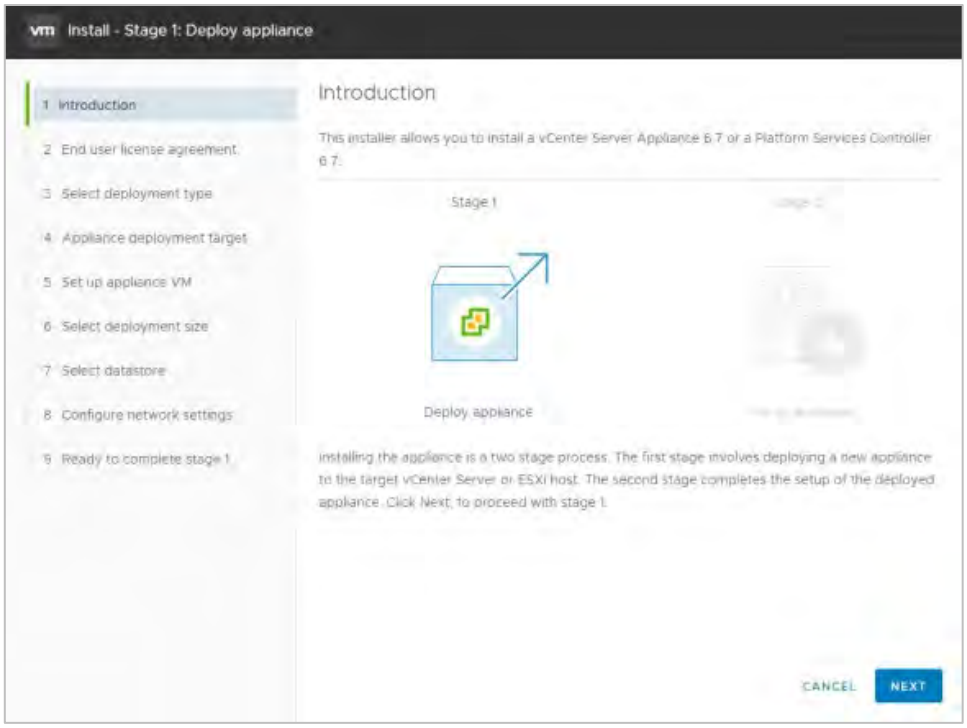
1. Mount the ISO on your computer. The VCSA installer is compatible with Mac, Linux, and Windows.
2. Browse to the corresponding directory for your operating system, e.g. `\vcsa-ui-installer\win32`. Right click Installer and select Run as administrator.



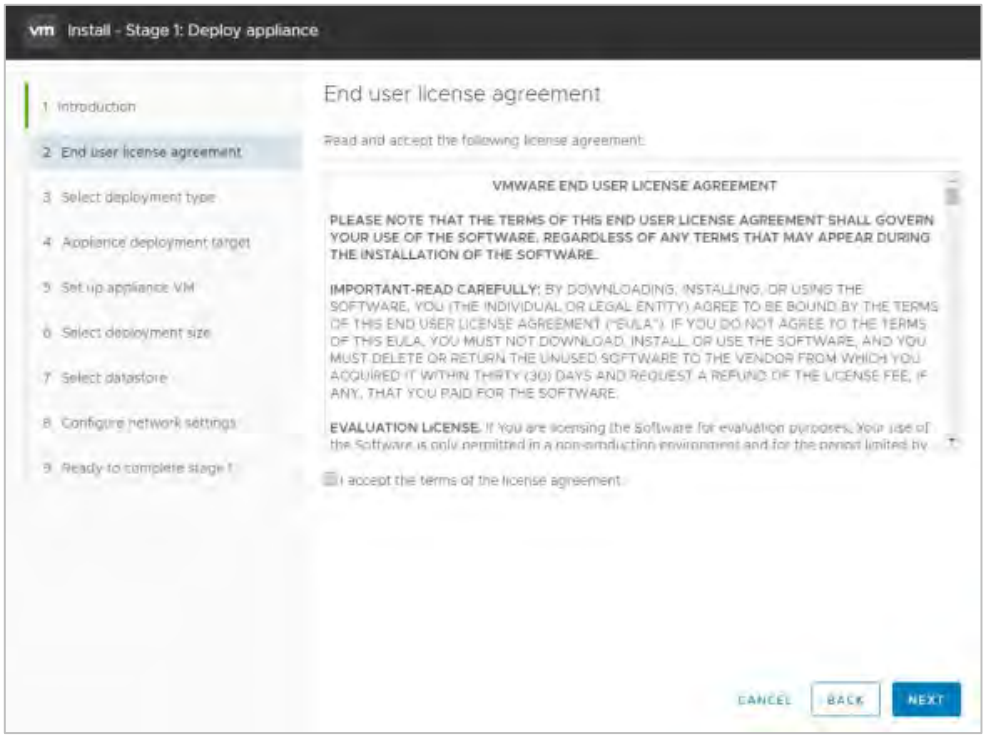
3. As you are installing a new instance, click Install.



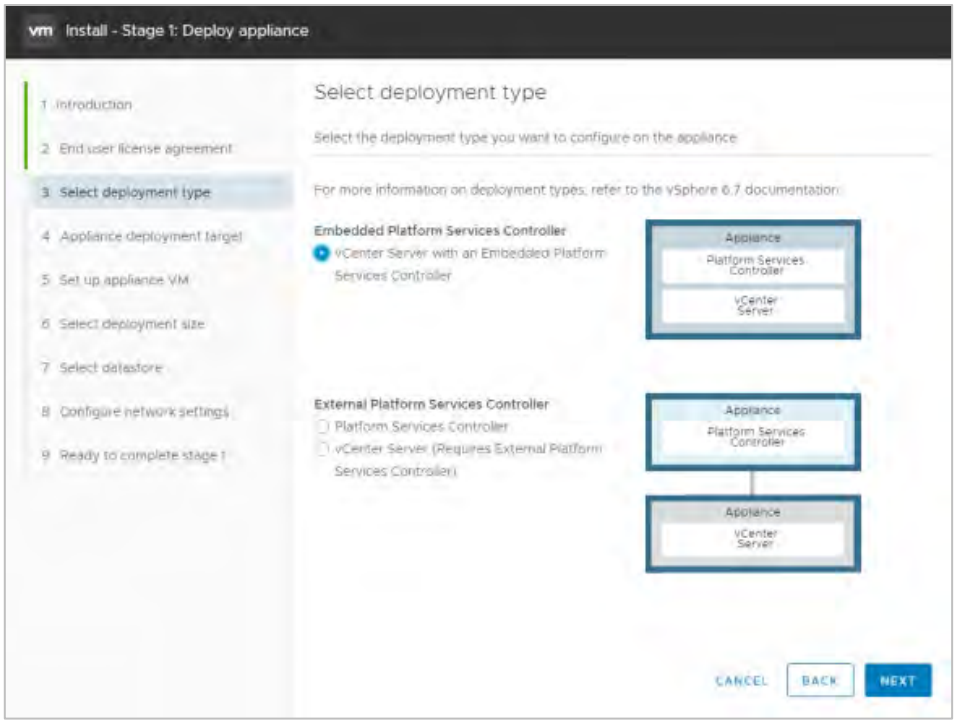
4. In Stage 1 of the install process, the installer deploys the appliance. Click Next to begin:



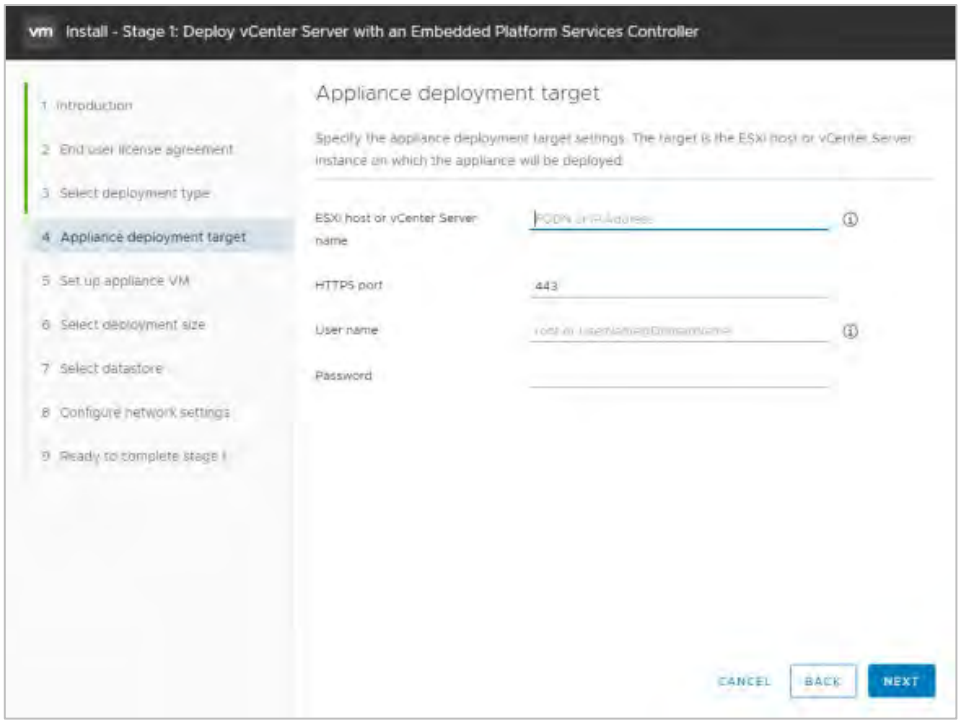
5. Read the EULA, then click Next to continue:



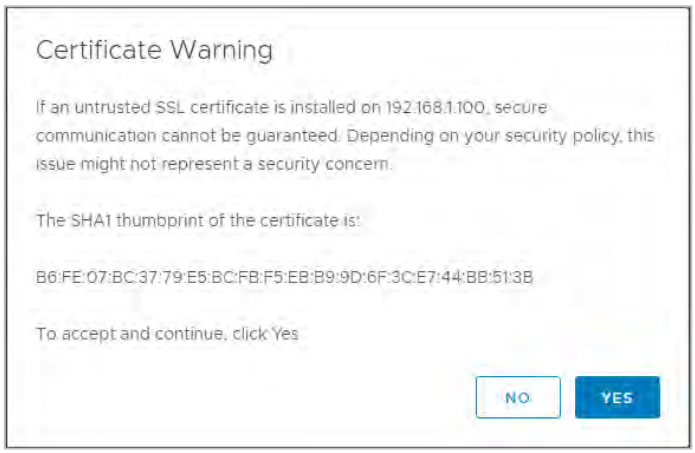
6. Select a deployment type. This example uses an embedded deployment combining the vCenter Server and Platform Services Controller in one appliance. Select Embedded Platform Services Controller, then click Next:



7. Select the ESXi host on which to install the VCSA as a guest. This must be a host that runs ESXi 5.5 or later. NVIDIA recommends 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 fully qualified domain name (FQDN) of the chosen host, then its root username and password., and click Next:

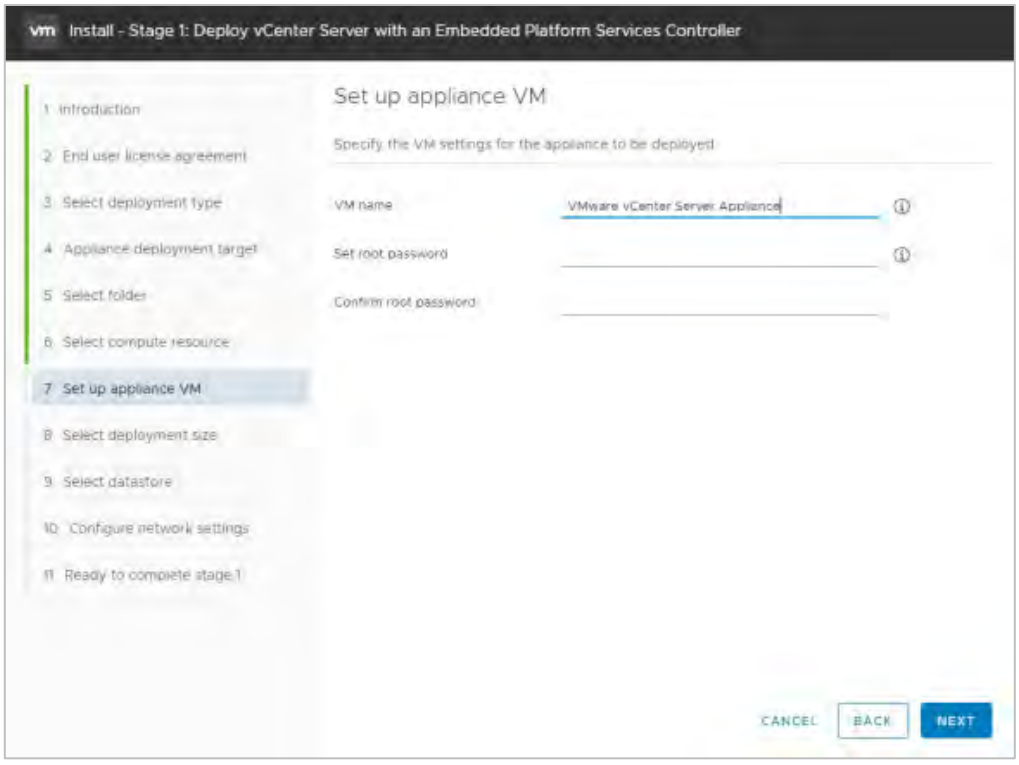


8. If your computer running the installer can access the host, the installer may display a certificate warning as it connects. This happens if the host is using a self-signed cert. If the host uses a signed certificate, this warning does not appear. If you get the warning, click Yes to continue:

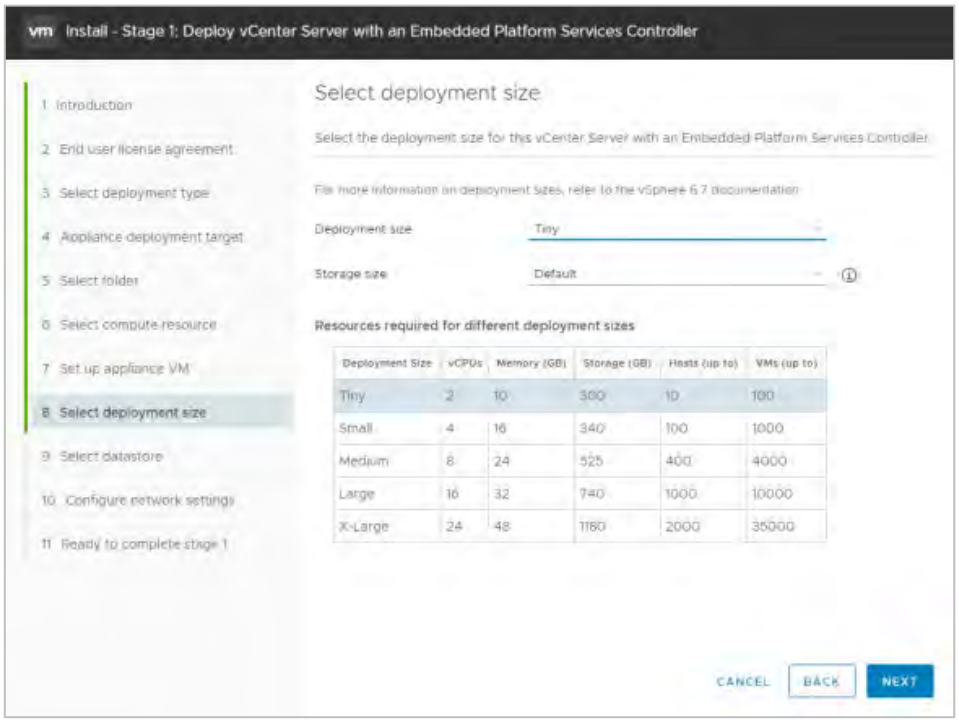


9. The installer validates the credentials you provided:

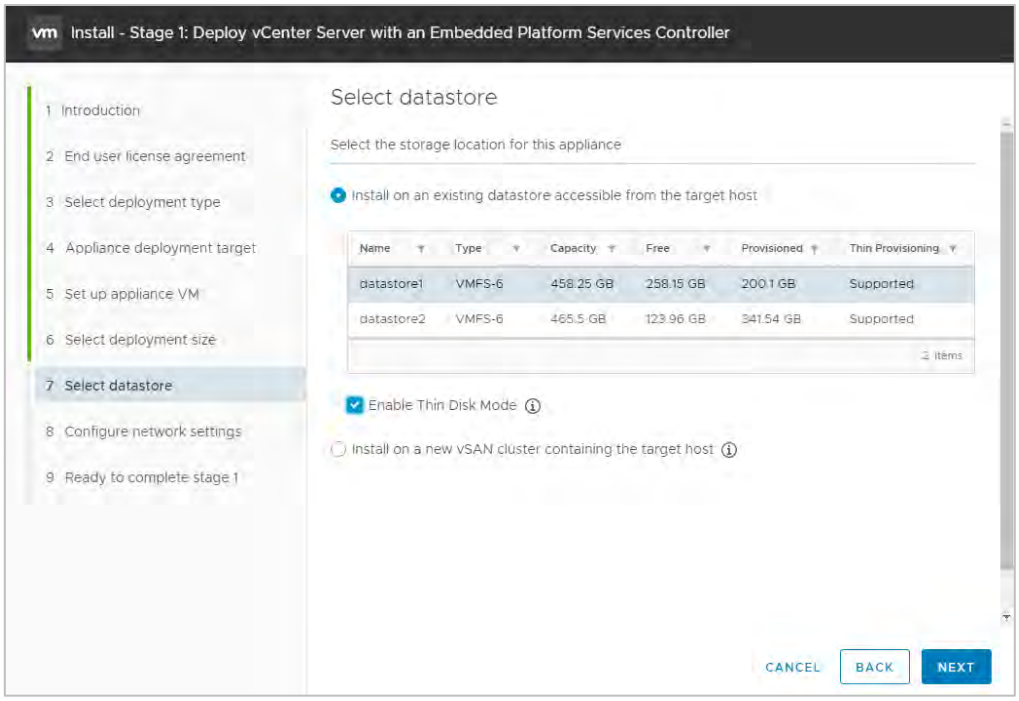
10. If the installer connects to the host successfully it prompts you to name the appliance, enter a root password for the appliance (twice, to check for typing errors), and click Next:



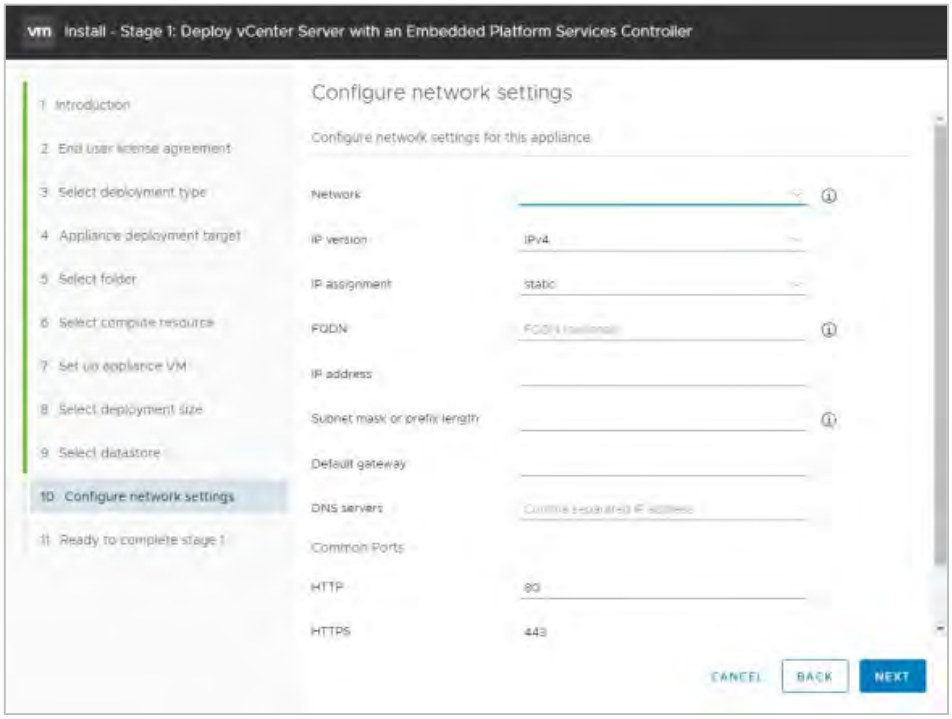
11. Select a deployment size appropriate to the number of hosts and virtual machines that that vCenter Server will manage, then click Next:



12. Select the datastore where the VCSA is to be deployed'; select thin provisioning if required, then click Next. Configure the network settings for the appliance and click Next.

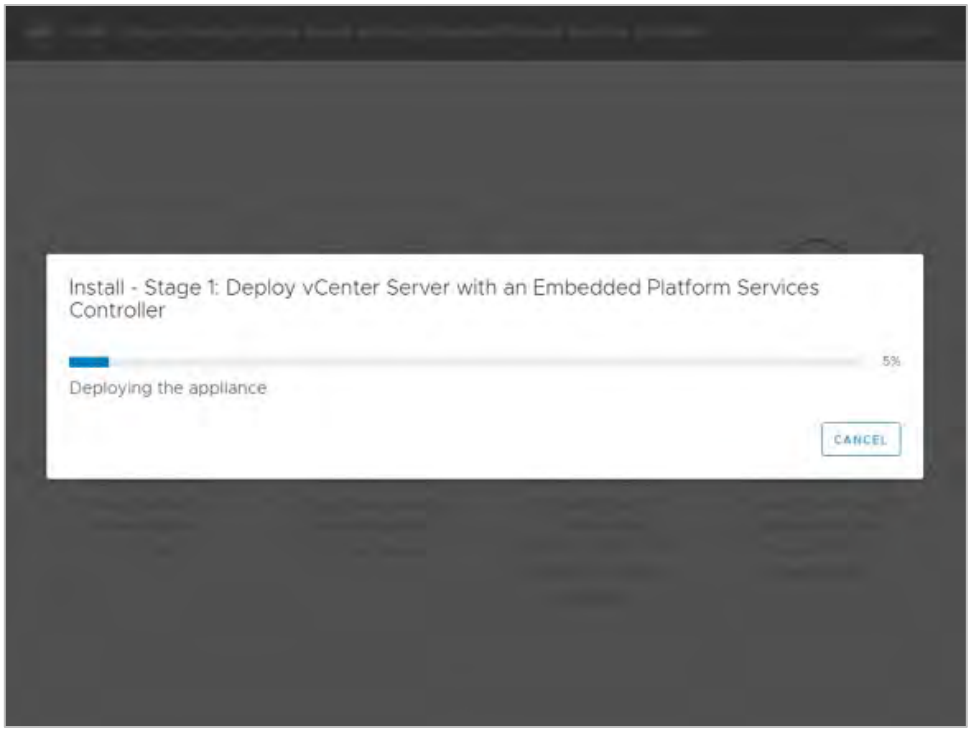


13. The installer displays a page for configuring network settings. This is a long page which you must scroll to display all of the settings.

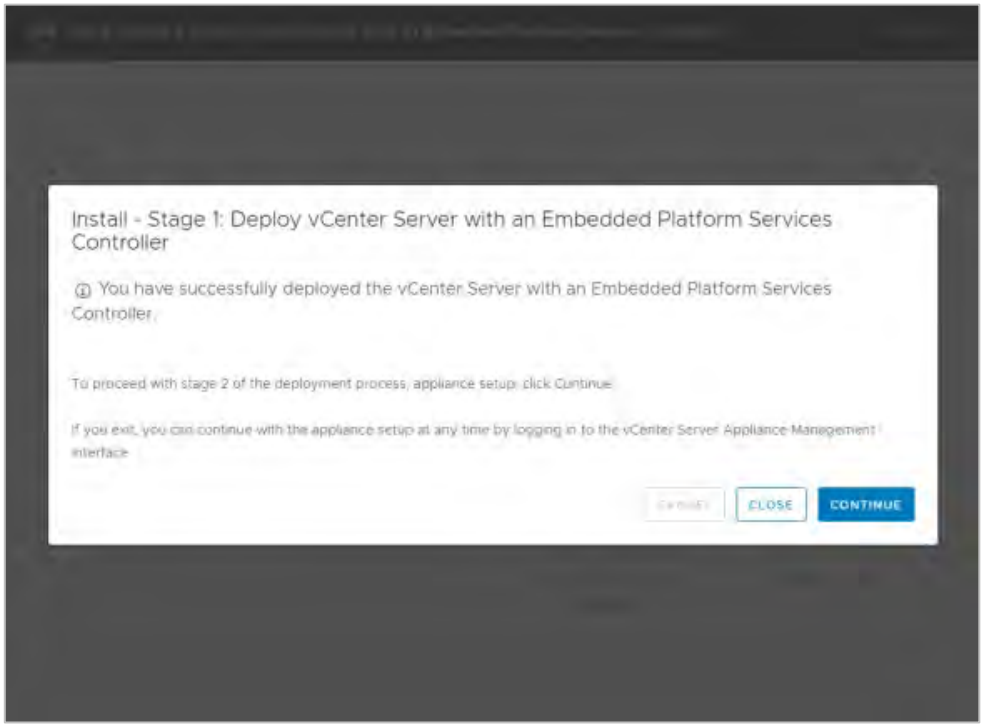


Before you configure these settings, choose an appropriate static IP address and enter it into local DNS (e.g. on the Domain Controller). Once you can resolve the address, enter the IP address its host name on the network setting page, then scroll down and enter remaining items. When the settings are complete, click Next.

14. The installer displays a summary page. Click Finish. The installer deploys the appliance.

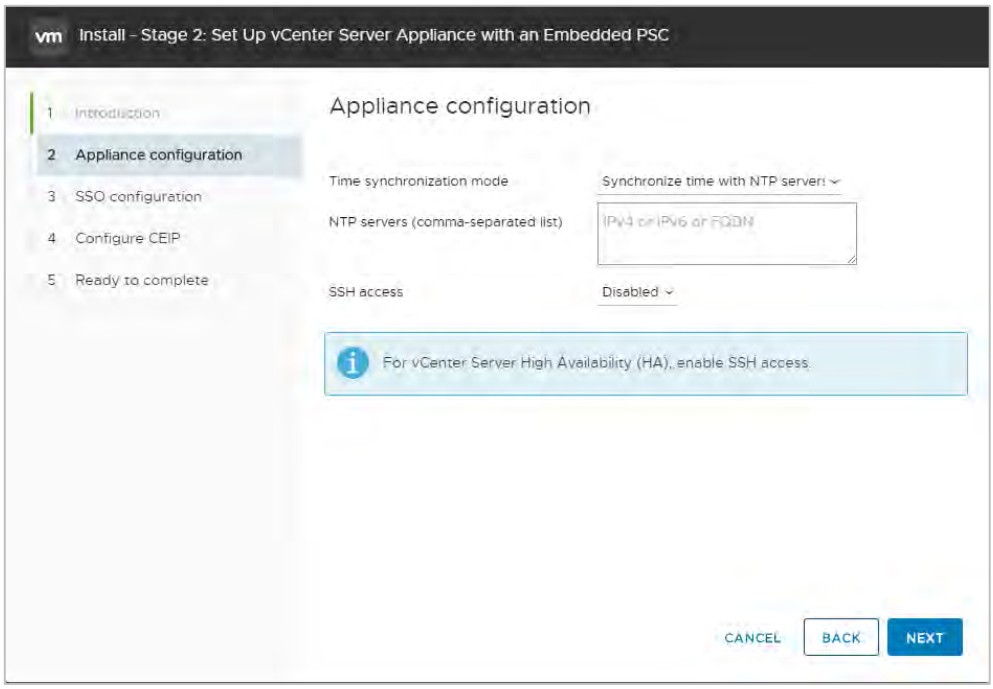


15. The VCSA is now deployed. Click Continue to proceed to the install process's Stage 2, setting up the VCSA.



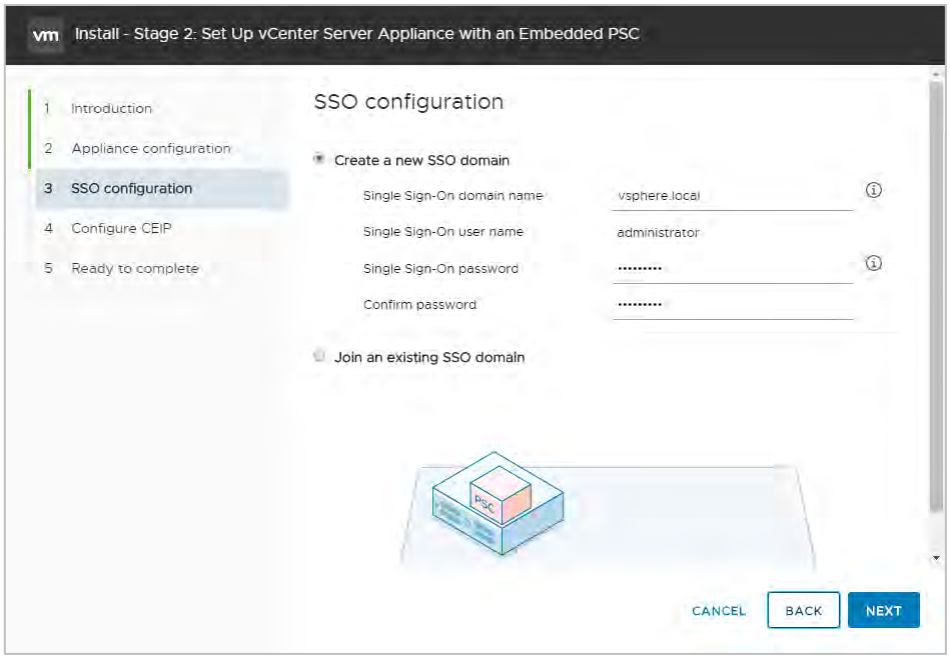
16. Click Next to begin the VCSA setup.

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

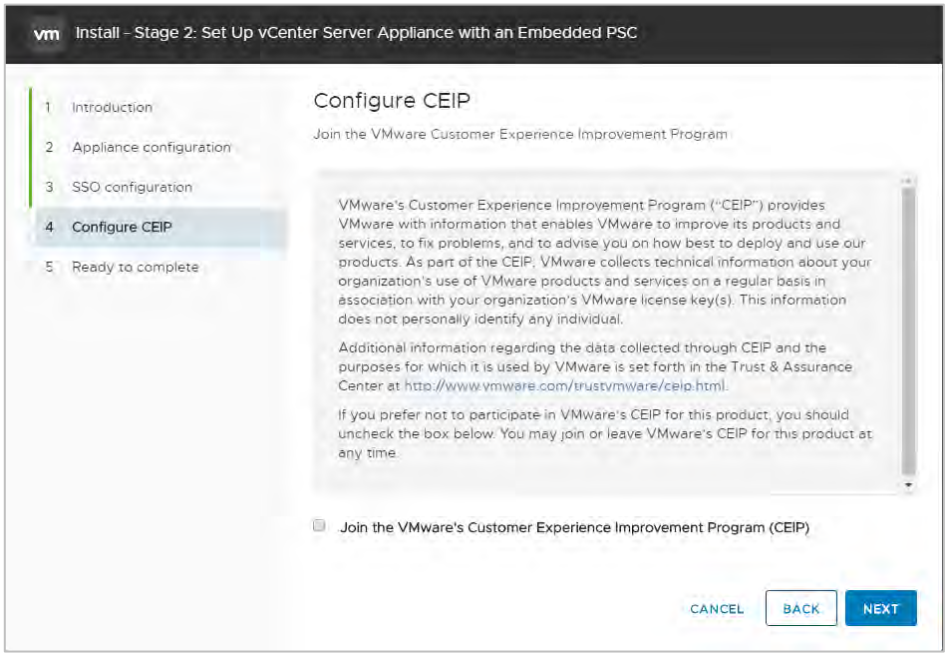


18. Enter a unique SSO domain name. The default name is `vSphere.local`. Configure a password for the SSO administrator, then click Next.

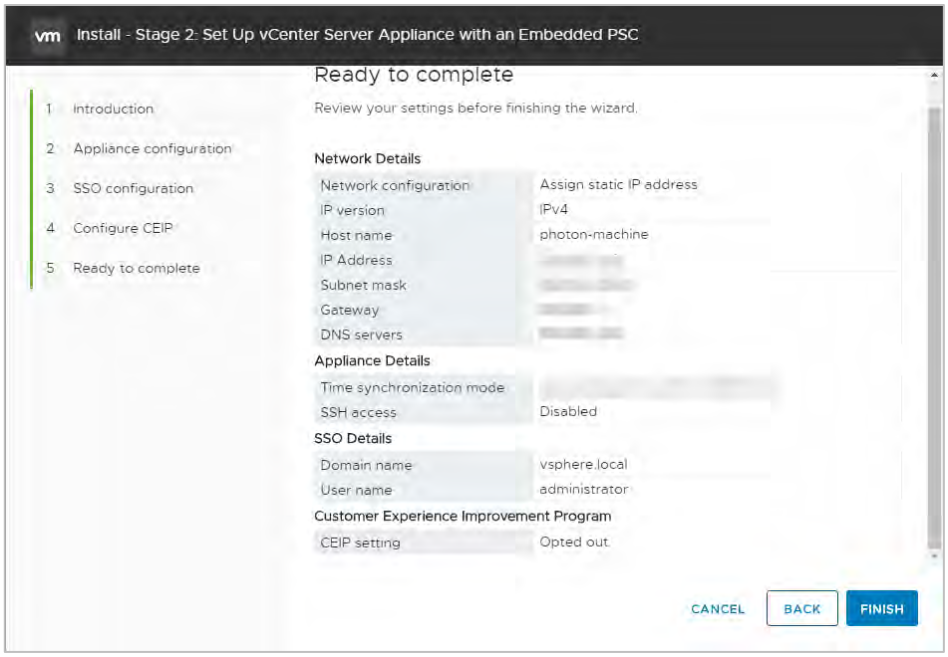
Do not make the SSO domain name the same as your Active Directory Domain.



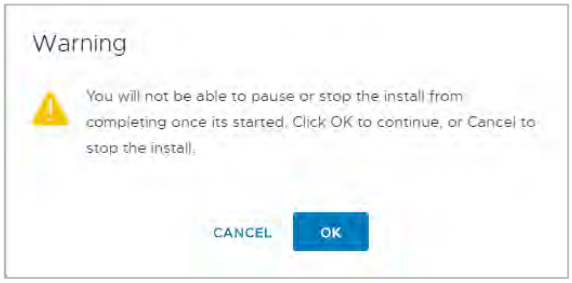
19. Check or clear the checkbox that opts in to the VMware Customer Experience Improvement Program, then click Next:



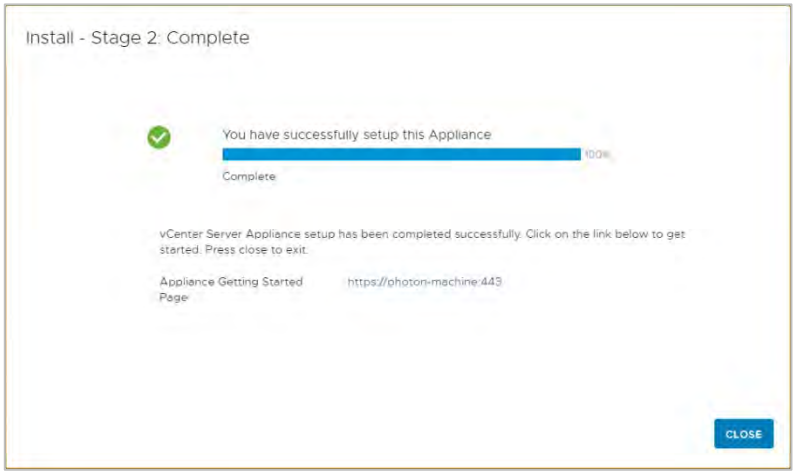
20. The installer displays a summary page. Review the details on this page, then click Finish.



21. The installer displays a warning that you cannot pause or stop the install once you start it. Click OK to acknowledge the warning a start the install.



22. When the install process is complete, click Close to close the installer:



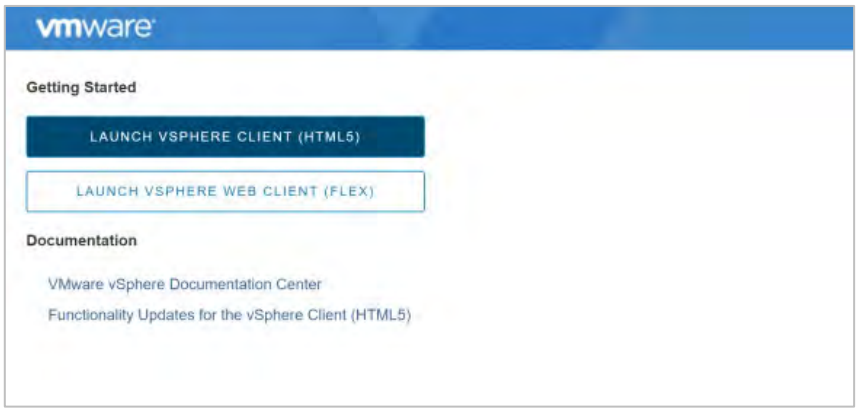
3.2 Post Installation

This section describes post install configuration of VCSA.

3.2.1 Adding Licenses to Your vCenter Server

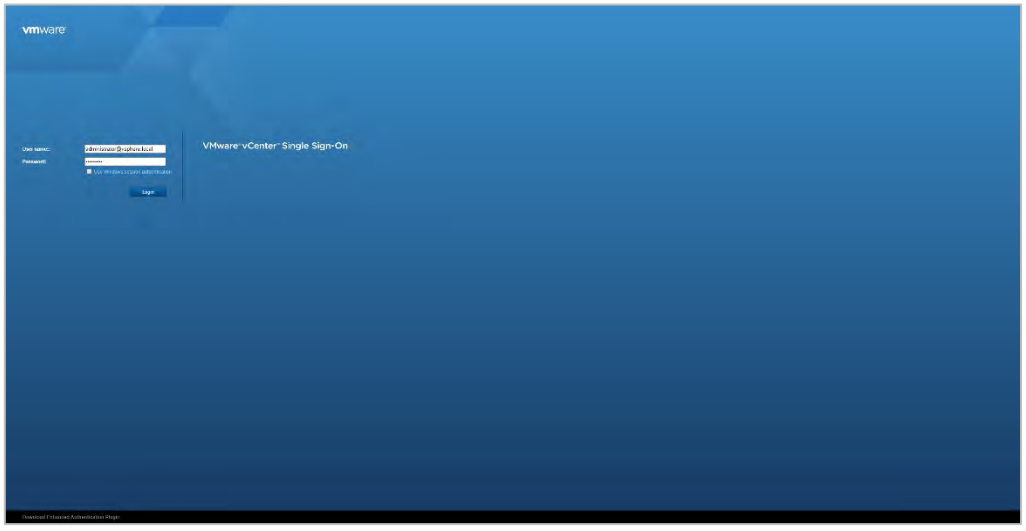
Logging into vCenter:

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).

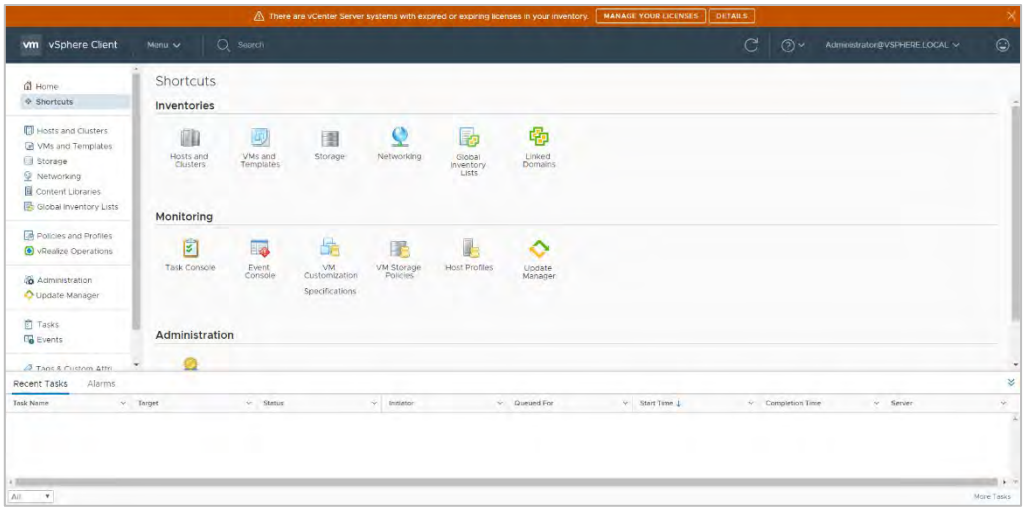


NVIDIA recommends selecting the HTML5 client. The web client will be depreciated in a future version, and the HTML5 client is at nearly full feature parity.

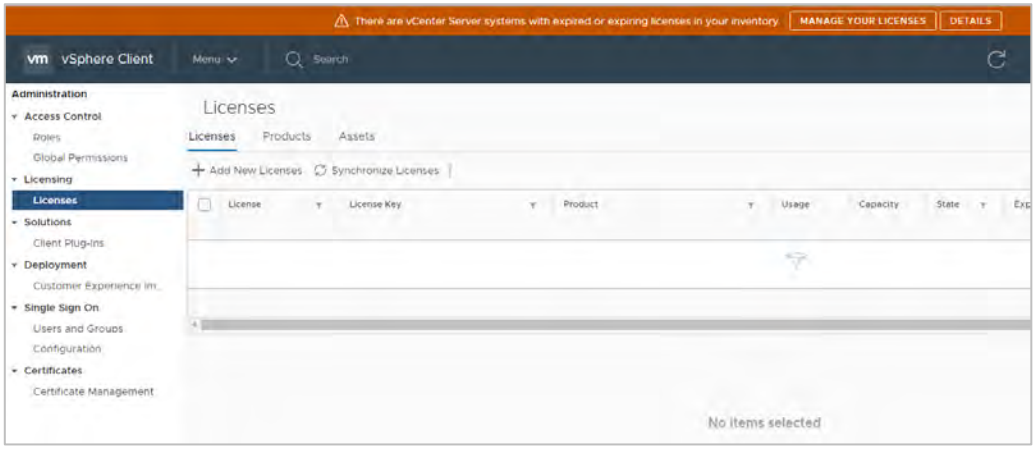
2. vCenter Server displays the VMware Single Sign-On page:



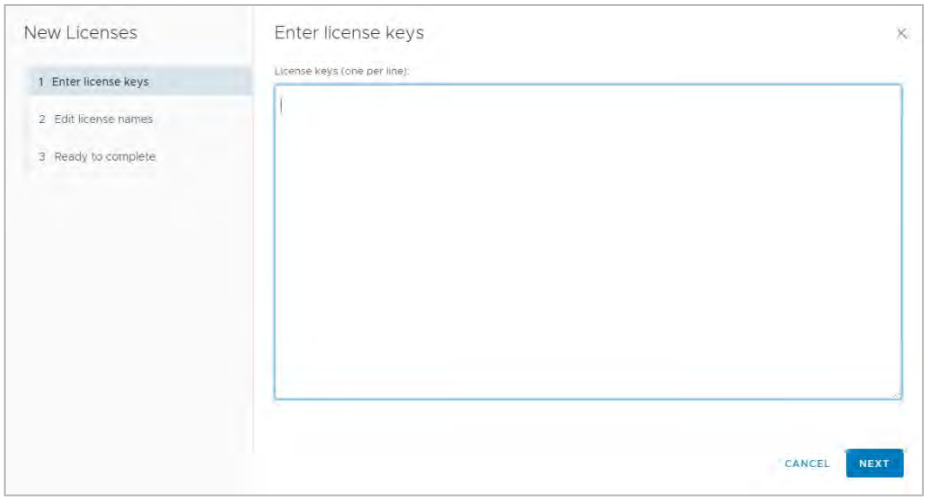
3. Enter the username and password that you specified during installation, then click the Login button. vCenter Server displays the VMware vSphere Web Client page:



4. You must apply a new vCenter license key within 60 days. If you have purchased vCenter Server, log in to your licensing portal. Select your license and log in to the vSphere Web Client using the SSO administrator login. (If the license key does not appear then check with your VMware account manager.) Click the Menu drop-down, then click Administration.

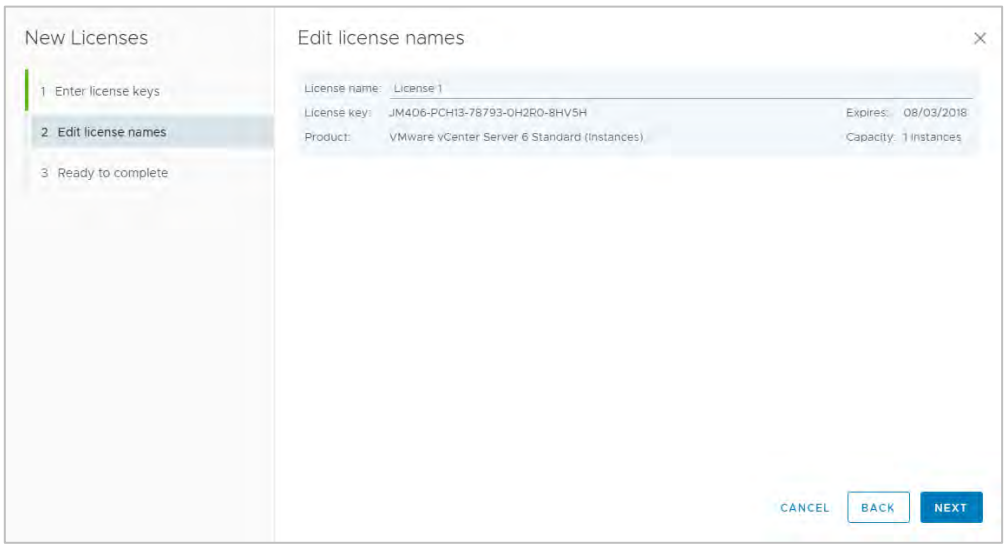


5. Select Licenses from the left-hand menu, then select the Licenses tab. Click Add New Licenses to open the New Licenses dialog:

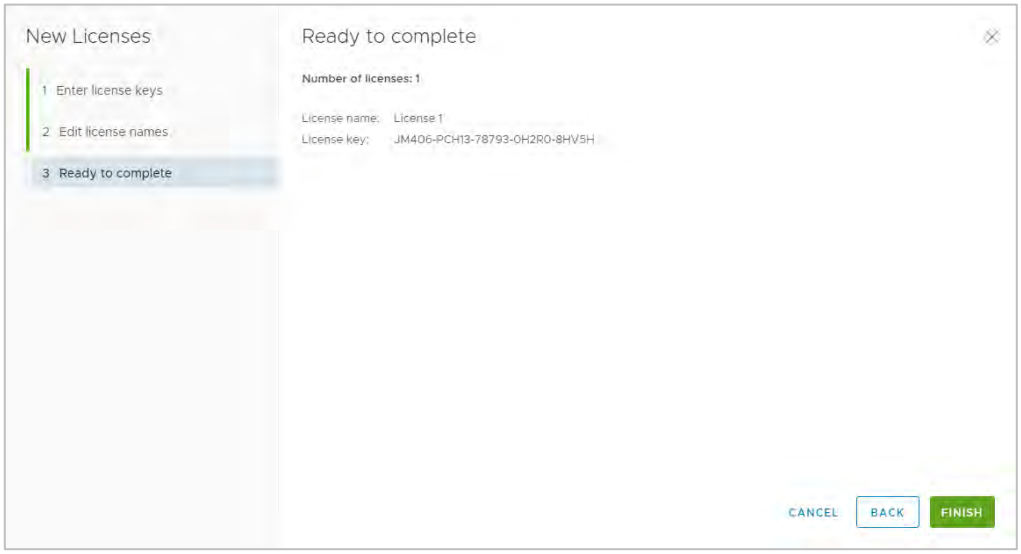


6. Enter the vCenter Server Standard license key provided at the vSphere beta program website:

7. Enter a unique name for the license in the License Name field, then click Next.



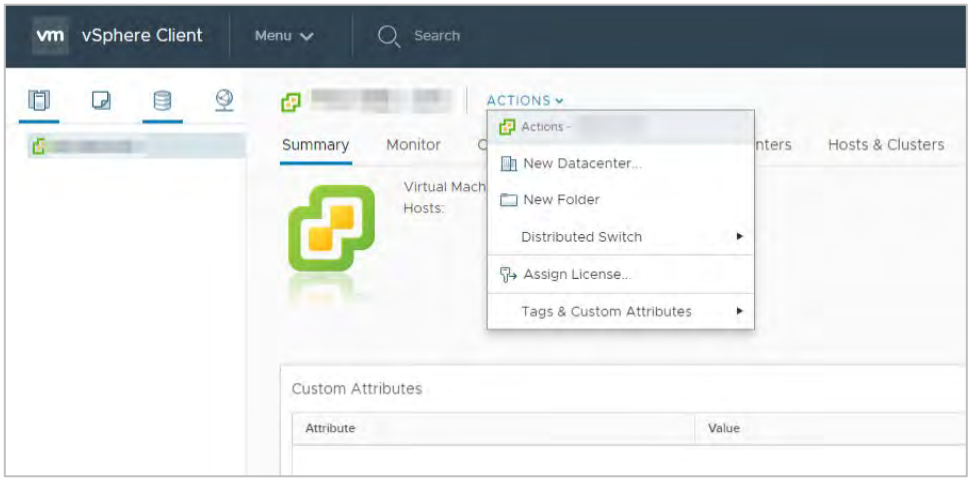
8. Review your selections, then click Finish to close the Enter New License dialog and return to the VMware vSphere Web Client page:



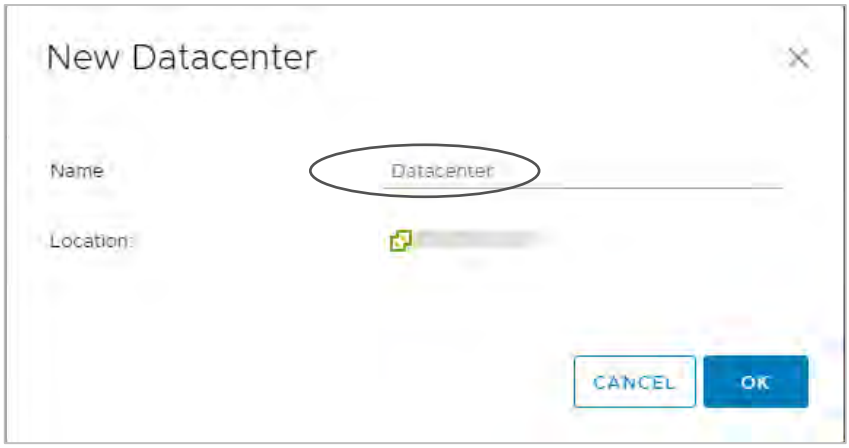
3.2.2 Adding a Host

To add a host in vCenter:

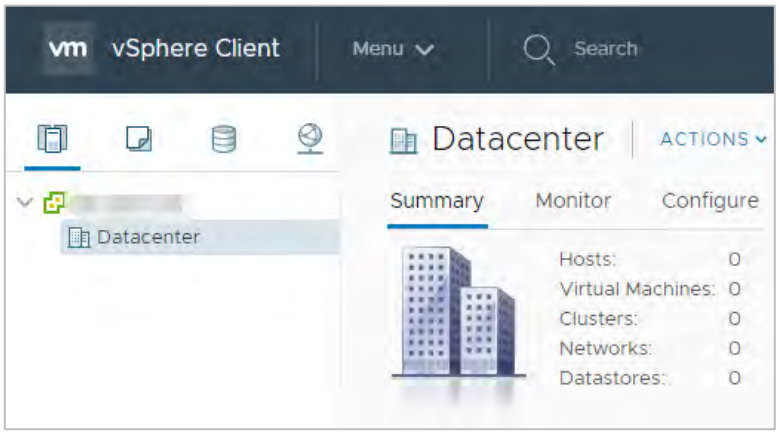
1. Select the Home icon (an outline of a house) on the VMware vSphere Web Client page.
2. Select Hosts and Clusters.
3. Click the ACTIONS dropdown and select New Datacenter.



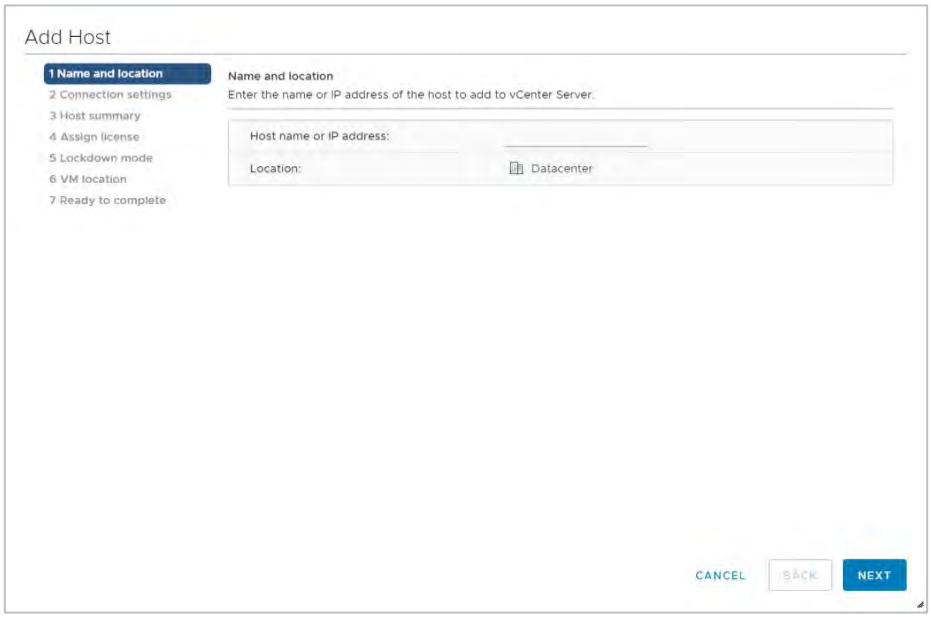
4. vCenter displays the New Datacenter dialog:



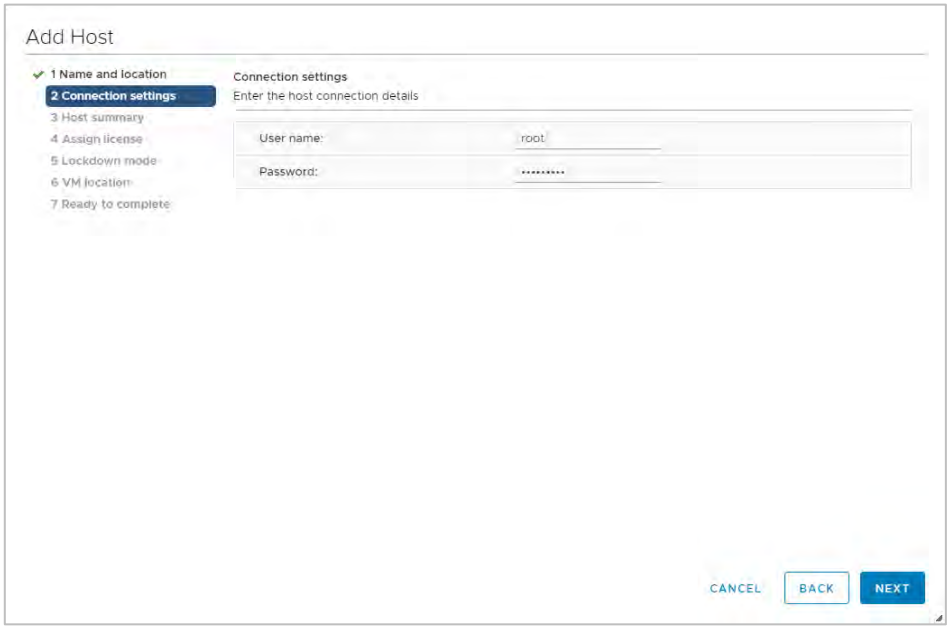
5. Enter a name for the datacenter in the Datacenter name field, then click OK. vCenter displays the new datacenter in the left panel of the page.



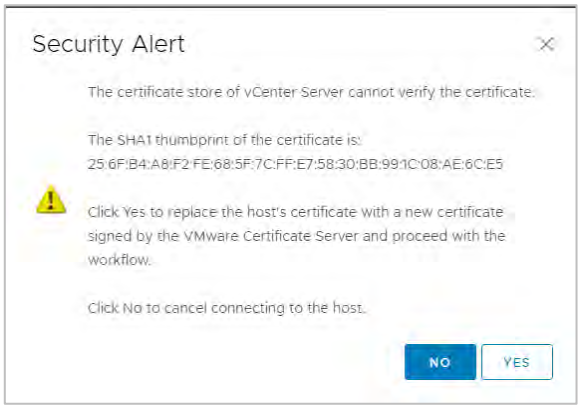
6. Click the ACTIONS dropdown and select Add a Host. vCenter opens the Name and location dialog:



7. Enter the host name or IP address of the vSphere host and click Next. vCenter displays the Connection settings dialog:



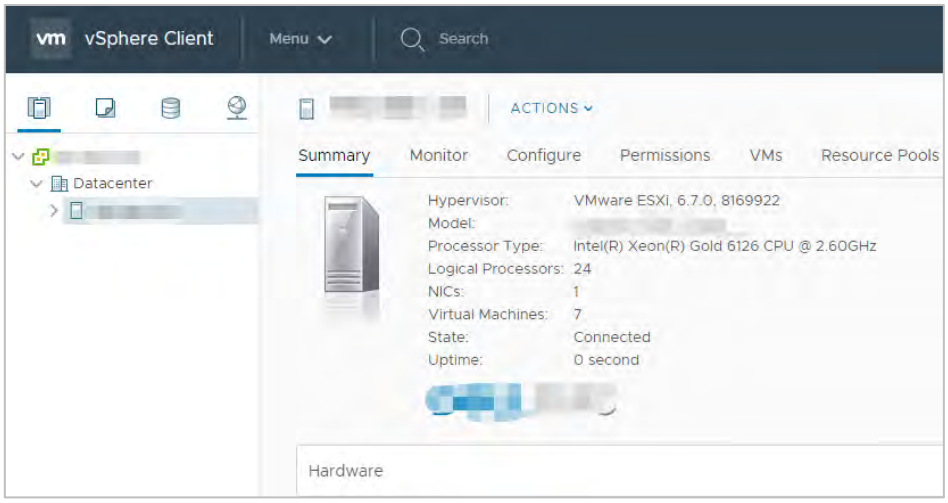
8. Enter the administrator account credentials in the Username and Password fields, then click Next. vCenter displays the Security Alert dialog"



9. Click Yes to replace the host certificate. vCenter displays the Host summary dialog.
10. Review the settings.

    If any settings are incorrect, click the Back button to return to an earlier screen and correct them.

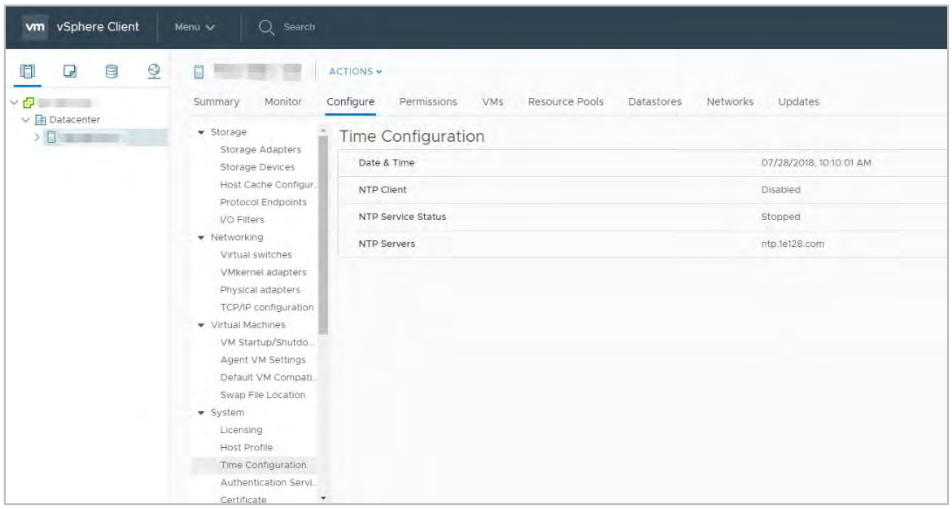
    If all of the settings are correct, click Next to continue. vCenter displays the Assign license dialog.
11. Confirm the license selection and click Next. vCenter displays the Lockdown mode dialog.
12. Accept the default setting (Disabled) and click Next. vCenter displays the VM location dialog.
13. Select a cluster or accept the default option and click Next to continue. vCenter displays the Ready to complete dialog.
14. Click Finish to add the new host. vCenter now displays the new host in the left-hand panel when you click the datacenter name.



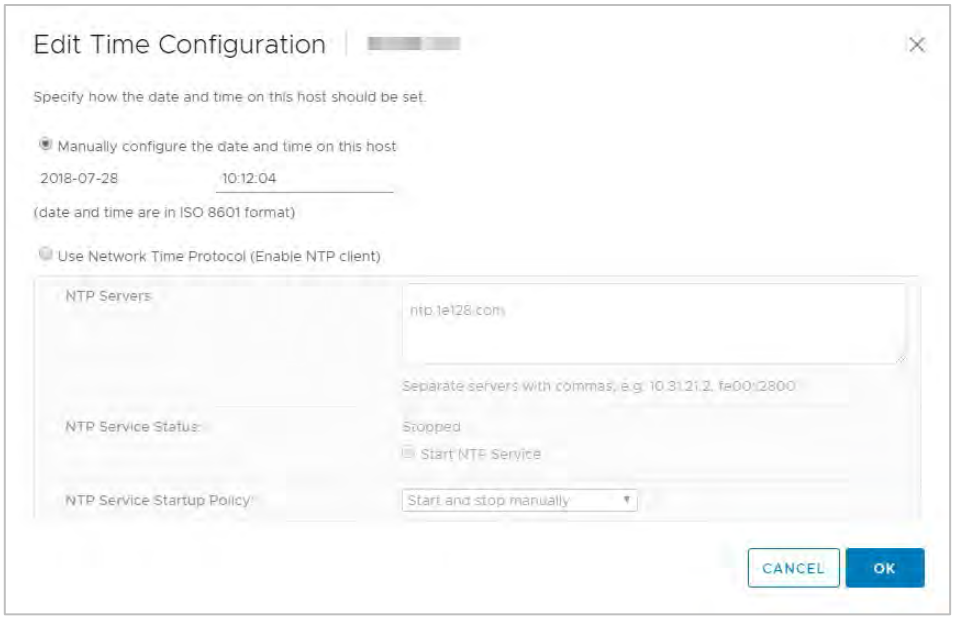
3.2.3 Setting the NTP Service on All Hosts

To ensure that time is accurate for all guests:

1. Click the first host object in the menu on the left:



2. Click Configure > System > Time Configuration > Edit
3. Enter a valid time server and click OK:

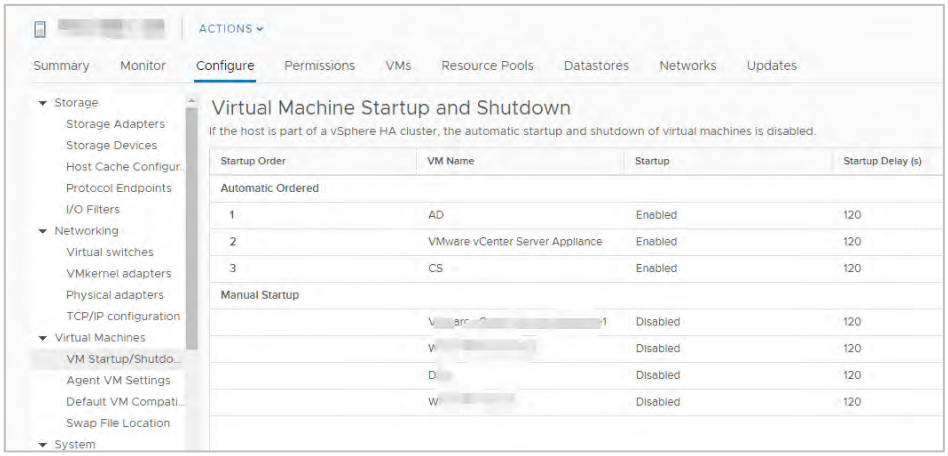


4. Click the next host object in the menu on the left, then repeat steps 2 and 3. Repeat until you have set the time for each host object defined on the server.

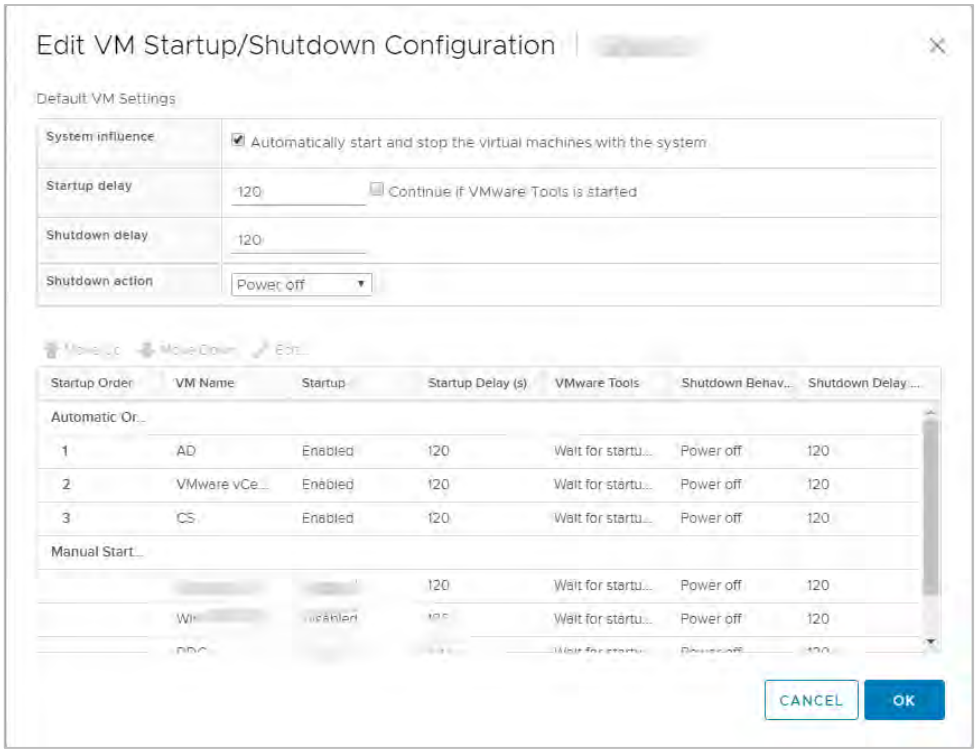
3.2.4 Setting a vCenter Appliance to Auto-Start

To set a vCenter Appliance to start automatically:

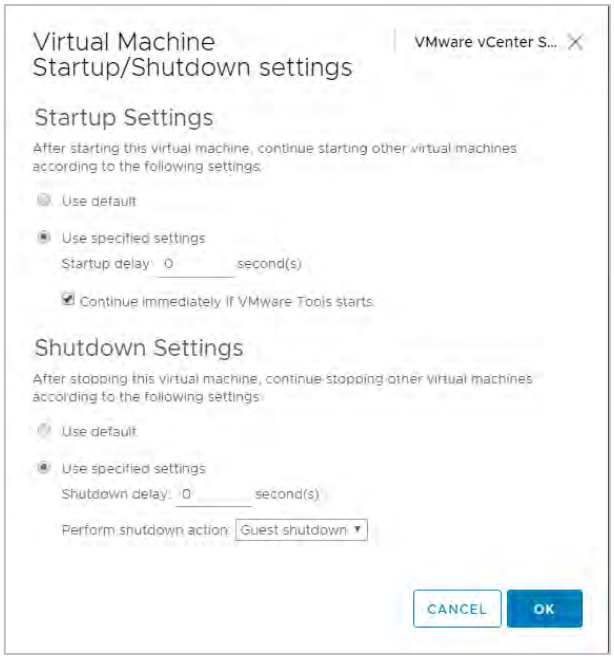
1. Select the host for the appliance on the vSphere Web Client, then select Configure > Virtual Machines > VM Startup/Shutdown.



2. Click the Edit button. vCenter displays the Edit VM Startup and Shutdown dialog:



3. Select vCenter Appliance, then press the Up Arrow key to move that virtual machine up to the Automatic Startup section of the appliance table. Then click the Edit button.
4. Select or set the following options:
   - Select Use specified settings.
   - Set Startup delay to 0 seconds.
   - Check the checkbox for Use specified settings > Continue immediately if VMware Tools starts.
   - Select Use specified settings.
   - Set Shutdown delay to 0 seconds.
   - Set Perform shutdown action to Guest shutdown.



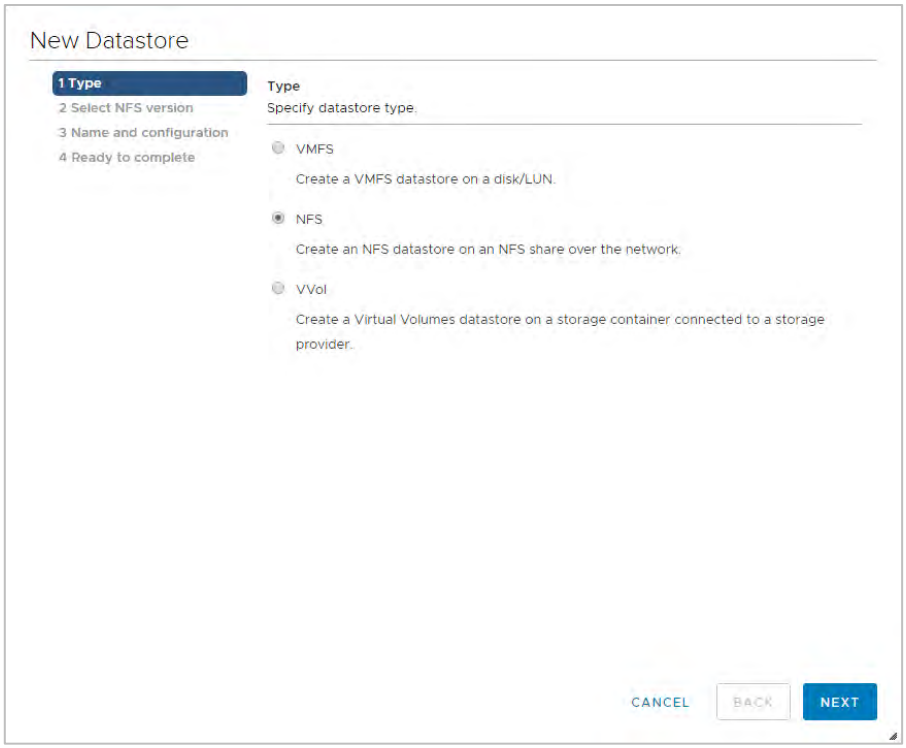
5. Click OK to apply the configuration settings.

> 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.

3.2.5 Mounting an NFS ISO Datastore

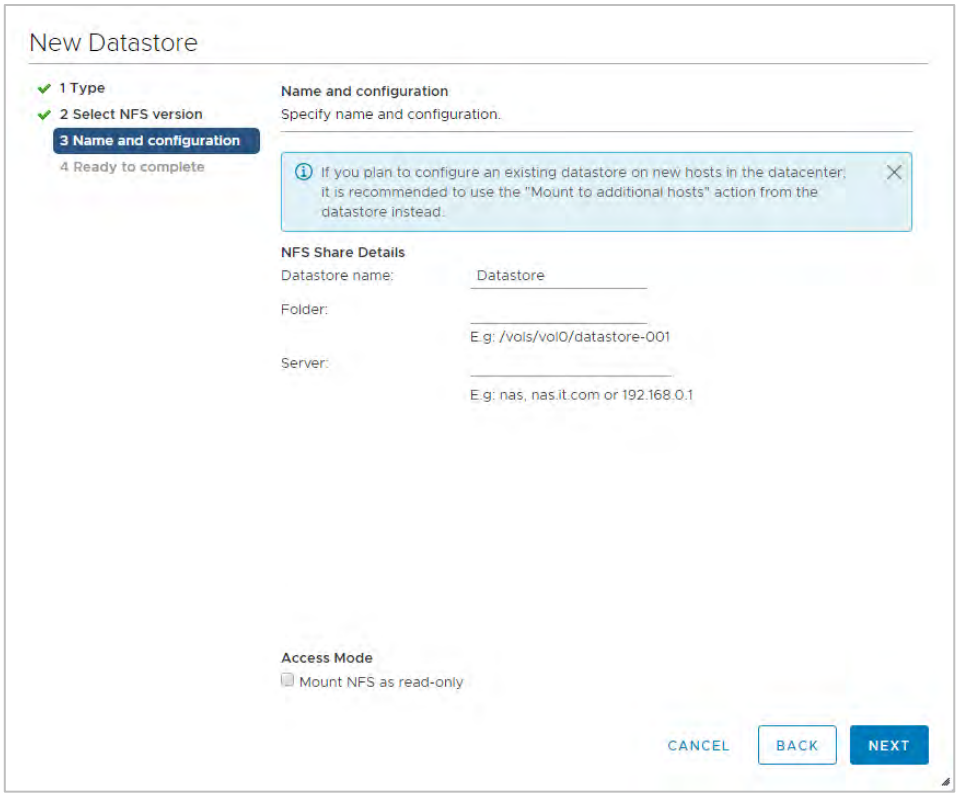
To mount an NFS ISO datastore:

1. In the main vSphere Web Client window, select Hosts and Clusters, then select the host.
2. Open the Actions dropdown, the select Storage > New datastore. vCenter Server displays the New Datastore window displays with the Type tab selected.



3. Select NFS, then click Next to continue. vCenter Server displays the Select NFS version tab.
4. Select the correct NFS version, then click Next to continue. vCenter Server displays the Name and configuration tab.
5. Enter the NFS exported folder path in the Folder field, and the NFS server address in the Address field.

   Because this an ISO datastore, consider making it read-only by checking the Mount NFS as read-only checkbox.
6. Click Next to continue. vCenter Server displays the Host accessibility tab.
7. Select the host that is to use the new datastore,
8. Click Next to continue. vCenter Server displays the Ready to complete tab.



9. Review the settings.

   If any settings are incorrect, click Back to return to an earlier page and fix them.

   If all of the settings are correct, click Next.

10. Click Finish to complete the procedure for adding the NFS ISO datastore.

The new datastore is now accessible as an installation source for virtual machine CD drives.

Chapter 4. Building Horizon Composer Server

This chapter covers installing VMware Horizon 7 Composer Server, including:

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

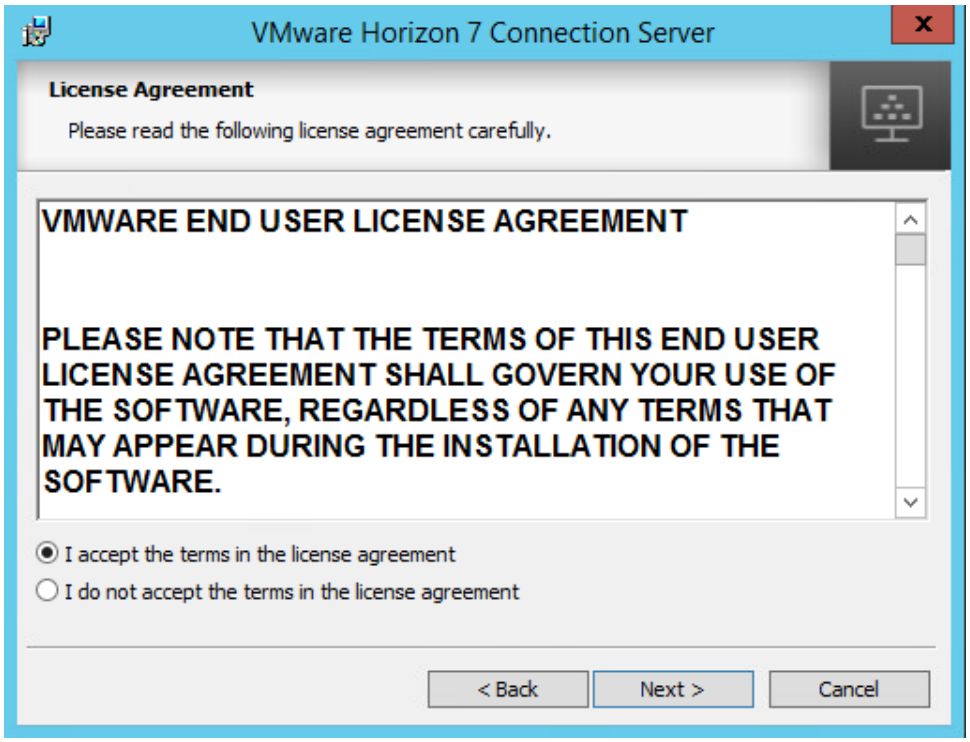
4.1 Installing Horizon

The server on which you install Horizon View must meet the requirements listed in General Prerequisites. As of this deployment guide's publication date, the current version of the Horizon Connection Server is version 7.5.1, and is installed by `VMware-viewconnectionserver-x86_64-7.5.1-9122465.exe`.

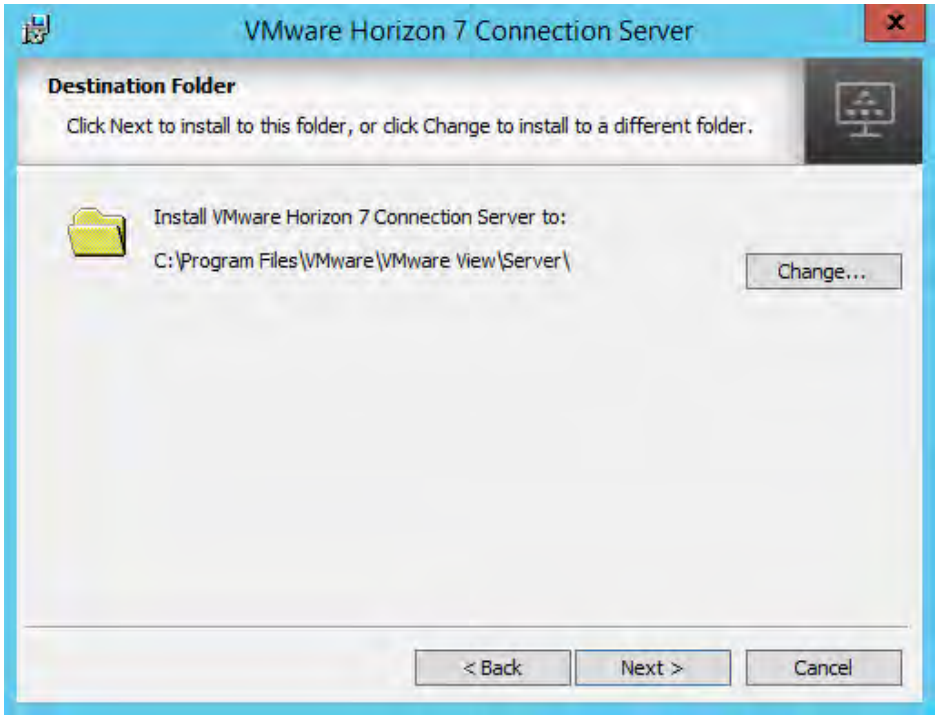
To install View 7.5.1:

1. Launch the installer. It displays the Welcome to the Installation Wizard screen.

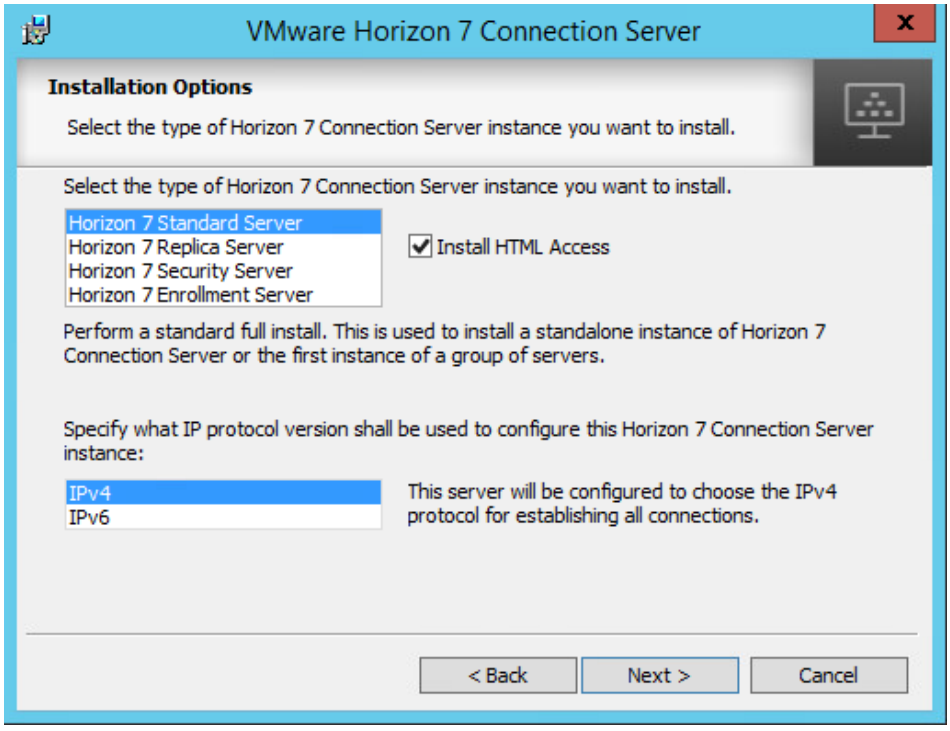
2. Click Next to display the End User License Agreement screen.



3. Click the I accept the terms of the license agreement radio button to accept the agreement, then click Next to continue.



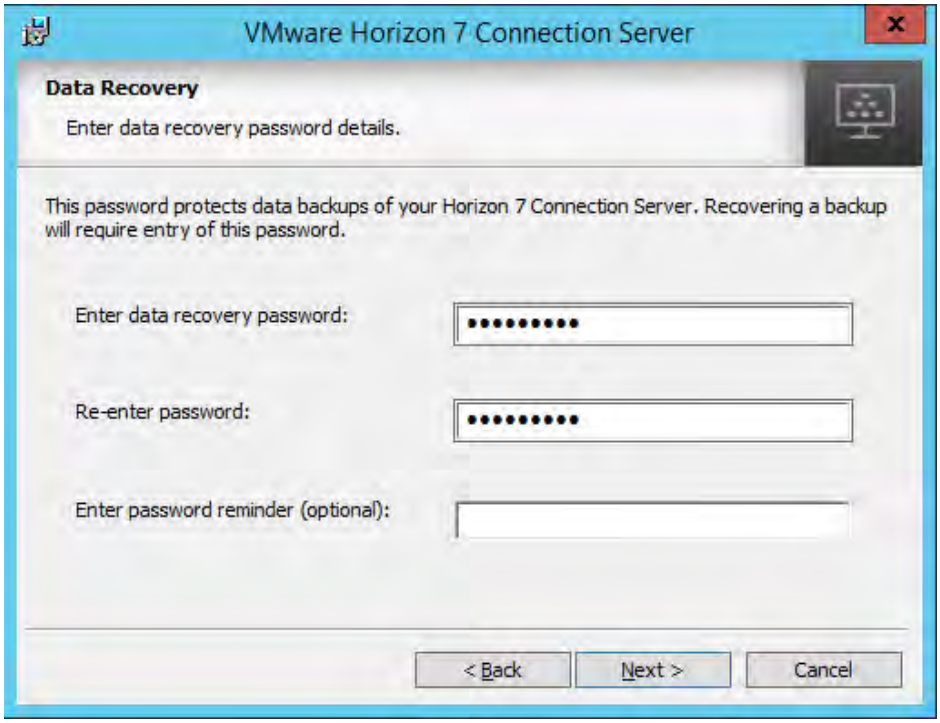
4. The Horizon installer displays the Destination Folder window. Choose an install location, then click Next.
5. The Horizon installer displays the Installation Options screen:



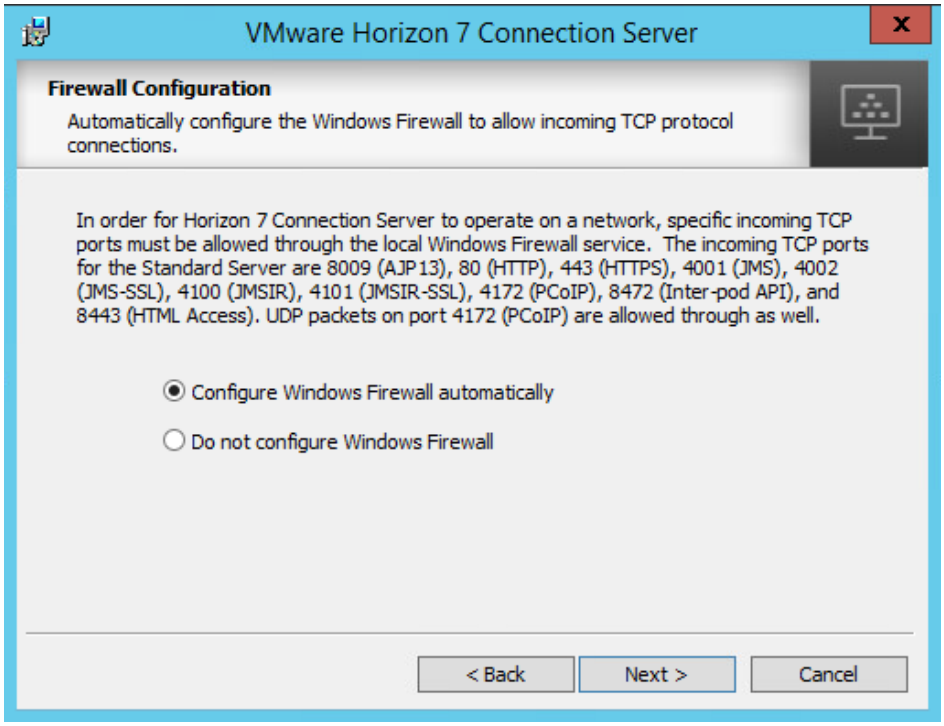
- Set the type of Horizon 7 Connection Server instance to Horizon 7 Standard Server.
- Check the Install HTML Access checkbox,
- Select IP protocol version IPv4.

- Click Next to continue.

6. The Horizon installer displays the Data Recovery screen.

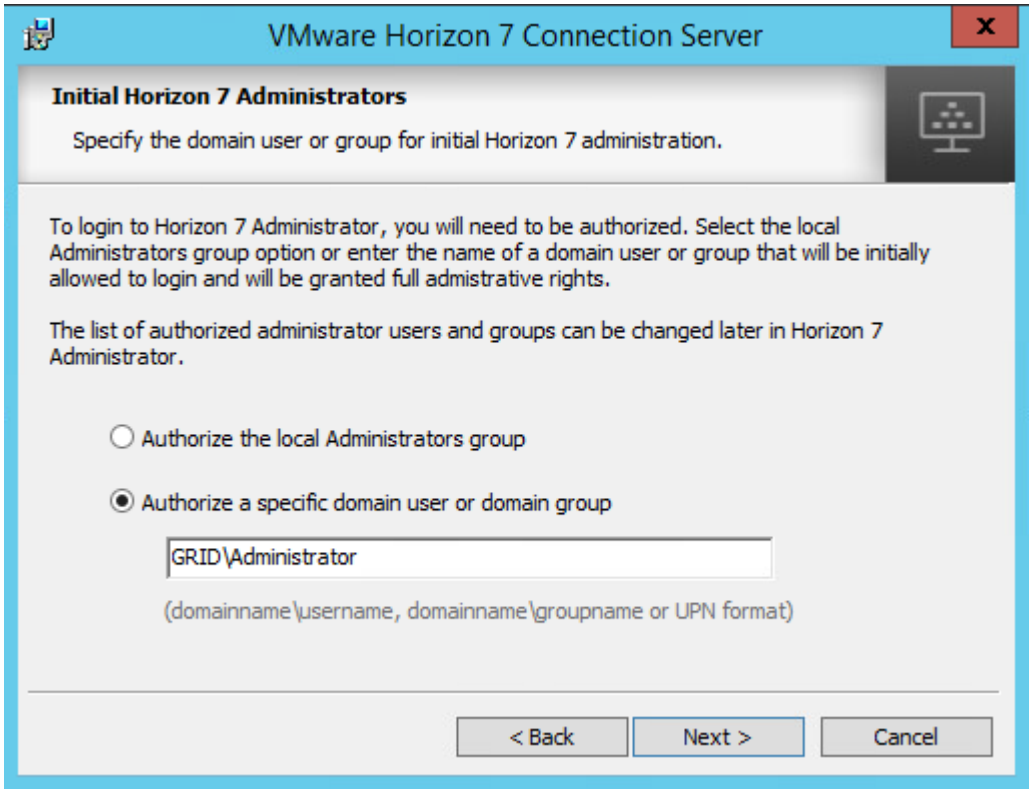


7. Enter a password in the Enter data recovery password screen and re-enter it in the Re-enter password screen; then click Next to continue. The Horizon installer displays the Firewall Configuration screen:

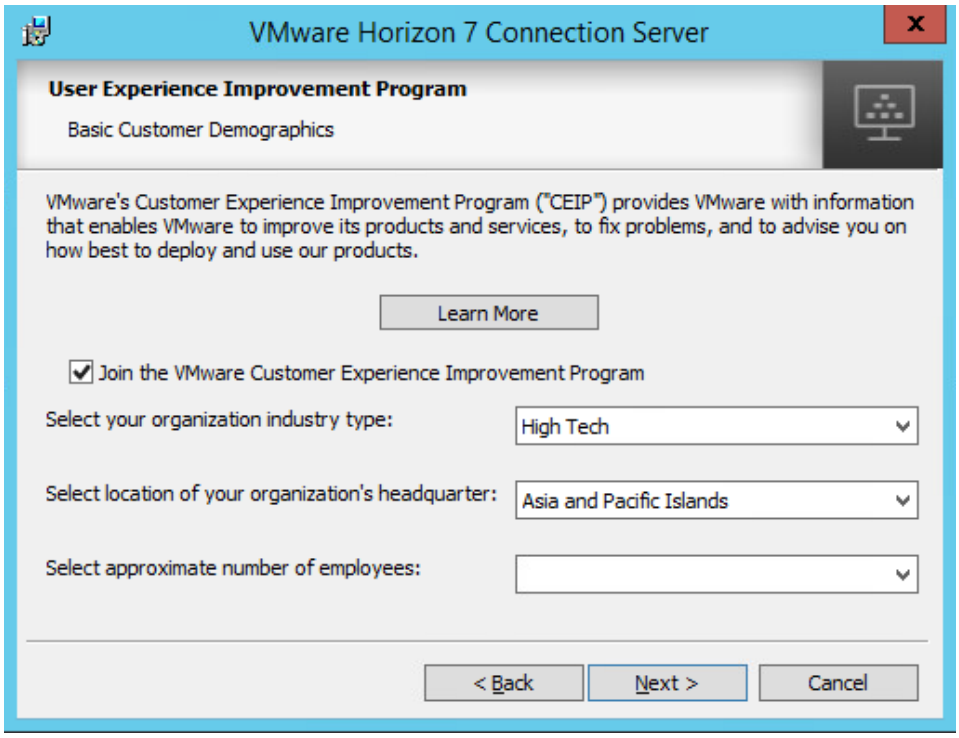


CAUTION: To prevent unauthorized access, choose a root password that contains at least eight (8) characters and contains a mix of lower case and capital letters, digits, and special characters.

8. Check the Configure Windows Firewall automatically checkbox.
9. Click Next to continue. The Horizon installer displays the initial Horizon Administrators screen.

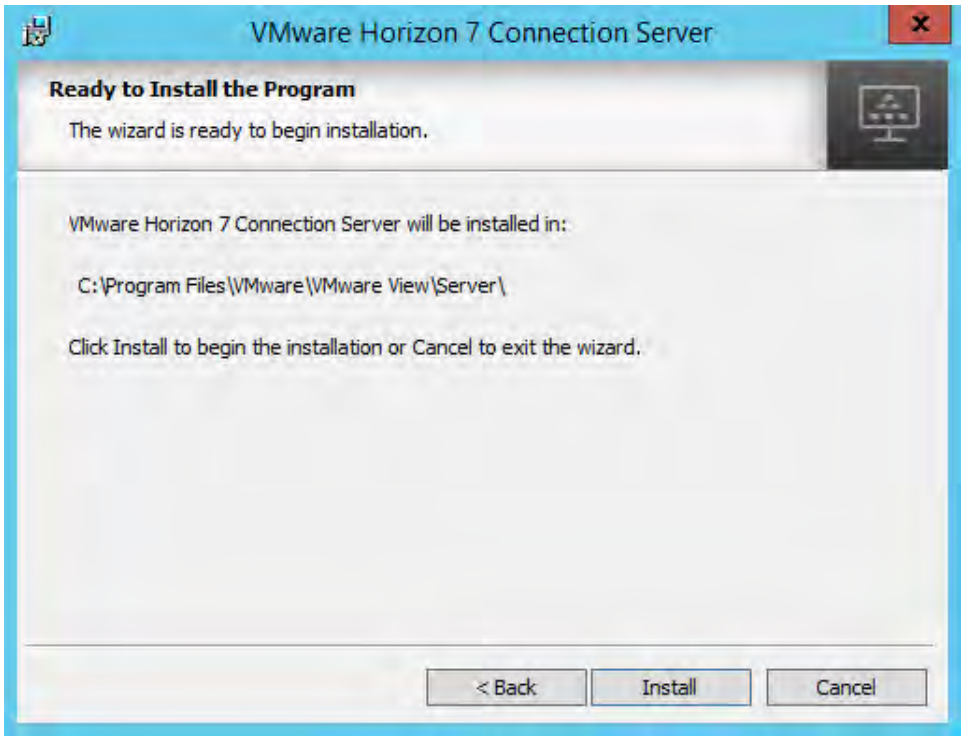


10. Select either the local Administrators group or a specific domain user or group, as appropriate, then click Next to continue.
11. The Horizon install displays the User Experience Improvement Program screen:



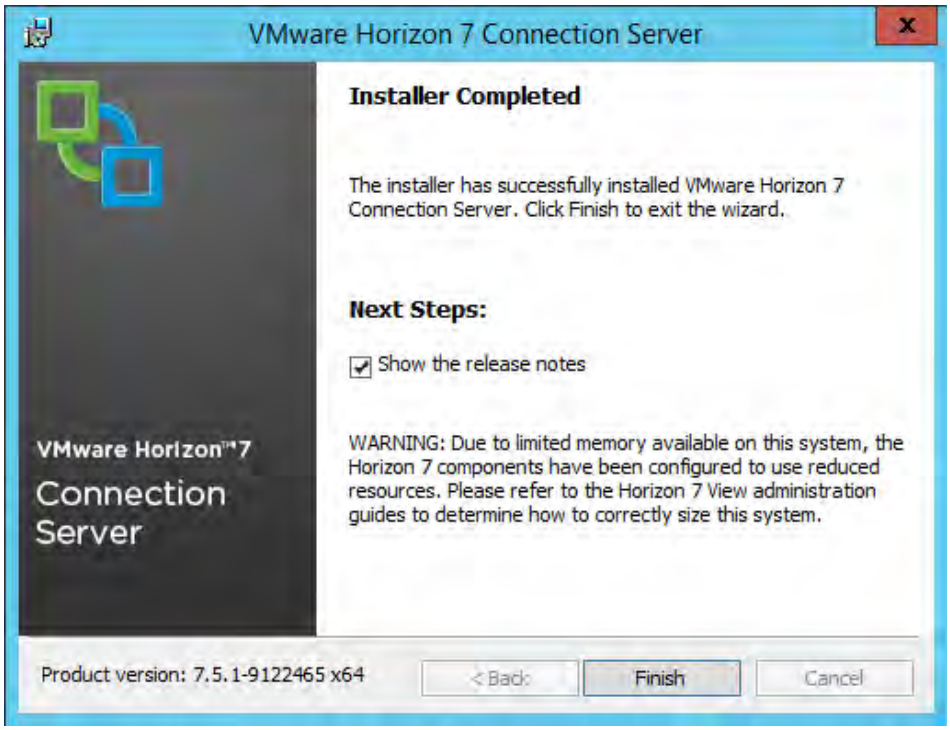
12. Check the Join the VMware Customer Experience Improvement Program checkbox if you want to join; clear the checkbox if you do not.

    Select appropriate values in the remaining fields, then click Next to continue. The Horizon install displays the Ready to Install the Program screen:



    Click Install to complete the installation.

13. The Horizon installer displays the Installer Completed screen.

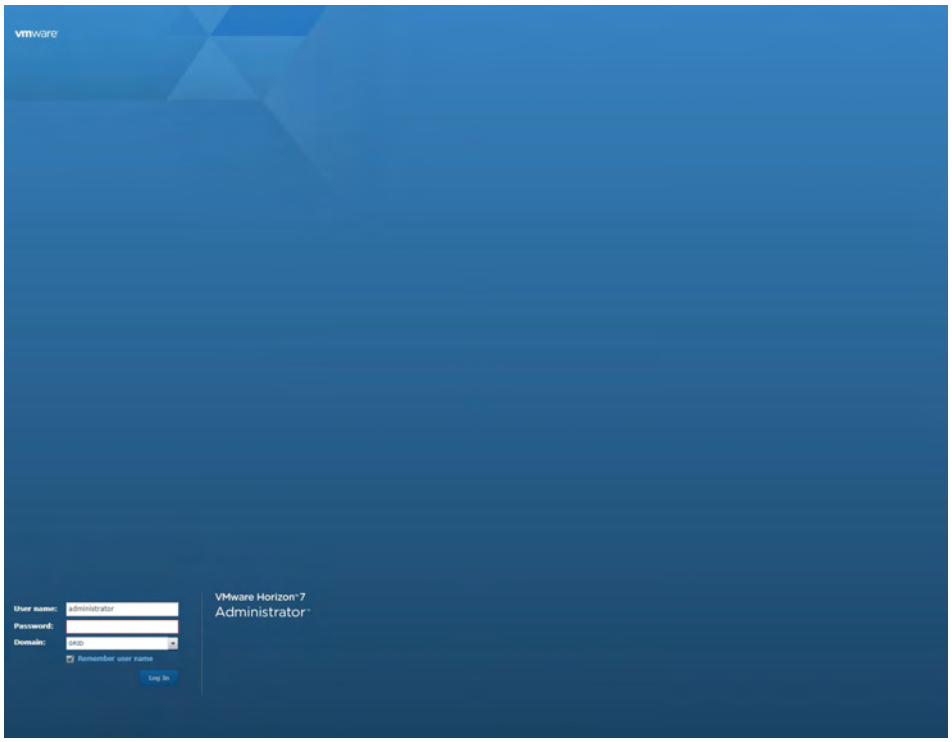


Click Finish to exit the installer.

14. Open the browser page at:

    `https://<host>/admin`

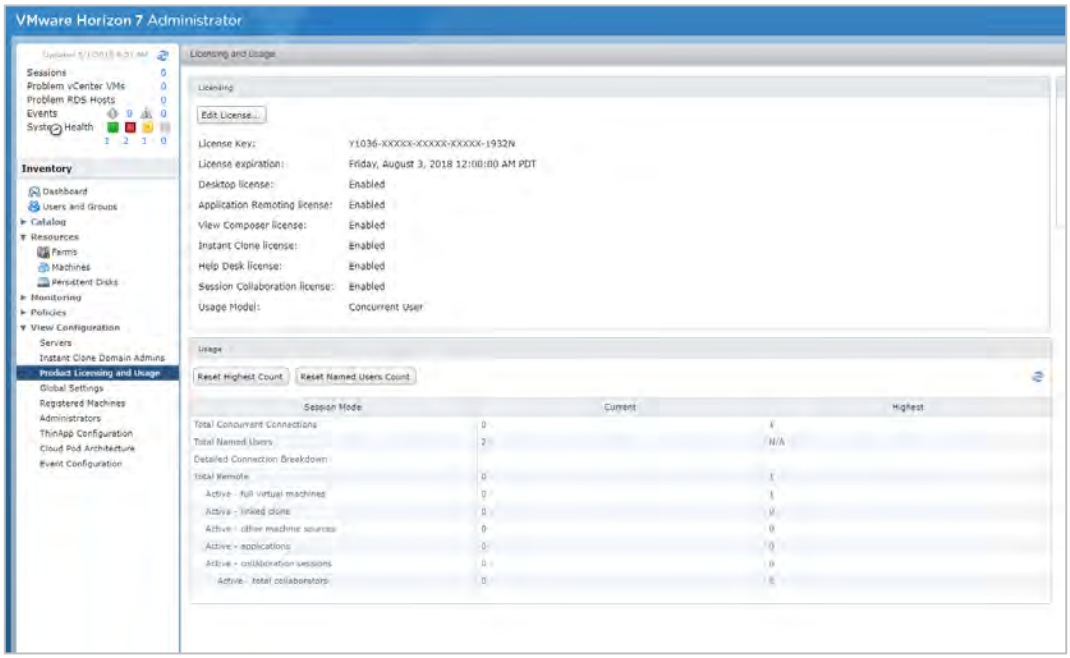
    Where `<host>` is the IP address or hostname of the server on which you installed Horizon:



4.1.1 Registering the Horizon License

To register the Horizon license:

1. Connect to the Horizon web console to open the Horizon Administrator:



2. Select Horizon Configuration > Product Licensing and Usage in the left-hand pane. Horizon displays the Licensing and Usage window.
3. Select Edit License. Horizon Administrator displays the Edit License dialog:

4. Enter your license key in the License serial number field.
5. Click OK to decode and apply the license.

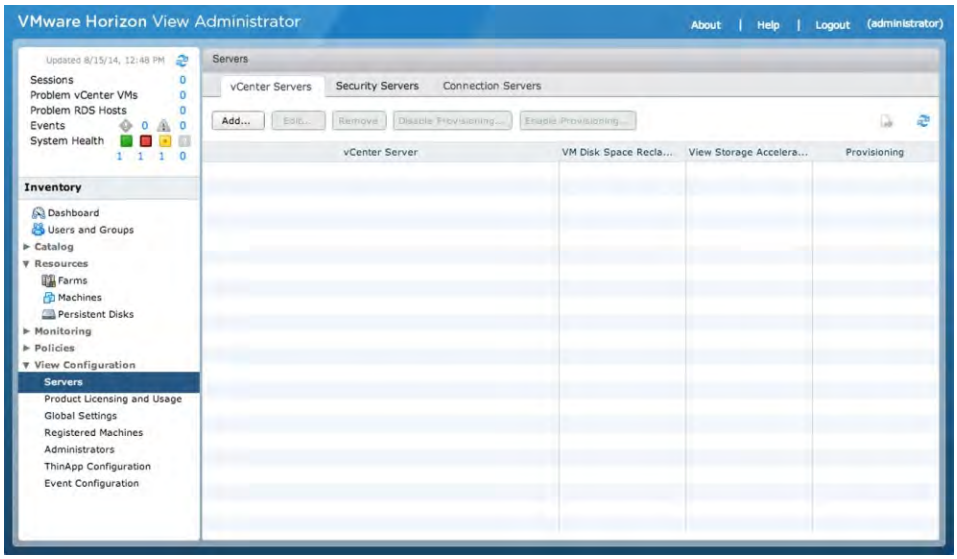
> You must obtain the license from the VMware Beta site from which you downloaded the software.

6. Verify that the license is enabled by looking for the “Enabled” messages in the “Licensing” part of the Licensing and Usage window.

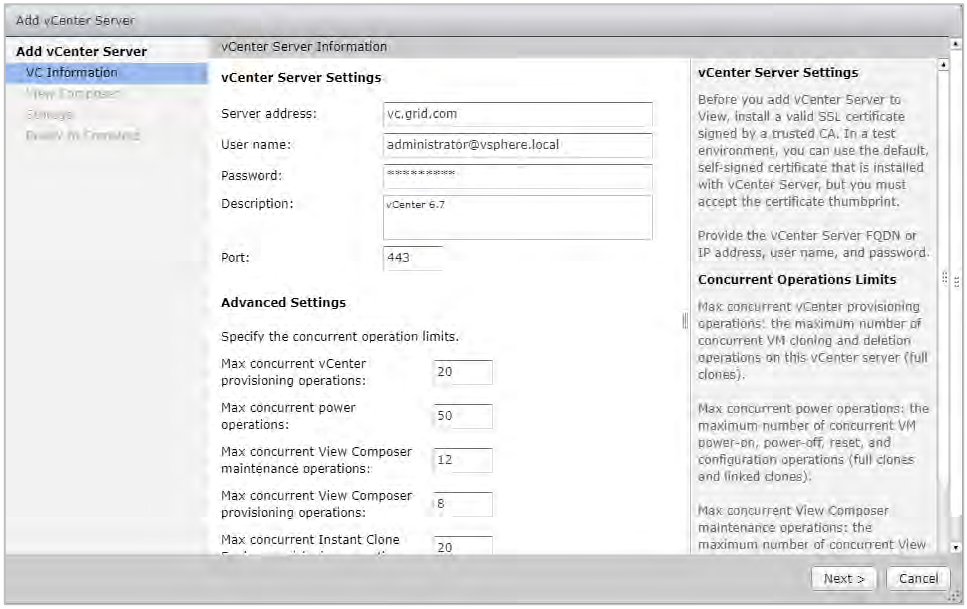
4.2 Registering vCenter Server

To register vCenter server:

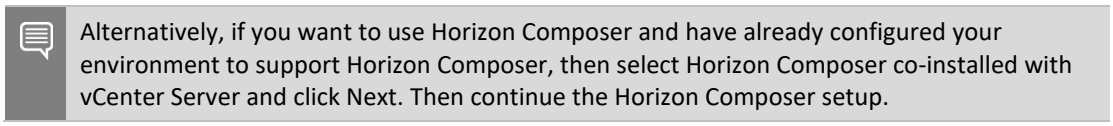
1. In the left-hand pane of the Horizon Administrator window, Select Horizon Configuration > Servers. Horizon Administrator displays the Servers window:



2. Click Add... to register a new vCenter server. Horizon Administrator displays the Add vCenter Server screen with the VC Information tab selected.



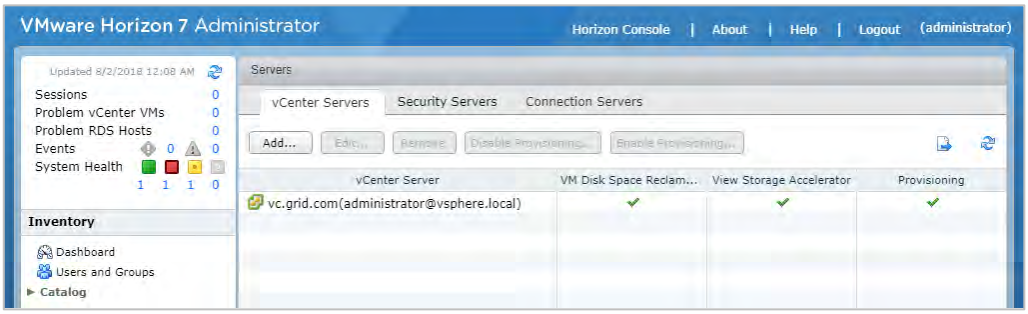
3. Enter the vCenter server address and user credentials (not the local desktop user credentials) in the appropriate fields.
4. Click Next to display the Horizon Composer Settings tab.
5. Click Do not use Horizon Composer, then click Next to display the Storage tab.



Alternatively, if you want to use Horizon Composer and have already configured your environment to support Horizon Composer, then select Horizon Composer co-installed with vCenter Server and click Next. Then continue the Horizon Composer setup.

6. Accept the default storage settings.
7. Click Next to display the Ready to Complete window.
8. Click Finish.

The vCenter server is now added, and displays in the Horizon configuration:



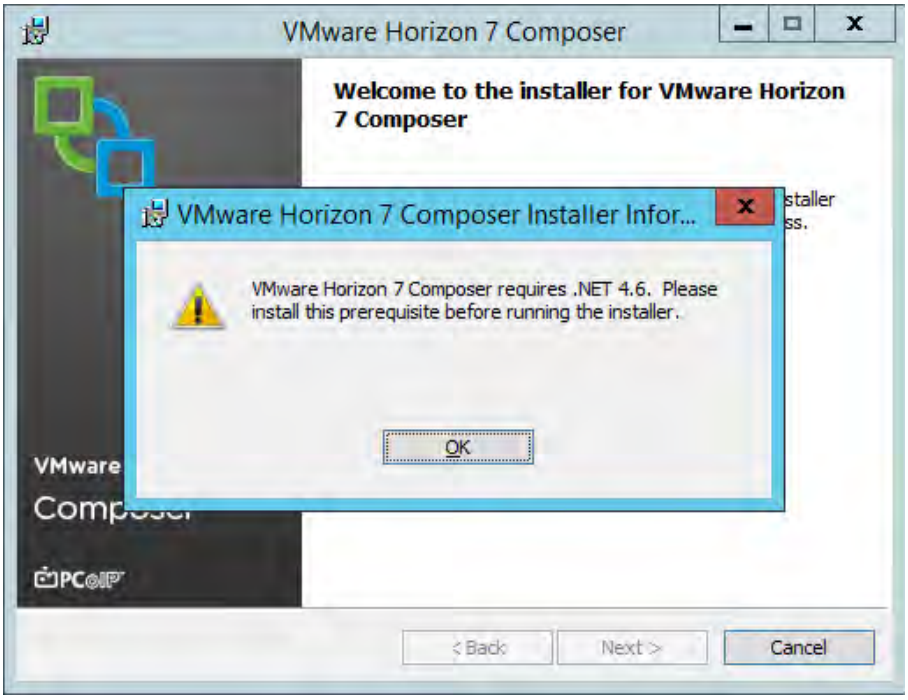
4.3 Installing Horizon Composer Server

The server on which you are installing Horizon must meet the requirements listed in General Prerequisites. You also need access to a SQL server. For the installation that was used as a basis for these instructions, NVIDIA installed SQL Express 2012 on a separate Microsoft Windows Server 2012R2 guest on the management cluster. As of this deployment guide's publication date, the current version of the Horizon Composer Server is version 7.5.1, and the filename of the installer is `VMware-viewcomposer-7.5.1-8971623.exe`.

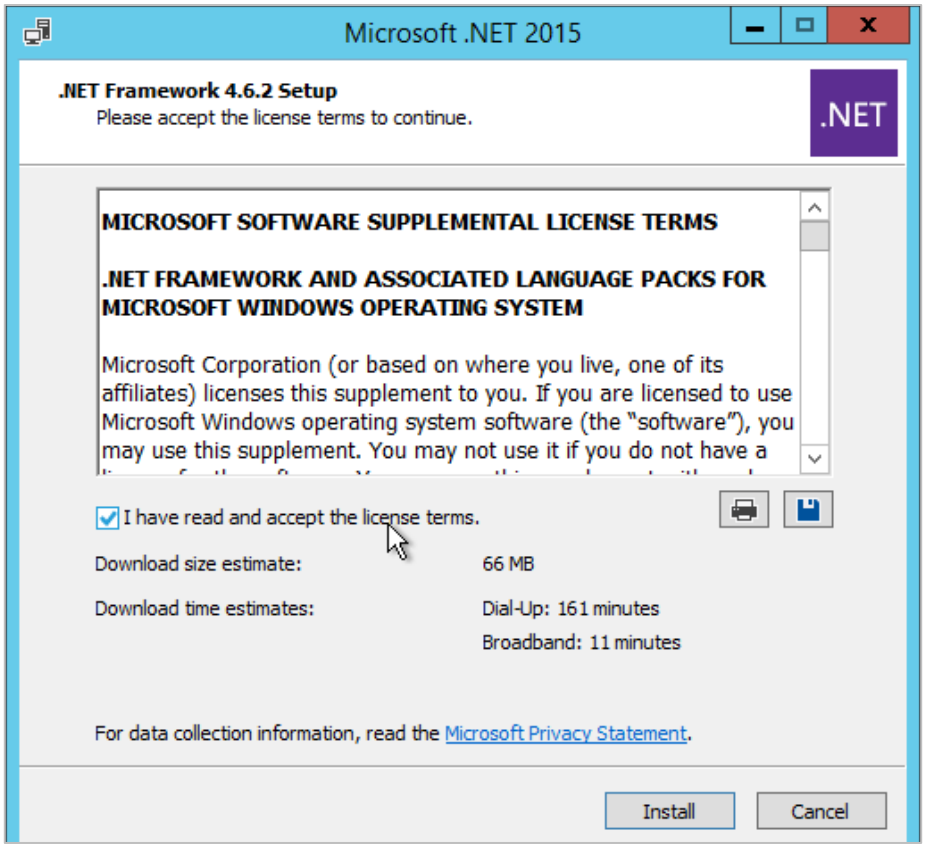
The purpose of the Composer Server is to create linked clones. The requirements are:

- A Microsoft Windows Server 2012R2 for Horizon Horizon Composer
- A Microsoft Windows Server 2012R2 for SQL 2012

Before install Horizon Composer 7.5 please make sure to install .NET 4.6 framework.

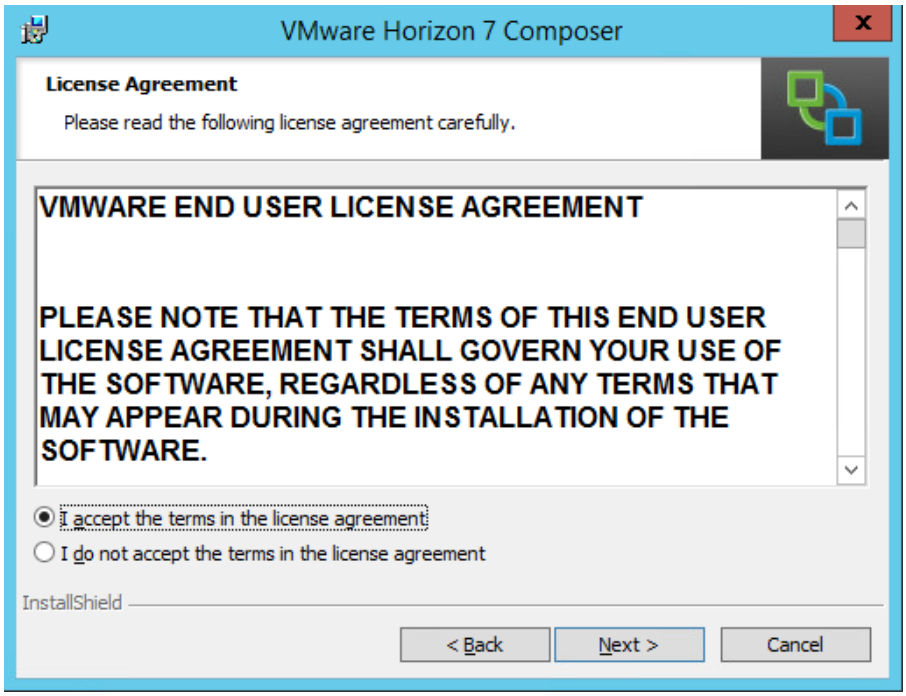


1. The message box notifies you that Horizon 7 Composer requires .NET 4.6 to run. Click OK to close the message box and continue. The Horizon Composer installer displays the .NET Framework Setup screen.

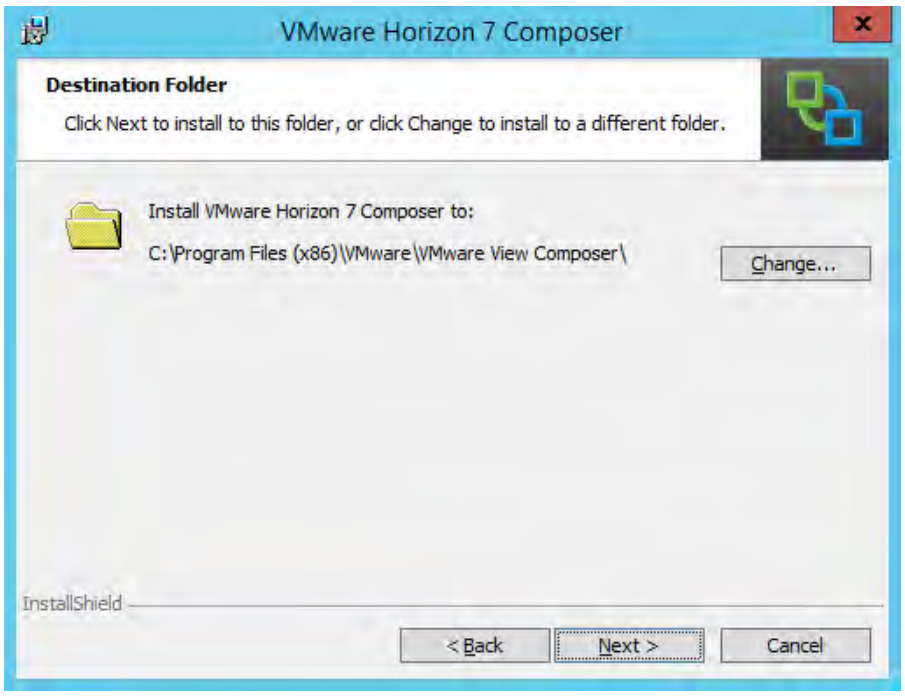


2. Check the “I have read…” checkbox, then click Install. The VMware Horizon Composer installer starts its installation wizard:

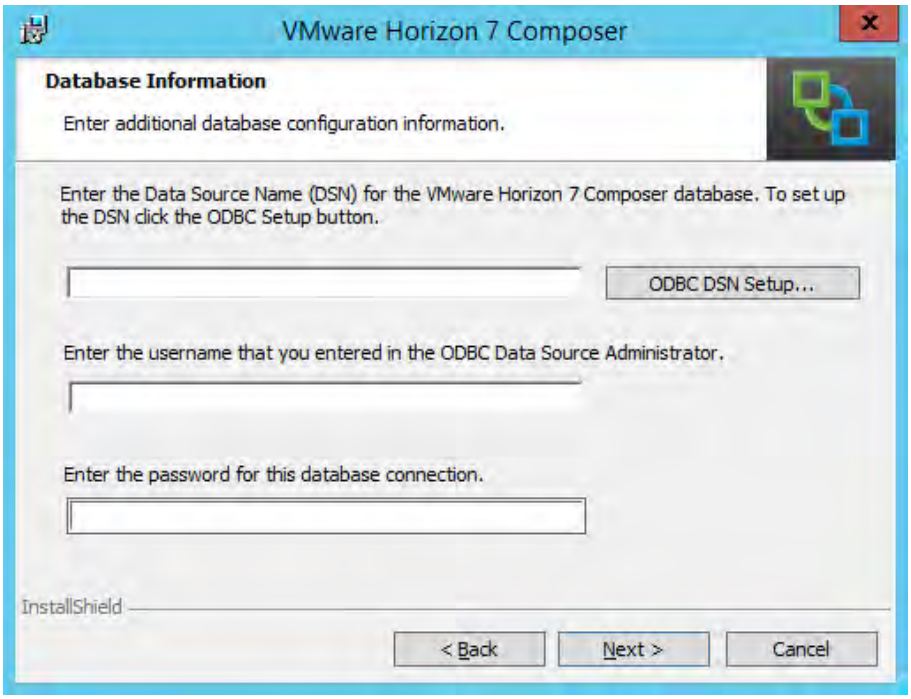
3. Click Next. The wizard displays the License Agreement screen.



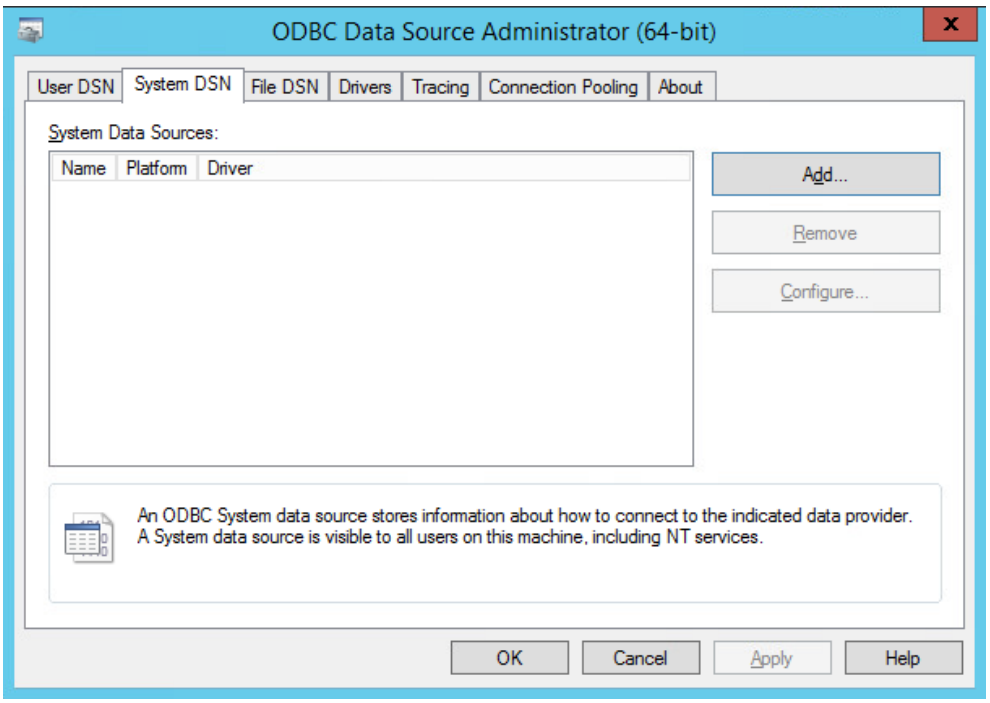
4. Select the "I accept…" radio button, then click Next to continue. The wizard displays the Destination Folder screen.



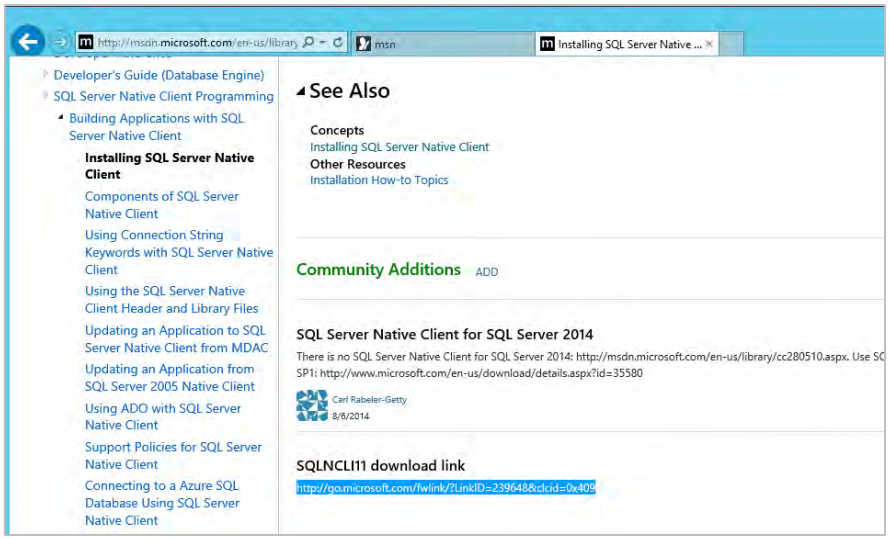
5. Choose an install location, then click Next. The wizard displays the Database Information screen:



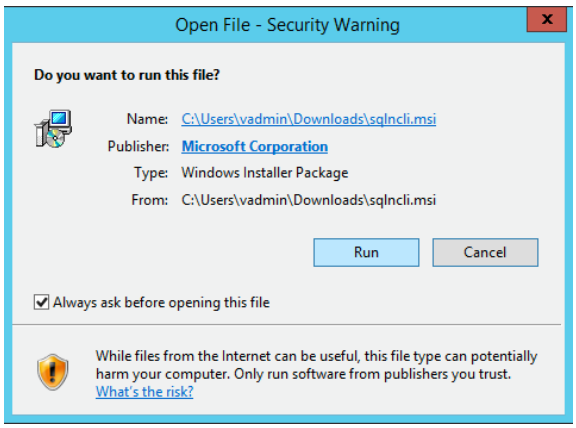
6. To set up a connection to a SQL server, click the ODBC DSN Setup… button. The VMware Horizon Composer wizard opens the ODBC Data Source Administrator installation wizard in a new window:



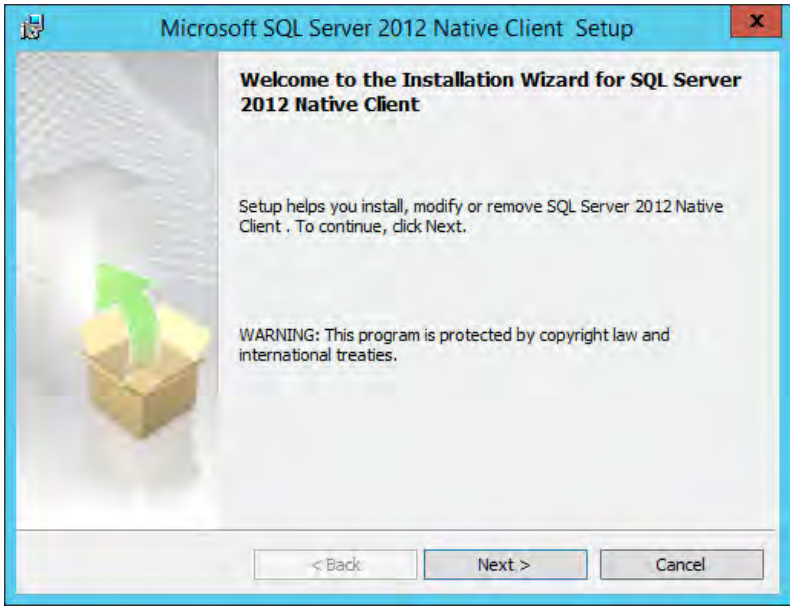
7. Click the Add… button. You may need to download and install the SQL Server Native Client.



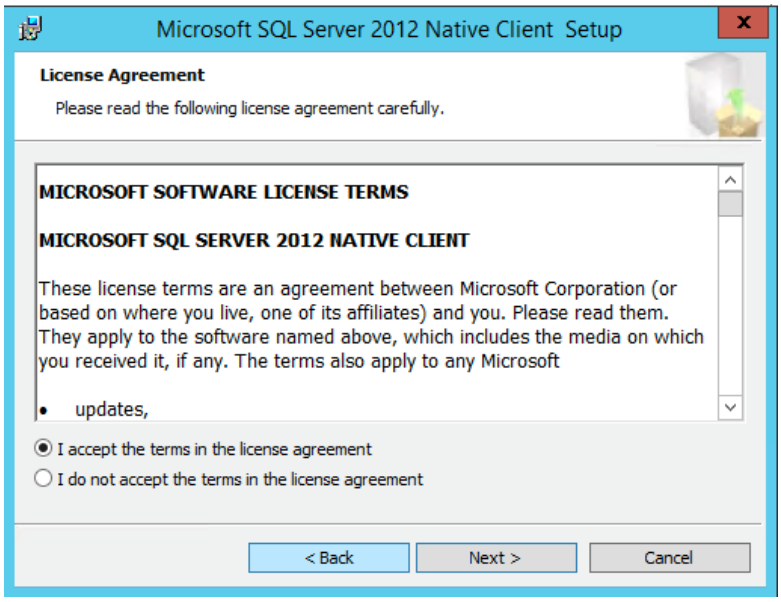
8. Accept any security warnings after you download and begin the install.



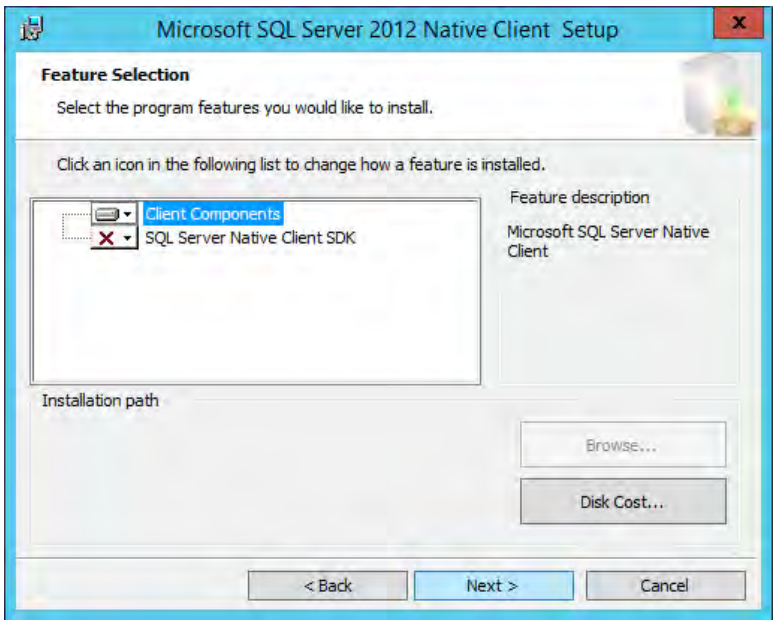
9. The ODBC Data Source Administrator wizard starts the SQL Server Native Client installation wizard.



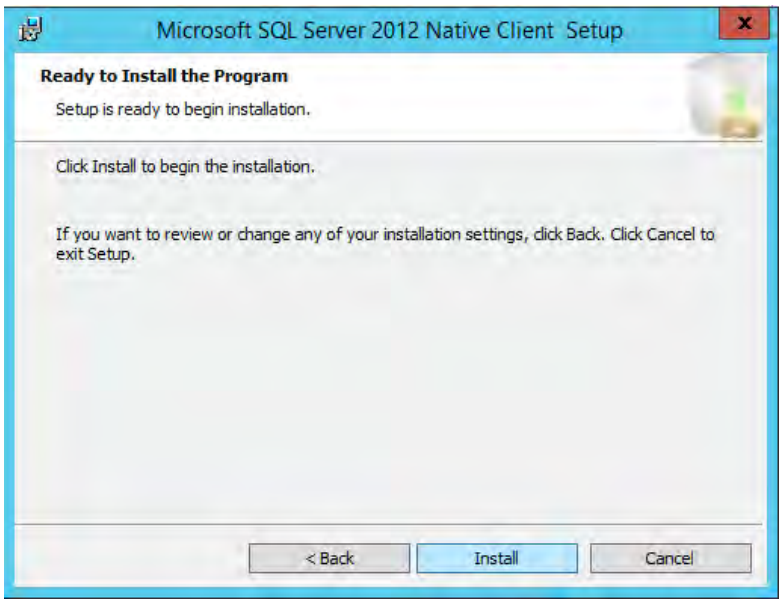
10. Follow the wizard's guidance to continue the installation. Accept the defaults for all options.
11. When you are done selecting features the wizard displays the License Agreement screen.



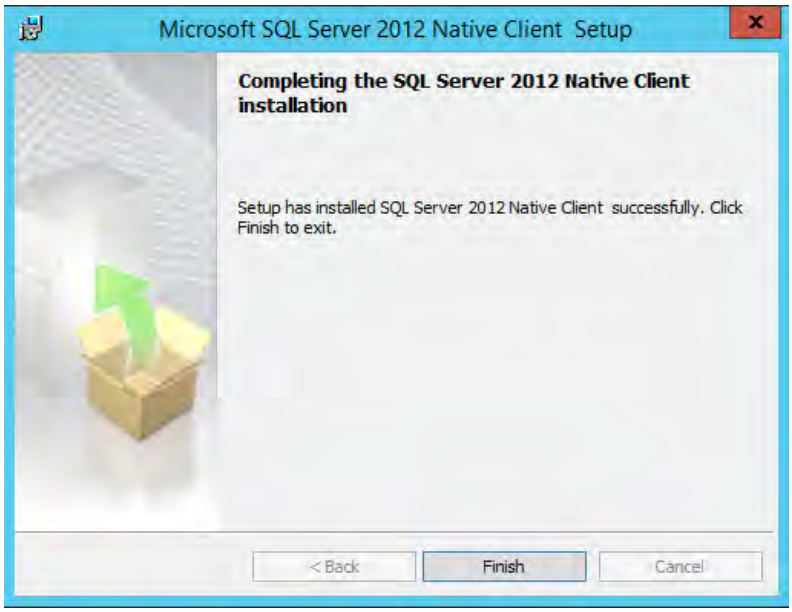
12. Click the "I accept…" radio button, then click Next. The wizard displays the Feature Selection screen.



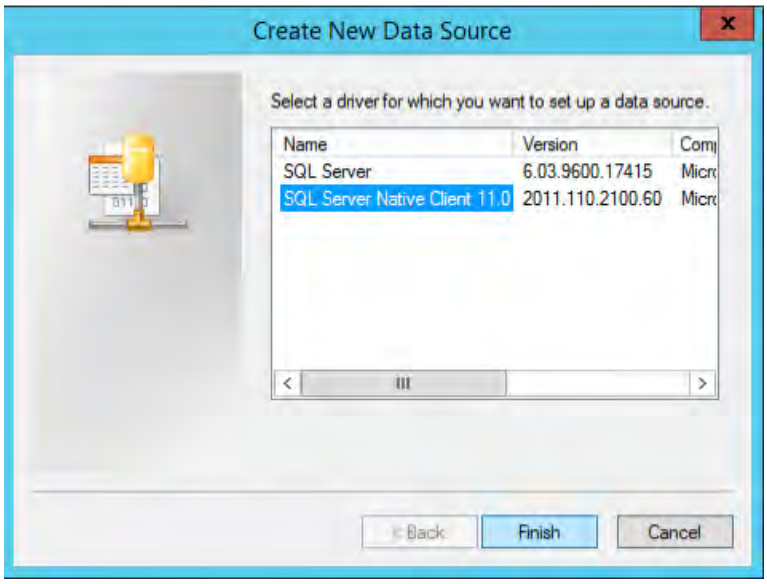
13. Click Next, accepting the defaults. The wizard displays the Ready to Install the Program screen.



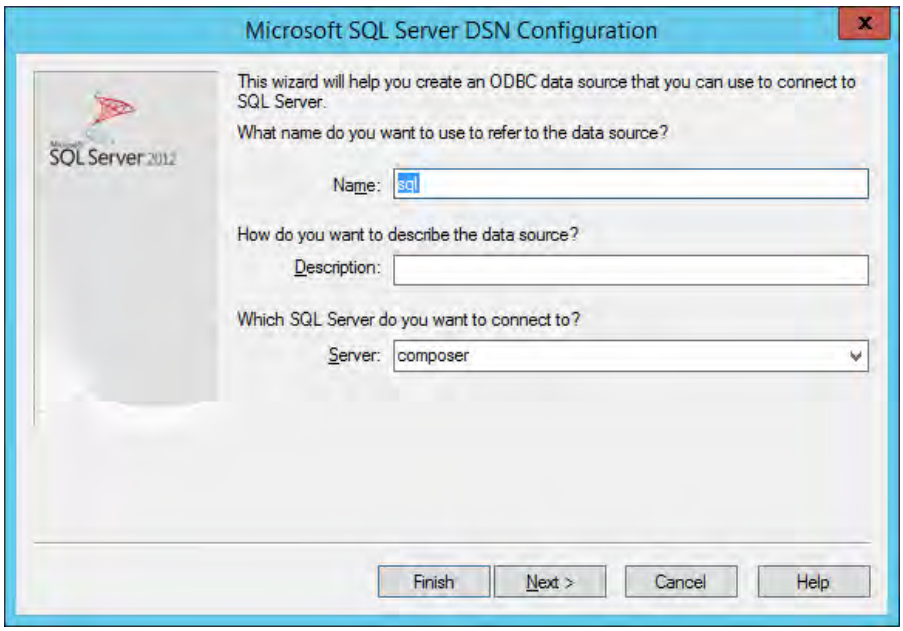
14. Click Install. The wizard displays the Completing the SQL Server 2012 Native Client Installation screen.



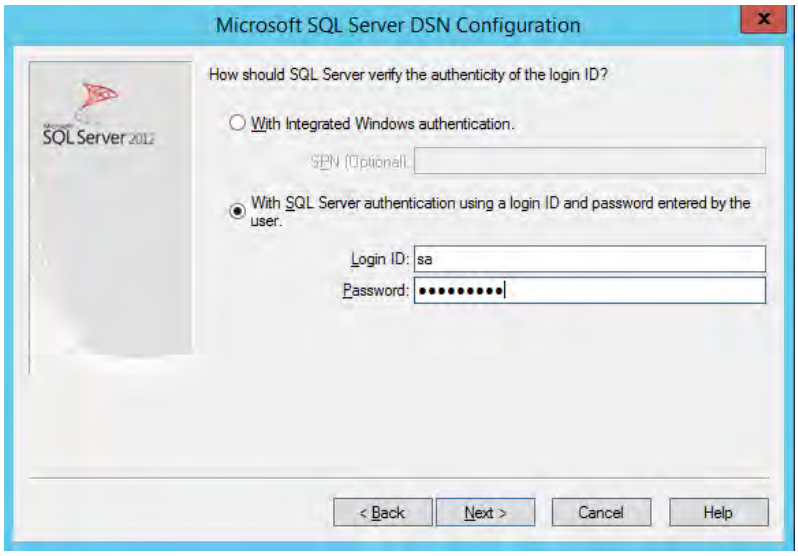
15. When the install is complete, the wizard enables the screen's Finish button. Click Finish to close the SQL Server Native Client wizard and return to the ODBC Data Source Administrator installation wizard. That wizard now displays the Create New Data Source screen.



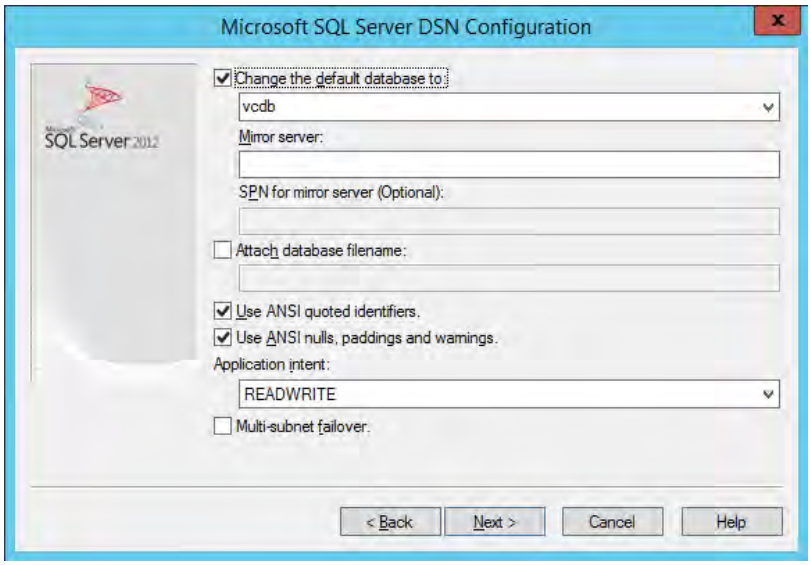
16. Select the SQL server you want to connect. Click Next to continue. The wizard displays a screen that prompts you to name and describe a new data source:



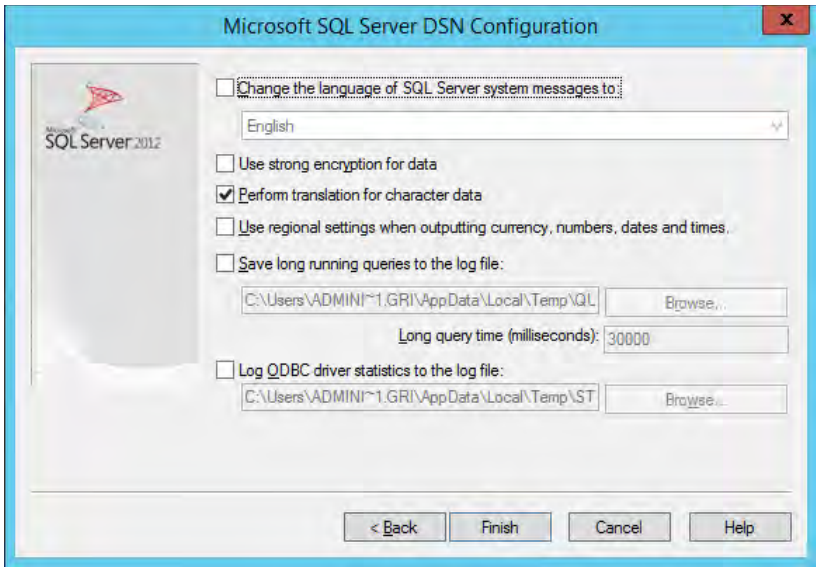
17. Enter a name and description for the new data source and select the appropriate server, then click Next. The wizard prompts you for logon credentials:



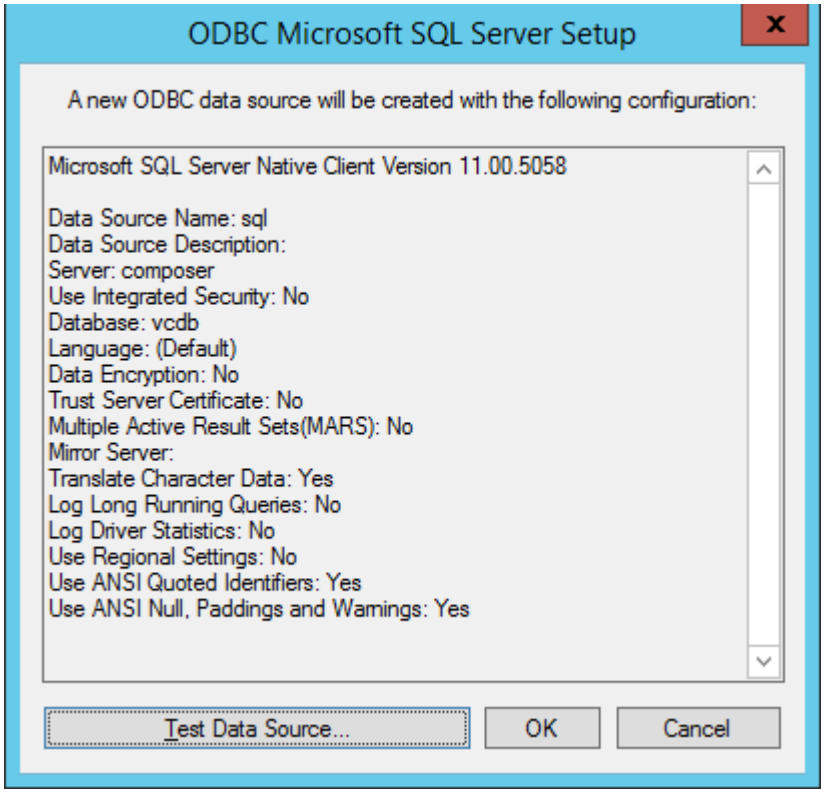
18. Enter the credentials for the SQL account (not the Windows user account). For a remote SQL database, only a SQL accounts will work. The SQL account must be the database's db_owner. Then click Next. The wizard prompts you to specify properties for the database:



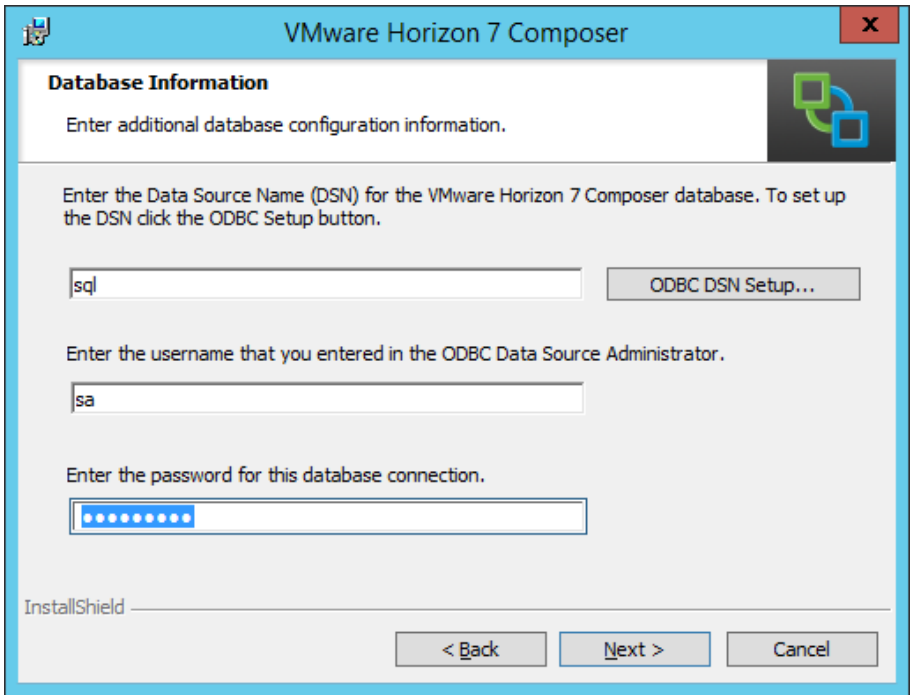
19. Check the Change the default name checkbox and name your database for the Horizon Composer. Click Next to continue. The wizard displays another screen that prompts you to specify database properties.



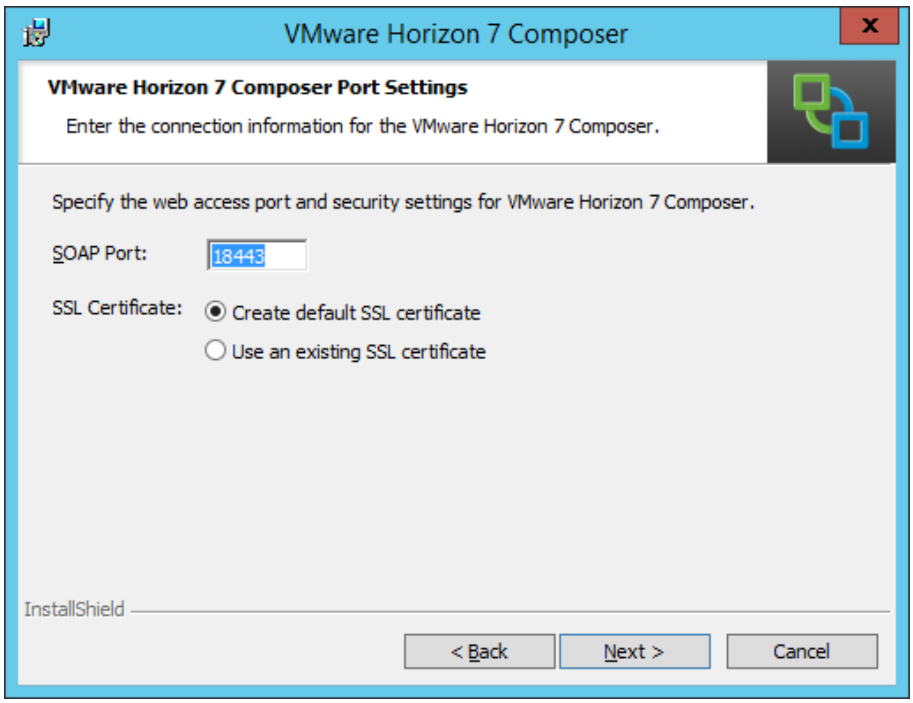
20. Click Finish to continue. The wizard displays a summary of properties you have specified for the new ODBC data source.



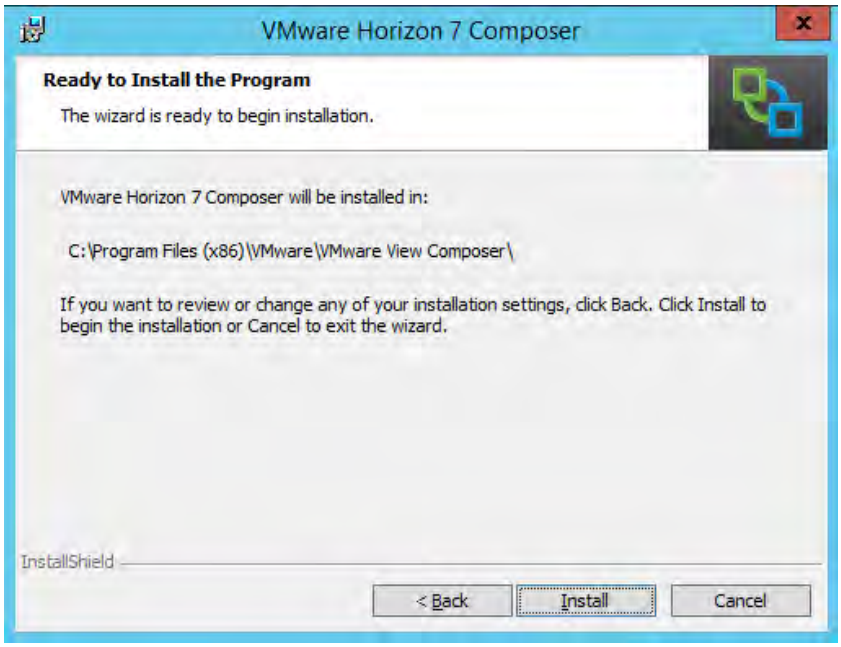
21. Review your configuration, then click Test Data Source… . If the test is successful, the wizard displays a “successful configuration” message. If not, go back and correct your settings.
22. Return to the VMware Horizon Composer wizard:



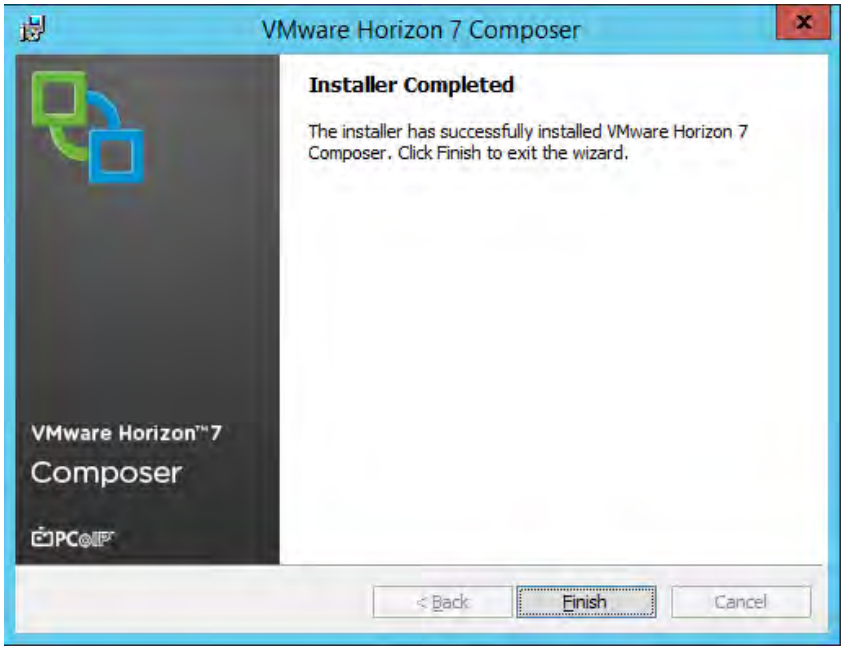
23. Enter the database name, SQL administrator account name, and password to complete installation. Click Next to continue. The wizard displays the Composer Port Settings screen:



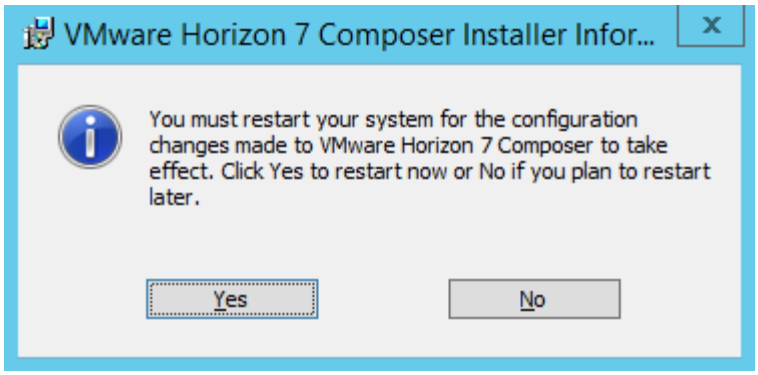
24. Select a web access port number. (Use the default if possible.) Click Next to continue. The wizard displays the Ready to Install the Program screen.



25. Click Install to continue. The wizard displays the Installer Completed screen.



26. When the installation is complete, the wizard enables the Finish button. Click Finish to close the wizard.
27. You must reboot the system to make the installed software and data source available. The wizard displays a “You must restart your system…” message box. Click Yes to and reboot the server.

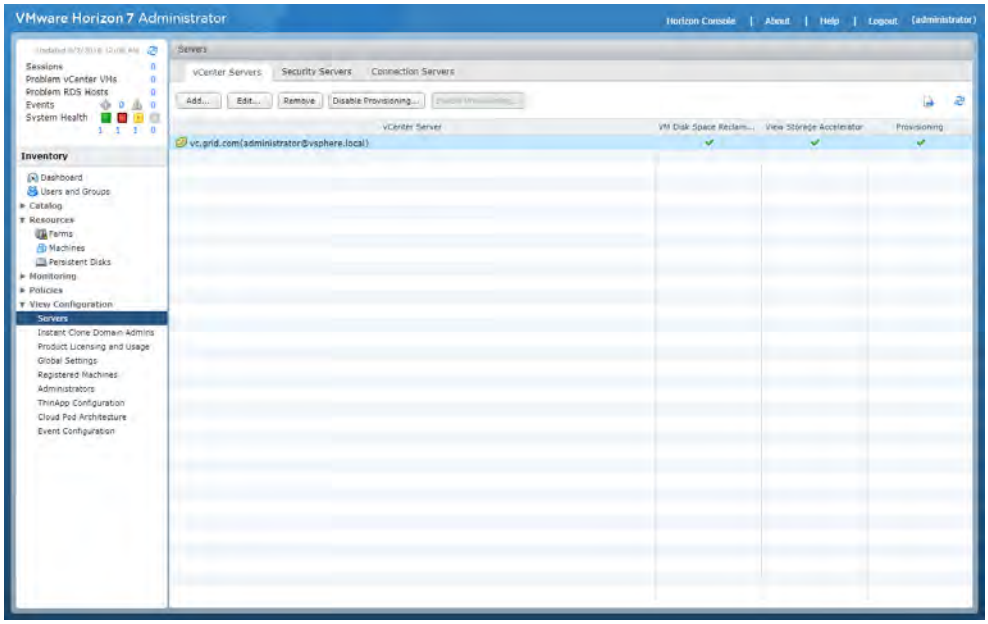


28. When the server reboot is complete, proceed to register Composer to Horizon. (It may take a few minutes for all of the services to start up, so be patient.)

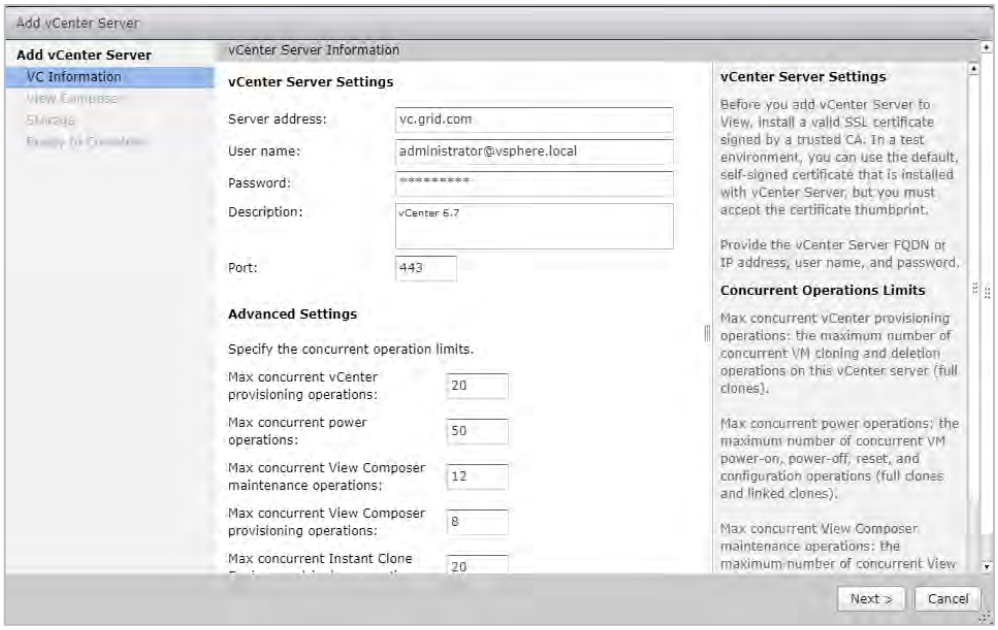
4.4 Registering Horizon Composer Server

To register Horizon Composer Server:

1. Log in to Horizon Server. Horizon Server displays the VMware Horizon Administrator window:



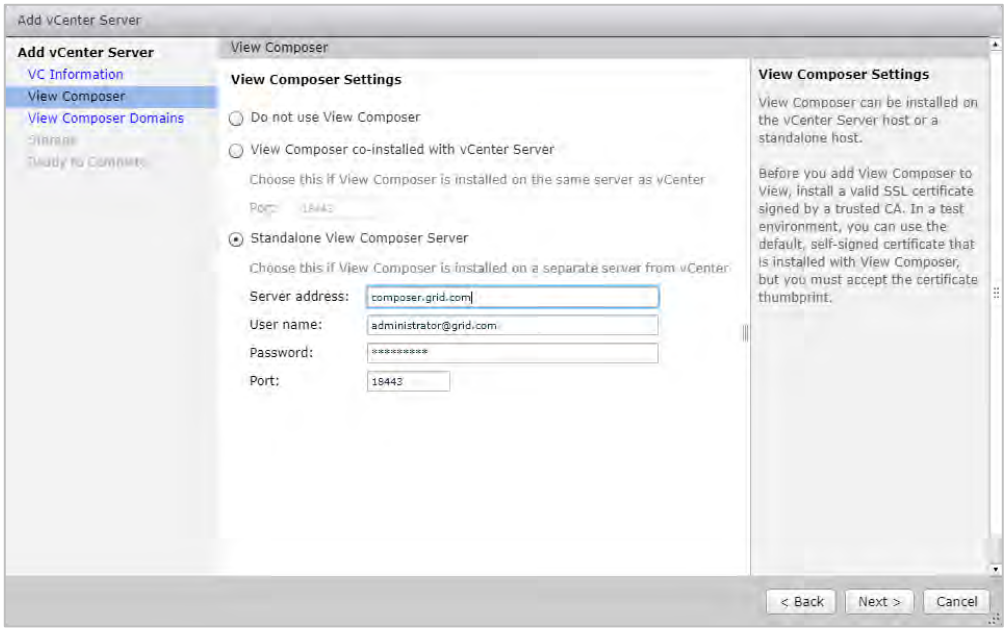
2. In the left-hand pane, under Horizon Configuration, select Servers. In the right-hand pane, select Edit to add a Horizon Composer to an existing vCenter Server, or Add to add a new vCenter Server with a Horizon Composer. Horizon Server opens the first window of the Register Horizon Composer wizard.



3. Enter the vCenter Server address, the administrator account username, and its password. Customize the configuration with the advanced settings if necessary; then click Next. The wizard opens the Invalid Certificate Detected dialog:

4. Click Horizon Certification to display the invalid certificate.

5. Click Accept to continue. The wizard displays the Horizon Composer Settings screen.



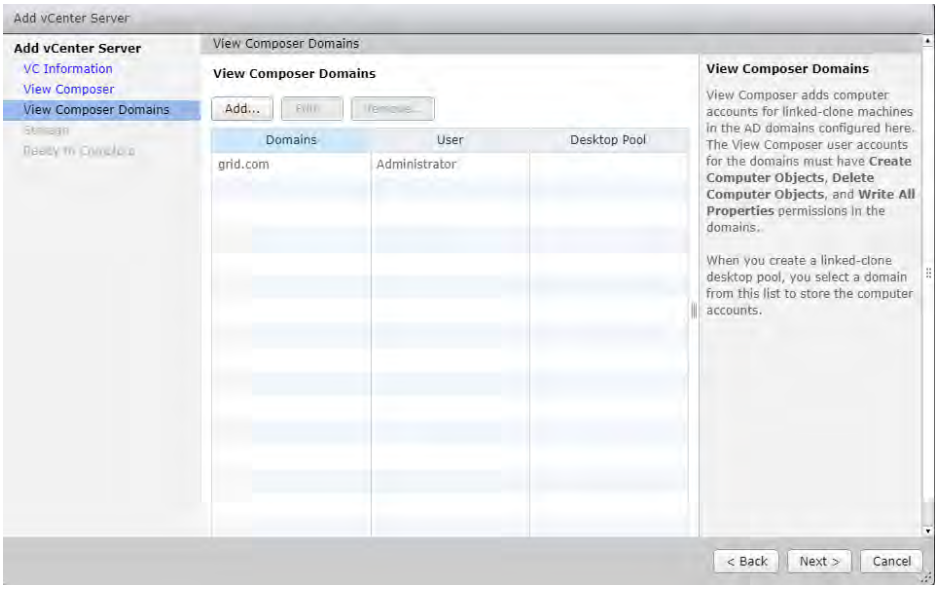
6. If you are using a standalone Horizon Composer Server, enter the server address, user account name and password.

   If you are co-installing Horizon Composer Server with vCenter Server, select the Horizon Composer co-installed option.

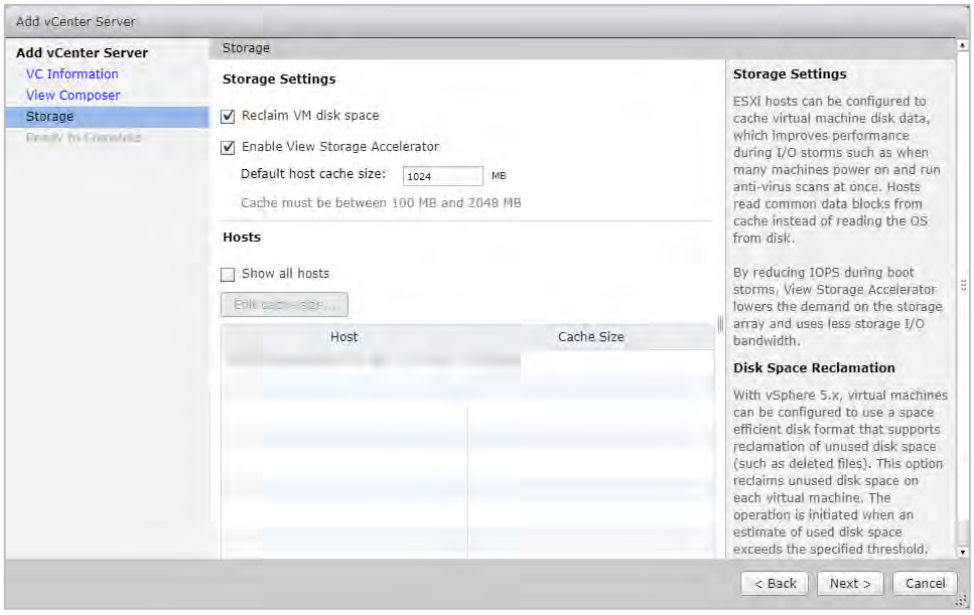
   Click Next to continue. The wizard opens the Invalid Certificate Detected dialog:

7. Click View Certificate. The wizard displays the invalid certificate.

8. Click Accept to accept the certificate. The wizard displays the Horizon Composer Domains screen.



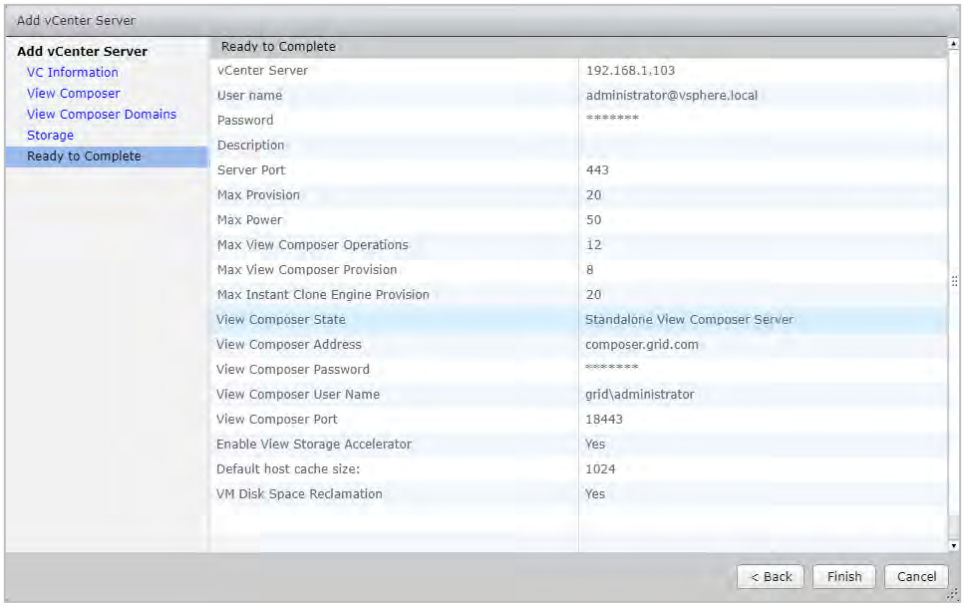
9. View the domains you have created, then click Next to continue.
10. Select View Storage Settings. The wizard displays the Storage Settings screen.



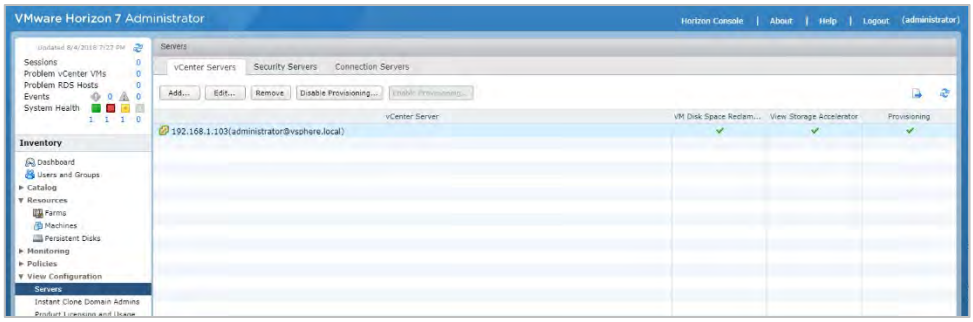
11. Check the Reclaim VM disk space and Enable Horizon Storage Accelerator checkboxes and set the Default host cache size.

    You can customize the cache sizes of each ESXi host at this point if necessary, by selecting Edit cache size.

    When you are done, click Next to continue. The wizard displays the Ready to Complete screen, which summarizes the configuration you have defined.



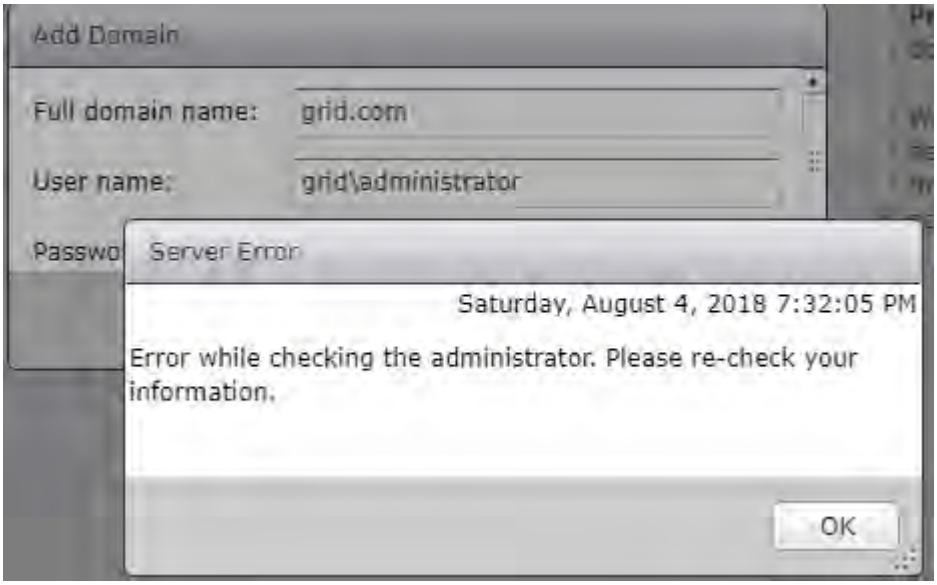
12. Review your configuration, then click Finish. This closes Horizon Composer registration wizard. The focus returns to the VMware Horizon Administrator window:



13. Verify that your vCenter Server with Horizon Composer is ready to go.

4.4.1 "Error while checking the administrator" message

Horizon Composer Server versions 7.5.0 and 7.5.1 have a bug which may cause this error to be reported while you are using the wizard to register a co-installed vCenter Server and Horizon Composer Server.



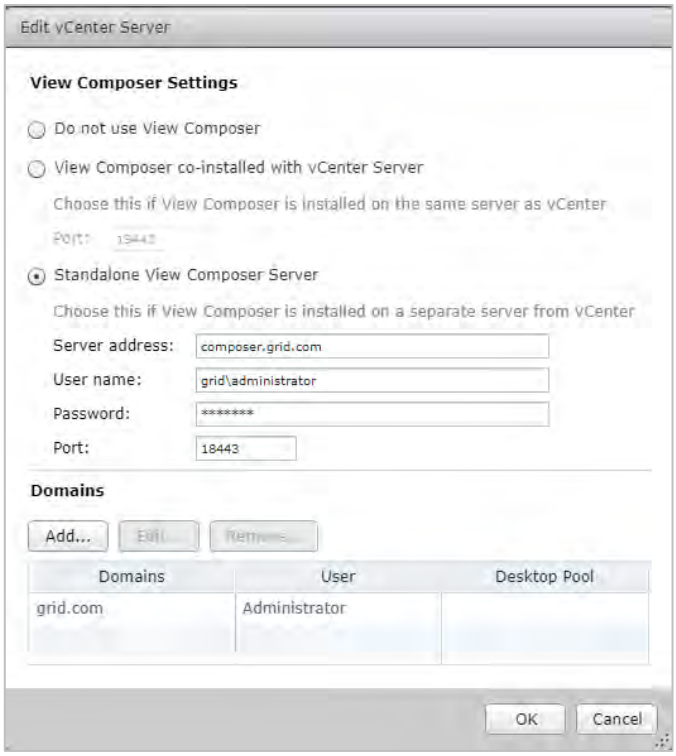
As of this deployment guide's publication date, this bug is not documented in the VMware release notes or knowledge base.

To work around the bug:

1. Register vCenter Server first, then click OK to finish:

2. Register Horizon Composer Server only, then click OK to finish:

3. Edit the Horizon Composer settings again to add Horizon Composer domains separately:


Edit vCenter Server
View Composer Settings
Do not use View Composer
View Composer co-installed with vCenter Server
Choose this if View Composer is installed on the same server as vCenter
Port: 18443
Standalone View Composer Server
Choose this if View Composer is installed on a separate server from vCenter
Server address: composer.grid.com
User name: grid\administrator
Password: *******
Port: 18443
Domains
Add...
Edit...
Remove...
Domains
User
Desktop Pool
grid.com
Administrator
OK
Cancel

Chapter 5. NVIDIA vGPU Manager VIB Installation

This chapter illustrates how the install the NVIDIA vGPU Manager VIB. It provides a step-by-step guide which includes downloading and preparing the VIB file for installation on the host.

For demonstration purposes, these steps use the VMWare vSphere web interface to upload the VIB to the server host.

The following paragraphs describe how to:

- Install the VIB
- Update the VIB
- Uninstall the VIB

5.1 Preparing the .vib File for Install

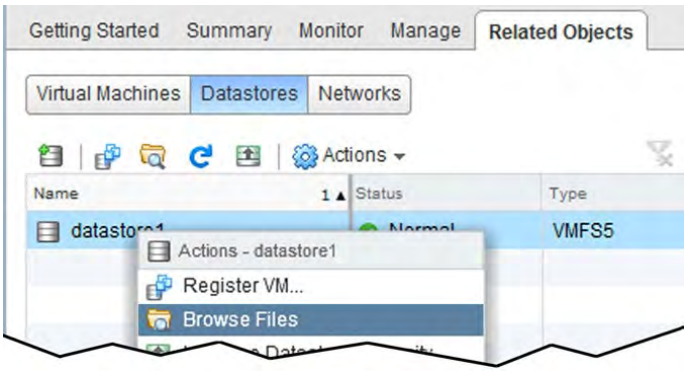
Before you perform the installation procedures, download the archive containing the VIB file and extract its contents to a folder. The file with that ends with `.vib` is the file that you must copy to the host datastore for installation.

5.1.1 Uploading the .vib File in vSphere Web Client

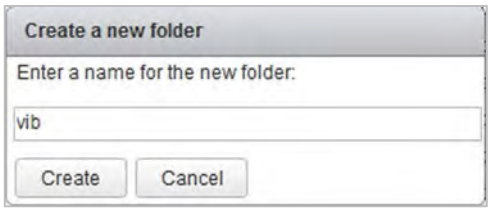
To upload the `.vib` file to the datastore using vSphere Web Client:

1. Click the Related Objects tab for the desired server.
2. Select Datastores. vSphere displays the names of the available datastores.
3. Either right click the appropriate datastore and select Browse Files or click the icon in the toolbar.

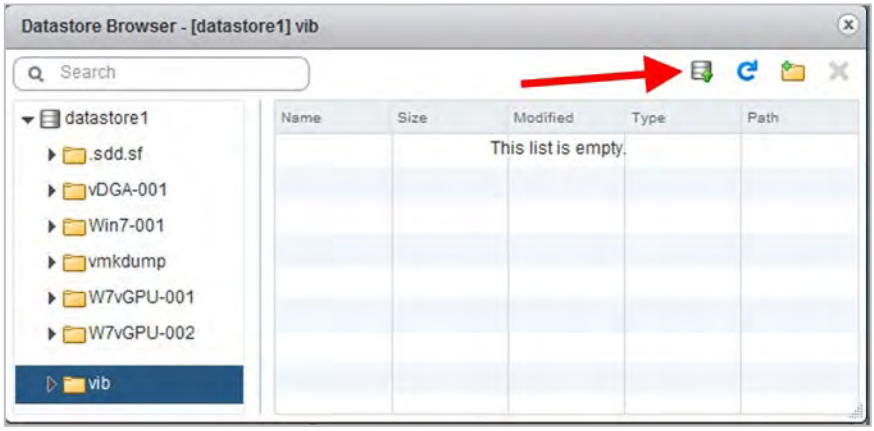
vSphere displays the Datastore Browser window:



4. Click the New Folder icon. vSphere displays the Create a new folder dialog:



5. Enter `vib` as the new folder's name, then click Create. vSphere creates the folder and closes the window.
6. Select the `vib` folder in the Datastore Browser window.
7. Click the Upload icon, indicated by the red arrow in this figure.



vSphere displays the Client Integration Access Control window.

8. Select Allow. vSphere uploads the `.vib` file to the datastore on the host.

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

5.2 Installing vGPU Manager with the .vib File

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

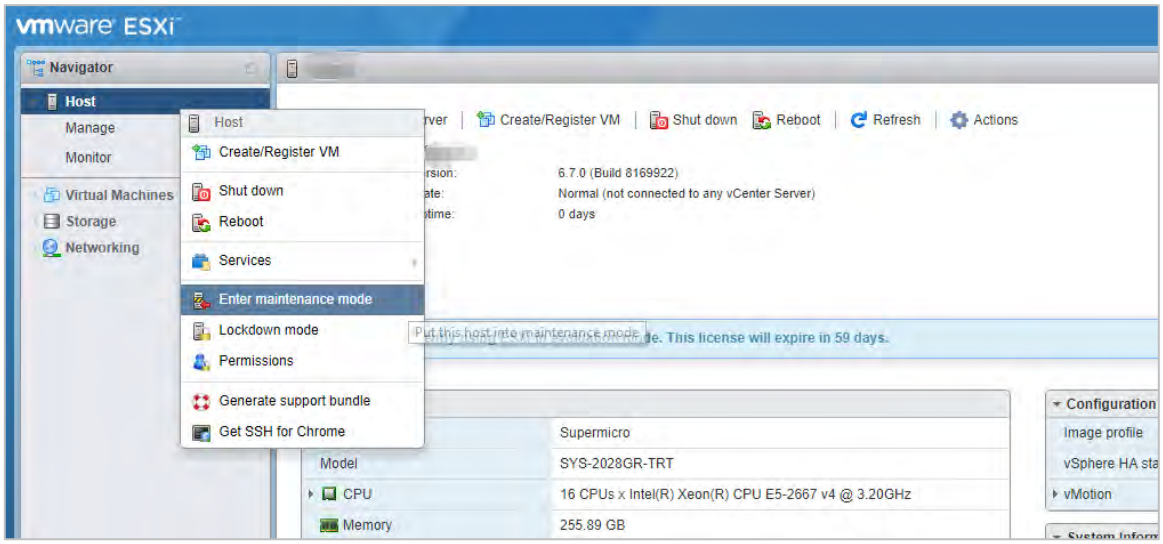
- As a `.vib` file, which must be copied to the ESXi host and then installed
- As an offline bundle that you can import manually as explained in Import Patches Manually in the VMware vSphere documentation

> CAUTION: Prior to vGPU release 11, 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. See Virtual GPU Software for VMware vSphere Release Notes for further details.

To install the vGPU Manager `.vib` file you must access the ESXi host via the ESXi Shell or SSH. See the VMware documentation for instructions 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. See the VMware documentation on how to place an ESXi host in Maintenance Mode.

1. To place the host in Maintenance Mode, right click the Host entry in the ESXi window's left-hand panel, then select 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 in Maintenance Mode disables any vCenter appliance running on this host until you exit Maintenance Mode, then restart that vCenter appliance.

2. Click OK to confirm your selection. This will place the ESXi host in Maintenance Mode.
3. Connect to the ESXi host over SSH.
4. Enter the `esxcli` command to install the vGPU Manager package:

```
$ esxcli software vib install -v <directory>/NVIDIA-
VMware_ESXi_6.7_Host_Driver-418.130-1OEM.670.0.0.8169922.vib
Installation Result    Message: Operation finished successfully.
   Reboot Required: false
   VIBs Installed: NVIDIA-VMware_ESXi_6.7_Host_Driver-418.130-
1OEM.670.0.0.8169922.vib
   VIBs Removed:
   VIBs Skipped:
```

Where `<directory>` is the absolute pathname of the directory that contains the `.vib` file. You must specify the absolute path even if the `.vib` file is in the current working directory.

5. 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.

6. Exit Maintenance Mode from the vSphere Web Client by right clicking the host and selecting Exit Maintenance Mode.

You also can exit from Maintenance Mode from the command prompt by entering this command:

$ esxcli system maintenanceMode set -- enable=false

This command does not return a response and does not refresh the vSphere Web Client UI. Click the Refresh icon in the upper right corner of the vSphere Web Client window to refresh the UI.

7. To reboot the host from the vSphere Web Client, right click the host and select Reboot.
8. The host displays a dialog which prompts you to describe a reason for the reboot. Make an appropriate entry in the Log a reason for this reboot operation field, then click OK to continue.

As an alternative to steps 6 and 7, you can reboot the host from the command prompt by entering this command:

$ reboot

This command does not return a response. It does not prompt you to log a reason for the reboot operation.

5.3 Updating vGPU Manager with the .vib File

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

To update the vGPU Manager VIB you must access the ESXi host with the ESXi Shell or SSH. See the VMware documentation for 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. See the VMware documentation for how to place an ESXi host in Maintenance Mode.

1. Enter the `esxcli` command to update the vGPU Manager package:

```
$ esxcli software vib update -v <directory>/NVIDIA-
VMware_ESXi_6.7_Host_Driver-418.130-1OEM.670.0.0.8169922.vib
Installation Result    Message: Operation finished successfully.
   Reboot Required: false
   VIBs Installed: NVIDIA-VMware_ESXi_6.7_Host_Driver-418.130-
1OEM.670.0.0.8169922.vib
   VIBs Removed: NVIDIA-vGPU-
VMware_ESXi_6.0_Host_Driver_390.57-1OEM.600.0.0.2159203
   VIBs Skipped:
```

   WxxxxxxxWhereWhere <`directory`> is the pathname of the directory that contains the `.vib` file.

2. Reboot the ESXi host and remove it from Maintenance Mode.

5.4 Verifying the Installation of vGPU Manager

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

1. Verify that the NVIDIA vGPU Manager was installed and loaded correctly by checking the list of kernels-loaded modules for the NVIDIA kernel driver.

```
$ vmkload_mod -l | grep nvidia
5    8420
```

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

2. Verify that the NVIDIA kernel driver can successfully communicate with the NVIDIA physical GPUs in your system by running the `nvidia-smi` command:

```
$ nvidia-smi
```

If the kernel driver an communicate with the physical GPUs, `nvidia-smi` displays a listing of the GPUs in your platform. The listing is similar to this:

```
Fri Jul 20 17:56:22 2018
+-----------------------------------------------------+
| NVIDIA-SMI 418.130     Driver Version: 418.130      |
|-------------------------------+----------------------+----------------------+
| GPU  Name        Persistence-M| Bus-Id        Disp.A | Volatile Uncorr. ECC |
| Fan  Temp  Perf  Pwr:Usage/Cap|         Memory-Usage | GPU-Util  Compute M. |
|===============================+======================+======================|
|   0  M60           On         | 0000:85:00.0     Off |                  Off |
| N/A   23C    P8    23W / 150W |     13MiB /  8191MiB |      0%      Default |
+-------------------------------+----------------------+----------------------+
|   1  M60           On         | 0000:86:00.0     Off |                  Off |
| N/A   29C    P8    23W / 150W |     13MiB /  8191MiB |      0%      Default |
+-------------------------------+----------------------+----------------------+
|   2  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 instructions.

`nvidia-smi` also lets you monitor GPUs by entering this command:

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

The -l switch makes nvidia-smi run in a loop, refreshing the display at a fixed interval.

5.5 Uninstalling vGPU Manager

To uninstall vGPU Manager:

1. Run `esxcli` to determine the name of the vGPU driver bundle.

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

NVIDIA-VMware_ESXi_6.7_Host_Driver-418.130-1OEM.670.0.0.8169922
```

2. Run this command to uninstall the driver package:

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

The command displays this message if the uninstall process is successful:

```
Removal Result
   Message: Operation finished successfully.
   Reboot Required: false
   VIBs Installed:
   VIBs Removed: NVIDIA-VMware_ESXi_6.7_Host_Driver-418.130-
1OEM.670.0.0.8169922
   VIBs Skipped:
```

3. Reboot the host to complete the uninstall process.

5.6 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.

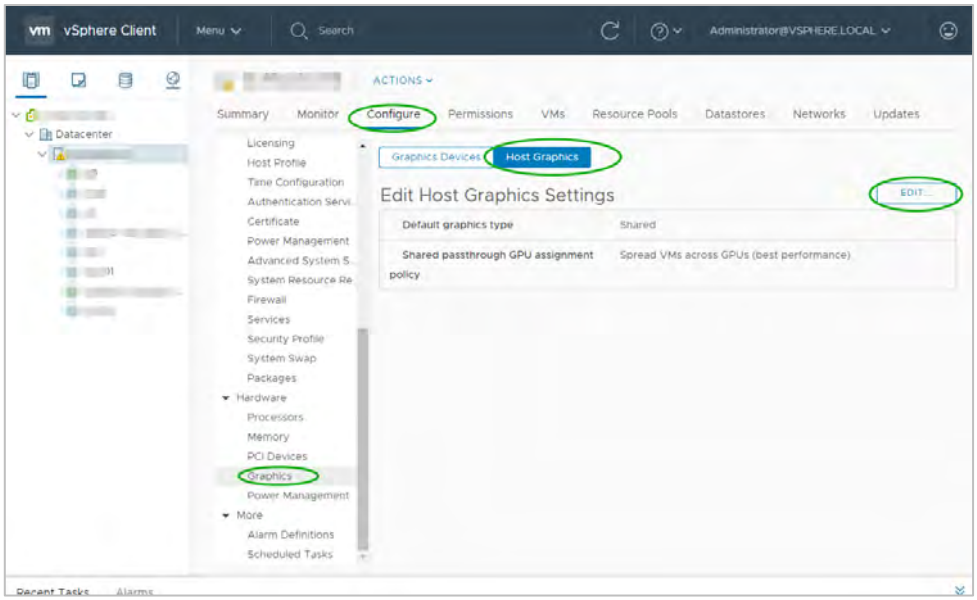
After this VIB is installed, the default graphics type is Shared, which only provides vSGA functionality. To enable vGPU support for VMs in VMware vSphere 6.5, you must change the default graphics type to Shared Direct. If you do not change the default graphics type, VMs to which a vGPU is assigned fail to start, and vSphere displays the error message: “The amount of graphics resource available in the parent resource pool is insufficient for the operation.”

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 you continue configuration.

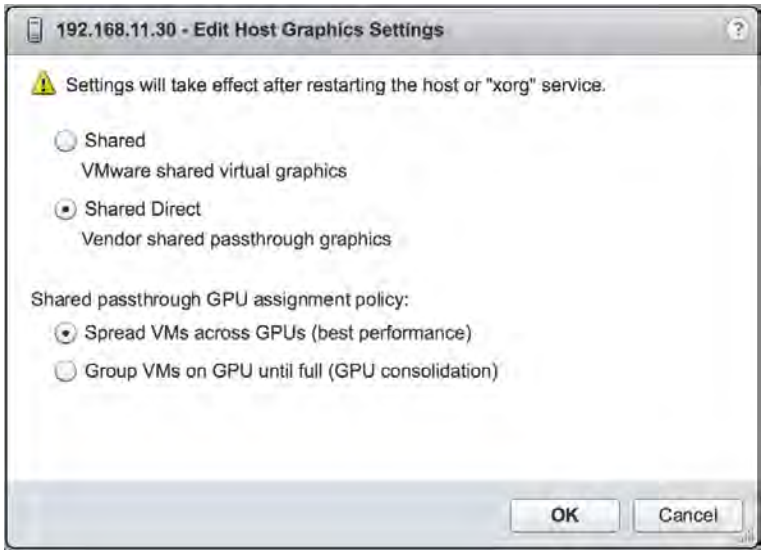
To change the default graphics type:

1. Ensure that the ESXi host is running and that all VMs on the host are powered off.
2. Log in to vCenter Server using the vSphere Web Client.
3. In the window’s left-hand pane, select your ESXi host and click the Configure tab.
4. Select Graphics from the menu (the middle pane), then click the Host Graphics tab. On the Host Graphics tab, click Edit.



vSphere opens the Edit Host Graphics Settings dialog box.

5. Select Shared Direct in the dialog box, then click OK.



This dialog box also lets you change the allocation scheme for vGPU-enabled VMs. For more information, see Modifying GPU Allocation Policy on VMware vSphere.

6. vSphere changes the default graphics type to Shared Direct.
7. Restart the ESXi host or stop and restart X Server and `nv-hostengine` on the ESXi host.

   To stop and restart X Server and `nv-hostengine`:

   a. Stop the Xorg service:

```
$ /etc/init.d/xorg stop
```

b. Stop `nv-hostengine`:

```
$ nv-hostengine -t
```

c. Wait one second to allow `nv-hostengine` to stop.

d. Restart nv-hostengine:

```
$ nv-hostengine -d
```

e. Start the Xorg service.

```
$ /etc/init.d/xorg start
```

8. Configure vGPU as explained in Configuring a vSphere VM with Virtual GPU.

For more information, see the following topics in the VMware vSphere documentation:

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

5.7 Changing the vGPU Scheduling Policy

GPUs, starting with the NVIDIA Maxwell™ graphic architecture, implement a **best-effort vGPU scheduler** that aims to balance performance across vGPUs. The best effort-scheduler allows one 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, starting with the NVIDIA Pascal™ architecture, also support **equal-share** and **fixed-share vGPU schedulers**. These schedulers impose a limit on GPU processing cycles used by a single vGPU, which prevents graphics-intensive applications running in one VM from affecting the performance of graphics-light applications running in other VMs. The best-effort scheduler is the default scheduler for all supported GPU architectures.

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

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

The GPUs that are based on the NVIDIA Turing™ architecture are the NVIDIA T4, RTX6000 and RTX8000.The GPU that is based on the NVIDIA Ampere™ architecture is the NVIDIA A100.

5.7.1 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.

5.7.2 RmPVMRL Registry Key

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

Type

`Dword`

Contents

Value	Meaning
`0x00` (default)	Best effort scheduler
`0x01`	Equal share scheduler with the default time slice length
`0x00TT0001`	Equal share scheduler with a user-defined time slice length TT
`0x11`	Fixed share scheduler with the default time slice length
`0x00TT0011`	Fixed share scheduler with a user-defined time slice length TT

Examples

The default time slice length depends on the maximum number of vGPUs per physical GPU allowed for the vGPU type.

Maximum Number of vGPUs	Default Time Slice Length
Less than or equal to 8	2 ms
Greater than 8	1 ms

TT

Two hexadecimal digits in the range 01 to 1E that set the length of the time slice in milliseconds (ms) for the equal share and fixed share schedulers. The minimum length is 1 ms and the maximum length is 30 ms.

If *TT* is 00, the length is set to the default length for the vGPU type.

If *TT* is greater than 1E, the length is set to 30 ms.

Examples

This example sets the vGPU scheduler to equal share scheduler with the default time slice length.

```
RmPVMRL=0x01
```

This example sets the vGPU scheduler to equal share scheduler with a time slice that is 3 ms long.

```
RmPVMRL=0x00030001
```

This example sets the vGPU scheduler to fixed share scheduler with the default time slice length.

```
RmPVMRL=0x11
```

This example sets the vGPU scheduler to fixed share scheduler with a time slice that is 24 (0x18) ms long.

```
RmPVMRL=0x00180011
```

5.7.3 Changing the vGPU Scheduling Policy for All GPUs

You can change the vGPU scheduling policy only on GPUs based on the NVIDIA Pascal, NVIDIA Volta, NVIDIA Turing, and NVIDIA Ampere 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. Set the RmPVMRL registry key to the value that sets the GPU scheduling policy that you want.

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

Value **-** The value that sets the vGPU scheduling policy that you want, for example:

- **`0x01`** **-** Sets the vGPU scheduling policy to Equal Share Scheduler.
- **`0x11`** **-** Sets the vGPU scheduling policy to Fixed Share Scheduler.
- For all supported values, see RmPVMRL Registry Key.

2. Reboot your hypervisor host machine.

5.7.4 Changing the vGPU Scheduling Policy for Specific GPUs

You can change the vGPU scheduling policy only on GPUs based on the NVIDIA Pascal, NVIDIA Volta, NVIDIA Turing, and NVIDIA Ampere architectures.

To change the vGPU scheduling policy for specific GPUs:

1. Open a command shell as the root user on your hypervisor host machine. You can use secure shell (SSH) for this purpose on all supported hypervisors.
2. Enter the `lspci` command and pipe its output to `grep` to display information only for NVIDIA GPUs:

   ```
   # lspci | grep NVIDIA
   ```

   The output of grep shows the PCI domain and bus/device/function (BDF) of each NVIDIA GPU. Note this information for each GPU whose scheduling behavior you want to change.

   For example, the GPUs listed here have the PCI domain 0000 and BDFs 85:00.0 and 86:00.0:

   - 0000:85:00.0 VGA compatible controller: NVIDIA Corporation GM204GL [M60] (rev a1)
   - 0000:86:00.0 VGA compatible controller: NVIDIA Corporation GM204GL [M60] (rev a1)
3. Enter an `esxcli set` command with the module parameter `NVreg_RegistryDwordsPerDevice` to set the PCI and `RmPVMRL` registry keys for each GPU:

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

   Where:

   - `<domain>` is the PCI domain of the GPU.
   - `<bdf>` is the PCI device BDF of the GPU.
   - `<value>` is the value that sets the vGPU scheduling policy you want, from list in section 5.7.2 above.

   For example, add this entry to the file `/etc/modprobe.d/nvidia.conf` to change the scheduling behavior of two GPUs:

   ```
   options nvidia NVreg_RegistryDwordsPerDevice=
   "pci=0000:85:00.0;RmPVMRL=0x01;pci=0000:86:00.0;RmPVMRL=0x11"
   ```

   The changes are:

   - For the GPU at PCI domain 0000 with BDF 85:00.0, use the equal-share scheduler.
   - For the GPU at PCI domain 0000 with BDF 86:00.0, use the fixed-share scheduler.
4. Reboot your hypervisor host machine.

5.7.5 Restoring Default vGPU Scheduling Policies

To restore default vGPU scheduling policies:

1. Open a command shell as the root user on your hypervisor host machine. You can use secure shell (SSH) for this purpose on all supported hypervisors.
2. Unset the `RmPVMRL` registry key.
3. Set the module parameter to the empty string:

```
# esxcli system module parameters set -m nvidia -p "<module>="
```

Where <module> is the module parameter to set, which depends on whether the scheduling behavior was changed for all GPUs or specified GPUs:

- For all GPUs, set the module parameter NVreg_RegistryDwords.
- For specified GPUs, set the module parameter NVreg_RegistryDwordsPerDevice.

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="
```

4. Reboot your hypervisor host machine.

5.8 Disabling and Enabling ECC Memory

Some GPUs that support NVIDIA vGPU software support error correcting code (ECC) memory with NVIDIA vGPU. ECC memory improves data integrity by detecting and handling double-bit errors. However, not all GPUs, vGPU types, and hypervisor software versions support ECC memory with NVIDIA vGPU.

On GPUs that support ECC memory with NVIDIA vGPU, ECC memory is supported with C-series and Q-series vGPUs, but not with A-series and B-series vGPUs. Although A-series and B-series vGPUs start on physical GPUs on which ECC memory is enabled, enabling ECC with vGPUs that do not support it might incur some costs.

On physical GPUs that do not have HBM2 memory, the amount of frame buffer that is usable by vGPUs is reduced. All types of vGPU are affected, not just vGPUs that support ECC memory.

The effects of enabling ECC memory on a physical GPU are as follows:

- ECC memory is exposed as a feature on all supported vGPUs on the physical GPU.
- In VMs that support ECC memory, ECC memory is enabled, with the option to disable ECC in the VM.
- ECC memory can be enabled or disabled for individual VMs. Enabling or disabling ECC memory in a VM does not affect the amount of frame buffer that is usable by vGPUs.

GPUs based on the Pascal GPU architecture and later GPU architectures support ECC memory with NVIDIA vGPU. These GPUs are supplied with ECC memory enabled.

Tesla M60 and M6 GPUs support ECC memory when used without GPU virtualization, but NVIDIA vGPU does not support ECC memory with these GPUs. In graphics mode, these GPUs are supplied with ECC memory disabled by default.

Some hypervisor software versions do not support ECC memory with NVIDIA vGPU.

If you are using a hypervisor software version or GPU that does not support ECC memory with NVIDIA vGPU and ECC memory is enabled, NVIDIA vGPU fails to start. In this situation, you must ensure that ECC memory is disabled on all GPUs if you are using NVIDIA vGPU.

5.8.1 Disabling ECC Memory

If ECC memory is unsuitable for your workloads but is enabled on your GPUs, disable it. You must also ensure that ECC memory is disabled on all GPUs if you are using NVIDIA vGPU with a hypervisor software version or a GPU that does not support ECC memory with NVIDIA vGPU. If your hypervisor software version or GPU does not support ECC memory and ECC memory is enabled, NVIDIA vGPU fails to start.

Where to perform this task from depends on whether you are changing ECC memory settings for a physical GPU or a vGPU.

- For a physical GPU, perform this task from the hypervisor host.
- For a vGPU, perform this task from the VM to which the vGPU is assigned.

Note: ECC memory must be enabled on the physical GPU on which the vGPUs reside.

Before you begin, ensure that NVIDIA Virtual GPU Manager is installed on your hypervisor. If you are changing ECC memory settings for a vGPU, also ensure that the NVIDIA vGPU software graphics driver is installed in the VM to which the vGPU is assigned.

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

```
# nvidia-smi -q

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

Timestamp                           : Mon Jul 13 18:36:45 2020
Driver Version                      : 450.55

Attached GPUs                       : 1
GPU 0000:02:00.0

[...]

    Ecc Mode
        Current                     : Enabled
        Pending                     : Enabled

[...]
```

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

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

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

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

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

id is the index of the GPU or vGPU 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
```

3. Reboot the host or restart the VM.
4. Confirm that ECC is now disabled for the GPU or vGPU.

```
# nvidia–smi –q

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

Timestamp                           : Mon Jul 13 18:37:53 2020
Driver Version                      : 450.55

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

    Ecc Mode
        Current                     : Disabled
        Pending                     : Disabled

[...]
```

5.8.2 Enabling ECC Memory

If ECC memory is suitable for your workloads and is supported by your hypervisor software and GPUs, but is disabled on your GPUs or vGPUs, enable it.

Where to perform this task from depends on whether you are changing ECC memory settings for a physical GPU or a vGPU.

- For a physical GPU, perform this task from the hypervisor host.
- For a vGPU, perform this task from the VM to which the vGPU is assigned.

Note: ECC memory must be enabled on the physical GPU on which the vGPUs reside.

Before you begin, ensure that NVIDIA Virtual GPU Manager is installed on your hypervisor. If you are changing ECC memory settings for a vGPU, also ensure that the NVIDIA vGPU software graphics driver is installed in the VM to which the vGPU is assigned.

1. Use nvidia-smi to list the status of all physical GPUs or vGPUs, and check for ECC noted as disabled.

```
# nvidia-smi -q

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

Timestamp                           : Mon Jul 13 18:36:45 2020
Driver Version                      : 450.55

Attached GPUs                       : 1
GPU 0000:02:00.0

[...]

    Ecc Mode
        Current                     : Disabled
        Pending                     : Disabled

[...]
```

2. Change the ECC status to on for each GPU or vGPU for which ECC is enabled.
   - If you want to change the ECC status to on for all GPUs on your host machine or vGPUs assigned to the VM, run this command:

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

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

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

     id is the index of the GPU or vGPU 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
```

3. Reboot the host or restart the VM.
4. Confirm that ECC is now enabled for the GPU or vGPU

```
# nvidia–smi –q

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

Timestamp                           : Mon Jul 13 18:37:53 2020
Driver Version                      : 450.55

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

    Ecc Mode
```

```
        Current                     : Enabled
        Pending                     : Enabled

[...]
```

Chapter 6. Deploying the NVIDIA vGPU Software License Server

This chapter covers deployment of the NVIDIA vGPU software license server, including:

- Platform Requirements
- Installing the Java Runtime Environment on Windows
- Installing the License Server Software on Windows

6.1 Platform Requirements

Before proceeding, ensure that you have a platform suitable for hosting the license server

6.1.1 Hardware and Software Requirements

- The hosting platform may be a physical machine, an on-premises virtual machine (VM), or a VM on a supported cloud service. NVIDIA recommends using a host that is dedicated solely to running the license server.
- The recommended minimum configuration is 2 CPU cores and 4 GB of RAM. A high-end configuration of 4 or more CPU cores with 16 GB of RAM is suitable for handling up to 150,000 licensed clients.
- At least 1 GB of hard drive space is required.
- The hosting platform must run a supported operating system.
- On Window platforms, .NET Framework 4.5 or later is required.

6.1.2 Platform Configuration Requirements

- The platform must have a fixed (unchanging) IP address. The IP address may be assigned dynamically by DHCP or statically configured but must be constant.
- The platform must have at least one unchanging Ethernet MAC address, to be used as a unique identifier when registering the server and generating licenses in the NVIDIA Licensing Portal.
- The platform's date and time must be set accurately. NTP is recommended.

6.1.3 Network Ports and Management Interface

The license server requires TCP port 7070 to be open in the platform's firewall, to serve licenses to clients. By default, the installer will automatically open this port. The license server's management interface is web-based and uses TCP port 8080. The management interface itself does not implement access control; instead, the installer does not open port 8080 by default, so that the management interface is only available to web browsers running locally on the license server host. Access to the management interface is therefore controlled by limiting remote access (via VNC, RDP, etc.) to the license server platform.

Note: If you choose to open port 8080 during license server installation, or at any time afterwards, the license server's management interface is unprotected.

6.2 Installing the NVIDIA vGPU Software License Server on Windows

The license server requires a Java runtime environment, which must be installed separately before you install the license server.

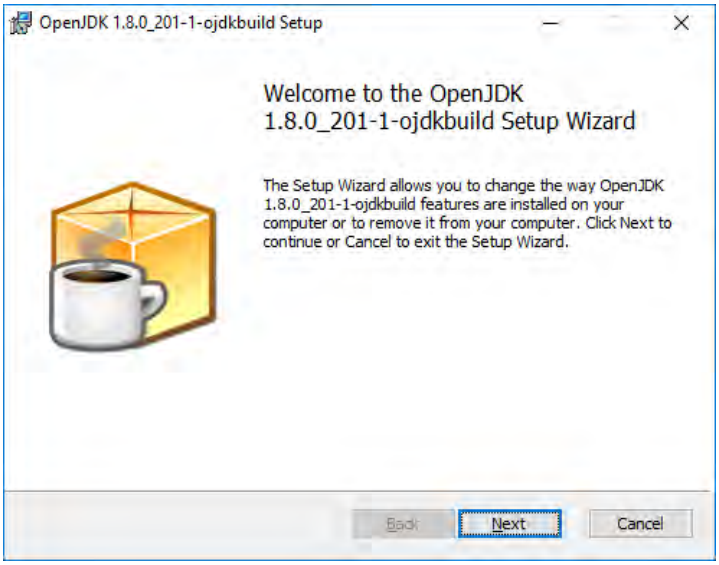
6.2.1 Installing the Java Runtime Environment on Windows

If a suitable Java runtime environment (JRE) version is not already installed on your system install a supported JRE before running the NVIDIA license server installer.

1. Download a supported 64-bit Oracle Java SE JRE or OpenJDK JRE.
   - Download Oracle Java SE JRE from the Java Downloads for All Operating Systems page.
     - Download Oracle Java SE JRE from the java.com: Java + You page
   - Download OpenJDK JRE from the Community builds using source code from OpenJDK project on GitHub.

1. Install the JRE that you downloaded.
   - Oracle Java SE JRE installation:

- OpenJDK JRE installation:



2. Set the JAVA_HOME system variable to the full path to the jre folder of your JRE installation.
   - **For 64-bit Oracle Java SE JRE**: C:\Program Files\Java\jre1.8.0_191
   - **For 64-bit OpenJDK JRE**: C:\Program Files\ojdkbuild\java-1.8.0-openjdk-1.8.0.201-1\jre

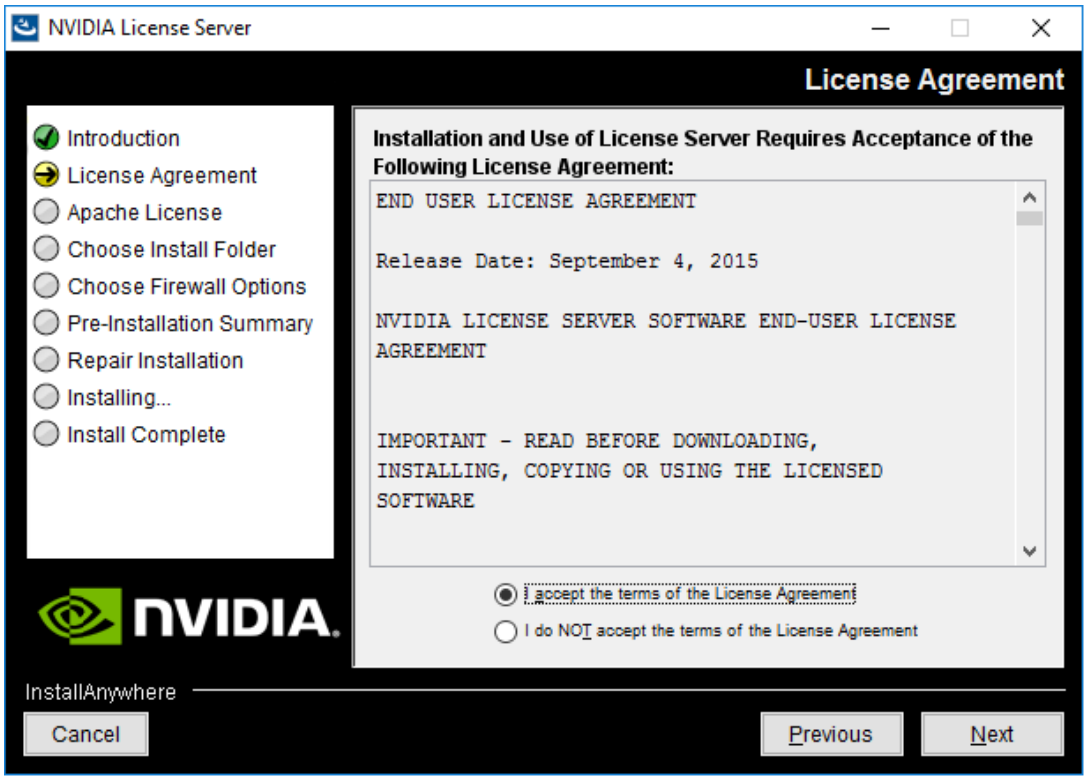
   Ensure that the path does not include any trailing characters, such as a slash or a space.

   If you are upgrading to a new version of the JRE, update the value of the JAVA_HOME system variable to the full path to the jre folder of your new JRE version.
3. Ensure that the Path system variable contains the path to the java.exe executable file.
   - **For 64-bit Oracle Java SE JRE**: C:\Program Files\Java\jre1.8.0_191\bin

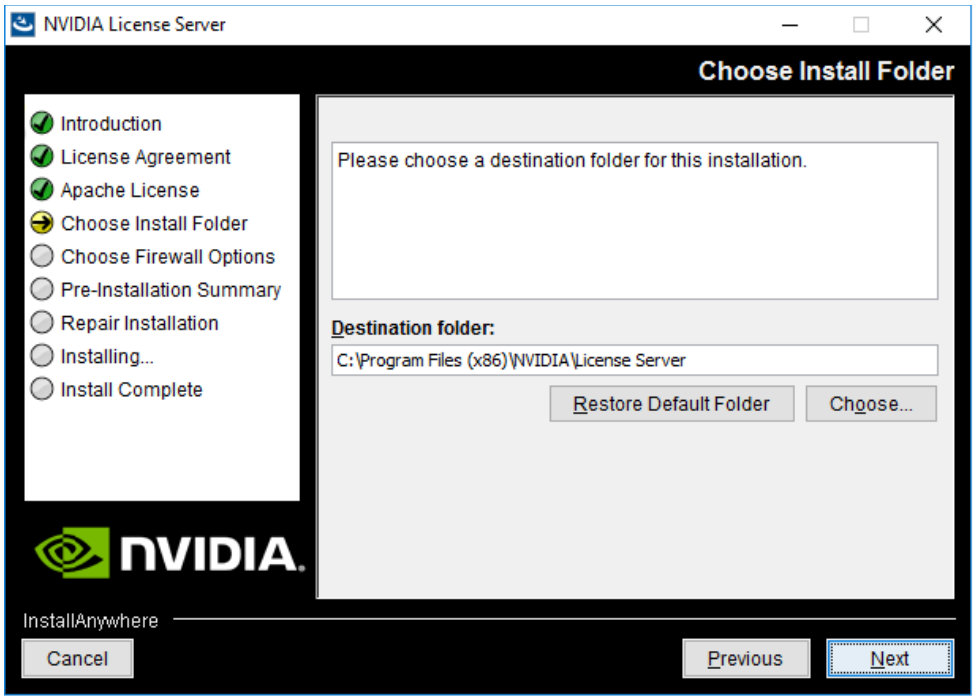
- **For 64-bit OpenJDK JRE**: C:\Program Files\ojdkbuild\java-1.8.0-openjdk-1.8.0.201-1\bin

6.2.2 Installing the License Server Software on Windows

1. Unzip the license server installer and run setup.exe.
2. Accept the EULA for the license server software and the Apache Tomcat software used to support the license server's management interface.



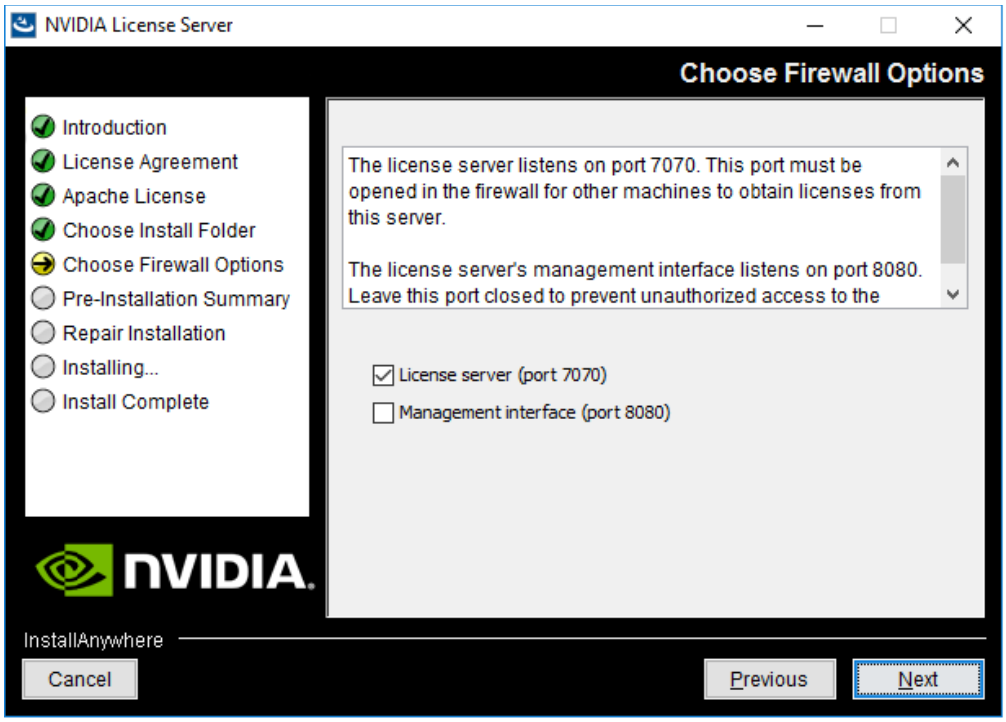
3. Choose the destination folder where you want the license server software to be installed.



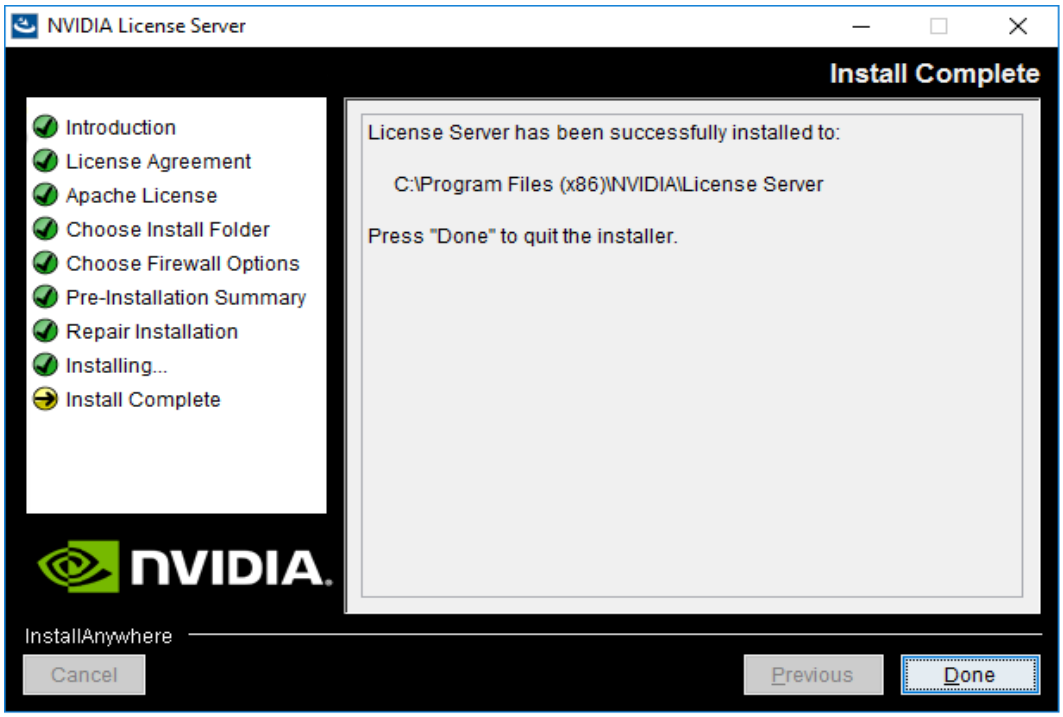
4. In the Choose Firewall Options dialog box, select the ports to be opened in the firewall.

   To enable remote clients to access licenses from the server and prevent remote access to the management interface, use the default setting, which sets ports as follows:

   - Port 7070 is open to enable remote clients to access licenses from the server.
   - Port 8080 is closed to ensure that the management interface is available only through a web browser running locally on the license server host.



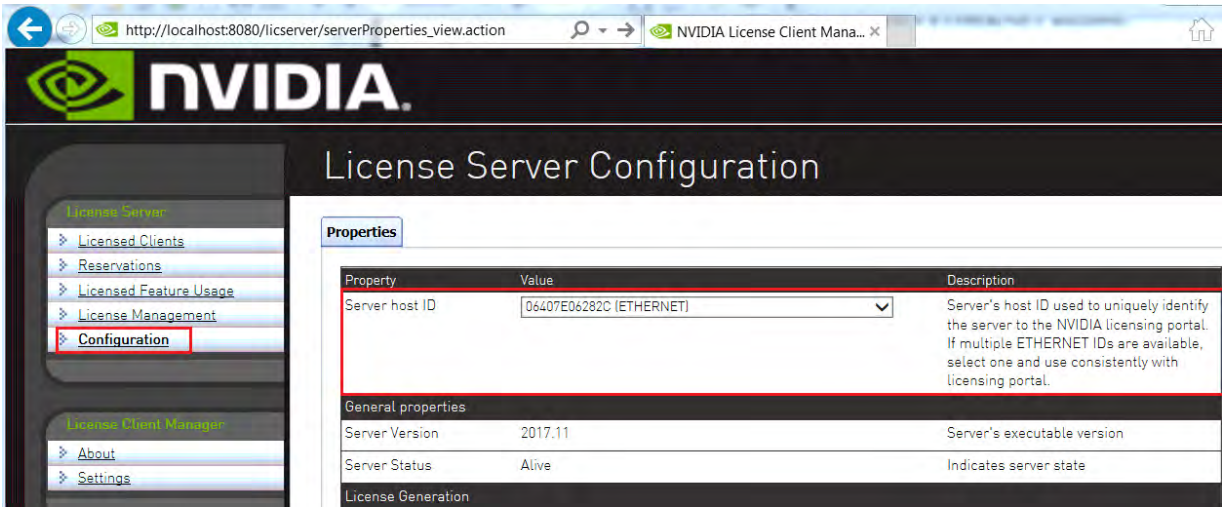
5. After installation has completed successfully, click Done to exit the installer.



6.2.3 Obtaining the License Server's MAC Address

The license server's Ethernet MAC address uniquely identifies your server to the NVIDIA Licensing Portal. You will need this address to register your license server with the NVIDIA Licensing Portal to generate license files.

1. Open a web browser on the license server host and connect to the URL `http://localhost:8080/licserver`.
2. In the license server management interface, select **Configuration**.
3. On the License Server Configuration page that opens, in the **Server host ID** drop-down list, select the platform's ETHERNET address.



6.2.4 Managing your License Server and Getting your License Files

To be able to download NVIDIA vGPU software licenses, you must create at least one license server on the NVIDIA Licensing Portal and allocate licenses to the server. After creating a license server and allocating licenses to it, you can download your license file.

6.2.4.1 Creating a Licenser Server on the NVIDIA Licensing Portal

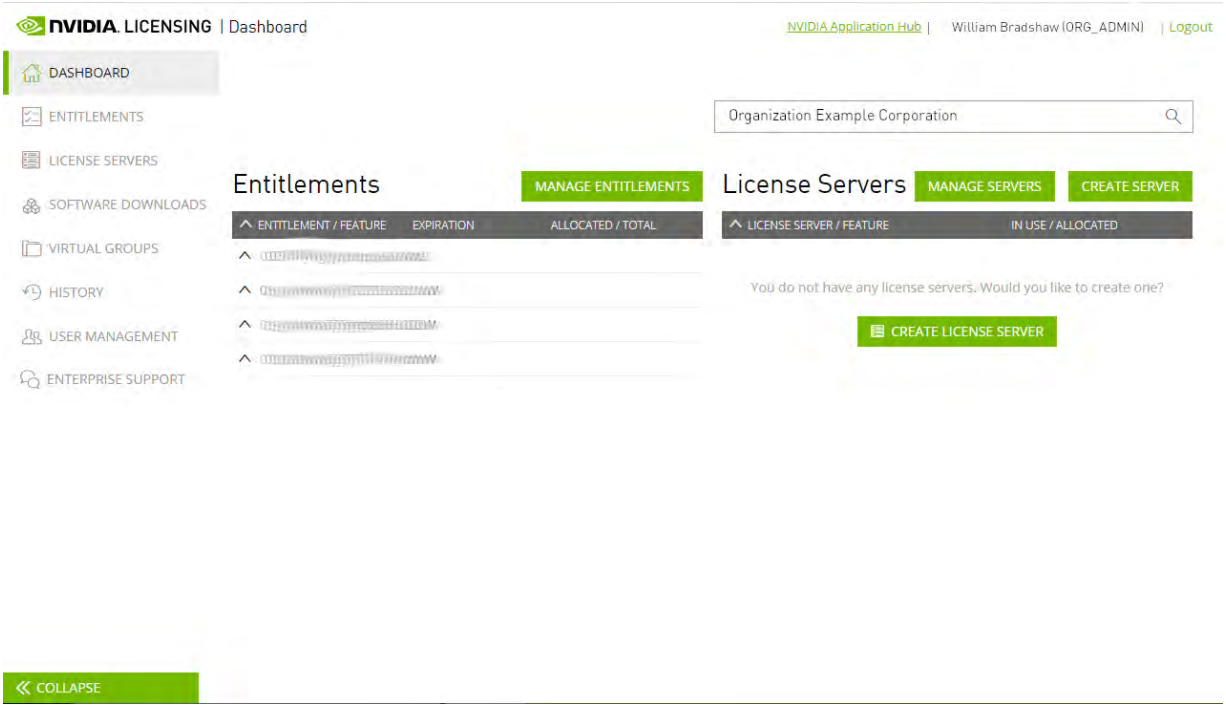
To be able to download NVIDIA vGPU software licenses, you must create at least one license server on the NVIDIA Licensing Portal. Creating a license server on the NVIDIA Licensing Portal registers your license server host with the NVIDIA Licensing Portal through the MAC address of the host.

1. In the NVIDIA Licensing Portal, navigate to the organization or virtual group for which you want to create the license server.

a. If you are not already logged in, log in to the NVIDIA Enterprise Application Hub and click **NVIDIA LICENSING PORTAL** to go to the NVIDIA Licensing Portal.

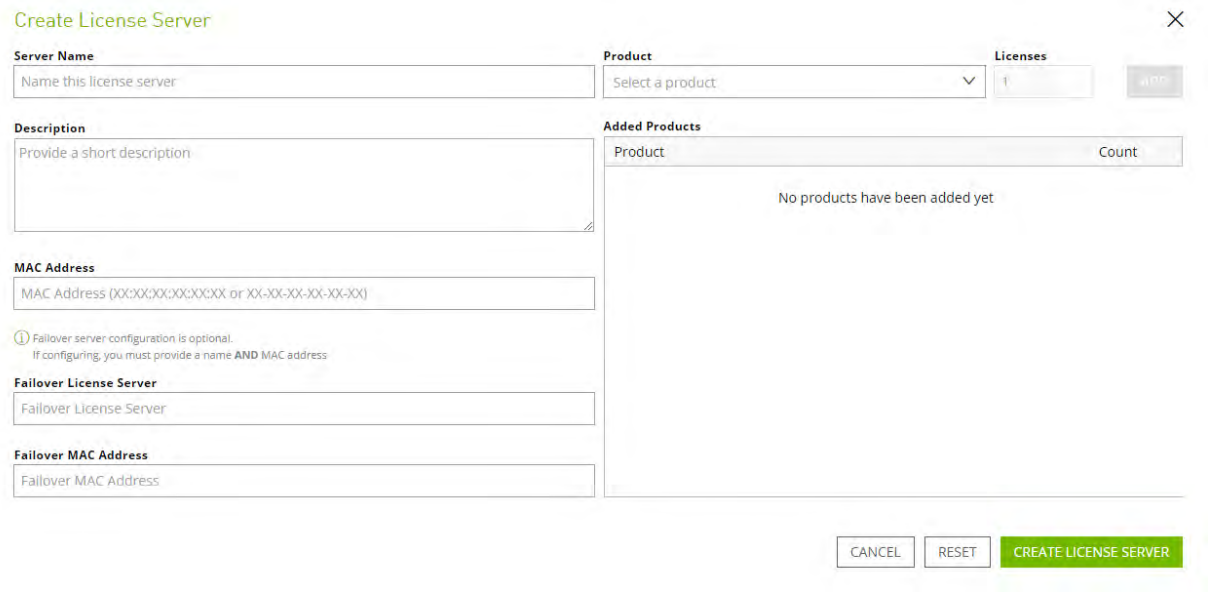
b. **Optional:** If your assigned roles give you access to multiple virtual groups, select the virtual group for which you are creating the license server from the list of virtual groups at the top right of the page.

If no license servers have been created for your organization or virtual group, the NVIDIA Licensing Portal dashboard displays a message asking if you want to create a license server.



2. On the NVIDIA Licensing Portal dashboard, click **CREATE LICENSE SERVER**.

   The Create License Server pop-up window opens.



3. Provide the details of your license server.
    a. In the **Server Name** field, enter the host name of the license server.
    b. In the **Description** field, enter a text description of the license server. This description is required and will be displayed on the details page for the license server that you are creating.
    c. In the **MAC Address** field, enter the MAC address of your license server.
4. Add the licenses for the products that you want to allocate to this license server. For each product, add the licenses as follows:
    a. From the **Product** drop-down list, select the product for which you want to add licenses.
    b. In the **Licenses** field, enter the number of licenses for the product that you want to add.
    c. Click **ADD**.
5. Leave the **Failover License Server** and **Failover MAC Address** fields unset.
6. Click **CREATE LICENSE SERVER**.

6.2.4.2 Downloading a License File

Each license server that you create has license file associated with it. The license file contains all the licenses that you allocated to the license server. After downloading the license file, you can install it on the license server host associated with the license server on the NVIDIA Licensing Portal.

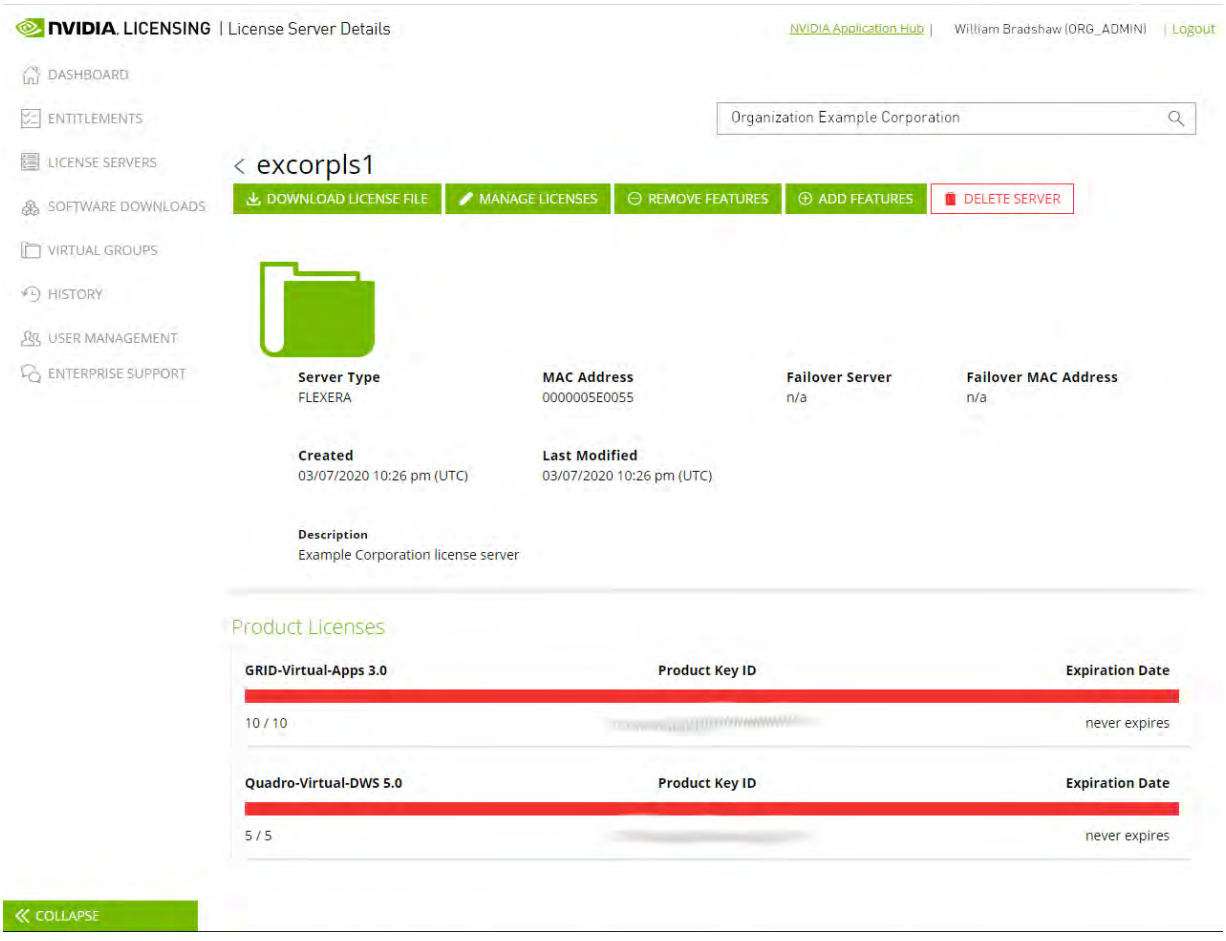
1. In the NVIDIA Licensing Portal, navigate to the organization or virtual group for which you want to download the license file.

a. If you are not already logged in, log in to the NVIDIA Enterprise Application Hub and click **NVIDIA LICENSING PORTAL** to go to the NVIDIA Licensing Portal.

b. **Optional:** If your assigned roles give you access to multiple virtual groups, select the virtual group for which you are downloading the license file from the list of virtual groups at the top right of the page.

2. In the list of license servers on the NVIDIA Licensing Portal dashboard, select the license server whose associated license file you want to download.

3. In the License Server Details page that opens, review the licenses allocated to the license server.



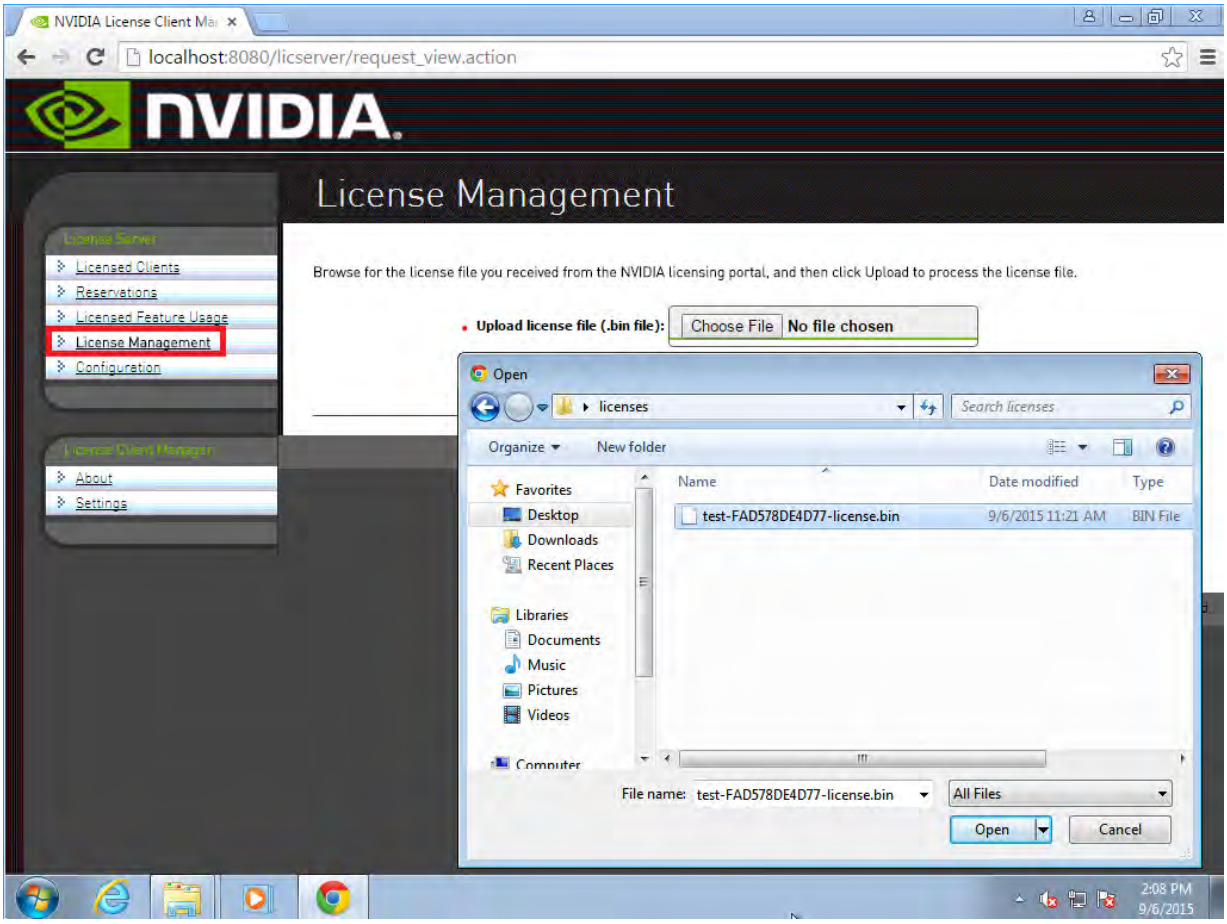
4. Click **DOWNLOAD LICENSE FILE** and save the .bin license file to your license server for installation.

6.2.5 Installing a License

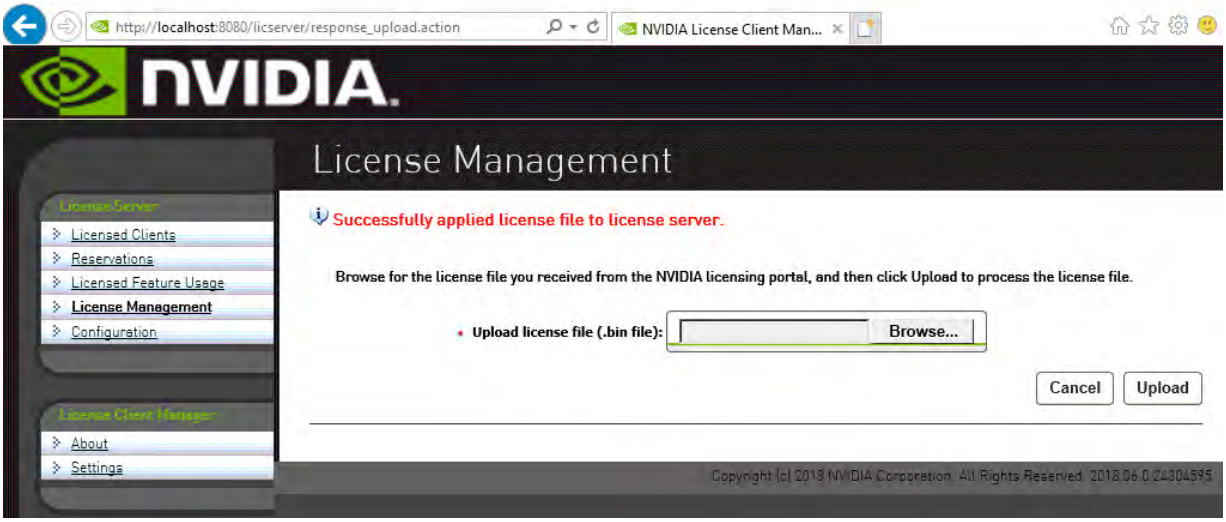
NVIDIA vGPU software licenses are distributed as .bin files for download from the NVIDIA Licensing Portal.

Before installing a license, ensure that you have downloaded the license file from the NVIDIA Licensing Portal.

1. In the license server management interface, select **License Management**.
2. On the License Management page that opens, click **Choose File**.



3. In the file browser that opens, select the .bin file and click **Open**.
4. Back on the License Management page, click **Upload** to install the license file on the license server. The license server should confirm successful installation of the license file.



Note: For additional configuration options including Linux server deployment, securing your license server, and license provisioning, refer to the Virtual GPU Software License Server User Guide.

Chapter 7. Selecting the Correct vGPU Profiles

A **vGPU profile** defines the properties of a vGPU. Profiles are grouped into series that are compatible with various physical GPU architectures.

Because NVIDIA vGPUs are a licensed feature, they require software licenses to be used on a virtual machine. The type of license required for a vGPU depends on the series of the profile used. Table 7-1 summarizes the vGPU profile series, their licenses, and their characteristics. A complete list of vGPU profiles may be found in Virtual GPU Software User Guide.

Table 7-1

vGPU Profile Series	Designed for	Required License
Q-series	Creative & technical professionals who need Quadro performance & features	Quadro Virtual Data Center Workstation (vDWS)
C-series	Intensive server workloads: artificial intelligence (AI), deep learning, or high-performance computing (HPC)	Virtual Compute Server license or Quadro vDWS
B-series	Virtual desktops for business professionals & knowledge workers	NVIDIA GRID® Virtual PC license or Quadro vDWS
A-series	Streaming or session-based services for virtual applications	GRID Virtual Applications

It is critical to choose the right profiles to give your users the best virtual GPU service.

7.1 The Role of the vGPU Manager

You use NVIDIA vGPU Manager to customize several properties of vGPU profiles to suit your users' needs, and to assign an appropriate vGPU profile to each vGPU.

Each virtual machine (VM) has specified amount of dedicated graphics memory. You must set the amount of dedicated graphics memory assigned to each vGPU profile to ensure that a VM which use it has the resources it needs to handle the expected graphics load.

7.2 Bringing Quadro to Virtualization

The Q-series vGPU profiles undergo the same rigorous application certification process as the NVIDIA® Quadro® line of adapters for professional graphics applications. As a result, you can expect 100% compatibility and performance. with your applications. ISVs requiring Quadro GPUs can certify against the NVIDIA vGPU profiles as well.

7.3 Matching Profiles to User Needs

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

Continuing the example SOLIDWORKS use cases described in the section Size Your Environment, the VMs' video RAM and display needs can be matched to appropriate vGPU profiles:

- Entry level engineer or design reviewer
  - 8 GB RAM
  - 4 vCPUs (2.4 GHz)
  - 2-4 GB video RAM
  - One 1920×1080 display
- Mid-level engineer
  - 16 GB RAM
  - 4-8 vCPUs (2.6 GHz)
  - 4-8 GB video RAM
  - Two 1920×1080 displays and/or One 2560×1600
- Advanced engineer
  - 32-64 GB RAM
  - 8-12 vCPUs (3.2 GHz)
  - 8-16 GB video RAM
  - Two 2560×1600 displays and/or one 3840×2160

7.4 Creative and Technical Professionals

Creative and technical professionals with very demanding tasks 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. NVIDIA data center GPUs with NVIDIA Quadro Virtual Data Center Workstation (Quadro vDWS) software extends the benefits of virtualization to these users.

The keys to picking the proper profile for creative and technical professionals are 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 is critical to understand what applications your users require on their virtual machines. You must 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 you must estimate properly to ensure the best user experience.

Begin profile selection by considering the requirements of your users' primary applications. Professional application software vendors certify their products to run with NVIDIA Quadro to ensure the performance is tuned for maximum efficiency. Most vendors' web sites have a dedicated page about the proper GPU hardware to use; use those recommendations to select the appropriate vGPU profile to meet your end users' needs. To understand more about the graphics requirements of your users' applications, consult the application vendors.

7.5 Knowledge Worker Profiles

Today's digital workplace is changing due to the increased graphics requirements of 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 do not launch.

According to a study by Lakeside SysTrack Community data, the number of applications that are accelerated by graphics have doubled over the past five years. Below are some examples and their percentages of GPU demands based on the transition from a Microsoft Windows 7 environment to a Microsoft Windows 10 environment.

- Google Chrome: 36% increase
- Mozilla Firefox: 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 require a GPU for graphics acceleration, it is also clear that common business applications now need GPU resources to run efficiently.

Chapter 8. Application Based VDI Hosting Sizing

8.1 How Many Users Can I Support on a Host?

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

Customer: How many users can I support 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 digital twins. It 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 fall into the following categories, which are appropriate for different groups of NVIDIA vGPU profiles:

	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

	Light User	Medium User	Heavy User
Guaranteed performance	12 users per server P4-4Q 4 vCPU 8GB RAM	6 users per server P4-8Q 8v CPU 8GB RAM	3 users per server P40-24Q 12 vCPU 8GB RAM2
Typical deployment	16-24 users per server P4-2Q 4 vCPU 8-16GB RAM	12-18 users per server P4-2Q / P4-4Q 8 vCPU 16-32GB RAM	6-9 users per server P40-8Q / P40-12Q 12 vCPU >96GB RAM

Notice the two rows of configurations. It is *critical* to ensure that the customer understands and deploys against the appropriate level of performance

- **Guaranteed performance** describes resources and constrains that guarantee the user will experience full performance of the vGPU at all times.
- **Typical deployment** describes resources and constraints that guarantee the user a typical level of performance that may be reduced at certain times. Do not use these figures if a guaranteed level of performance is critical.

8.3 Application-Based Sizing

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

The first step is to consider the specifications of a physical workstation build.

8.3.1 Recommended System Requirements

8-1 lists recommended minimum, recommended, and optimal hardware resources for a workstation supported by vSphere Hypervisor.

Table 8-1 Hardware Resources for a Workstation

Resource	Requirement
CPU speed	Minimum: Hyperthreaded dual core * Recommended: Quad core * Optimal: 2x hyperthreaded six-core *
Platform	x64 with SSE2 extensions
Memory/RAM	Minimum: 4 GB Recommended: 16 GB Optimal: 32 GB

Resource	Requirement
Display properties	24-bit color depth
GPU	Good: NVIDIA® Quadro® P1000 Better: NVIDIA Quadro P2000 Best: NVIDIA Quadro P5000
Disk space	Recommended: 1 TB solid state drive (SSD)

The test keys on the “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 and establishing remote connections to executing the industry standard graphics performance benchmark 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. Scores were averaged across three runs. Table 8-2 describes the VM Configuration used:

Table 8-2 VM Configuration for vCPU Performance Testing

VM CONFIGURATION	
Operating System	Windows 10 (1903)
vCPUs	8
vMemory	16GB
Hard Disk	100GB
vGPU Driver Version	NVIDIA Virtual GPU Software 10.X

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	1920×1080
HYPERVISOR CONFIGURATION	
Hypervisor	VMware vSphere 6.7.0
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 10.X Software

A Proof of Concept (POC) is more likely to be successful when end users are classified as outlined above, as this allows standardization for optimal performance while minimizing management overhead. Customers often segment their end users into three user groups for each application and run similar user types on the same host. The below table outlines the most common definitions of user groups for Siemens NX, along with recommended configurations.

Table 8-3 Common User Group Definitions for Siemens NX

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. This CPU's 18 cores and high frequency of 3.0 GHz are well suited to getting optimal performance for each end user while supporting the largest user scale on the server, making it also a cost-effective solution for Siemens NX.

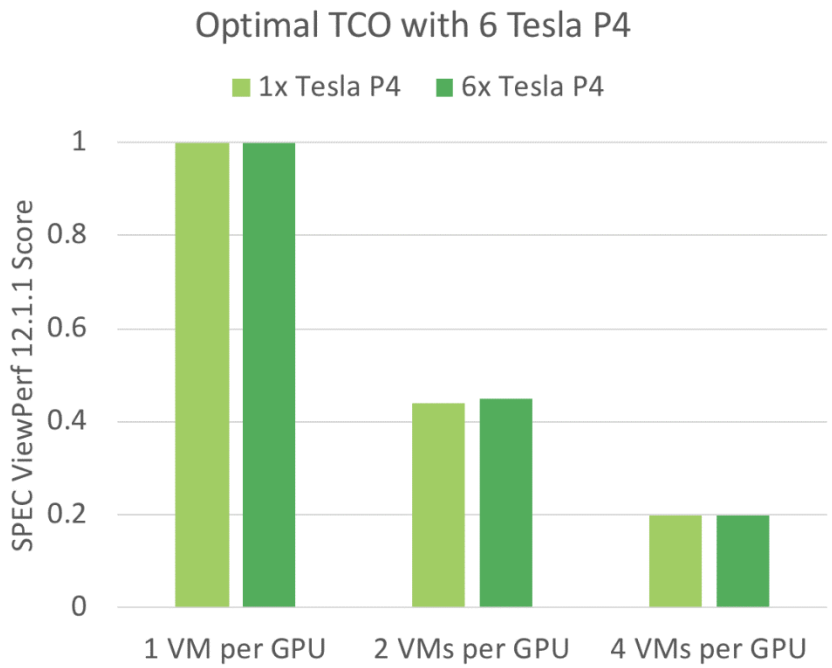
Based on SPECviewperf 12.1.1, the below section outlines what typical customers usually configure based on various needs.

P4 with Quadro vDWS for Light and Medium Users

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

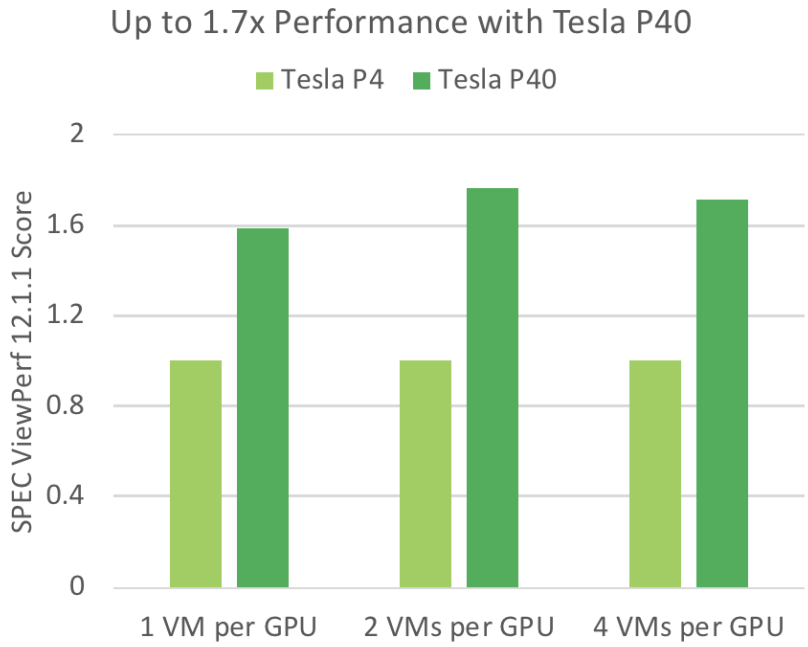
The form factor of the NVIDIA P4 (single width, half height and powered through the PCIe bus) allows the use of up to six NVIDIA P4s in a server with two Intel Xeon Gold 6154 CPUs. SPECviewperf 12.1.1 tests show that six NVIDIA P4 GPUs with two Intel Xeon Gold 6154 CPUs is a well-balanced configuration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

The below bar graph demonstrates that there are enough CPU resources available to host six NVIDIA 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.



NVIDIA P40 with Quadro vDWS for Heavy Users

NVIDIA recommends the NVIDIA P40 with Quadro vDWS for heavy users that require the additional performance of an NVIDIA P40 over an NVIDIA P4. The NVIDIA P40 is the optimal choice for heavy users due to its greater performance and larger frame buffer, while the NVIDIA P4 is recommended for light and medium users due to the possibility of hosting up to six NVIDIA P4s in a two-socket, 2U server.



Sufficient System Memory for each Individual User

While SPECviewperf 12.1.1 performs optimally with 8 GB of system memory per virtual machine, Siemens NX customers typically assign 16–32 GB of system memory to medium users for optimal performance. System memory requirements do not change with the transition to Quadro vDWS; therefore the same amount of system memory should be assigned to the Quadro vDWS accelerated virtual machine as is used in a physical workstation.

Flash Based Storage for Best Performance

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

A typical configuration for non-persistent virtual machines uses direct attached storage (DAS) on 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 for Quadro vDWS because its requirements vary based on multiple factors such as choice of hypervisor, persistent/non-persistent virtual machines, choice of storage solution and many more. Most customers are using 10 Gbit networking for optimal performance..

Chapter 9. Creating Your First vGPU Virtual Desktop

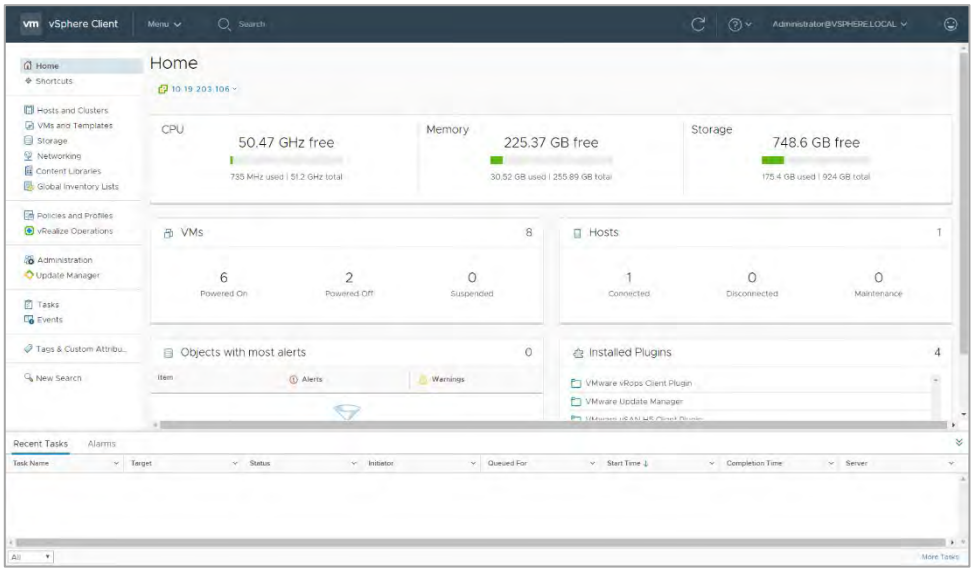
This chapter describes how to:

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

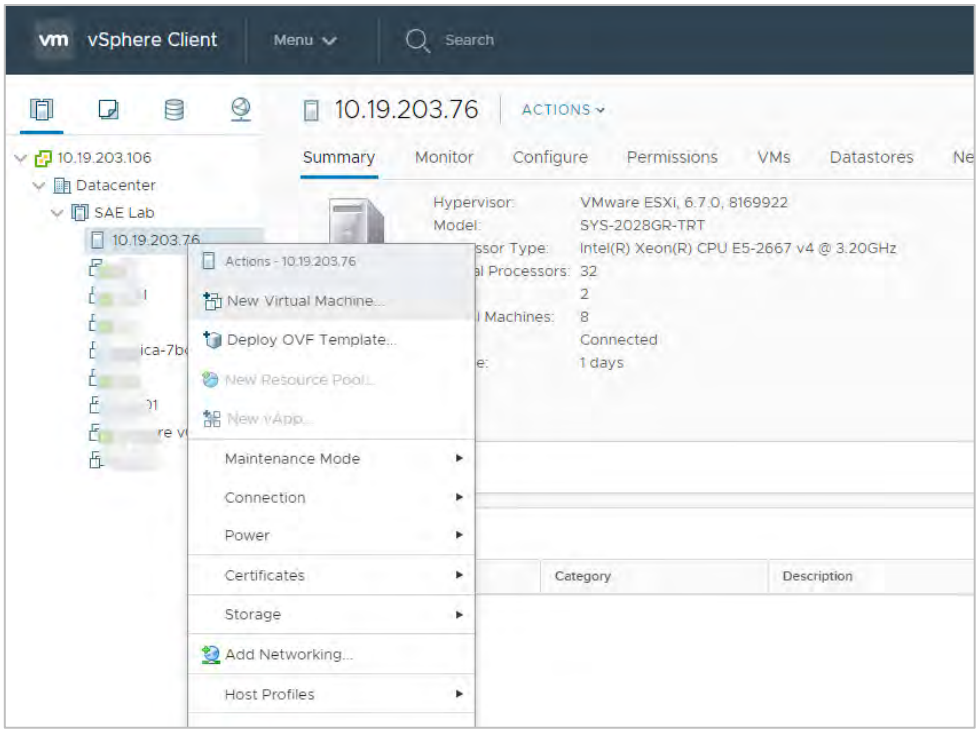
9.1 Creating a Virtual Machine

These instructions assist in making a VM from scratch to support NVIDIA vGPUs. Such a VM may be used as a gold master image to create additional VMs. Use the following procedure to configure a vGPU for a single guest desktop:

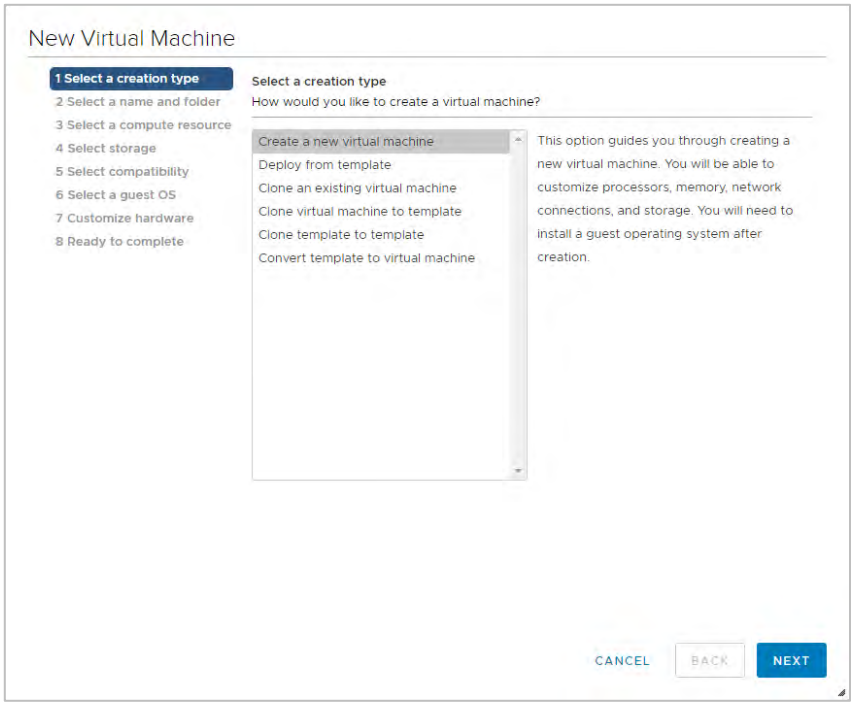
1. Open the vSphere Web Client:



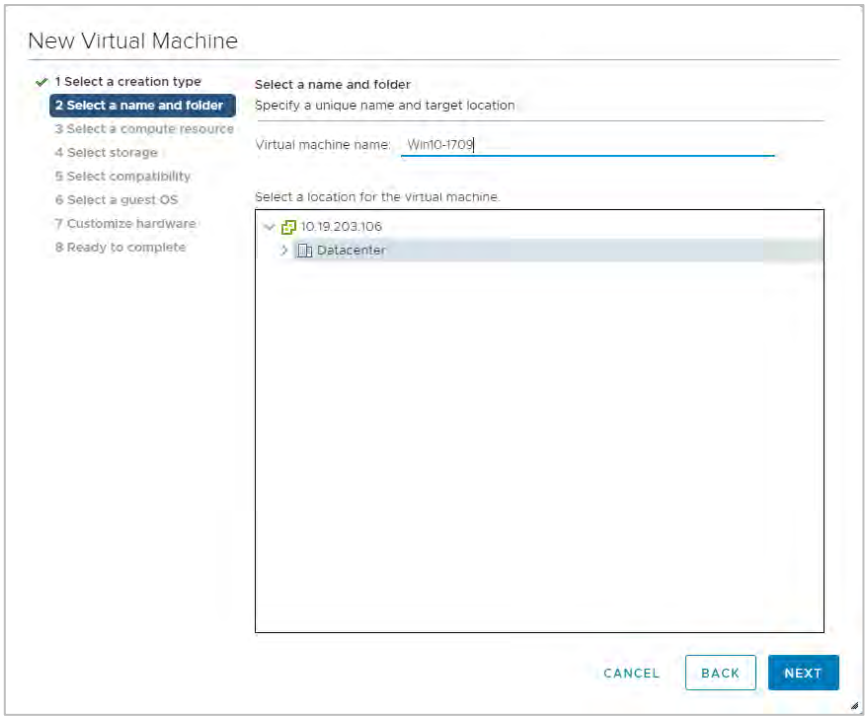
2. Browse to the host or cluster for which the VM is to be created. Right click the host or cluster and select New Virtual Machine:



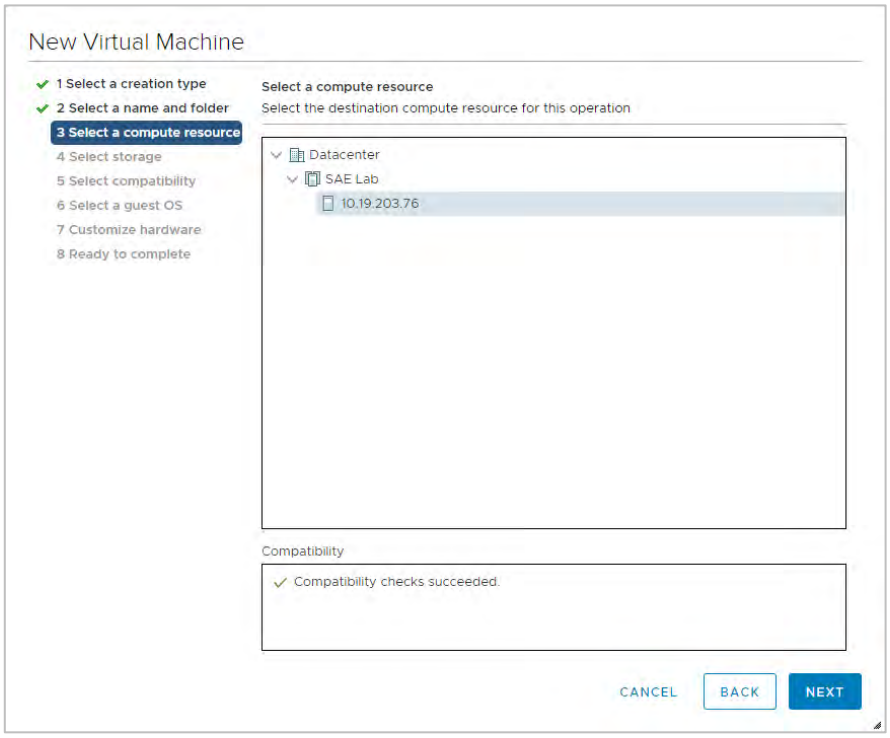
vSphere opens the New Virtual Machine window:



Click Next. vSphere prompts you to specify the new virtual machine's name and location:



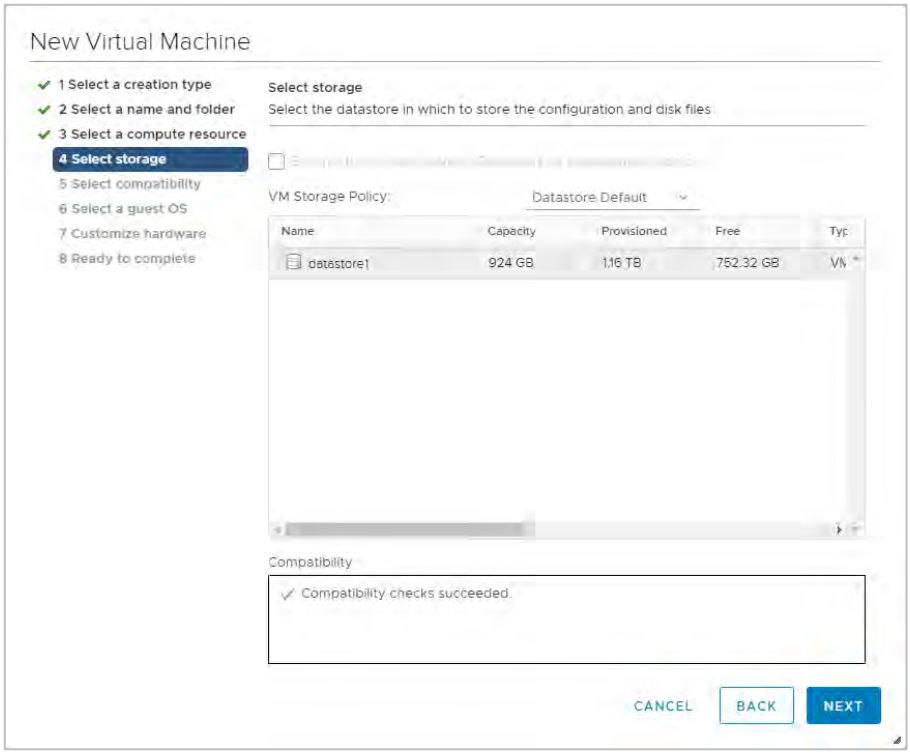
3. Enter a name for the virtual machine. Choose the location to host the virtual machine using the collapsible tree under the label “Select a location for the virtual machine,” then click Next to continue. vSphere prompts you to select a compute resource:



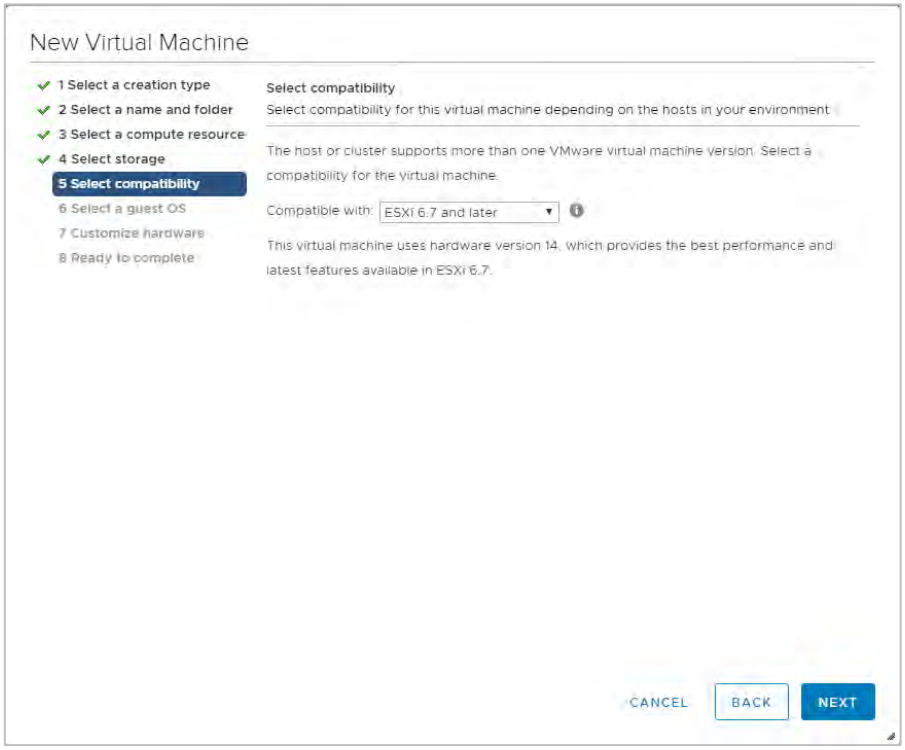
4. Select a compute resource to run the VM.

   The compute resource you select must include an installed, correctly configured NVIDIA adapter that supports vGPU operations.

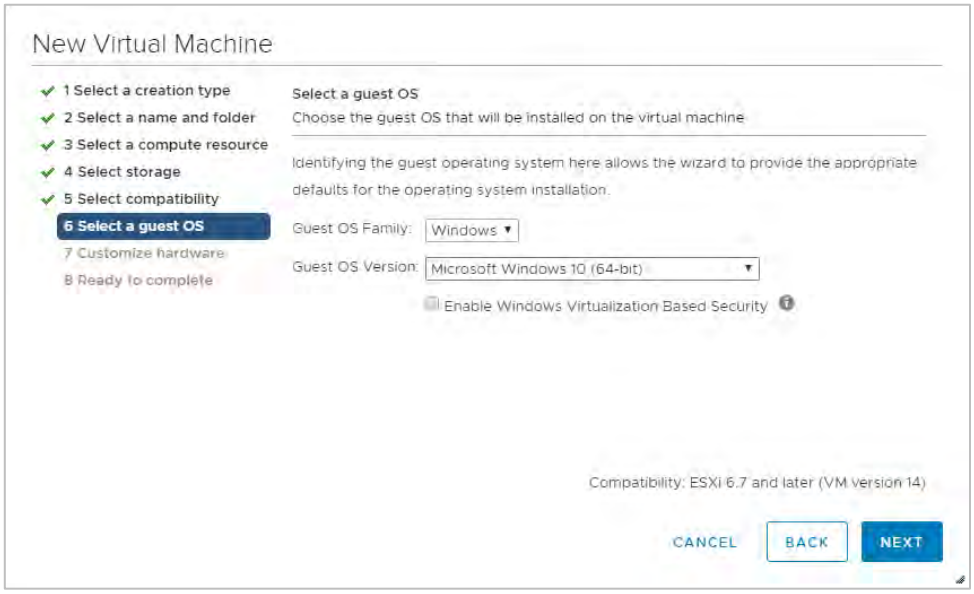
Click Next to continue. vSphere prompts you to select a storage resource:



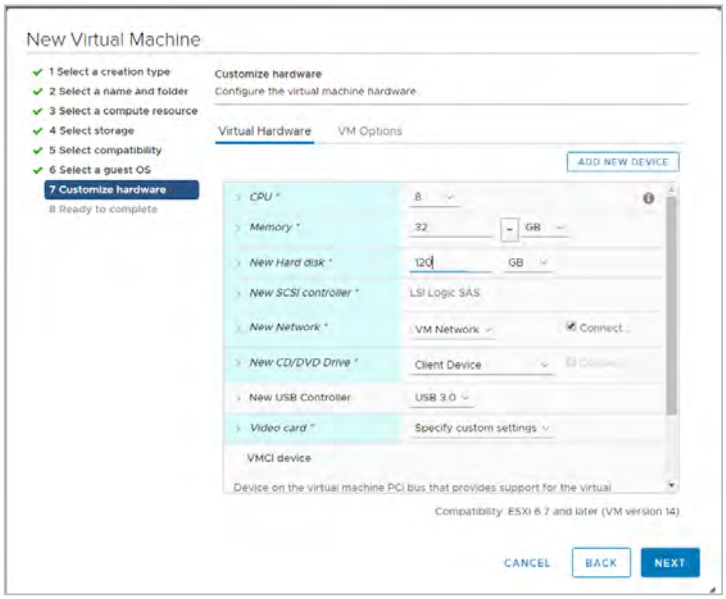
5. Select the datastore to host the virtual machine, then click Next to continue. vSphere prompts you to select compatibility for the vGPU:



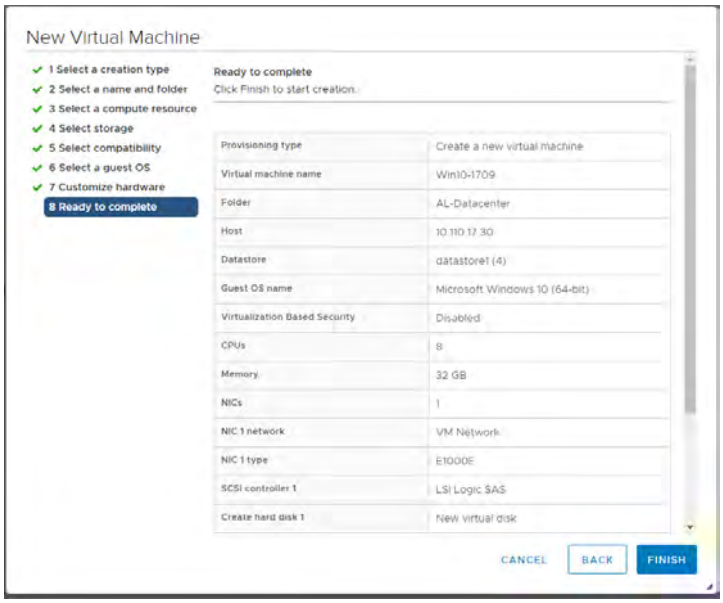
6. Compatibility settings allow the VM to run on different versions of vSphere. To run vGPUs, select `ESXi 6.0 and later`, then click Next to continue. vSphere prompts you to select a guest OS:



7. Select the appropriate Windows OS from the Guest OS Family and Guest OS Version dropdowns, then click Next to continue. vSphere prompts you to customize the vGPU's virtual hardware:



8. Set the virtual hardware based on your desktop workload requirements, then click Next to continue. vSphere displays a summary of the settings you have selected:

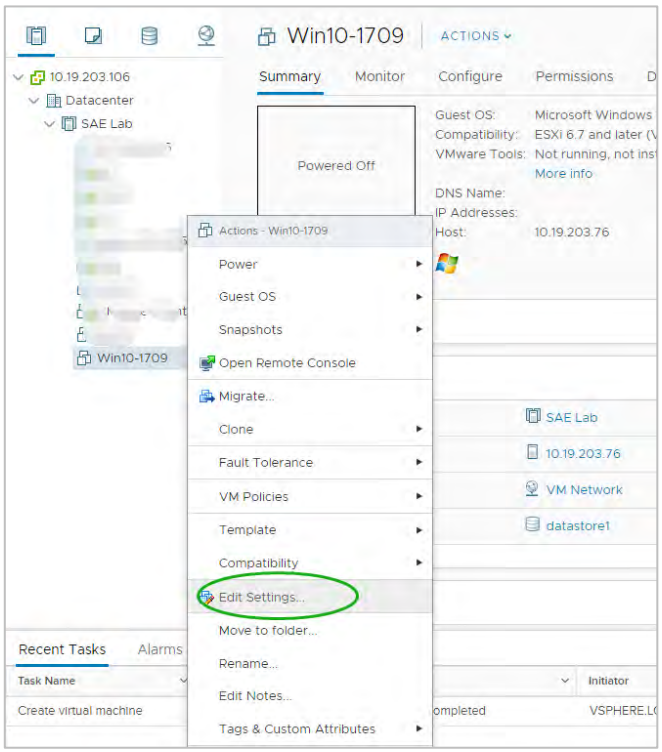


9. Review the new virtual machine's configuration. If any of the settings are wrong, click Back and correct them. When all of the settings are right, click Finish.

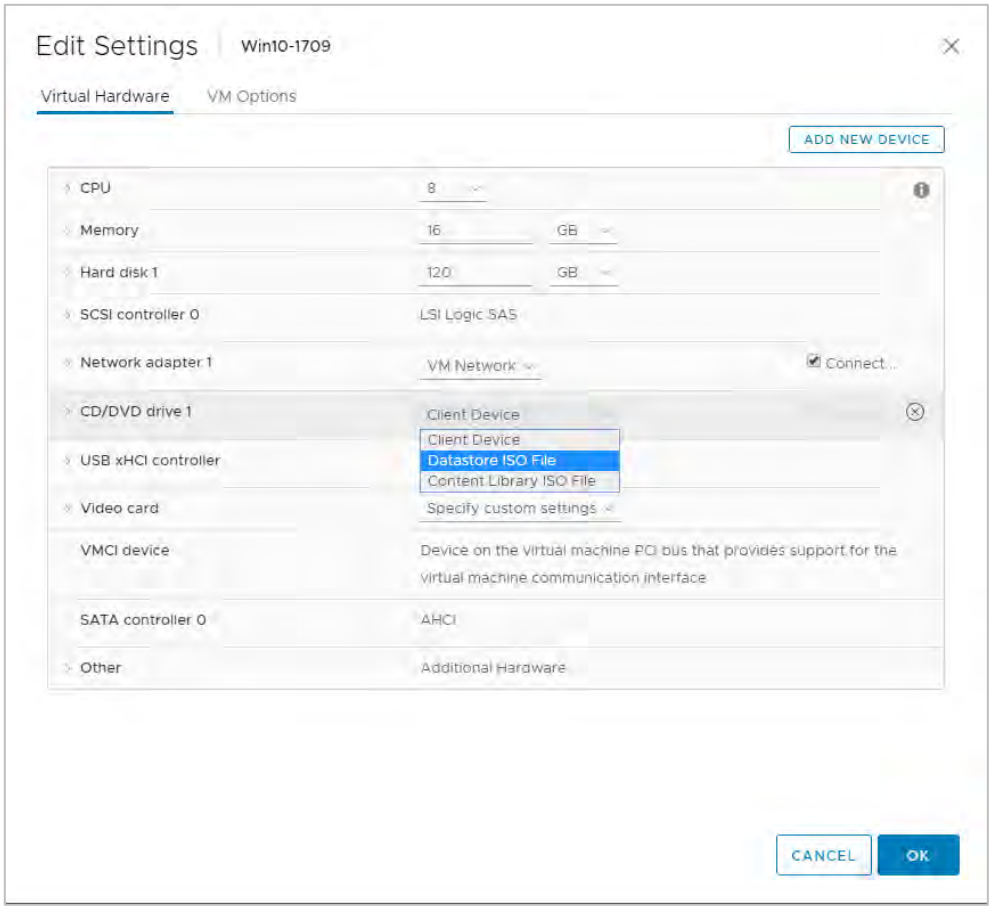
9.2 Installing Microsoft Windows

To install Microsoft Windows on the virtual machine:

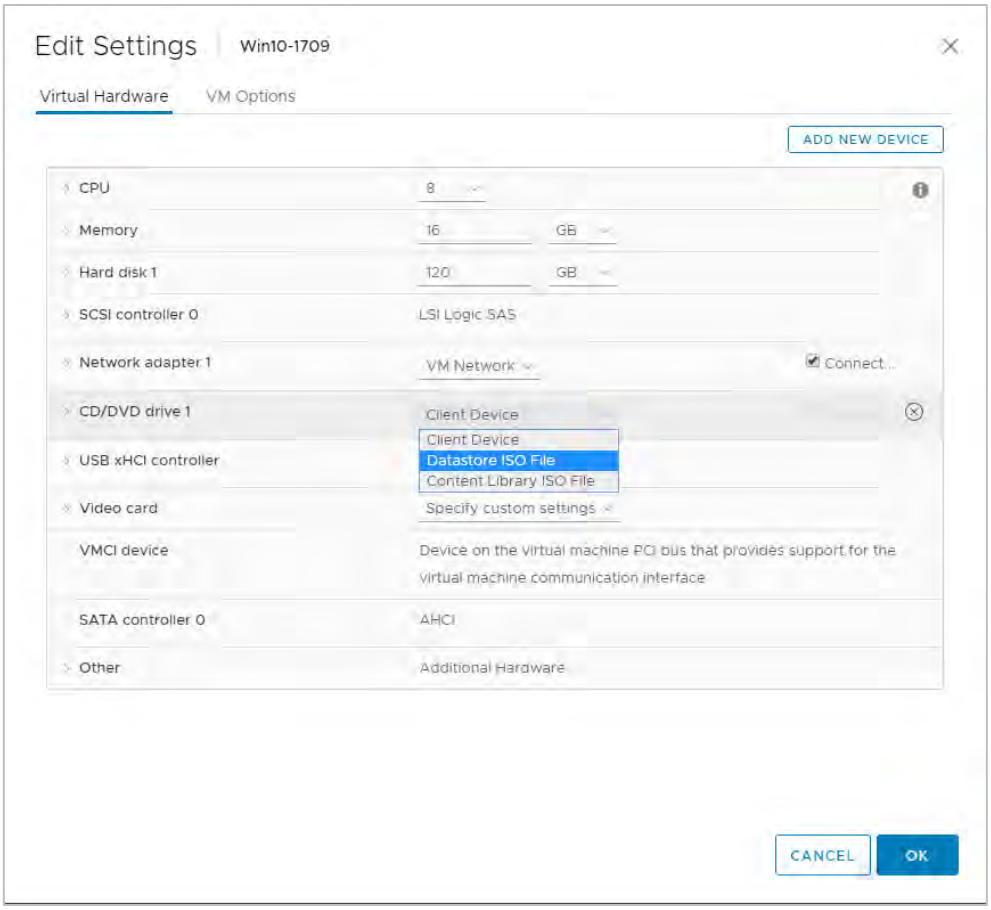
1. Select the virtual machine, right click it, and select Edit Settings:



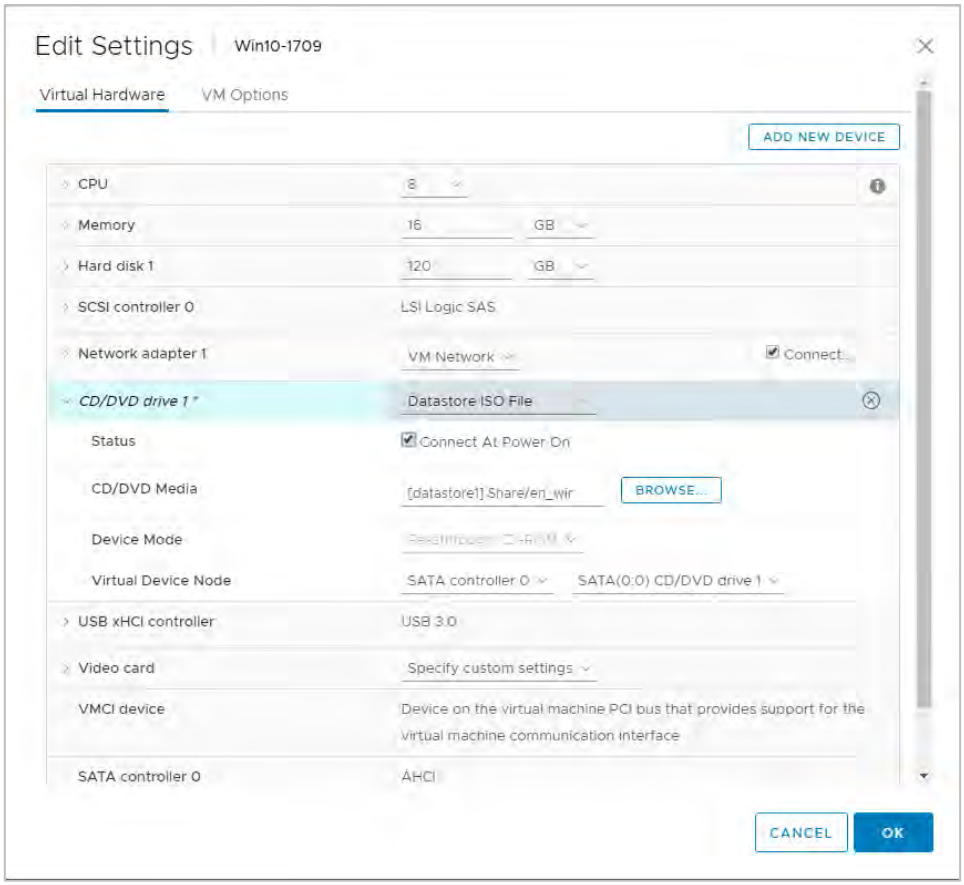
vSphere displays the Edit Settings dialog:



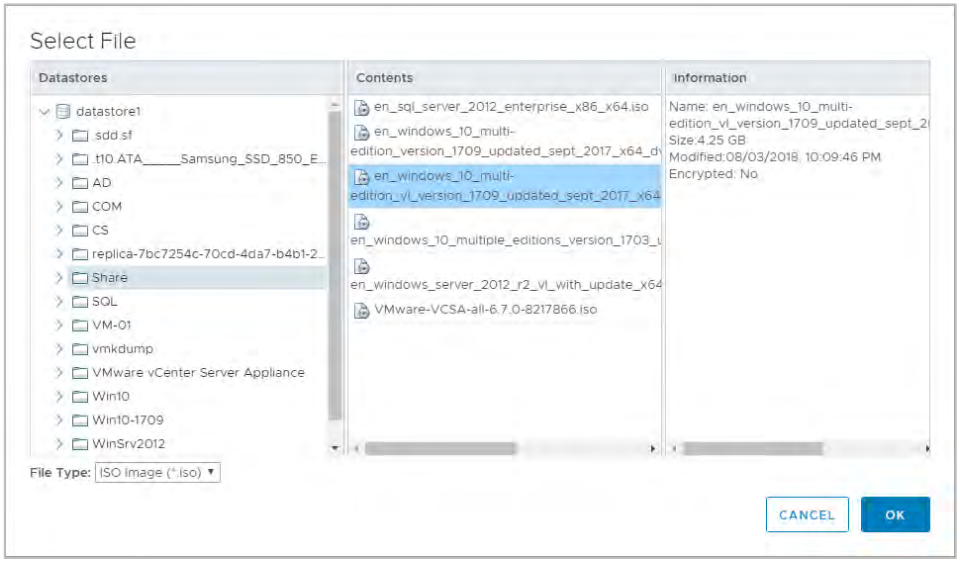
2. Click the dropdown list opposite the label “CD/DVD drive 1” and select an appropriate data source for CD/DVD media. (This example uses a Datastore ISO file.) Check the Connect checkbox to the right of the CD/DVD drive 1 dropdown to connect the ISO file to the VM’s virtual CD/DVD drive:



3. Click the caret next to “CD/DVD drive 1.” vSphere expands the tree entry to reveal the details of the virtual device.



4. Check the Connect At Power On checkbox to connect the ISO file to the VM's virtual CD/DVD drive at boot time, then click the Browse button to the right of "CD/DVD Media." vSphere opens a Select File dialog:



5. Navigate to and select the OS ISO file to be used for installation. Click OK to select the file.
6. Right click the virtual machine, then select Power>Power On to start the virtual machine, boot the ISO file, and install the operating system from it.

The virtual machine boots from the selected ISO file.

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

7. Perform a Custom (fresh) installation of Microsoft Windows 10 on the virtual machine.

   During installation, Windows reboots the VM several times.

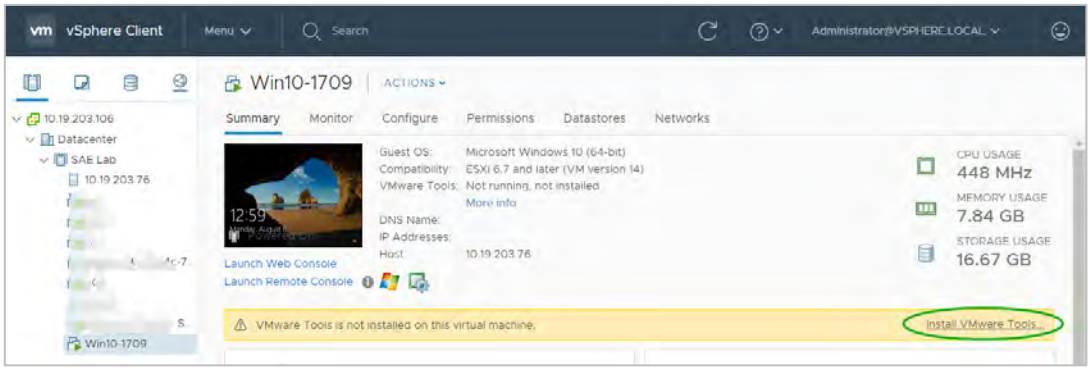
8. When Windows is done installing, disconnect the ISO from the VM.
9. Go through the initial Windows setup wizard to name the computer, create a local account, set the time zone, choose the update installation policy, etc.

Windows 10 is now installed on the virtual machine.

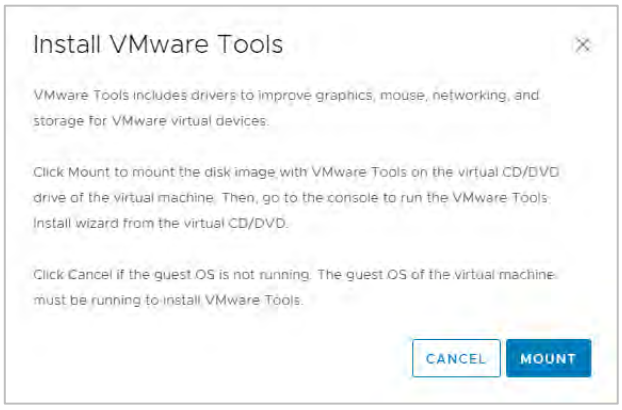
9.3 Installing VMware Tools

After you install Microsoft Windows on the virtual machine (VM), the next step is to install VMware Tools.

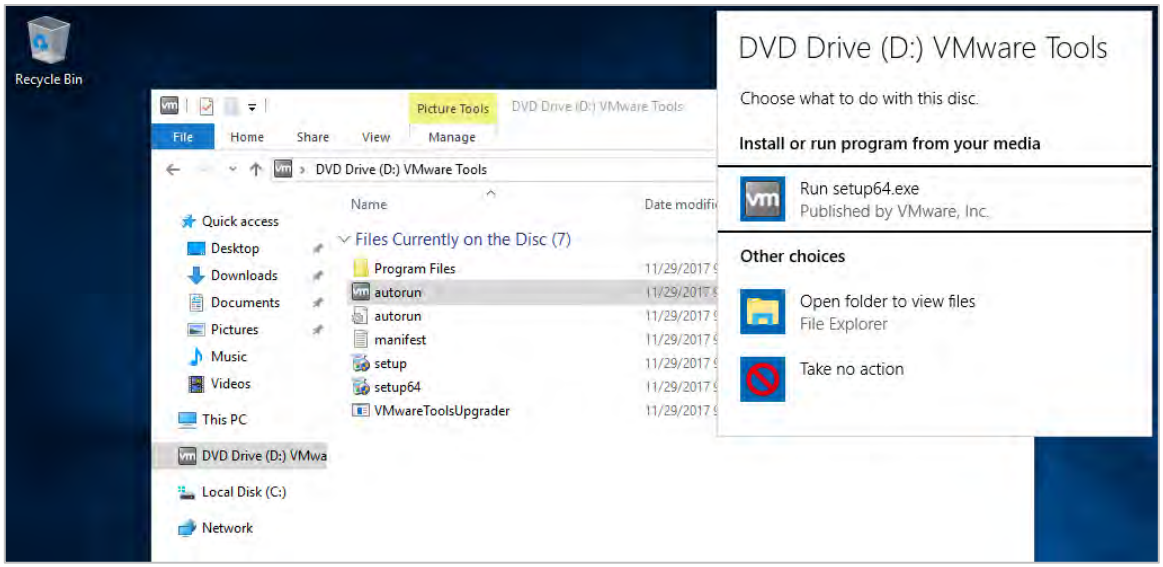
1. Select the Summary tab from the VM console.
2. Click the Install VMware Tools link in the yellow bar. The VM displays the Install VMware Tools window:



3. Click Install VMware Tools. The VM displays the Install VMware Tools window:

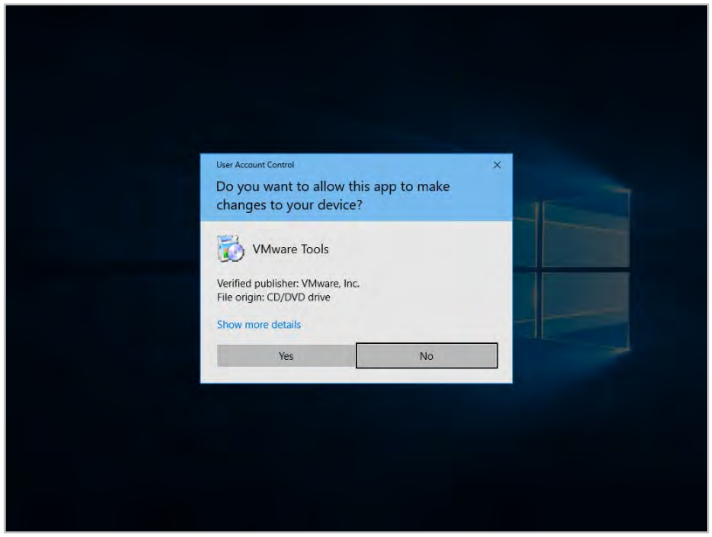


4. Click Mount. The VM displays an Open dialog:

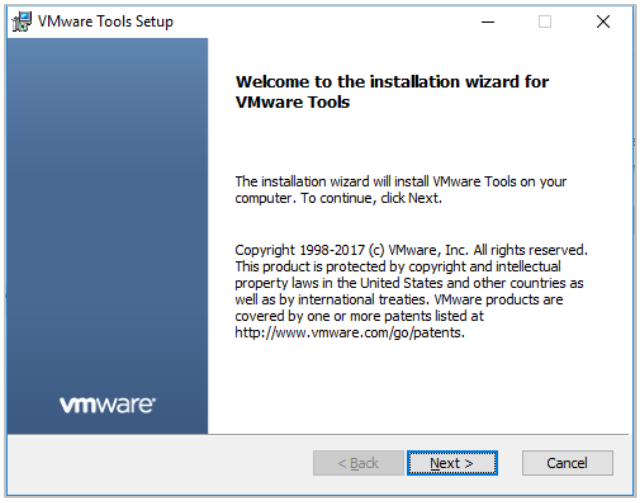


5. Back on the VM console, Windows 10 detects the mounted CD image and opens the AutoPlay dialog. If the AutoPlay dialog does not open, browse to the virtual CD-ROM in the VM and start the installer (`setup.exe` or `setup64.exe`) manually.

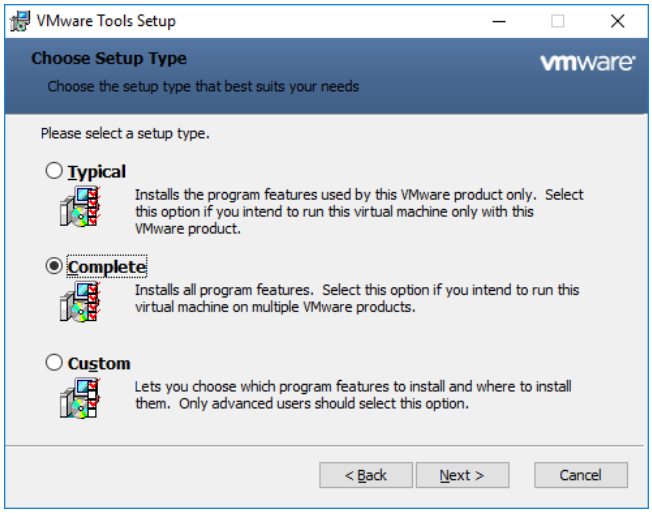
   Windows asks whether you want to allow the installer to make changes:



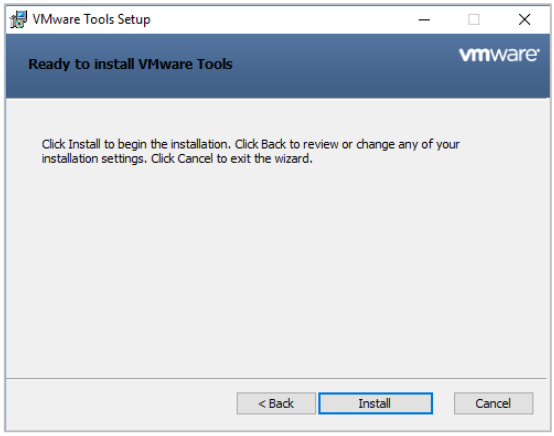
6. Click Yes to continue. If Windows displays the User Account Control (UAC) dialog, click Yes.
7. The installer displays the “Welcome…” screen:



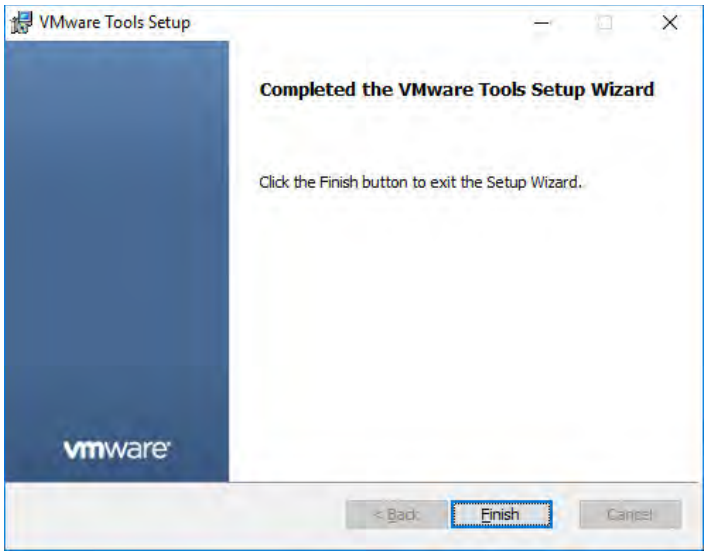
8. To begin the installation, click Next.
9. When the installer prompts you to select a setup type, select Complete, then click Next:



10. In the next screen, accept all defaults, then click Next.
11. When the installer displays the Ready to install screen, click Install to begin installation:



12. When the installation is complete, the installer displays the Completed screen:



13. Click Finish to close the VMware Tools installer. Reboot the virtual machine when prompted. This reboot is critical to ensure the tools are will load properly.

VMware Tools is now installed on the virtual machine.

9.4 Adding the Golden Master to the Domain

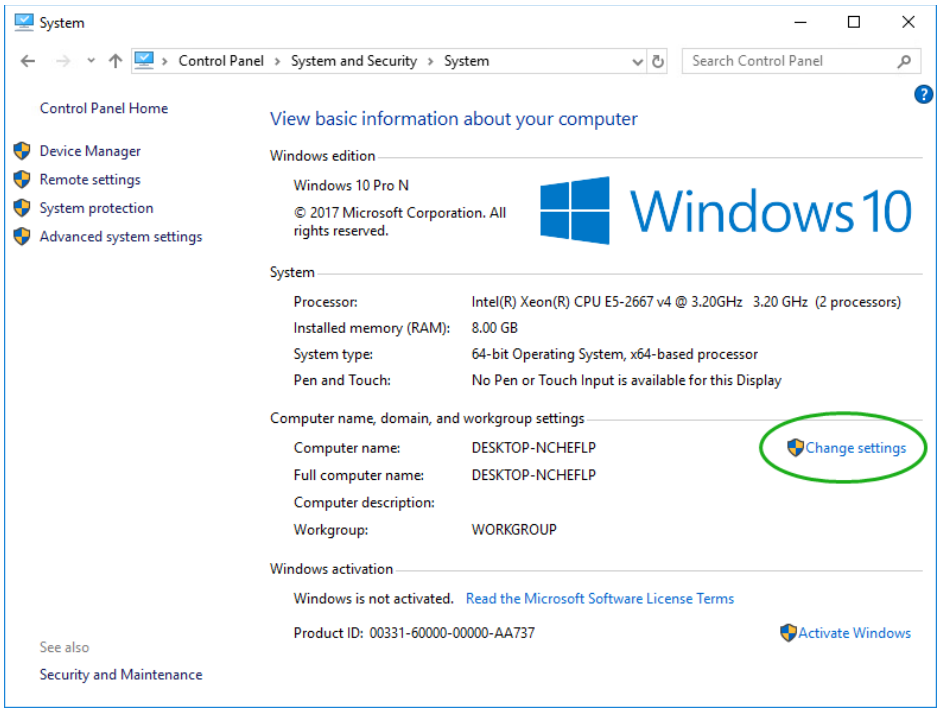
If you join a VM to a Windows Active Directory domain, you can manage it as you would any physical desktop in the domain.

Customize Windows on the virtual machine as follows:

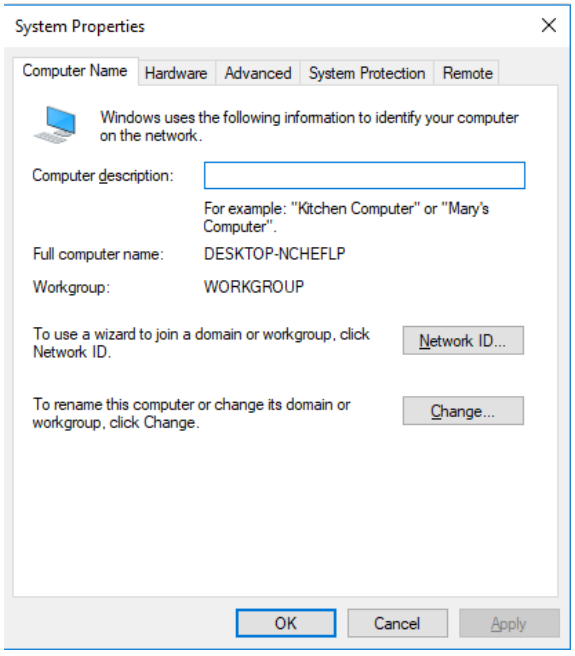
- Join the domain
- Add appropriate Domain groups to Local Administrators

Adding a VM to the domain:

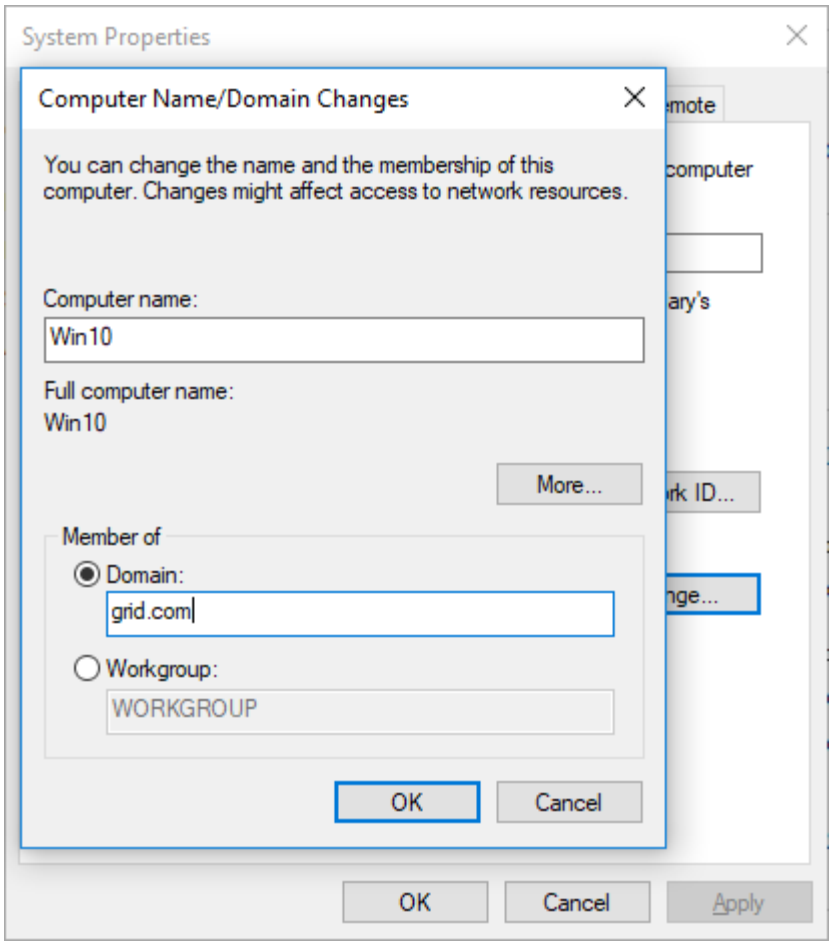
1. On the VM's Microsoft Windows, select Control Panel > System. Windows displays this dialog:



2. In the left-hand panel, click Advanced system settings. Windows displays the System Properties dialog:

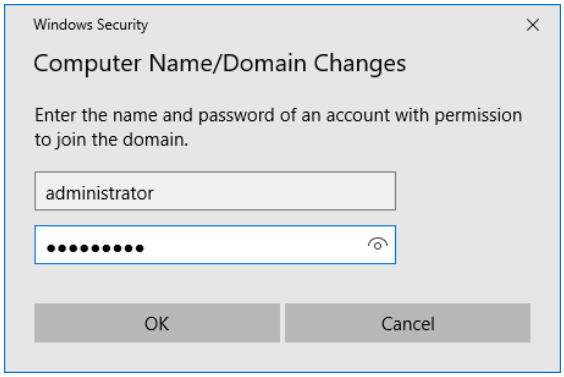


3. Select the Computer Name tab, then click Change. Windows displays the Computer Name/Domain Changes dialog:

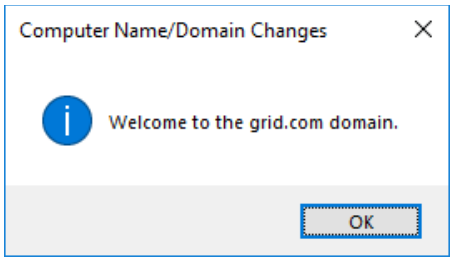


4. Enter an appropriate name in the Computer name field. Select Domain and enter an appropriate name in the Domain name field. (The names in the illustration above are examples, and are not necessarily appropriate for your VM.)

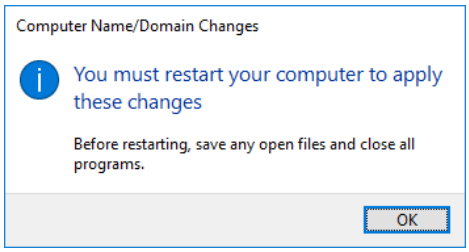
   Click OK. Windows opens a Windows Security dialog:



5. Enter your domain administrator credentials, then click OK. If the credentials are valid, Windows sets the domain name as you have specified it and opens a “Welcome…” dialog:



6. Click OK to close the dialog. Windows displays a “You must restart your computer…” dialog:



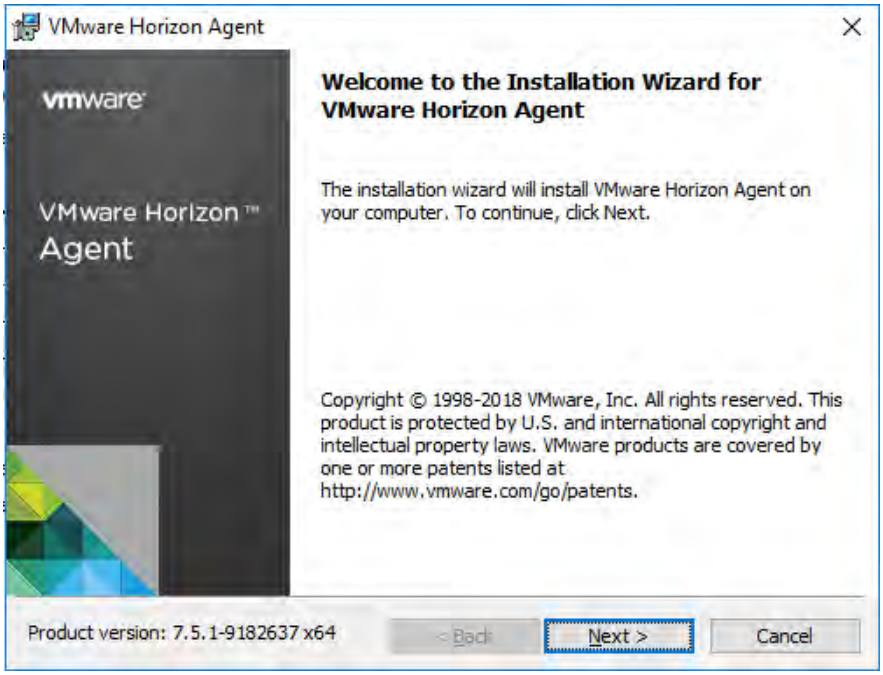
7. Click OK. The VM reboots.

9.5 Installing Horizon Agent

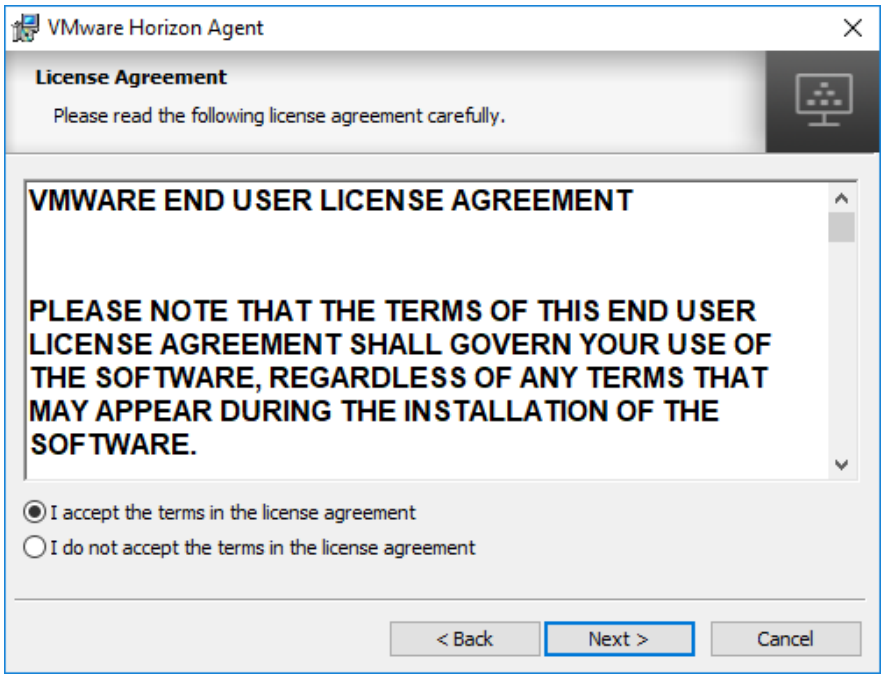
You must install the correct version of the Horizon Agent for your virtual machine.

To install Horizon Agent on your VM:

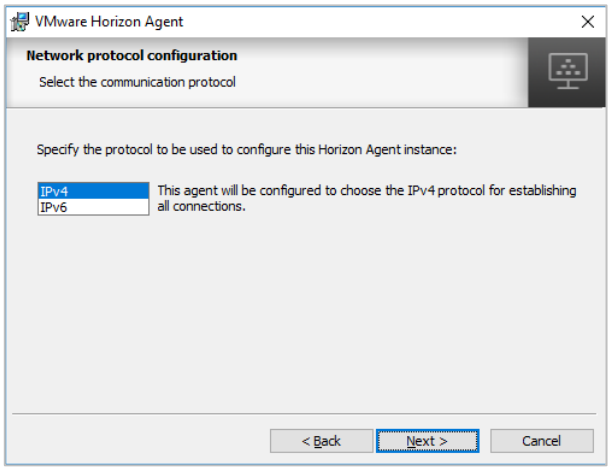
1. Several versions of the installer are provided with the Horizon 6.1 Early Access code. Find the correct version for your VM:
   - For 64-bit virtual machines, install `VMware-viewagent-x86_64-7.5.1-9182637.exe`.
2. Launch the installer. It displays a “Welcome…” screen:



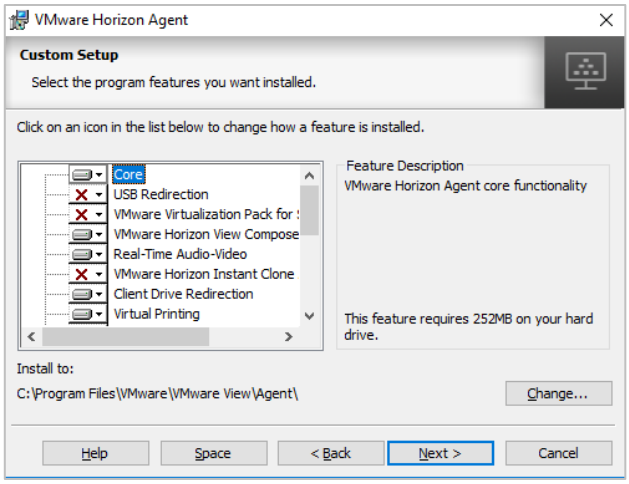
3. Click Next to continue. The installer displays the License Agreement screen:



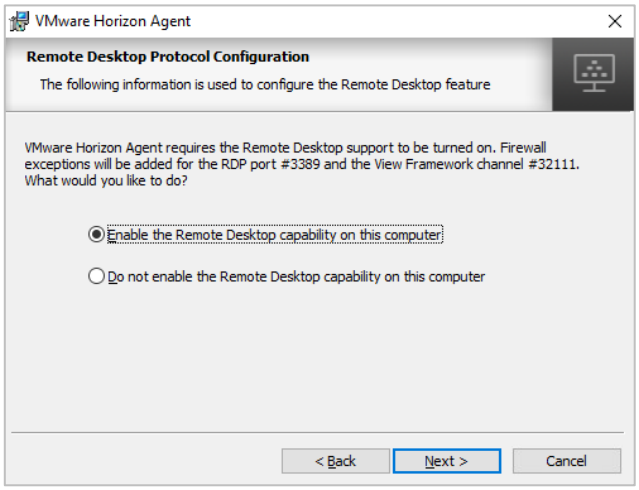
4. Select the “I accept the terms…” radio button, then click Next to continue. The installer displays the Network protocol configuration screen.



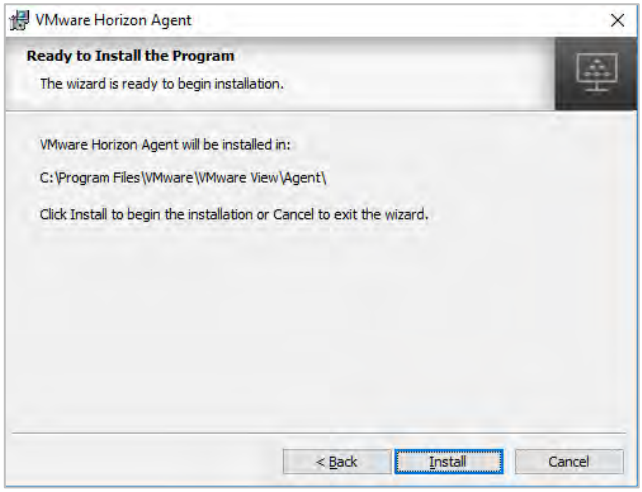
5. Choose the appropriate network protocol (typically IPv4), then click Next to continue. The installer displays the Custom Setup screen:



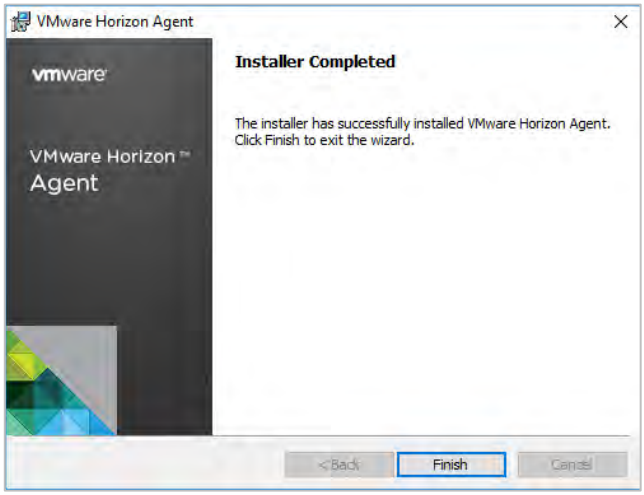
6. Accept all default values; click Next.
7. If the installer displays the Remote Desktop Protocol Configuration screen, select “Enable the Remote Desktop…,” then click Next:



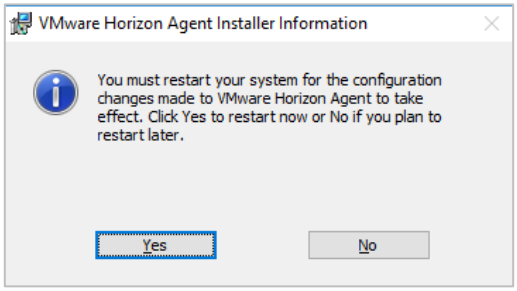
8. The installer displays the Ready to install the program screen:



9. Click Install to continue. The installer installs Horizon Agent on the VM. When it is done, it displays the Installer Completed screen:



10. Click Finish to close the installer. The installer prompts you to reboot:



11. Click Yes to reboot the VM.

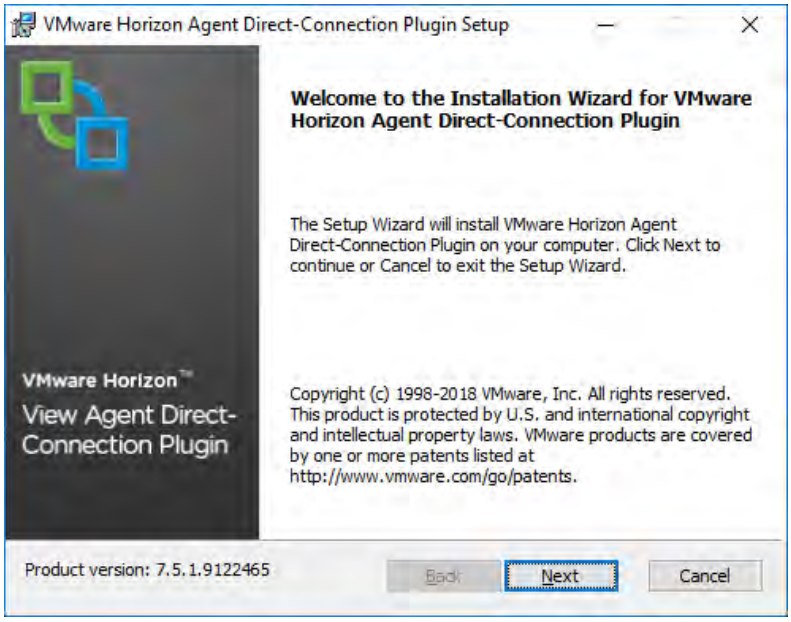
VMware Horizon Agent is now installed on the virtual machine.

9.6 Installing Horizon Direct Connection

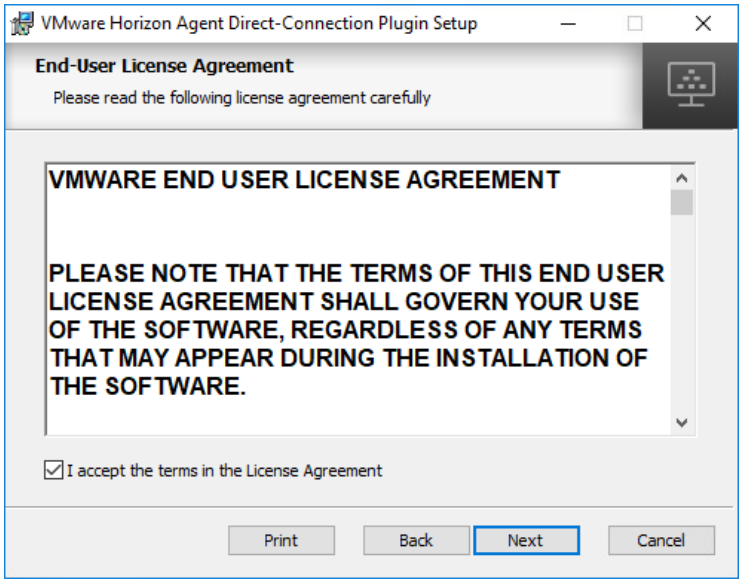
The Horizon Direct Connection agent is a helpful tool for debugging when you configure virtual machines with vGPU. Install the correct version for your VM.

To install Horizon Direct Connection agent on your VM:

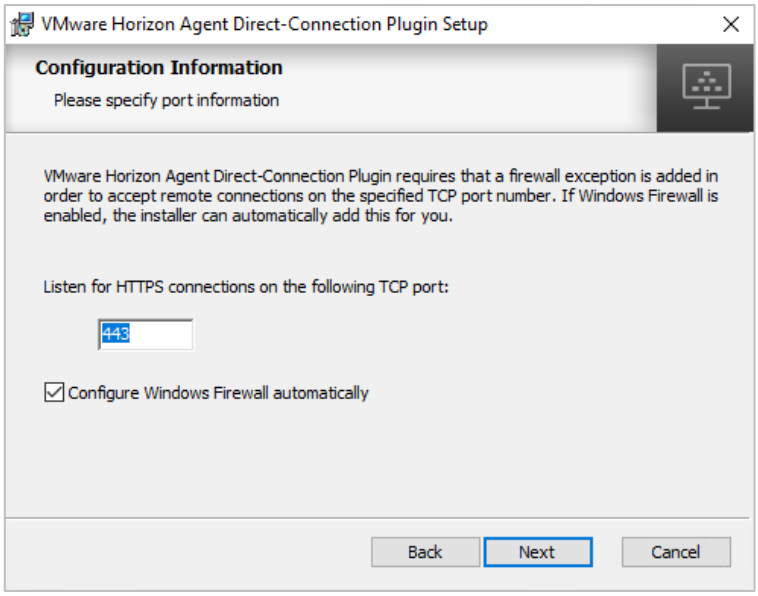
1. Several versions of the installer are provided with the Horizon 6.1 Early Access code. Find the correct version for your VM:
   - For 64-bit virtual machines, install `VMware-viewagent-direct-connection-x86_64-7.5.1-9122465.exe`.
2. Launch the installer. It displays a “Welcome…” screen.



3. Click Next to continue. The installer displays the End-User License Agreement screen:

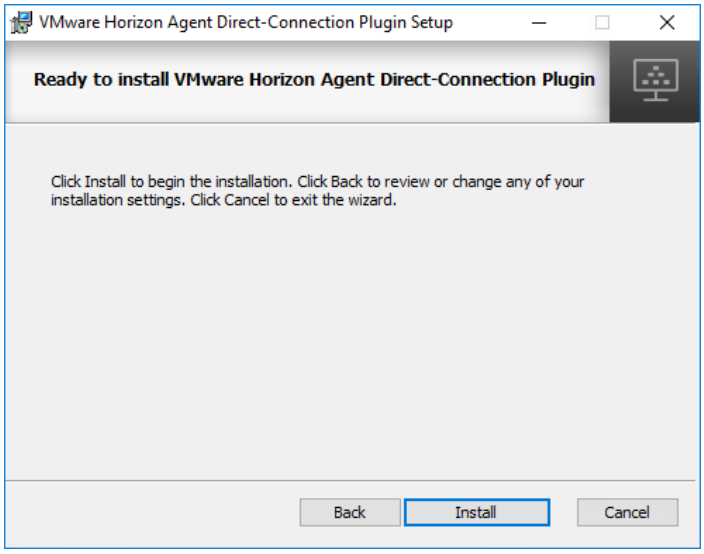


4. Check the I accept the terms in the license agreement checkbox, then click Next to continue. The installer displays the Configuration Information screen:

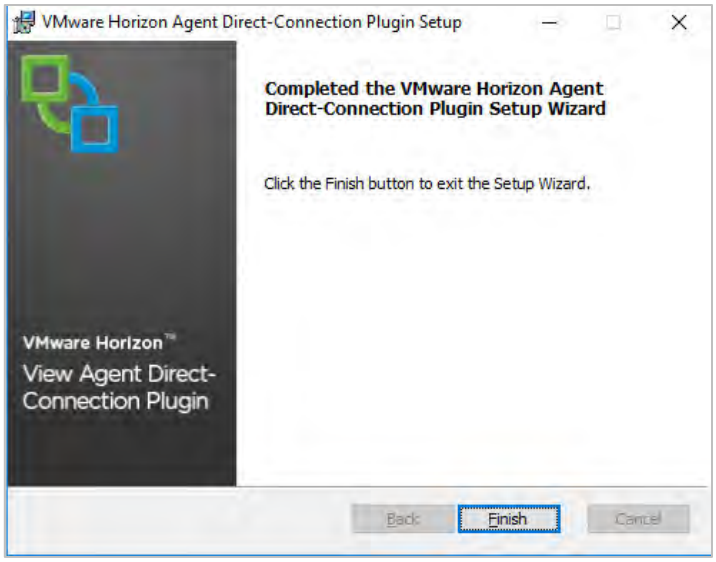


5. Enter an appropriate port number in the Listen for HTTPS connections field. The default HTTPS port number is 443; however, you may use a different port if you want.

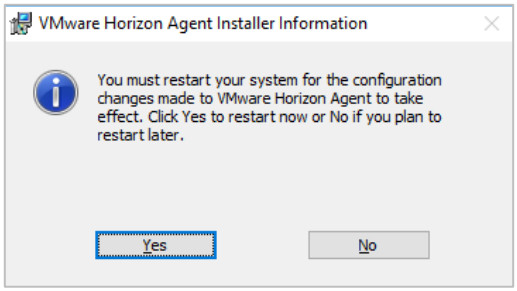
   Check the Configure Windows Firewall automatically checkbox. Then click Next to continue. The installer displays the Ready to install the program screen:



6. Click Install to continue. The installer begins performing the installation. When it is done, it displays the “Completed…” screen:



7. Click Finish. The installer prompts you to reboot:



8. Click Yes to reboot the VM.
9. When Windows is fully booted, test Direct Connection using the installed VMware Horizon client by adding a server, entering the IP address of the VM, and logging in to the VM with the appropriate credentials when you are prompted to do so.

Horizon Direct Connection is now installed on the virtual machine.

9.7 Optimizing Windows

The VMware OS Optimization Tool helps optimize Microsoft 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 across multiple systems, per VMware recommendations and best practices. Since most Windows system services are enabled by default, the optimization tool can easily be used to disable unnecessary services and enable features to improve performance.

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

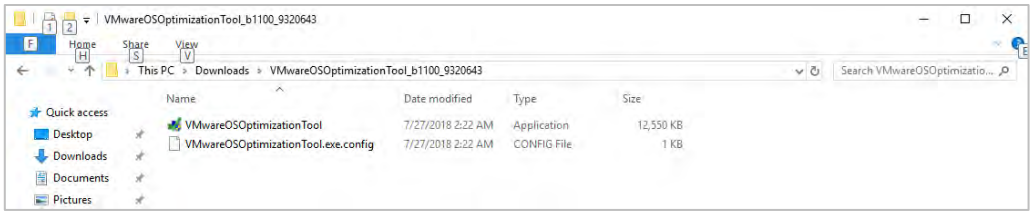
- Local analysis and optimization
- Remote analysis
- Optimization history recording and rollback
- Template management

Review the VMware Horizon Optimization Guide for Windows for OS-specific tweaks.

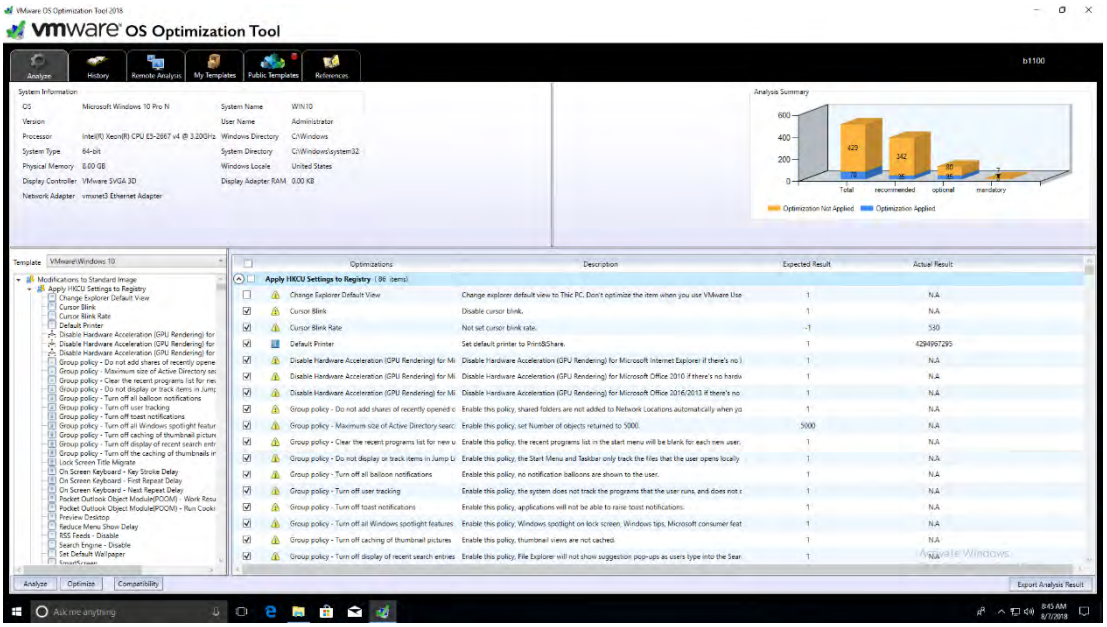
> Do not run the batch files recommended in this document without understanding the changes they make, as they may disable more functionality than anticipated.

To install the optimizer:

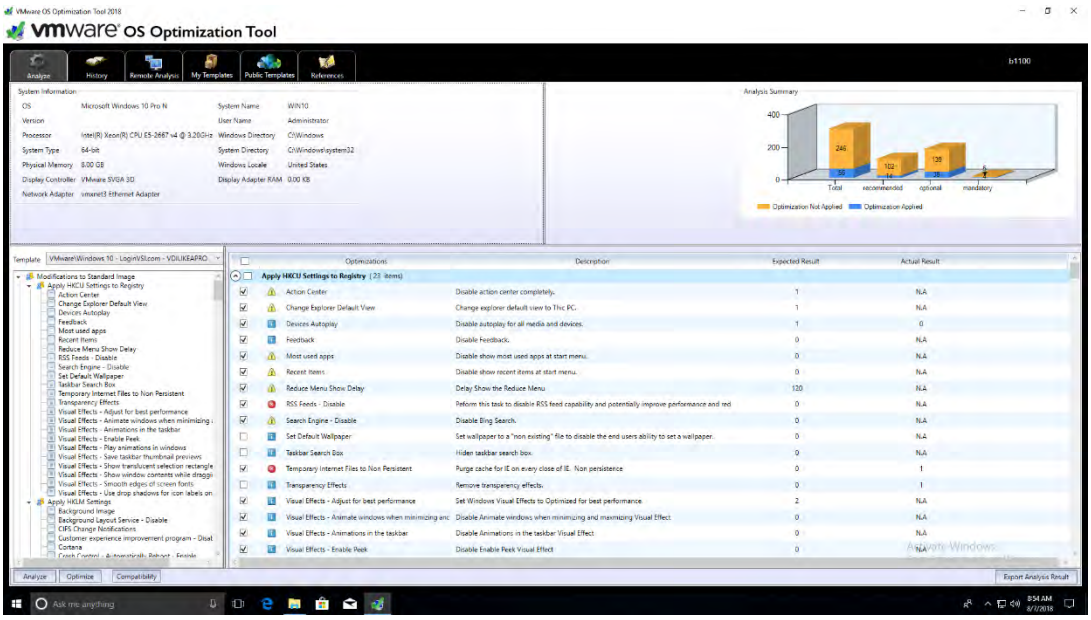
1. Download and browse to the folder containing the VMware OS Optimizer Tool, double click *VMwareOSOptimizerTool.exe* to run it.



2. The tool opens and displays the Analyze tab with System Information:

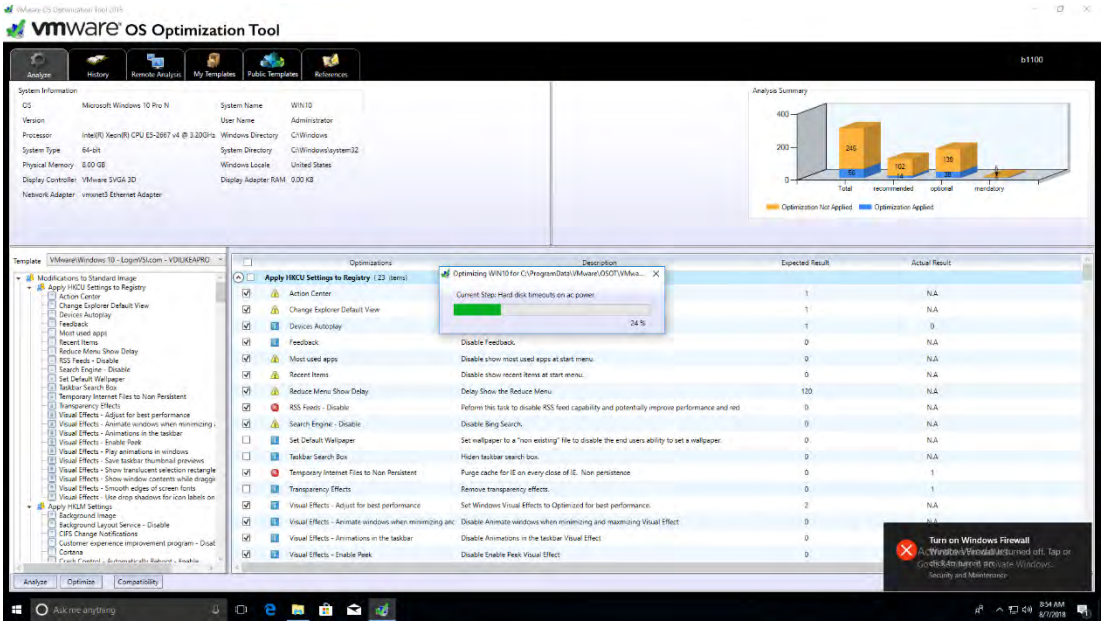


3. Click the Analyze button at the bottom left to begin.
4. Choose the template you want to use. These instructions use `Win10 - LoginVSI` as an example. You can also create your own template.
5. When the tool finishes analyzing the OS the main window shows the results, and displays an Optimize button at the bottom left:

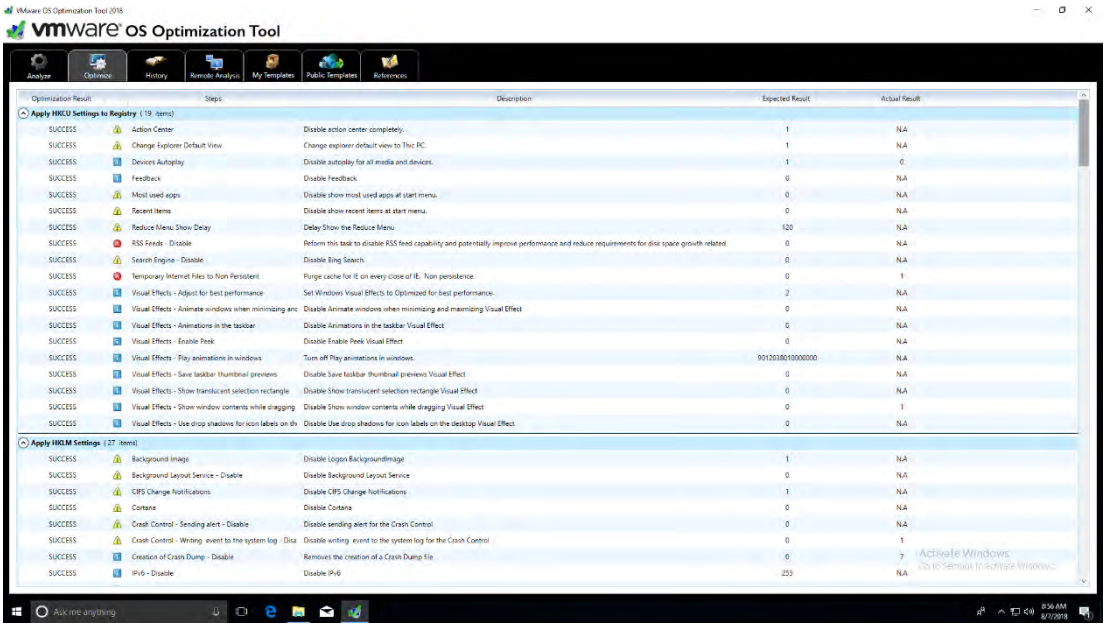


6. Scroll through the list of optimizations and select or deselect items depending on your needs. These instructions use the defaults.

   Click Optimize. The tool runs the selected scripts to optimize the OS:



7. When optimization is complete the tool displays its results:



9.8 Additional Virtual Machine Settings

Perform the following additional tasks on the virtual machine as required in preparation for configuring its vGPU:

1. Turn Off Windows Firewall for all network types.

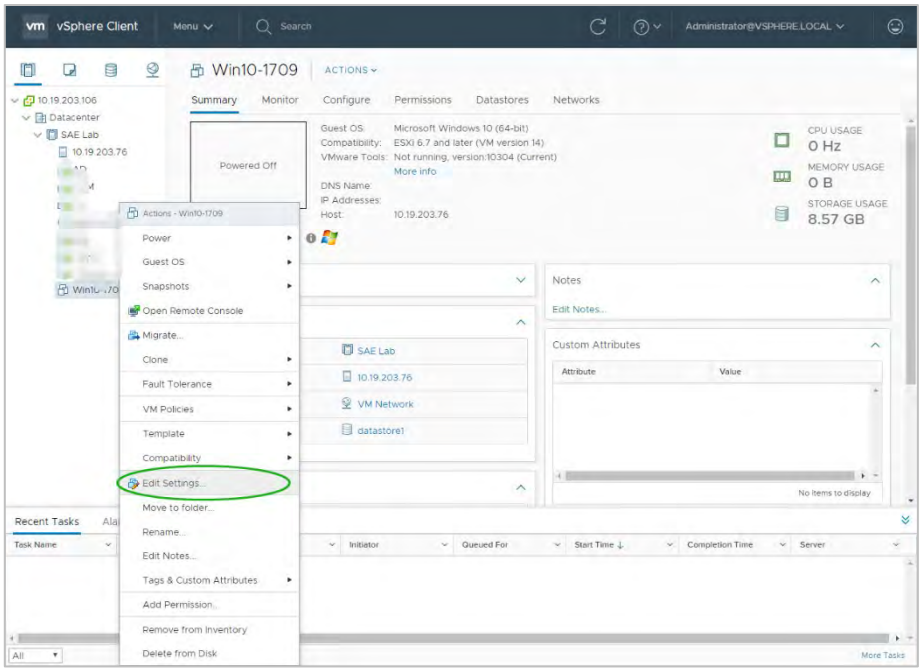
> CAUTION: These instructions assume that the VM is for proof-of-concept only and that disabling the firewall therefore poses only a minimal security breach. Always follow your established security procedures and best practices when setting up security for a production machine or for any environment that can be accessed from outside your network.

9.9 Enabling the NVIDIA vGPU

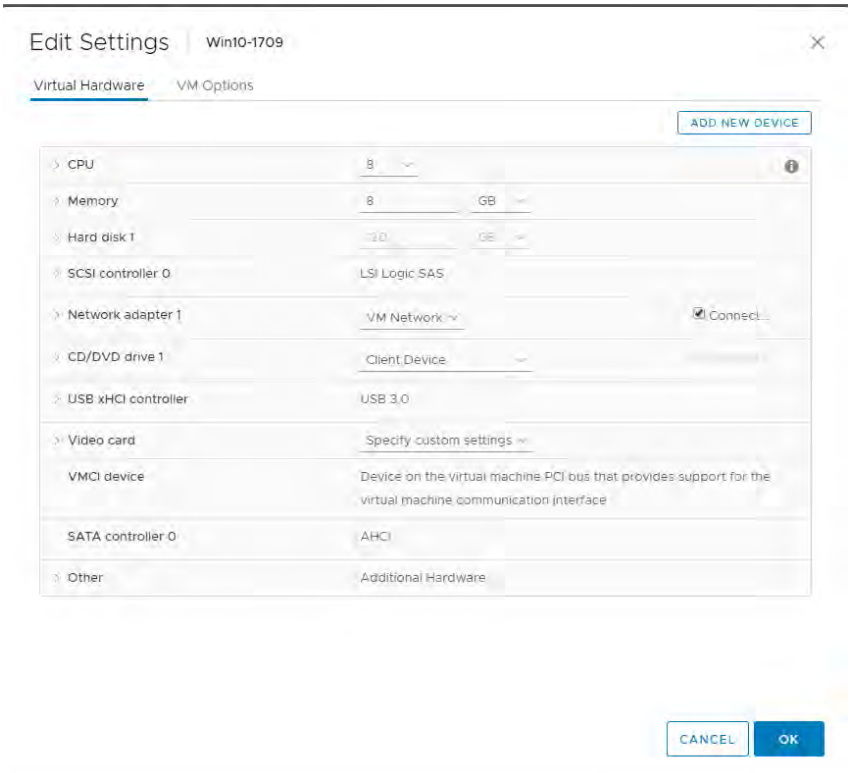
To enable vGPU support for your virtual machine, you must edit the virtual machine settings.

To enable vGPU support for your VM:

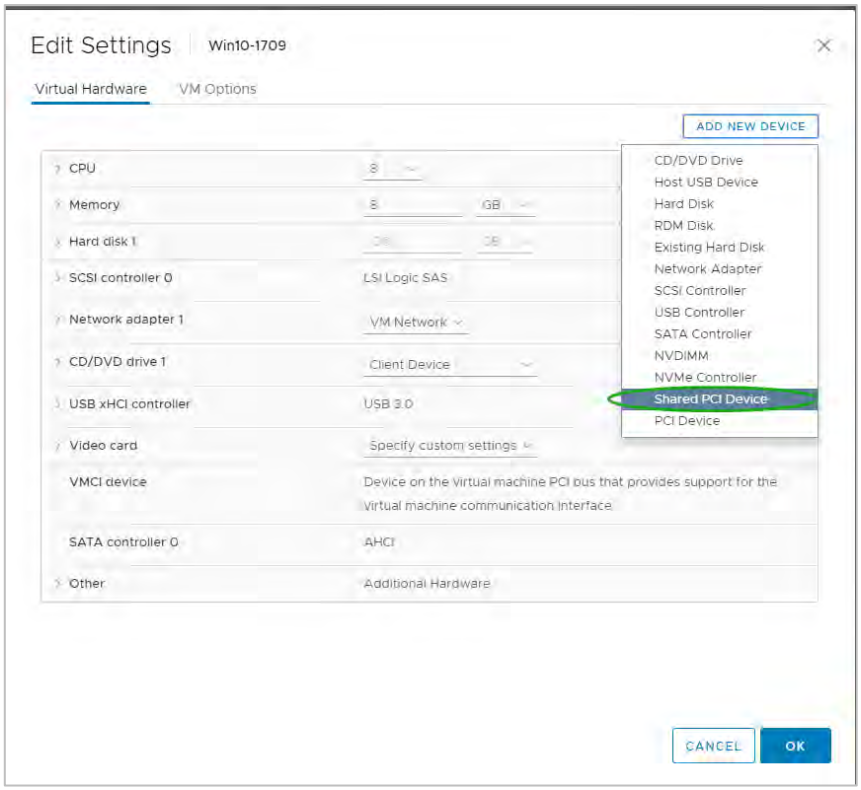
1. Power down the VM.
2. Click the VM in the inventory window. Right click the VM and select Edit Settings:



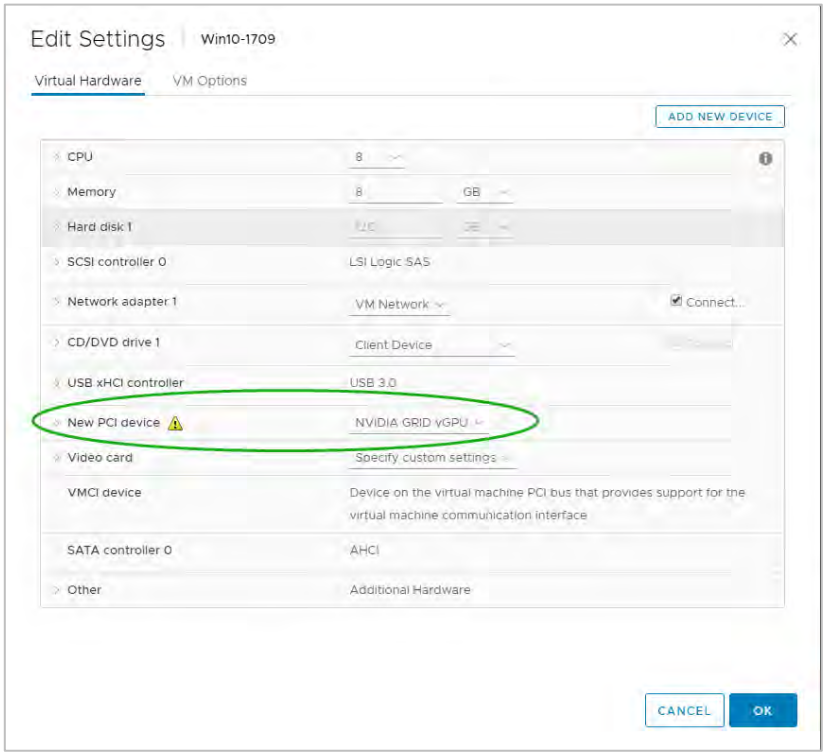
The Edit Settings dialog appears:



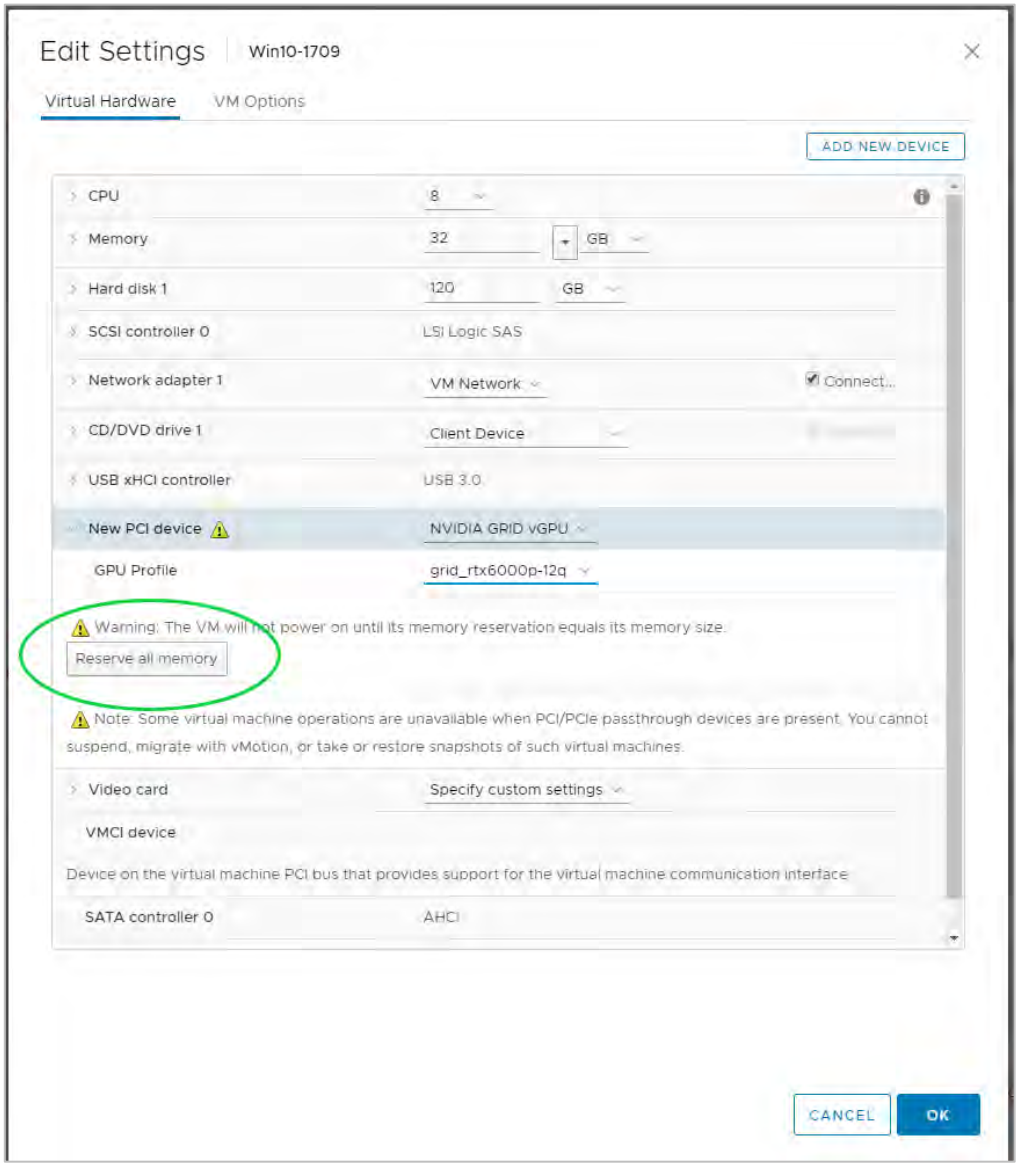
3. Click the Add New Device button. Navigation displays a dropdown menu of device types. Select Shared PCI device:



4. Select Shared PCI Device to continue. The new PCI device shows that an NVIDIA GRID® vGPU device has been added.



5. Expand the New PCI device, click the GPU Profile dropdown, and select the appropriate configuration. Then click the Reserve all memory button:



6. Click OK to complete the configuration.

9.10 Installing the NVIDIA vGPU Driver: Microsoft Windows

After you create a Microsoft Windows virtual machine on the hypervisor and boot the VM, you must install the NVIDIA vGPU software display driver to fully enable GPU operation.

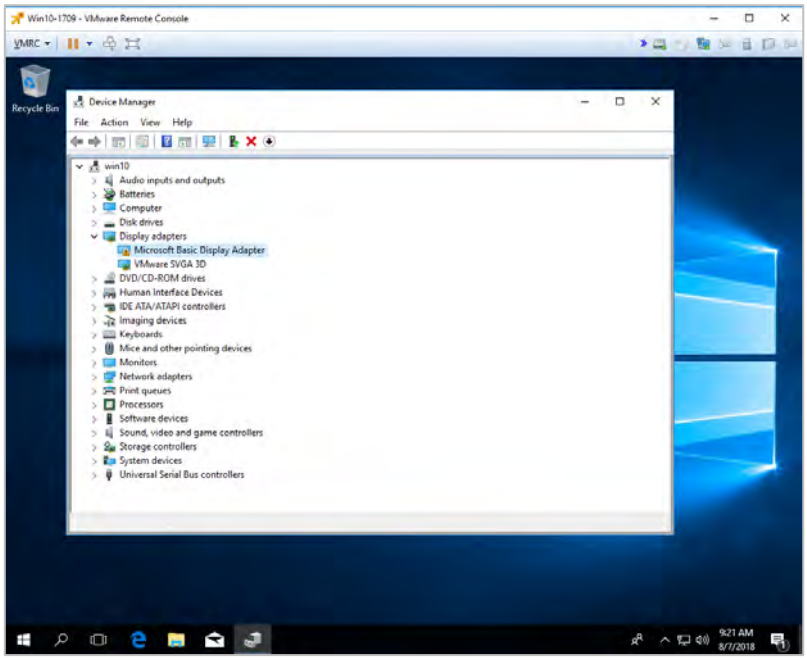
To install the NVIDIA driver in Microsoft Windows:

1. Start the virtual machine, then connect to it using either VMware Remote Console through the vSphere Web Client or VMware Horizon Client (via Direct Connection).

The first time you boot the VM after you enable a GRID vGPU, it displays a dialog warning requesting that you restart the computer to apply changes. Click Restart Later to continue booting the VM.

CAUTION: Do not reboot the VM if older NVIDIA drivers are installed. Doing so would produce a blue screen.

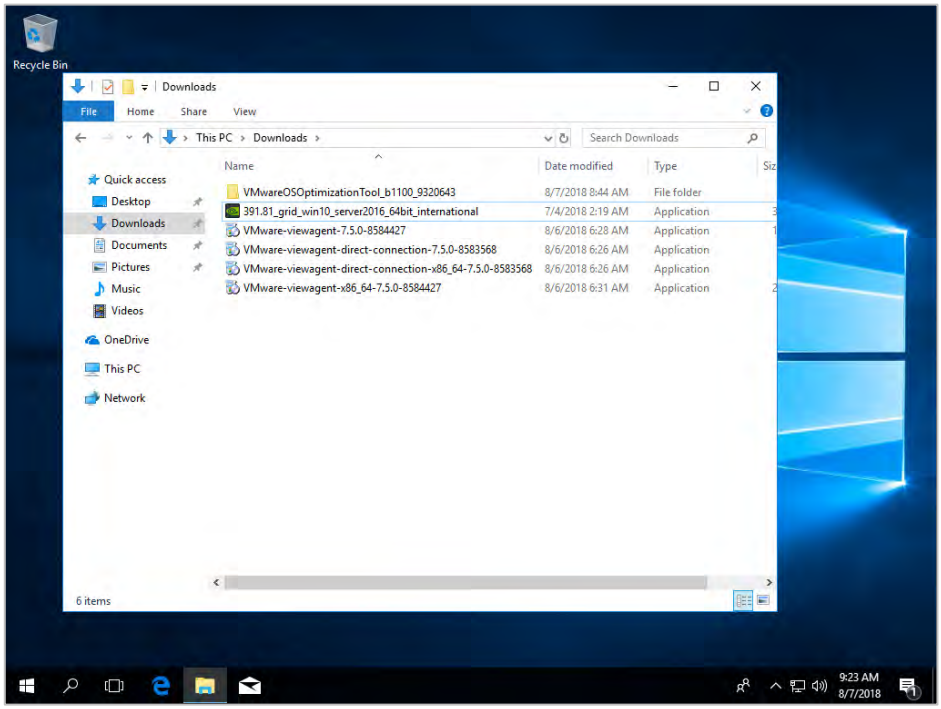
2. Log in to Windows and open Device Manager. Device Manager opens its window:



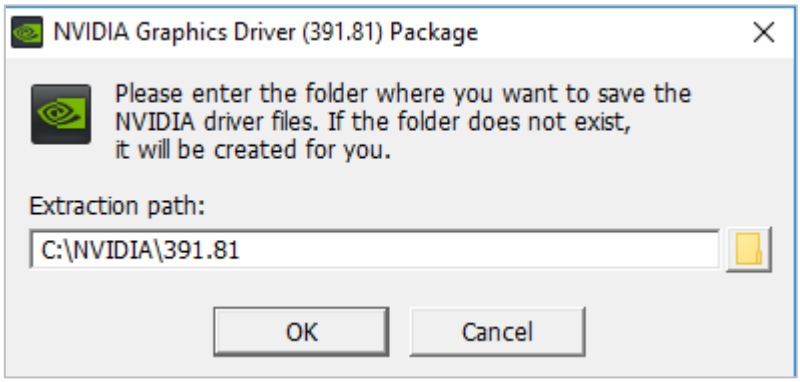
The window's "Display adapters" section shows a "Microsoft Basic Display Adapter" with an exclamation point on its icon to indicate a driver problem. This is normal.

3. Locate the NVIDIA driver and double click its Setup icon to launch it.

NVIDIA recommends that the installer be on a share volume that the VM can mount, for quick access.



4. The installer prompts you to specify a directory to hold the driver files:

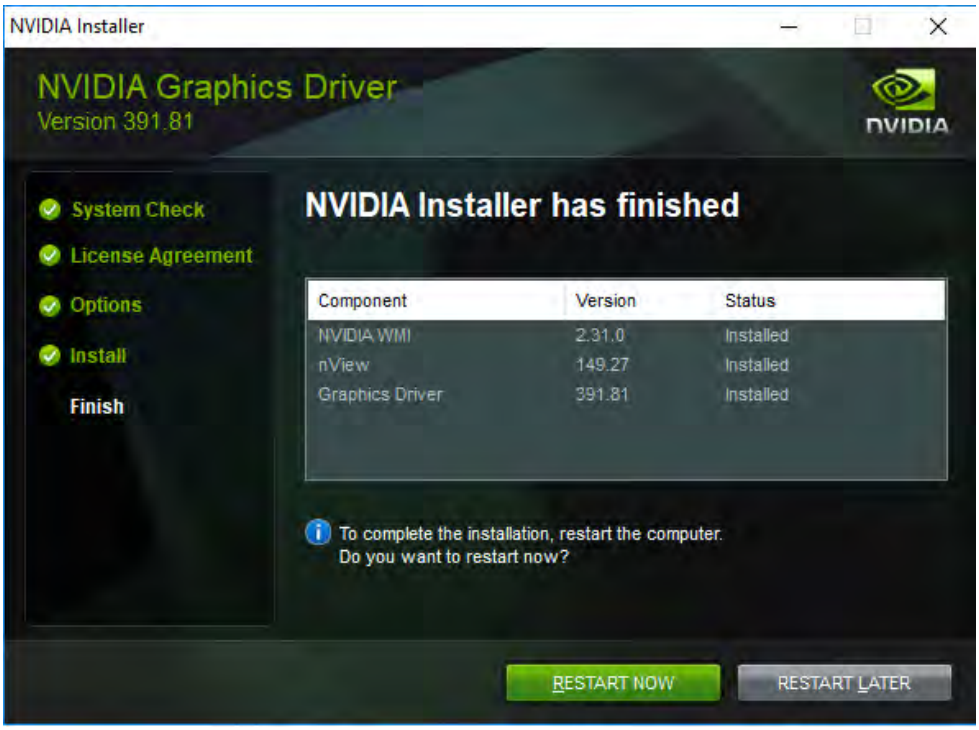


5. Click OK to accept the default and continue the install.The installer displays the NVIDIA software license agreement:

6. Click AGREE AND CONTINUE to continue. The installer displays the Installation Options screen:

7. Click the Custom (Advanced) radio button, then click Next. The installer displays the Custom installation options screen:

8. Check the Perform a clean installation checkbox, then click Next. The installer begins installing the driver. When it is done, it displays the “Installer has finished” screen:



9. Click Restart Now to restart the VM and complete the install.

After you restart the VM, the mouse cursor may not track properly using VNC or vSphere console. In that case, use View Client to Direct Connect.

9.11 Installing the NVIDIA vGPU Driver: Linux

After you create a Linux virtual machine on the hypervisor and boot the VM, you must install the NVIDIA vGPU software display driver to fully enable GPU operation.

9.11.1 Installing 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 it to access the desktop. Otherwise, use SSH to log in to the host remotely. If the Nouveau driver for NVIDIA graphics cards is installed, disable it before you install the NVIDIA vGPU driver.

Prerequisites

Installing the NVIDIA vGPU software display driver for Linux requires:

- Compiler toolchain
- Kernel headers

To install the NVIDIA vGPU driver for Linux

1. Copy the NVIDIA vGPU software Linux driver package to the guest VM or physical host where you are installing the driver.

   The package has this name or a similar one: `NVIDIA-Linux_x86_64-390.75-grid.run`.

2. 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:

     ```
     $ sudo init 3
     ```

   - On Ubuntu platforms:
     i. Enter Ctrl+Alt+F1 to switch to a console log-in prompt.
     ii. Log in and shut down the display manager:

     ```
     $ sudo service lightdm stop
     ```

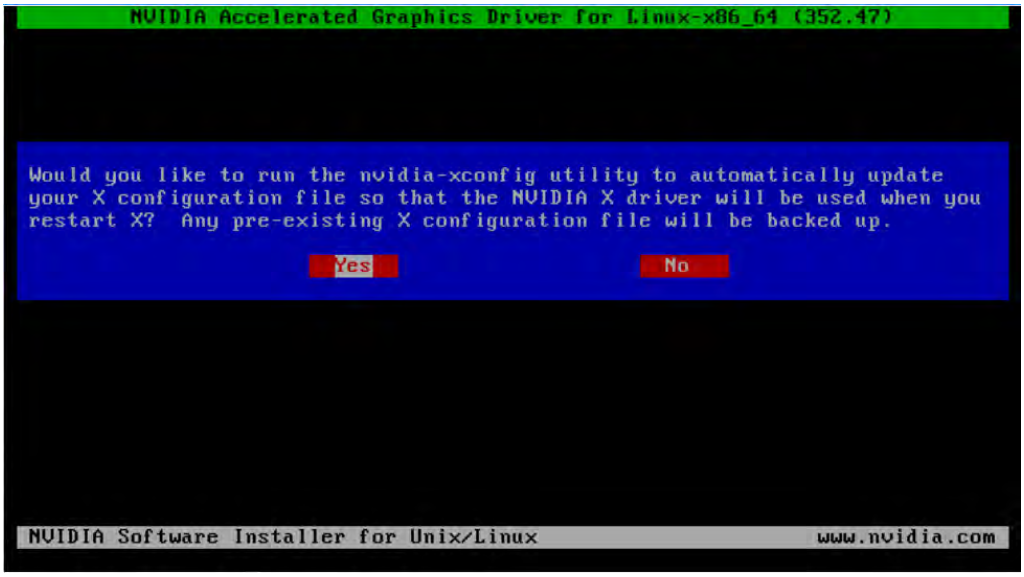
3. From a console shell, run the driver installer as the root user:

   ```
   # 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 that case rerun 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
```

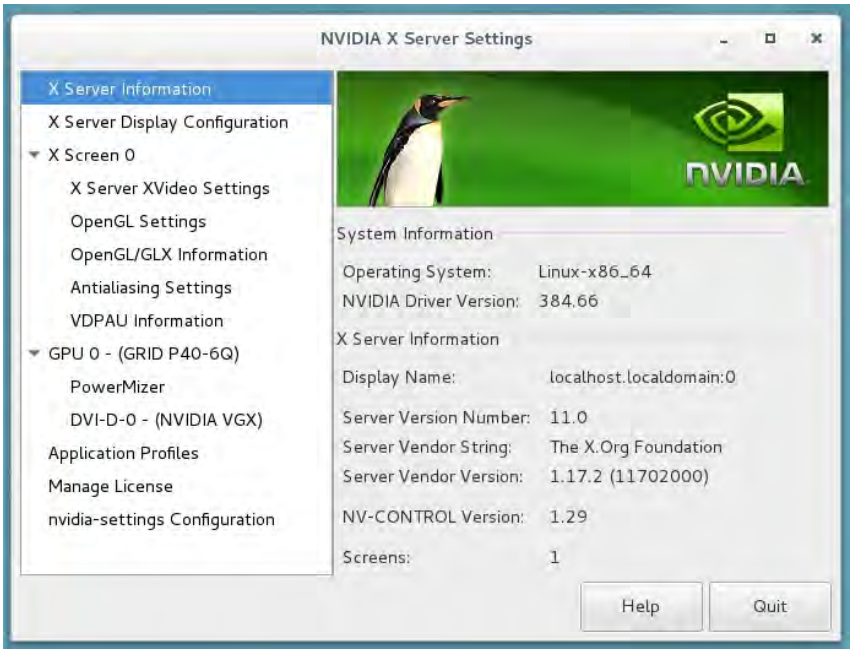
4. When the installer prompts you, accept the option to update the X configuration file `xorg.conf`:



5. When the install is finished, select OK to close the installer.
6. Verify that the NVIDIA vGPU driver is operational.
   a. Reboot the system and log in.
   b. Run `nvidia-settings`:

```
$ nvidia-settings
```

The NVIDIA X Server Settings dialog opens, showing that the NVIDIA driver is operational.



9.12 Licensing NVIDIA vGPU Software

NVIDIA vGPU software is a licensed product. It runs with reduced capability until you acquire a license.

The performance of an unlicensed vGPU is restricted as follows:

- Frame rate is capped at 3 frames per second.
- GPU resource allocations are limited, preventing some applications from running correctly.
- CUDA is disabled on vGPUs that support it.
- For vGPU software releases 6.0 and 6.1 only: Screen resolution is limited to 1280×1024 or less.

Acquiring a license removes these restrictions. A VM that is set up to use a licensed copy of NVIDIA vGPU software can run all versions of DirectX up to and including DirectX12 (and DirectX12 Raytracing on Turing architecture cards), OpenGL, and Vulkan graphics applications.

If licensing is configured, the VM obtains a license from the license server when a vGPU is booted. 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 be modified only if the license server address changes or the VM is reconfigured to use GPU passthrough.

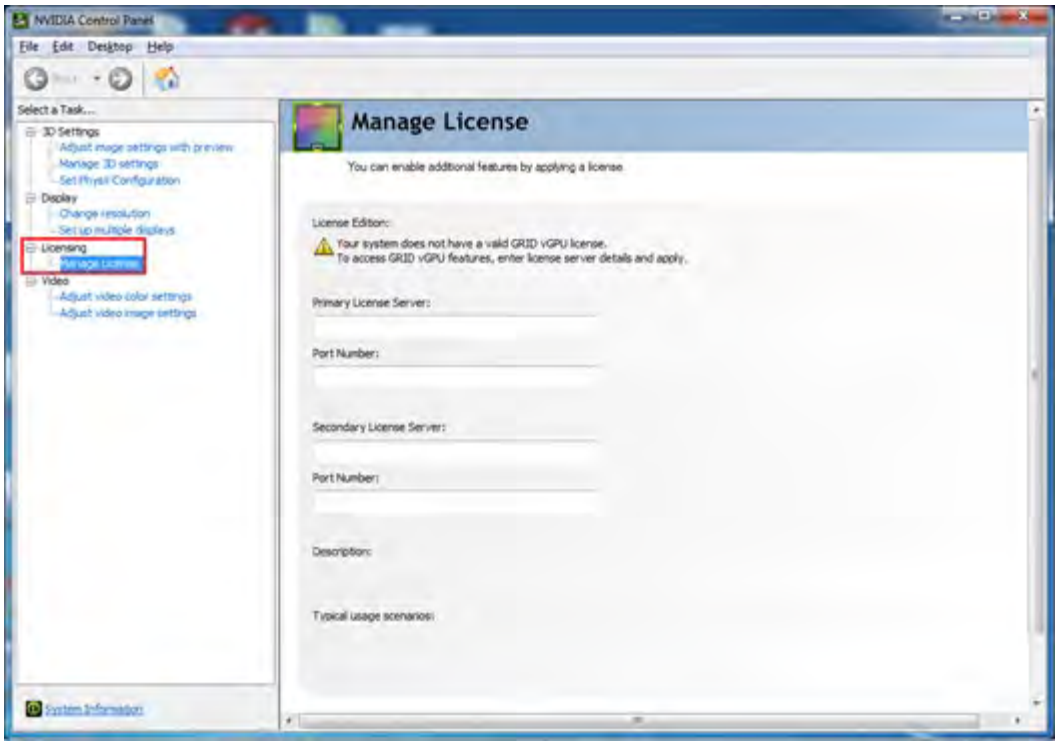
For complete information about configuring and using NVIDIA vGPU software licensed features, including vGPU software, see Virtual GPU Client Licensing User Guide.

9.12.1 Licensing NVIDIA vGPU Software on Windows

1. Right click the Windows desktop and select NVIDIA Control Panel from the menu. This opens NVIDIA Control Panel.

   Alternatively, 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.



In this example the Manage License task pane shows that NVIDIA vGPU is currently unlicensed.

> If the Licensing section and the Manage License task are not visible in NVIDIA Control Panel, the system has been configured to hide licensing controls. You must modify Windows Registry settings to change this. For information about Registry settings, see Virtual GPU Client Licensing User Guide.

3. In the Primary License 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, enter its address in this field.
4. Leave the Port Number field under Primary License Server unset. The port number 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, leave this field unset.
6. Leave the Port Number field under Secondary License Server unset. Like the primary license server port number, the port defaults to 7070.
7. Click Apply to apply the settings. NVIDIA Control Panel requests an appropriate license for the VM's vGPU from the configured license server.
8. If the VM successfully obtained a vGPU software license, it now supports 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 VM fails to obtain a license, see Virtual GPU Client Licensing User Guide for troubleshooting guidance.

9.12.2 Licensing NVIDIA vGPU Software on Linux

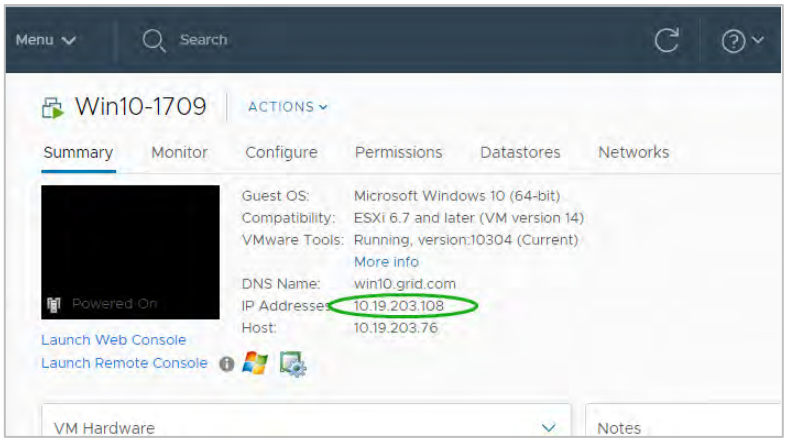
9. Start NVIDIA X Server Settings, using the method for launching applications in your Linux distribution. For example, on Ubuntu Desktop, open the Dash, search for NVIDIA X Server Settings, and click the NVIDIA X Server Settings icon. Linux launches NVIDIA X Server Settings.
10. Click Manage GRID License. The License Edition section of the NVIDIA X Server Settings window shows that NVIDIA vGPU software is currently unlicensed.
11. 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, enter its address in this field.
12. Leave the Port Number field under Primary Server unset. The port number defaults to 7070, which is the default port number used by NVIDIA vGPU software License Server.
13. In the Secondary Server field, enter the address of your secondary NVIDIA vGPU software License Server. If you have only one license server, leave this field unset.
14. Leave the Port Number field under Secondary Server unset. Like the primary license server port number, the port defaults to 7070.
15. Click Apply to apply the settings. NVIDIA X Server Settings requests the appropriate license for the VM's vGPU from the configured license server.
16. If the VM successfully obtained a vGPU software license, it now supports full frame rate, resolution, and display output capabilities. The VM is now capable of running the full range of DirectX and OpenGL graphics applications.
17. If the VM fails to obtain a license, see Virtual GPU Client Licensing User Guide for troubleshooting guidance.

9.13 Finalizing the Installation

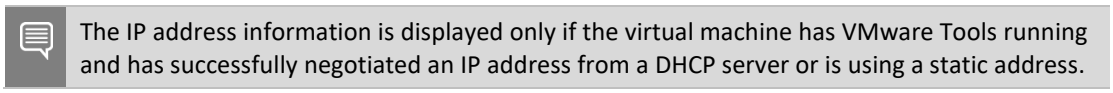
The final phase of NVIDIA vGPU configuration is to connect to the virtual machine and verify settings.

To finalize the installation:

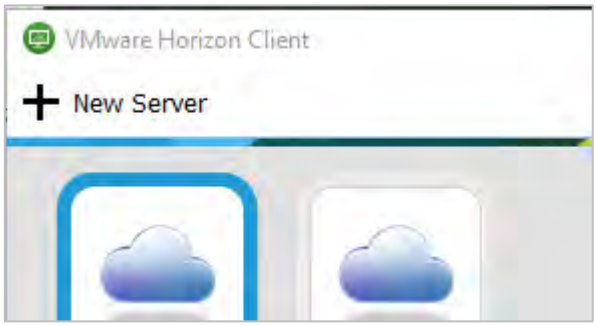
18. Select the virtual machine in the vSphere Web Client window, then select the Summary tab:



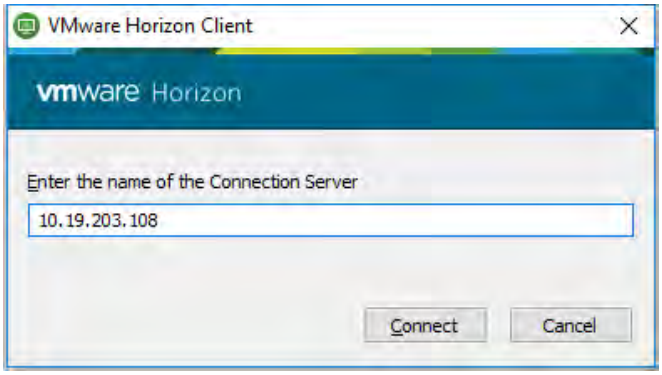
19. Note the VM's IP address (circled in the illustration above).



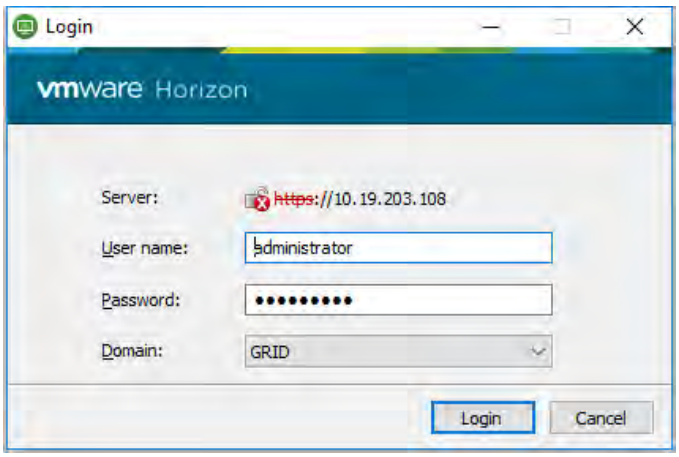
20. Start the VMware Horizon Client from the physical desktop to connect. Click New Server to register a new virtual machine connection.



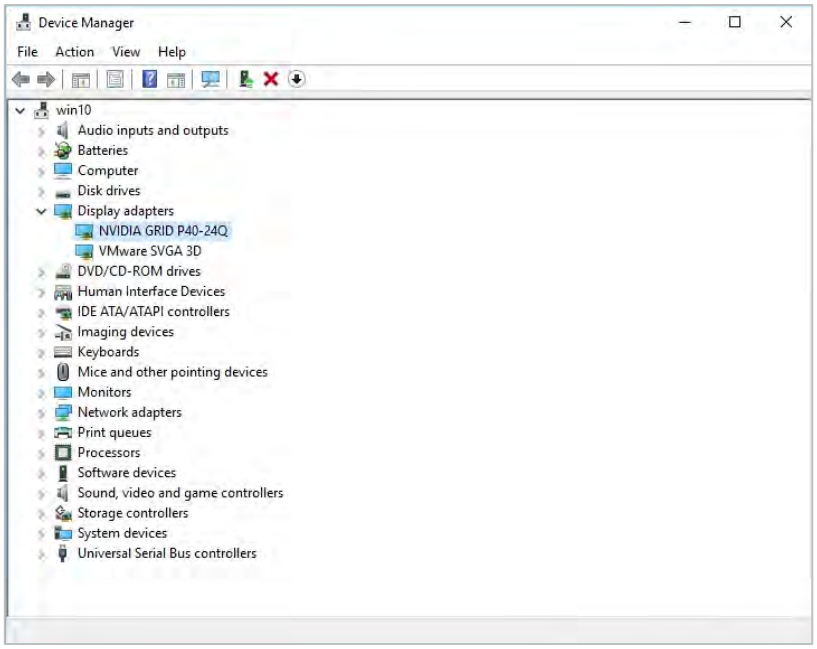
21. Enter the IP address of the VDI desktop to connect to in the Connection Server field, then click Continue. (You are connecting directly to a desktop, so don't enter the Horizon Connection Broker address at this time.)



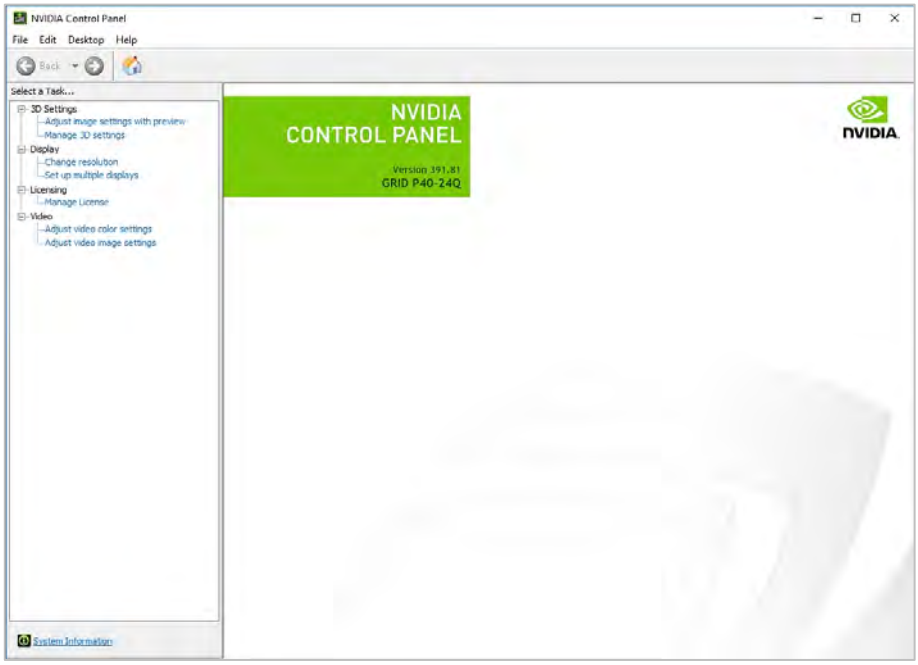
22. Click Connect. VMware Horizon Client displays the Login dialog:



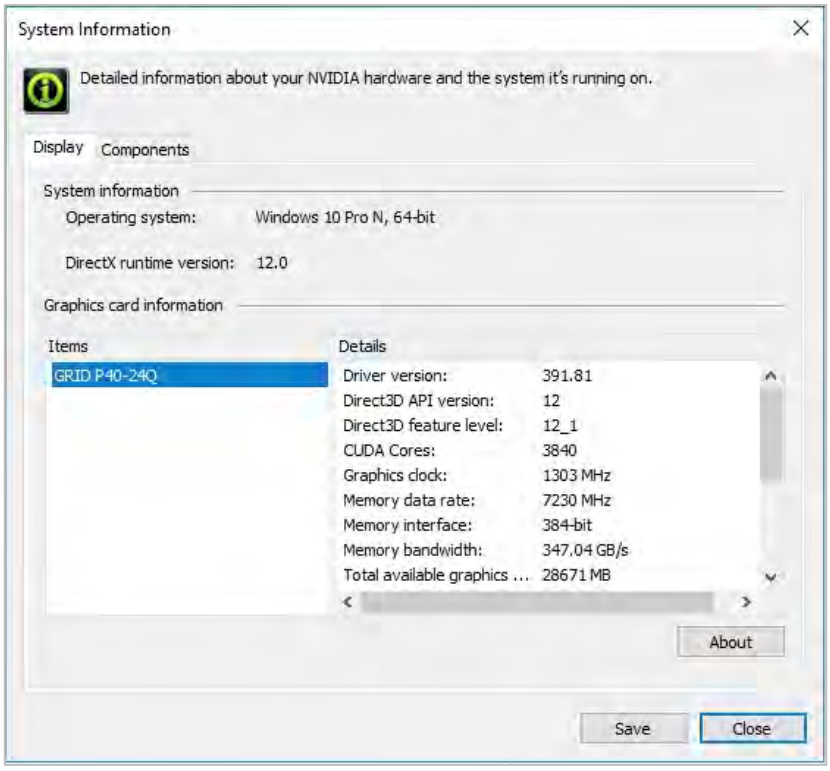
23. Enter the local username and password (or the domain user and password if the virtual machine is a domain member), then click Login. This will launch the VDI.
24. Log in to the virtual machine.
25. Open Device Manager on the virtual machine and expand the Display Adapters entry:



26. Confirm that the display adapter is now a vGPU display.
27. Right click the desktop and select NVIDIA Control Panel:



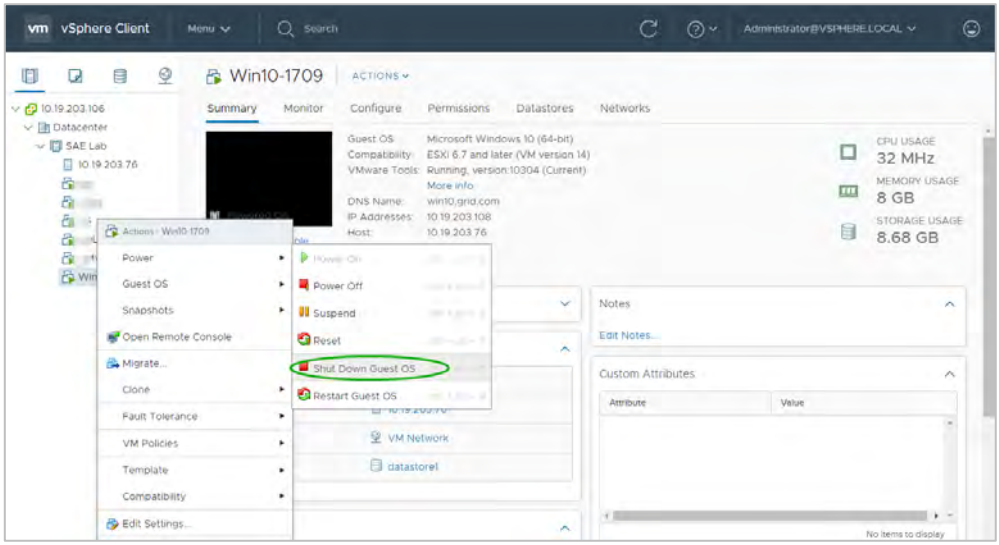
28. Click the System Information link in the bottom left corner of the window. NVIDIA Control Panel opens a System Information dialog:



29. Confirm that the DirectX and graphics card driver versions are correct, then click Close.
30. If you plan to use this VM as a gold master image, release the DHCP address before logging out and shutting down. Open a command prompt and enter:

```
C...> ipconfig /release
```

31. At this point you lose connectivity to the VM. Use vSphere Web Client to shut down the guest OS:



32. A vGPU enabled virtual machine is now confirmed. The next step is to build and deploy a Horizon vGPU Desktop Pool.

Chapter 10. Creating a Horizon vGPU Pool

A **pool** is a collection of virtual machines that can be controlled, authorized to users, etc. as a group. This chapter describes the following:

- Creating a template from an existing virtual machine
- Creating a Customization Specification
- Provisioning a single vGPU-enabled virtual machine from a template
- Creating Full and Linked Clones Horizon pools
- Enabling User Access to pools

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

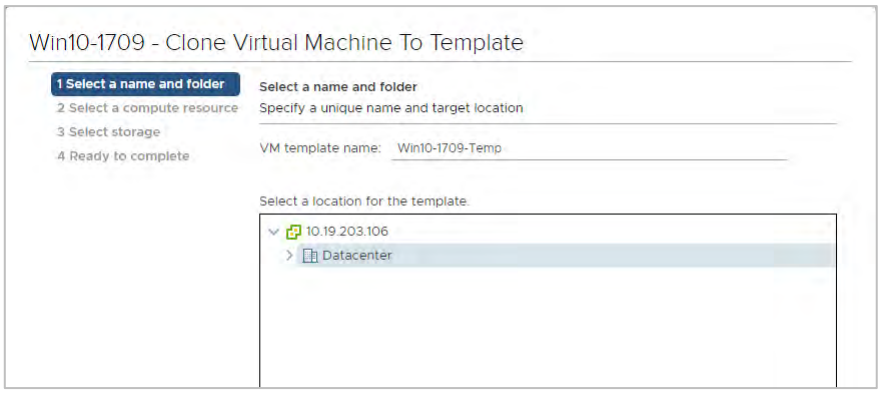
A complete demonstration of these features is beyond the scope of this document; however, converting a virtual machine to a template and then deploying virtual machines from that template is a fundamental operation that can reduce evaluation time.

10.1 Creating a Template

To create a template from an existing virtual machine:

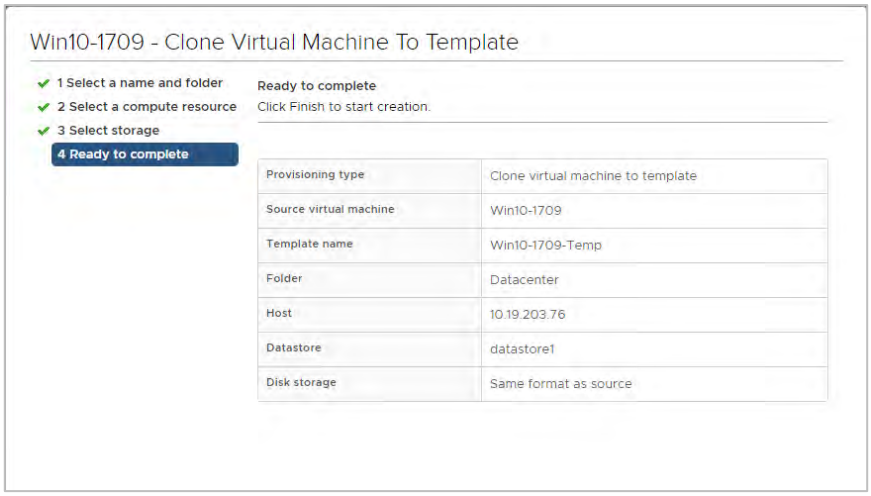
1. In vSphere Web Client, right click a virtual machine and select Clone - > Clone to Template...

   vSphere displays the Clone Virtual Machine To Template dialog with the Select a name and folder tab selected:



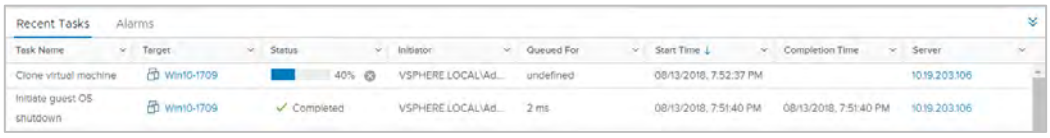
2. In the VM template name field, enter a name for the template.
3. Select either a datacenter or virtual machine folder in which to create the template.

4. Click Next to display the Select a compute resource tab.
5. Select the host or cluster in which to store the template.
6. Click Next to display the Select storage tab.
7. Select the storage type and target datastore.
8. Click Next to display the Ready to complete tab.

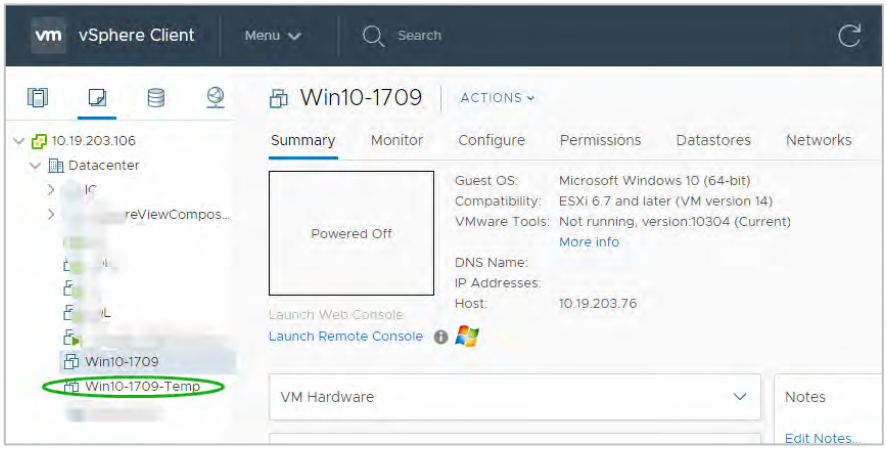


9. Review the settings, then click Finish to start cloning the virtual machine to a template.

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



10. When the cloning process is finished, vSphere opens a window that displays the template:

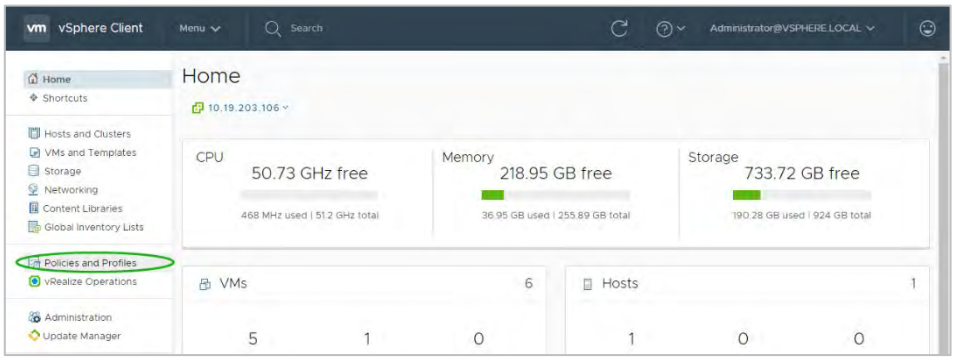


10.2 Creating a Customization Specification

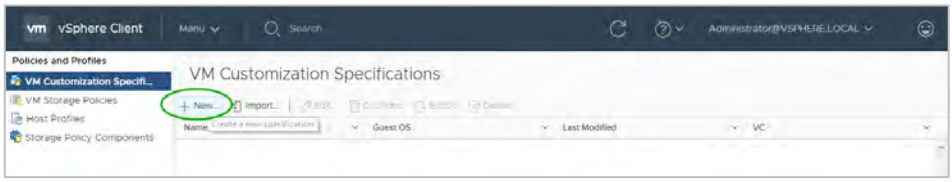
Before you can provision a virtual machine from a template you must create a **VM Customization Specification.** This will handle actions such as joining the domain, renaming the machine.

To create a VM Customization Specification:

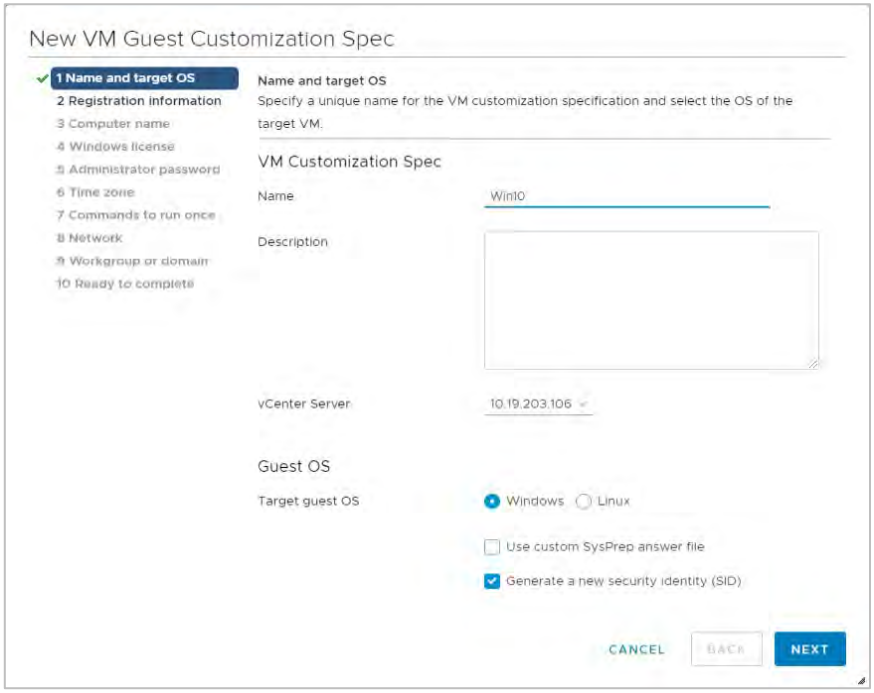
1. From the left-hand pane of the vSphere Client home view, click Policies and Profiles:



vSphere displays the VM Customization Specifications tab:



2. Click New. vSphere displays the New VM Guest Customization Spec screen:



3. For Target guest OS, select Windows.

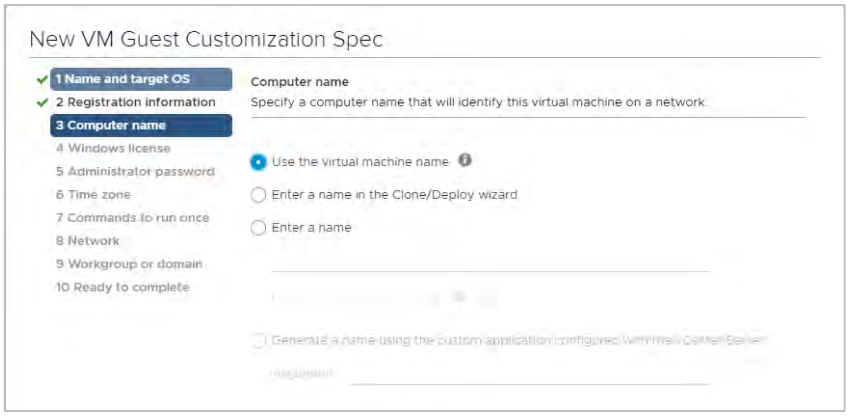
   Check the Generate New Security ID (SID) checkbox.

   If your organization currently utilizes an answer file, you can load that file by checking the Use custom SysPrep answer file checkbox.

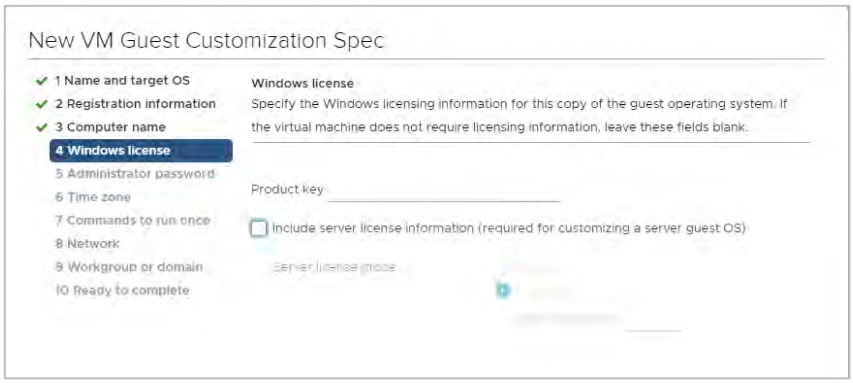
If your organization does not use this functionality, leave the Use custom SysPrep answer file checkbox cleared. Enter a name for the new specification in the Customization Spec Name field.

Click Next. vSphere displays the Set Registration Information tab:

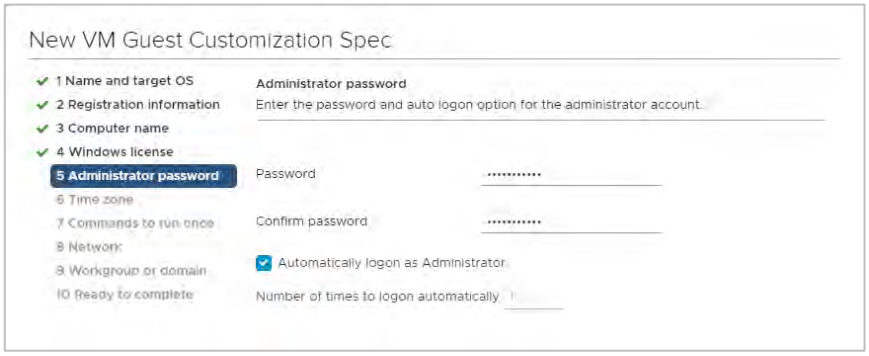
4. Enter the VM owner's name in the Name field and the owner's organization's name in the Organization field, then click Next to display the Computer Name tab:



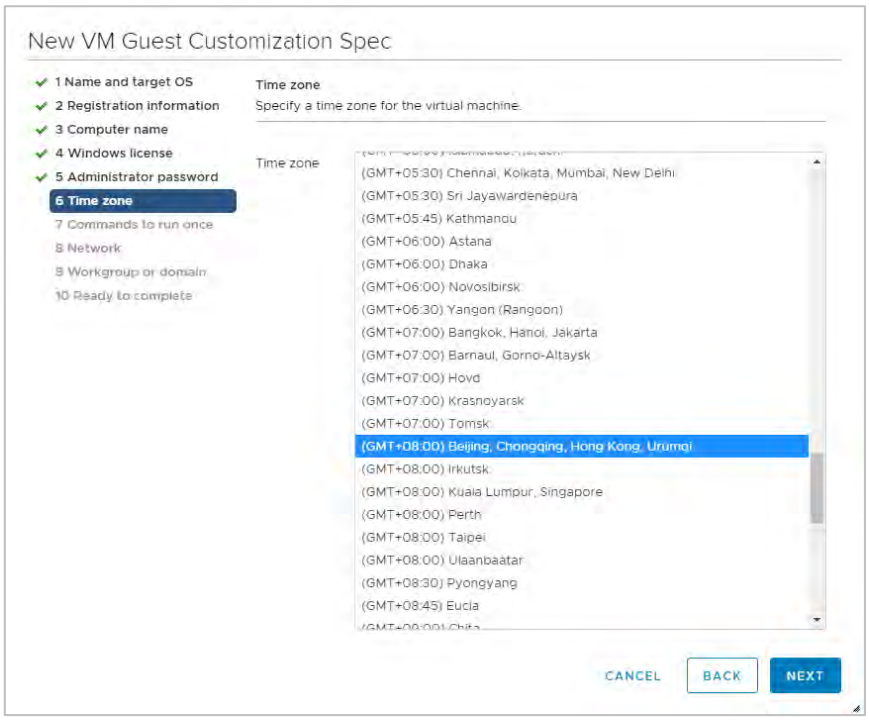
5. Enter a computer name for the VM, then click Next to display the Windows License tab:



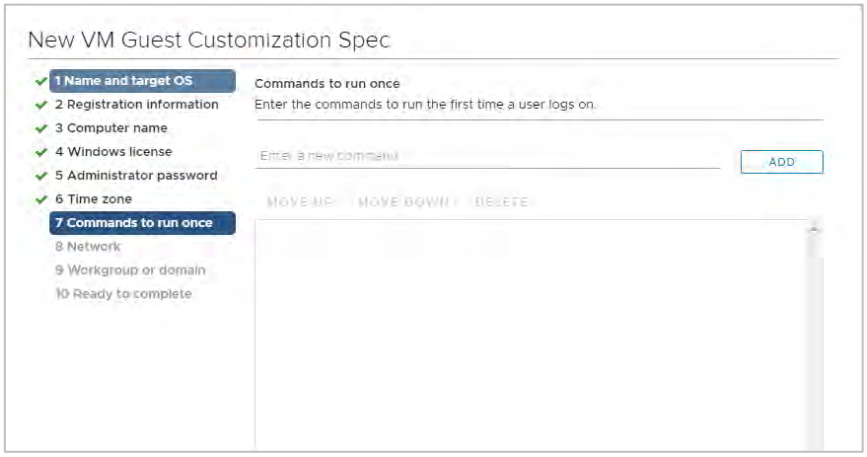
6. Enter your Windows product key information or leave this field blank. Then click Next to display the Administrator Password tab:



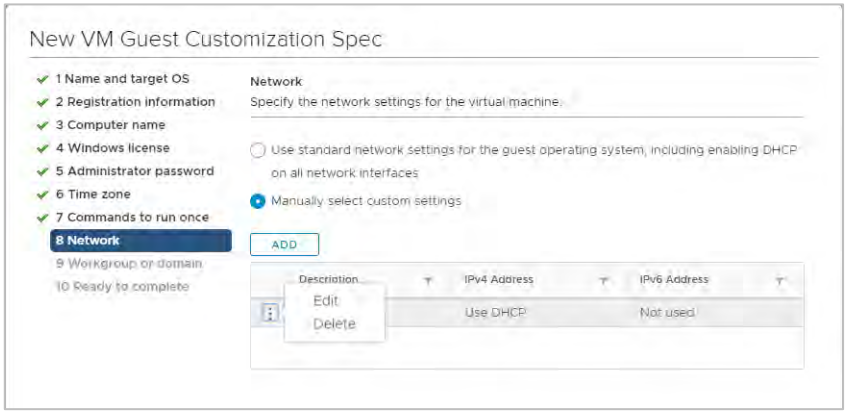
7. Enter and confirm the administrator password (it is case sensitive) in the Password and Confirm Password fields, then click Next to display the Time Zone tab:



8. Select your time zone from the Time Zone pulldown menu, then click Next to display the Commands to Run Once tab.

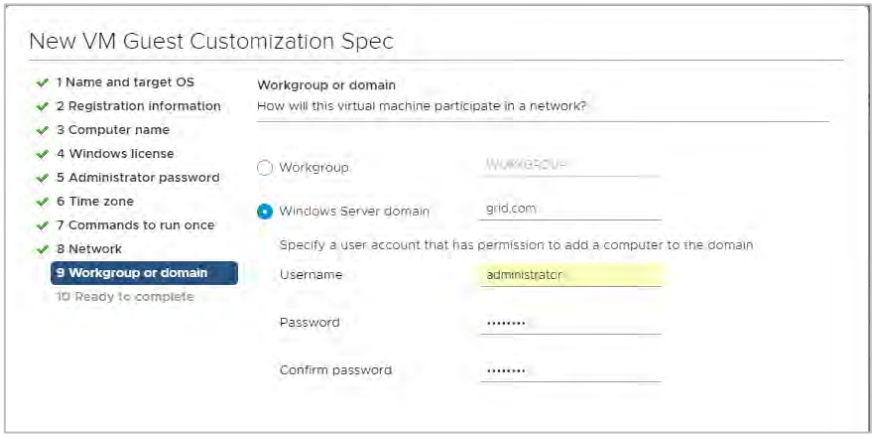


9. Enter any required one-time commands in this tab, then click Next to display the Network tab:

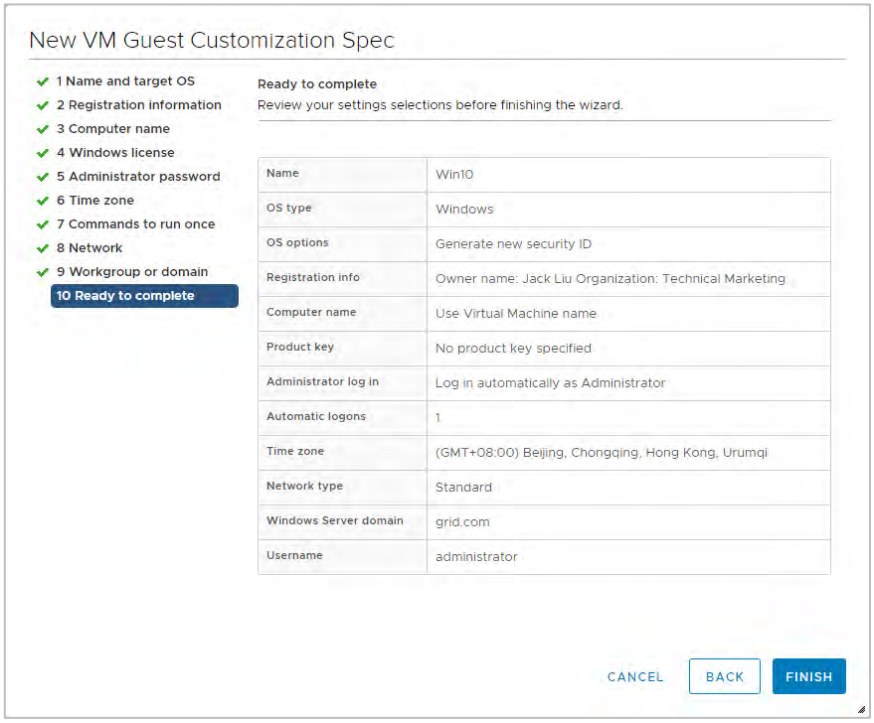


10. Click the appropriate radio button to either:
    - Use standard network settings: Automatically selects network settings.
    - Use manual network settings: Enter the network description, IPv4 address, and/or IPv6 address in the respective fields.

    Click Next to display the Workgroup or Domain tab:



11. Select the appropriate radio button to make the VM a member of either a workgroup or a domain. Enter the VM's workgroup or domain information in the appropriate fields Then click Next to display the Ready to complete tab.



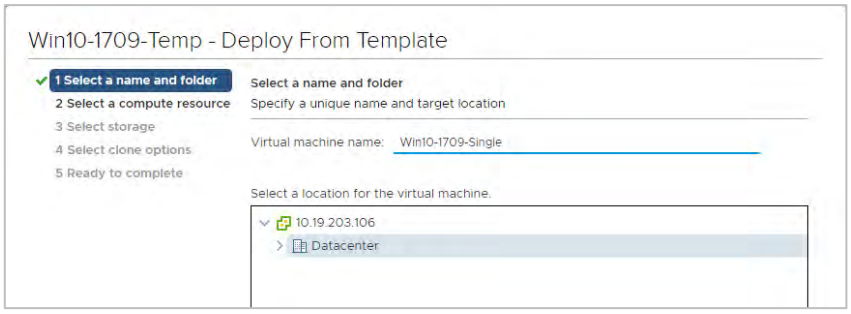
12. Review your settings. If any are incorrect, click Back to correct them. When all of the settings are correct, click Finish to view the new customization specification from the template.

> 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

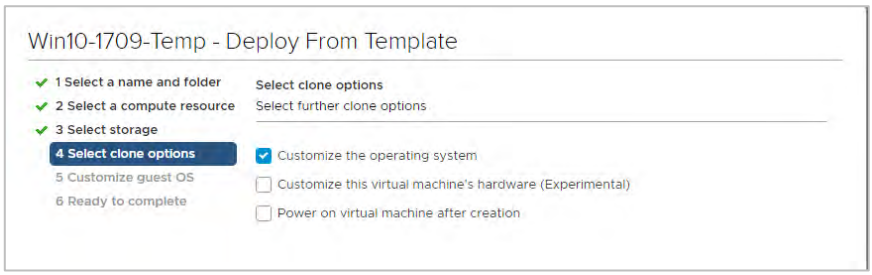
To provision a single vGPU-enabled virtual machine from a template:

1. From vSphere Client, right click the template and select New VM from this Template,,, vSphere displays the Deploy From Template window with the Select a name and folder tab selected.

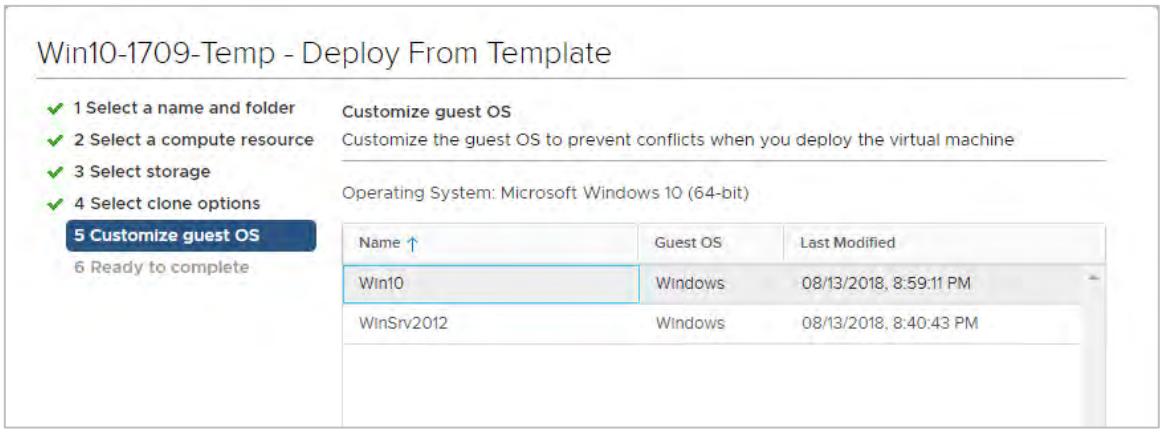


2. In the Virtual machine name field, enter a name for the new VM. Then click Next to display the Select a compute resource tab.

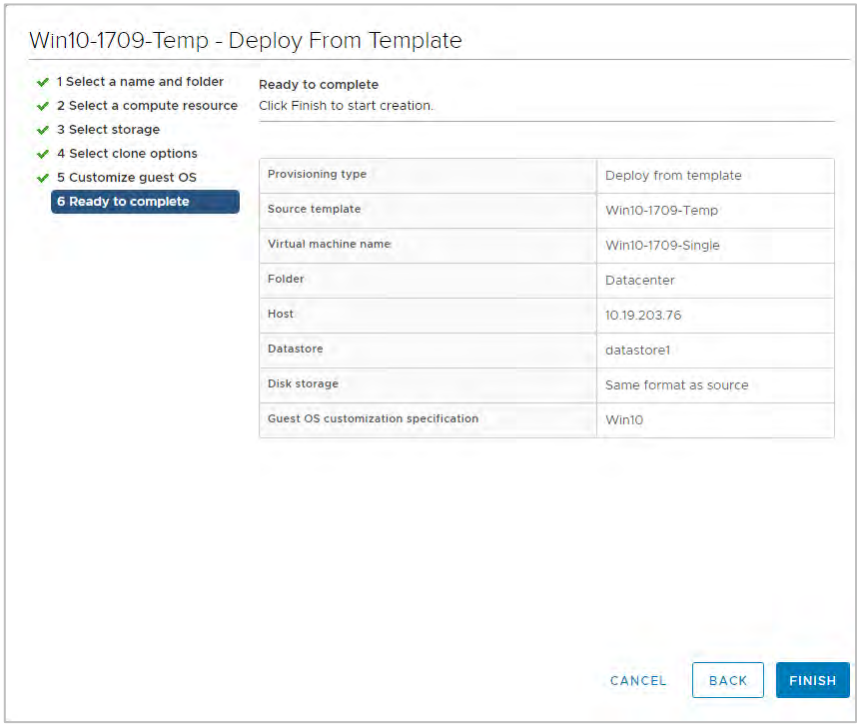
3. Under Select a location for the virtual machine, select the cluster or host that is to host the VM. Then click Next to display the Select storage tab.
4. Select the target storage type and datastore. Then click Next to display the Select clone options tab:



Check the Customize the operating system checkbox, then click Next to display the Customize guest OS tab:



5. Select the template to be used to create the VM. vSphere displays the Ready to complete tab of the Deploy From Template window, which displays the properties of the customization specification used:



6. Review the settings. If any of them are incorrect, click the Back button to fix them. If all of them are correct, click Finish to begin creating the virtual machine from the template.

   When vSphere finishes creating the VM, the VM becomes visible in the datacenter and on the host that it was cloned to.

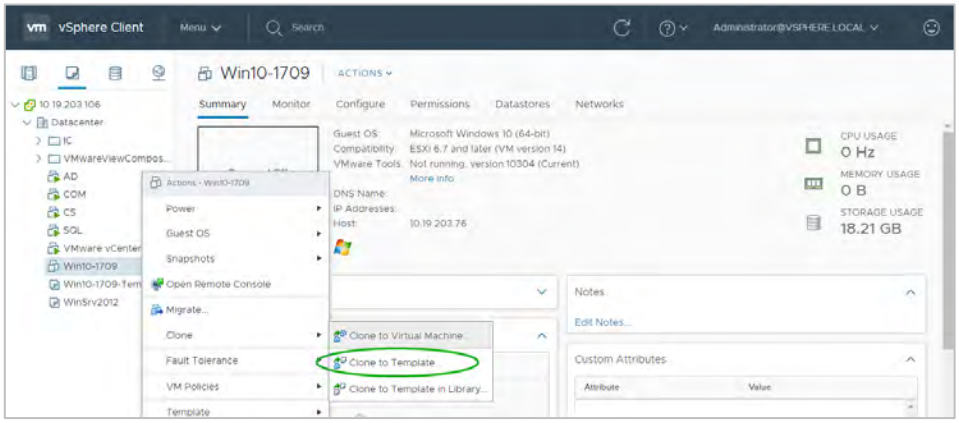
7. Add the new VM to a Horizon Desktop Pool, then authorize users and/or groups to use it. These steps are described later in section 10.6.

Horizon can leverage templates to create virtual machines automatically, on demand, to save time and resources. See the *Horizon* documentation for instructions on using this functionality.

The next sections describe how to install a desktop pool and then how to grant entitlements to users and groups to use that pool. You can create desktop pools from a gold master image with either Full Clones or Linked Clones with the help of Horizon Composer. Both use cases are explained in the following sections.

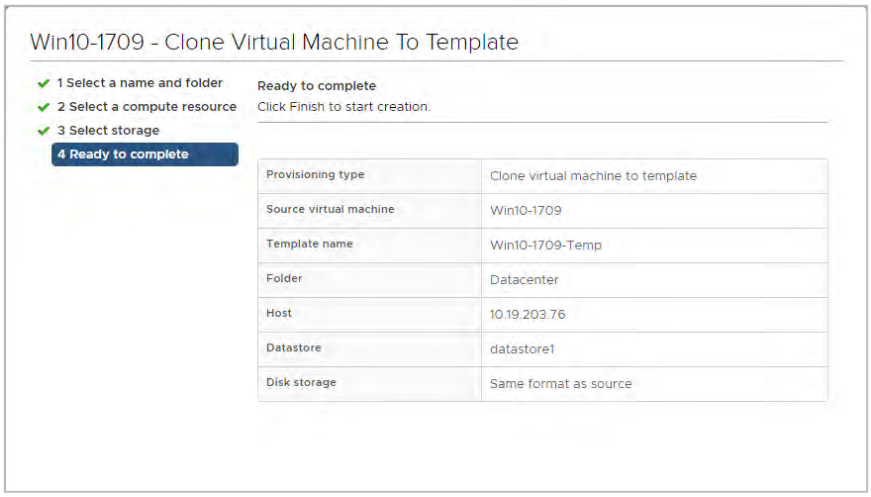
10.4 Creating Full Clone Desktop Pools

1. To create full clone desktop pools, you must have a template of the gold master image. Find the reference VM in the Navigator and right click it. Select Clone > Clone to Template.

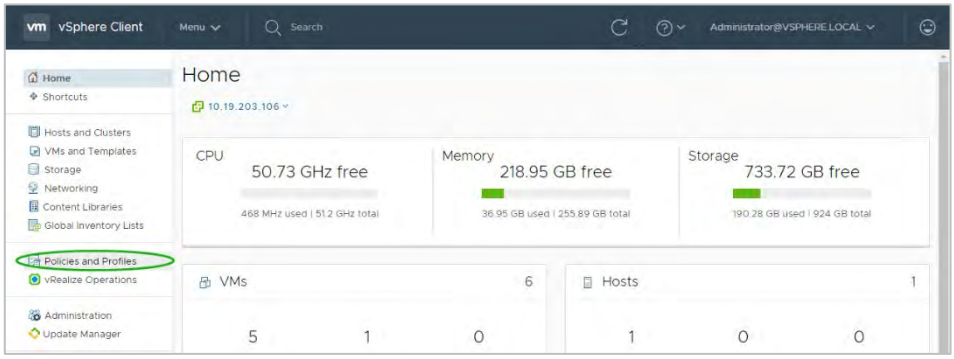


Alternatively, you can create a template by using the Template option to convert a VM to a template.

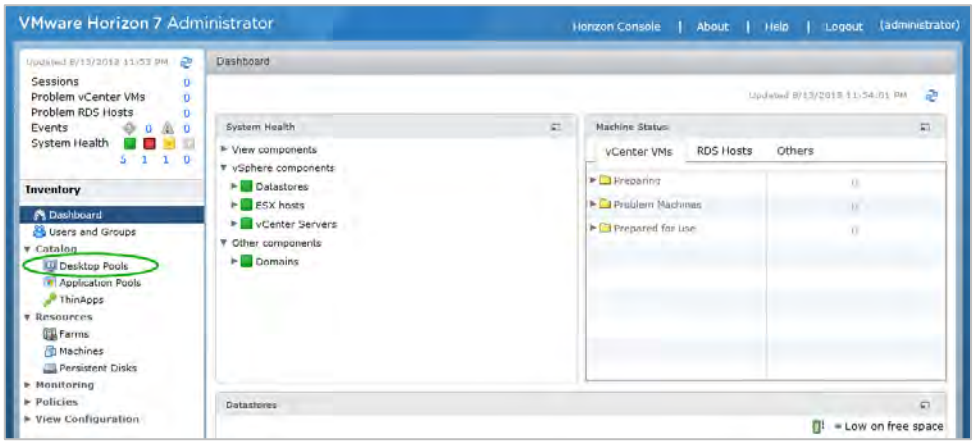
The Clone Virtual Machine To Template wizard walks you through the steps, described in Section 9.1. At the end of the process, click Finish to create the template:



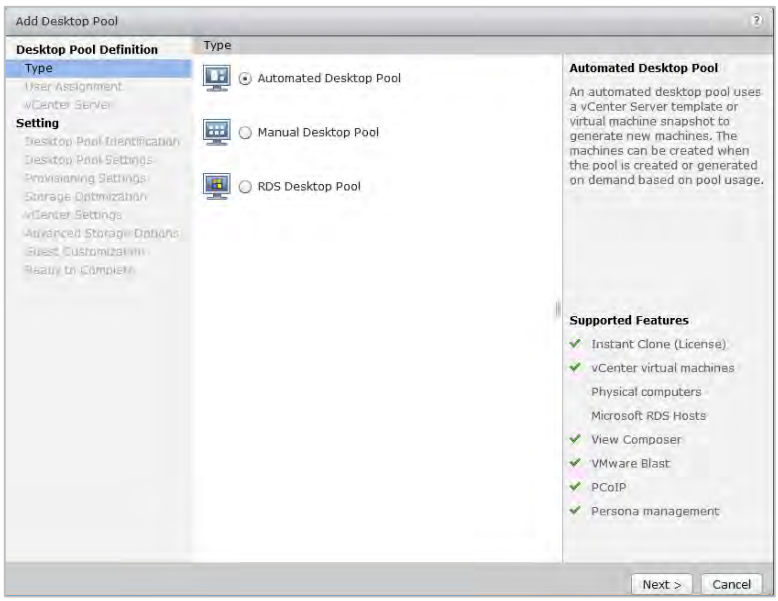
2. Create a Customization Specification file to deploy a full clone pool. The New VM Guest Customization Spec wizard walks you through the steps, described in Section 9.1, Creating a Virtual Machine:



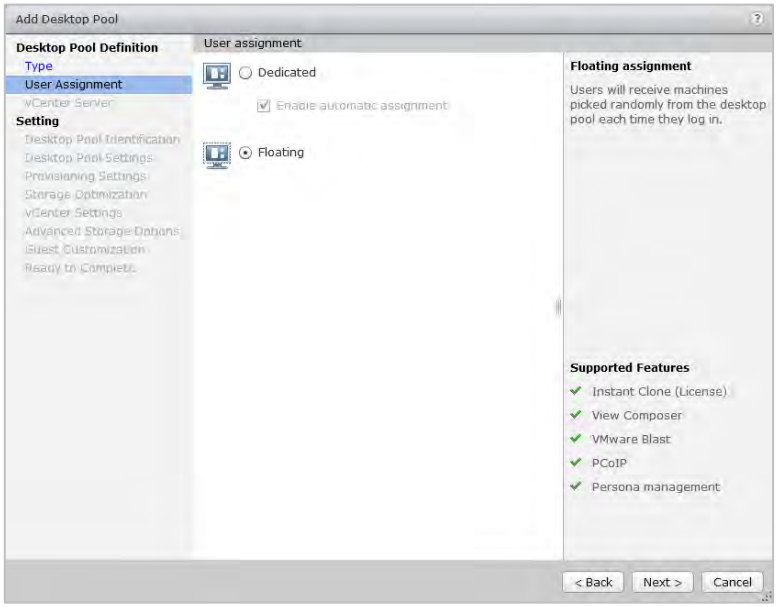
3. Log in to Horizon Administrator:



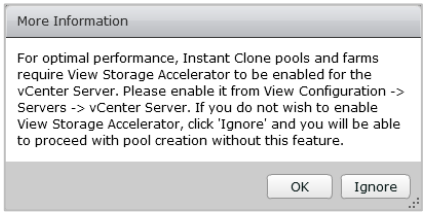
4. Select Desktop Pools in the Inventory pane, then click Add. Horizon Administrator displays the Type (type of desktop pool) screen:



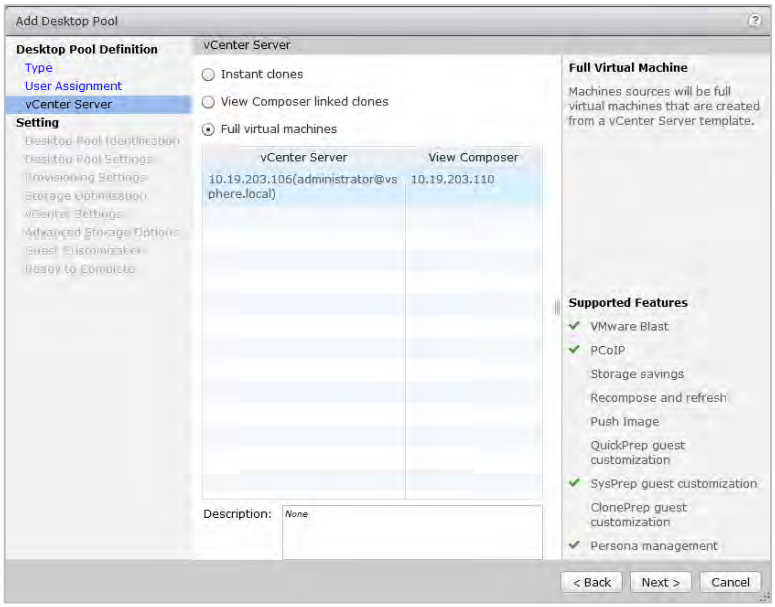
5. Select Automated Pool, then click Next. Horizon Administrator displays the User Assignment screen:



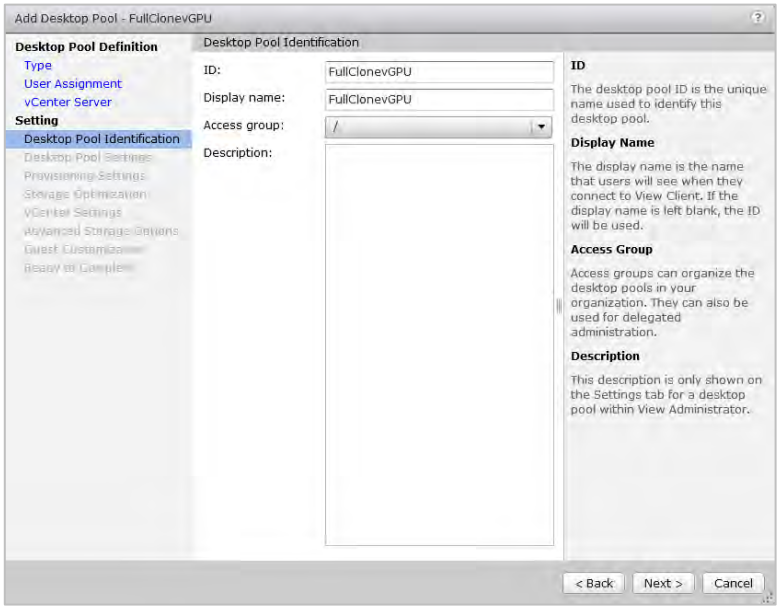
6. Select Floating, then click Next. Horizon Administrator displays a warning message:



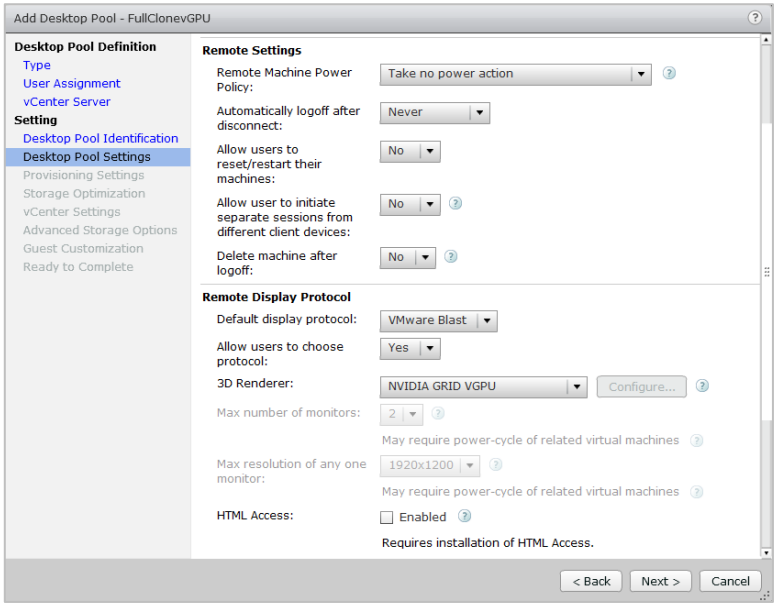
7. Ignore the warning and click OK. Horizon Administrator displays the vCenter Server screen:



8. Select Full Virtual Machines, then click Next. Horizon Administrator displays the Desktop Pool Identification screen:



Enter a unique ID and a display name for the desktop pool, then click Next. Horizon Administrator displays the Remote Settings and Remote Display Protocol screen:

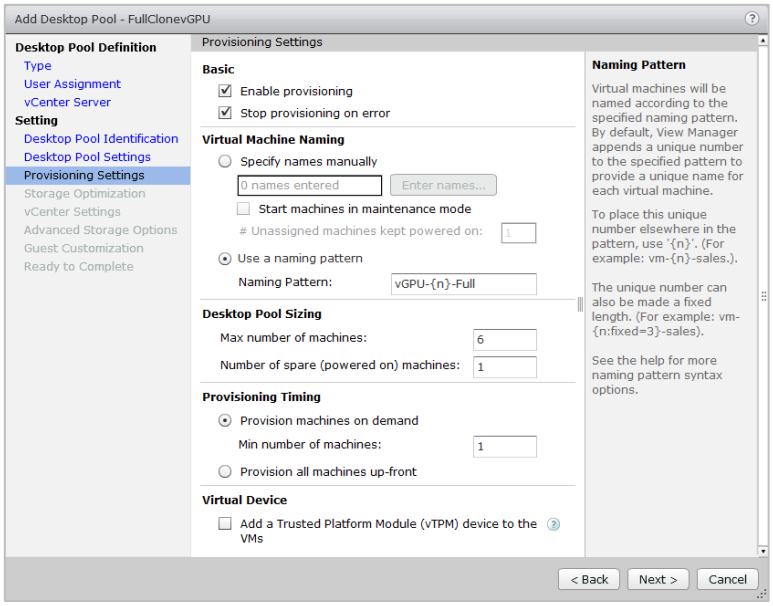


9. Set Default display protocol to Blast.

   Set Allow users to choose protocol to Yes.

   Set 3D Renderer to NVIDIA GRID VGPU. (HTML access is supported in this version if it is desired.)

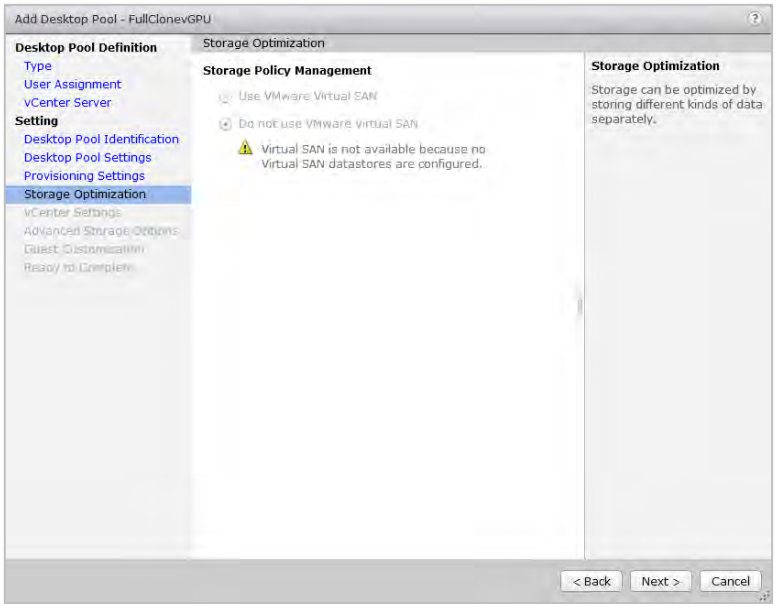
Click Next. Horizon Administrator displays the Provisioning Settings screen:



10. In the Select a naming pattern field, enter a naming pattern to help differentiate the VMs in this pool from those in other pools.

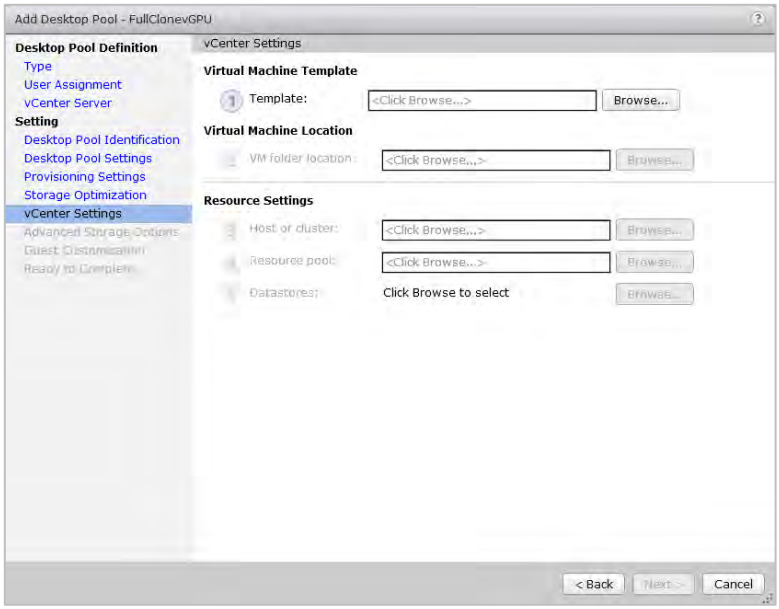
    Set Desktop Pools Sizing and Provisioning Timing to values that make sense for your infrastructure.

    Click Next. Horizon Administrator displays the Storage Policy Management screen:

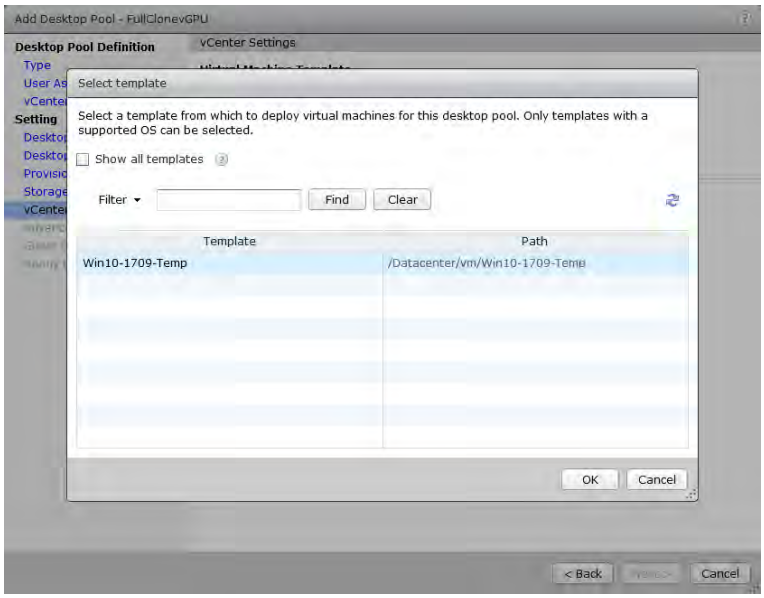


11. If a virtual SAN is configured, select the appropriate storage policy. If no virtual SAN is configured, both options are disabled so the screen allows no action.

Click Next. Horizon Administrator displays the vCenter Settings screen:

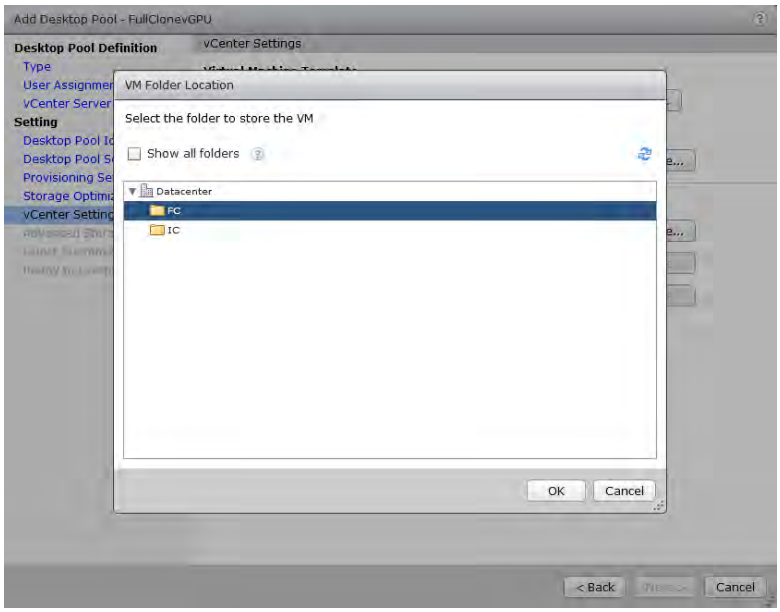


12. Click the Virtual Machine Template > Browse button to display the Select Template dialog:

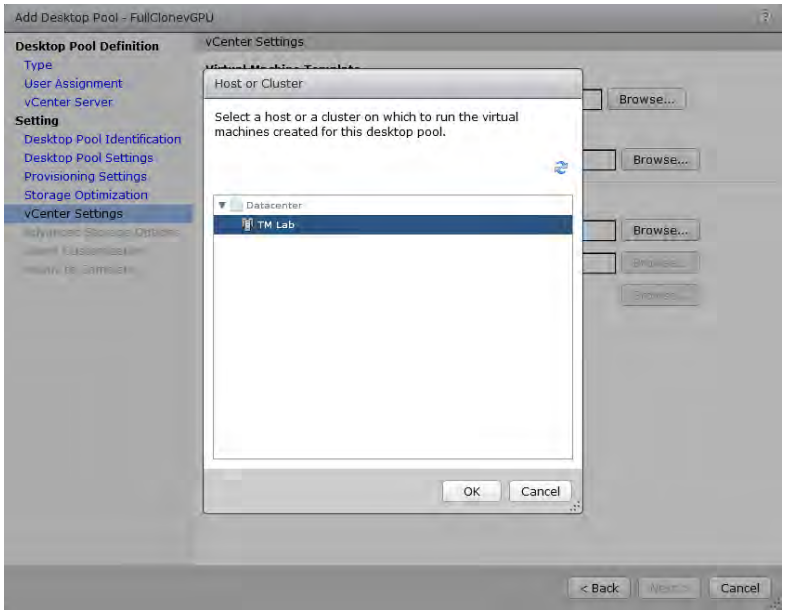


13. Select the template created earlier for the gold master image, then click OK to close the dialog:

14. Click the Virtual Machine Location > Browse button to display the VM Folder Location dialog:

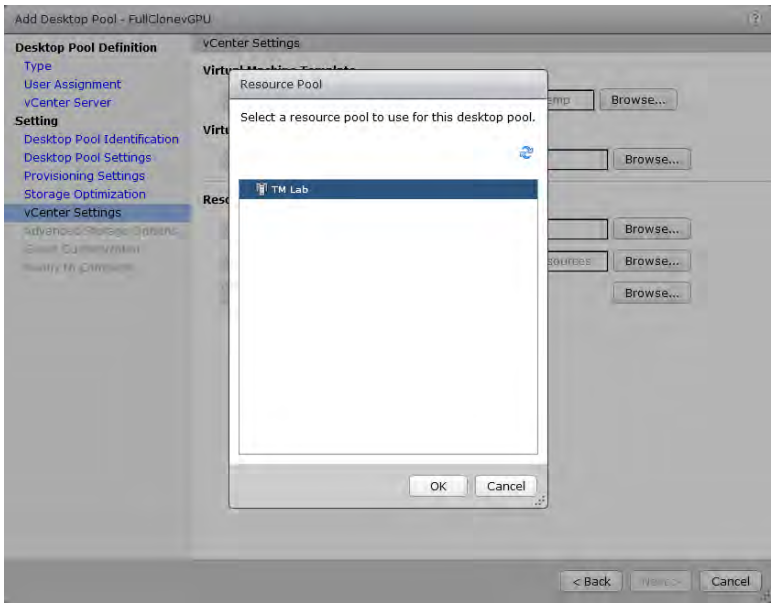


15. Select a folder to store the VMs, then click OK to close the dialog:
16. Click the Host or Cluster > Browse button to display the Host or Cluster dialog:

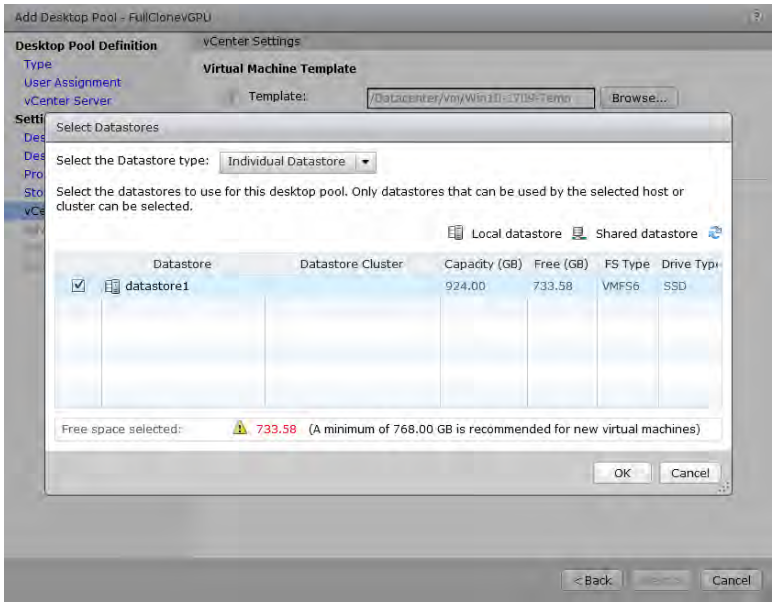


17. Select a Host or Cluster, then click OK to close the dialog:

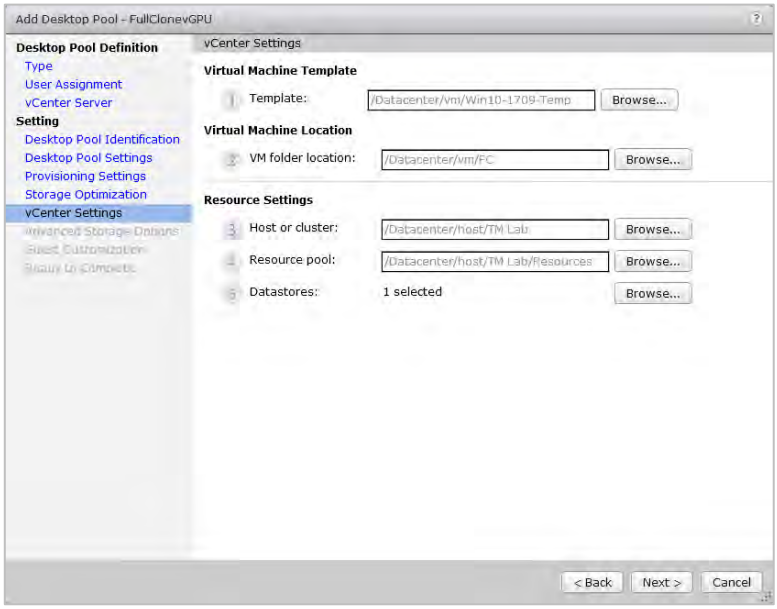
18. Click the Resource pool > Browse button to display the Resource Pool dialog:



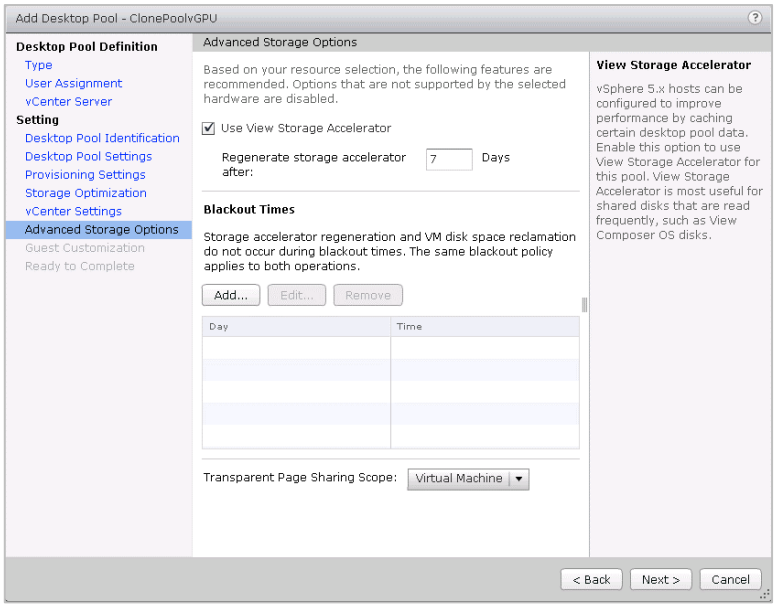
19. Select a Resource Pool, then click OK to close the dialog:
20. Click the Datastores > Browse button to display the Select Datastores dialog:



21. Select a datastore from the Desktop Pool, then click OK to close the dialog. Horizon Administrator displays the vCenter Settings screen:

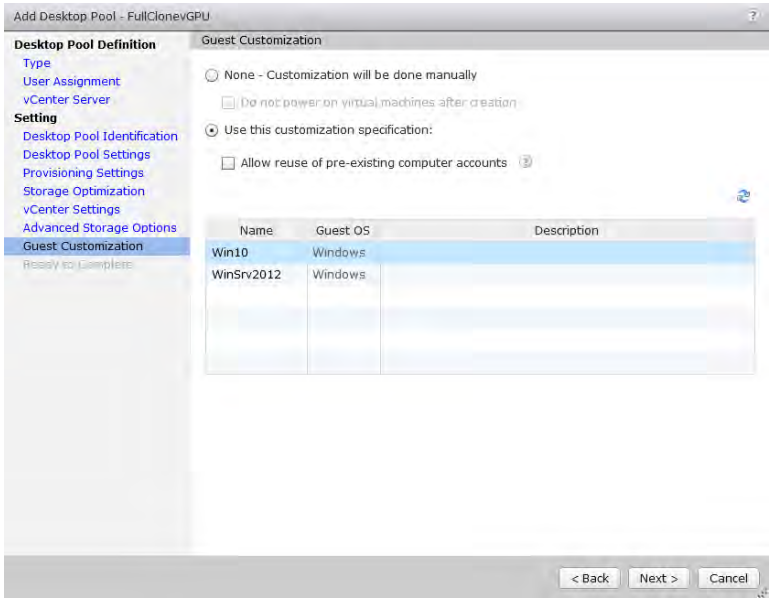


22. Review the settings you have chosen. If any are incorrect, click Back to correct them. When all of the settings are correct, click Next to continue. Horizon Administrator displays Advanced Storage Options screen:

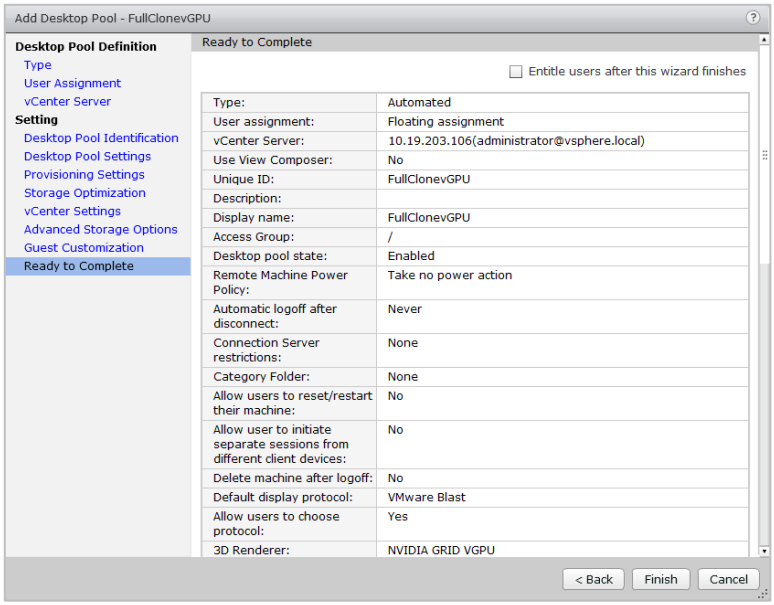


23. Check the Use Horizon Storage Accelerator checkbox if it is available.

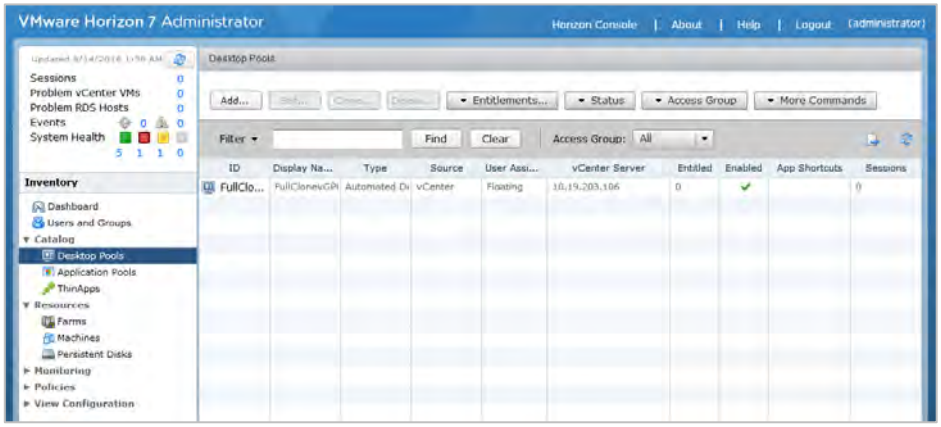
Enter blackout times for non-refresh of Horizon Storage Accelerator cache to minimize impact on production users, then click Next to continue. Horizon Administrator displays the Guest Customization screen:



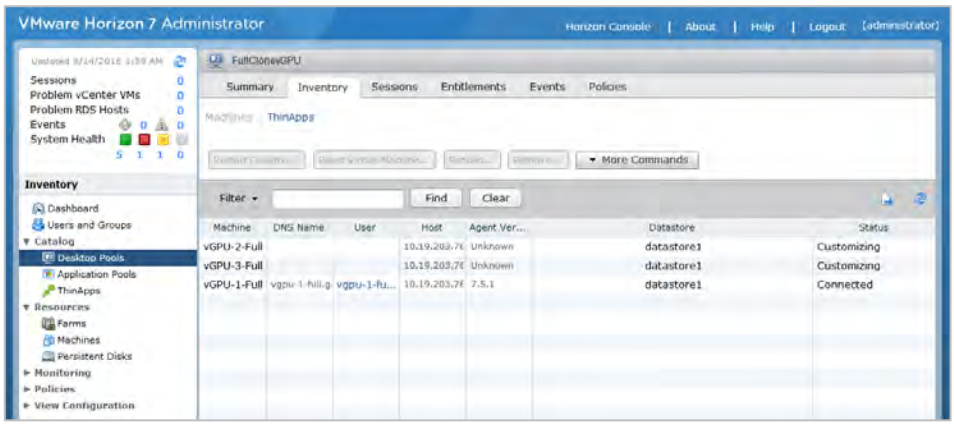
24. To simplify deployment, click Use this customization specification, then click Next to display the Ready to Complete dialog:



25. Review your selections. If any are incorrect, click Back to correct them. When your selections are all correct, click Finish to deploy the Desktop Pool. Horizon Administrator displays the Desktop Pools window, which now shows the pool you have defined:



26. Look Click the Status button to display the pools' status:



The Status column, near the right edge of the window, should say that the status of the pool you defined is "Connected." This means that the desktops in the pool are ready for use.

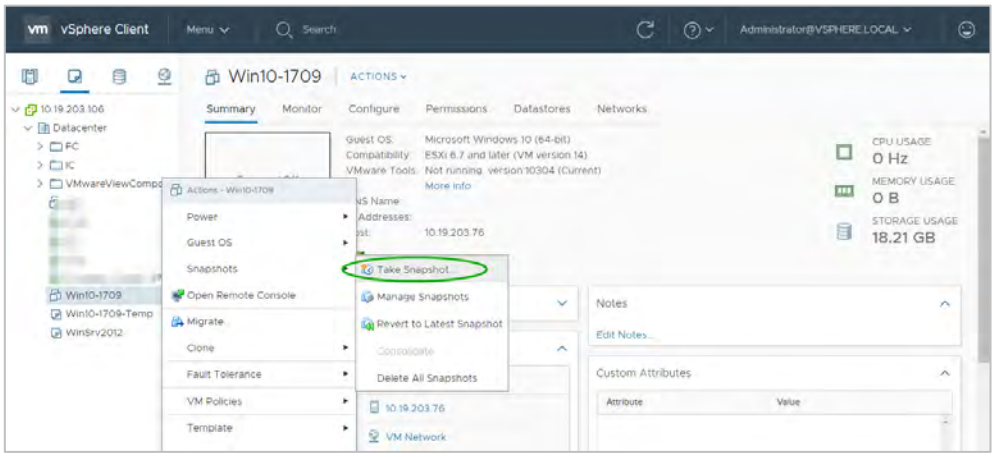
10.5 Creating Linked Clone Desktop Pools

Creating a linked pool of VDI desktops is very similar to creating a full clone pool. A linked pool leverages the Horizon Composer to deploy multiple desktops from one gold master image. Linked clones consume less disk space than full clones, and are easier to manage, upgrade, and deploy with little impact on the end user.

As for a clone pool, you begin by selecting a VM as gold master image. Instead of using a template of the VM, though, you take a snapshot of it.

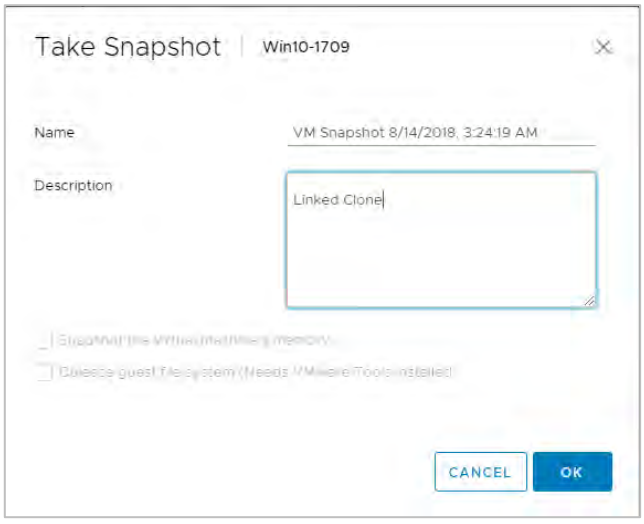
To create a linked clone desktop pool:

1. Start vSphere Web Client:

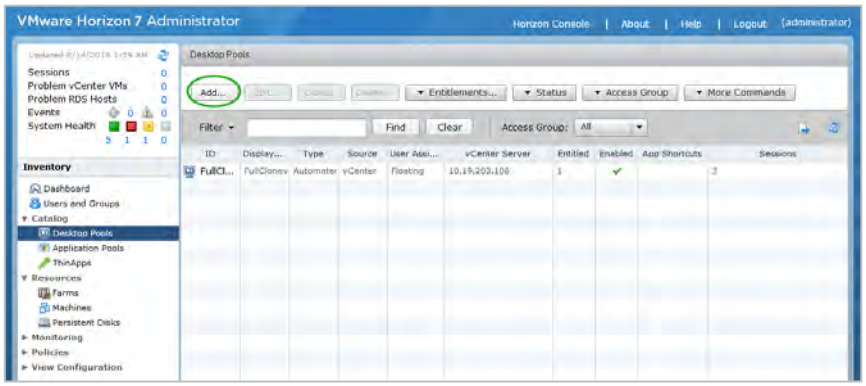


2. Right click the gold master image VM.

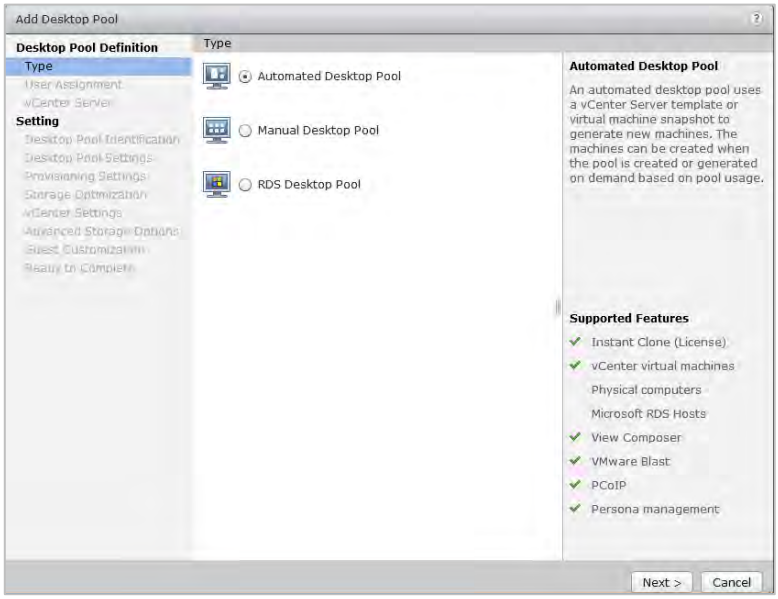
   Select Snapshots > Take Snapshot to open the Take Snapshot dialog:



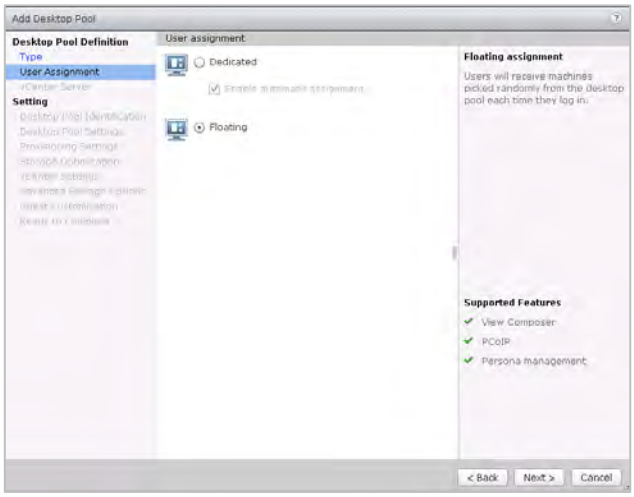
3. Name the snapshot to conform to your VM naming standards. Give the snapshot a meaningful description, then click OK to create the snapshot.
4. Start VMware Horizon Administrator:



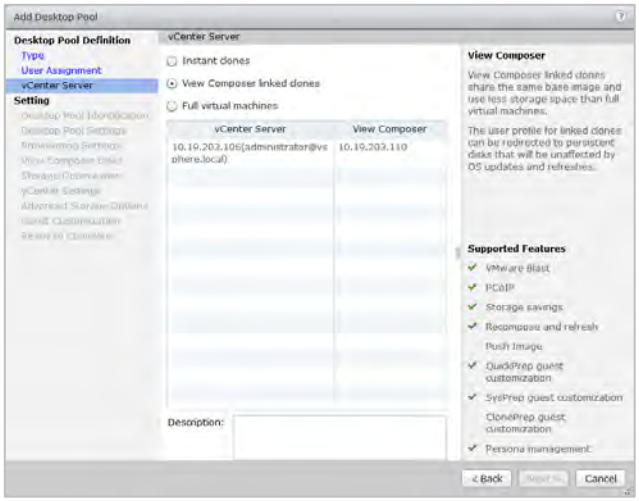
5. Select Desktop Pools in the Inventory pane, then click the Add button. Horizon Administrator starts the Desktop Pool wizard:



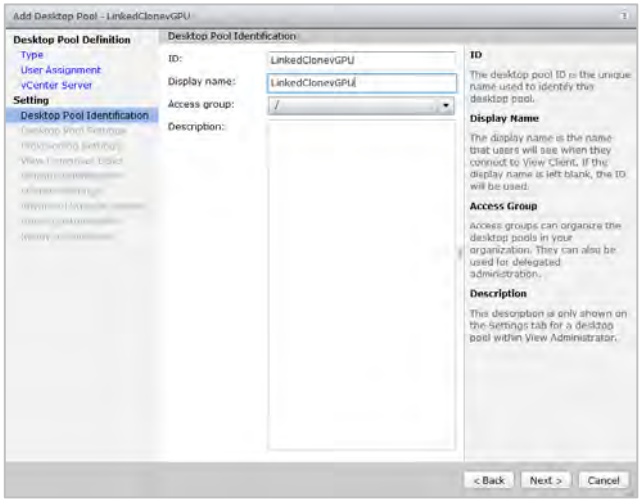
6. Select Automated Desktop Pool, then click Next. The wizard displays the User assignment screen:



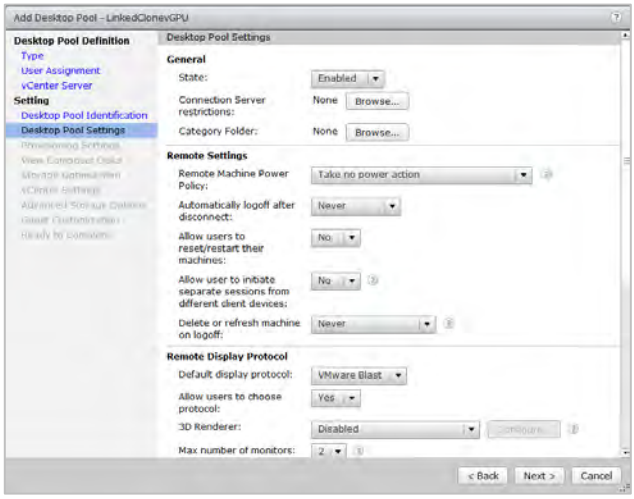
7. Click Floating, then click Next. The wizard displays the vCenter Server screen:



8. Select Horizon Composer linked clones. Verify the vSphere Server and Horizon Composer are correct, then click Next. The wizard displays the Desktop Pool Identification screen:



9. Enter a unique ID and a display name for the linked pool, then click Next. The wizard displays the Desktop Pool Settings screen:

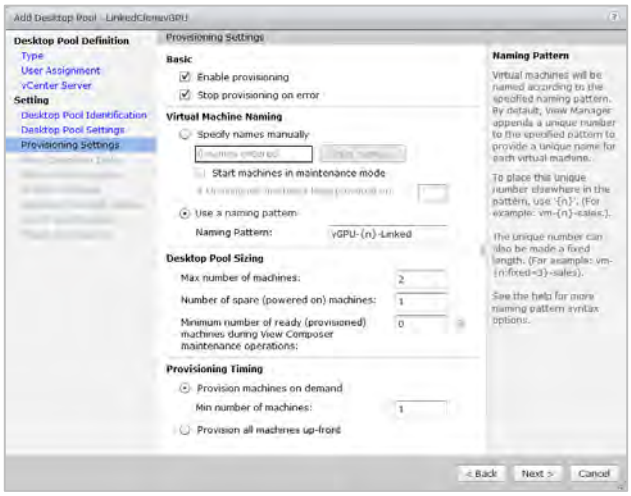


10. Set Remote Display Protocol to Blast.

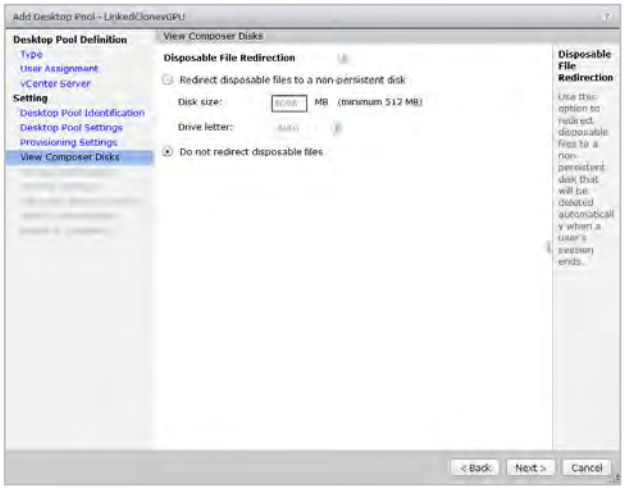
    Set Allow users to choose protocol to Yes.

    Set 3D Renderer to NVIDIA GRID VGPU. (In this version, HTML Access is fully supported and can be set to Enabled.)

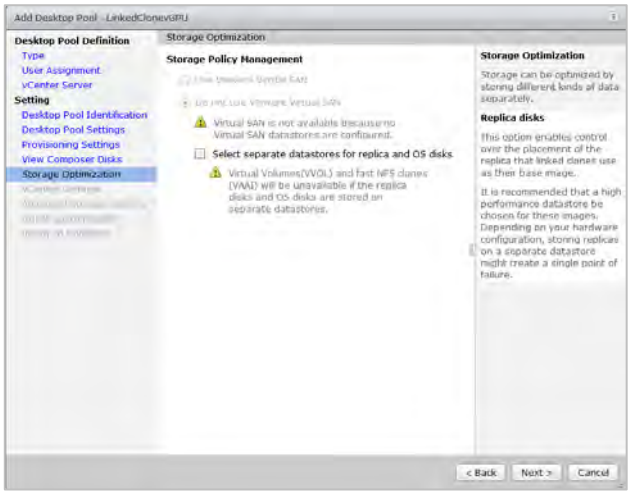
    Click Next. The wizard displays the Provisioning Settings screen:



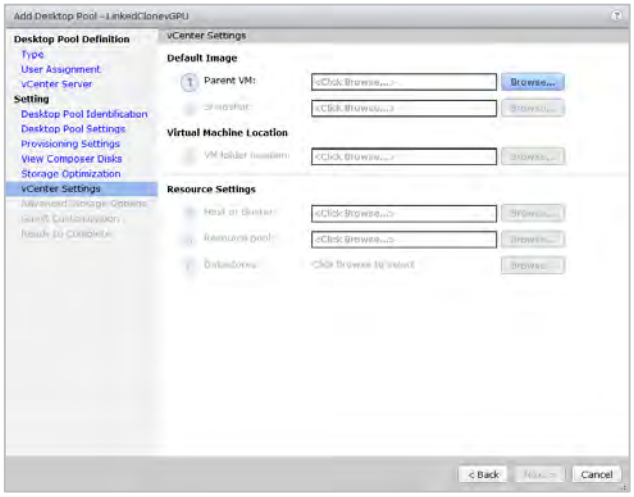
11. Configure the provisioning settings to conform to your infrastructure and testing needs, then click Next. The wizard displays the Horizon Composer Disks screen:



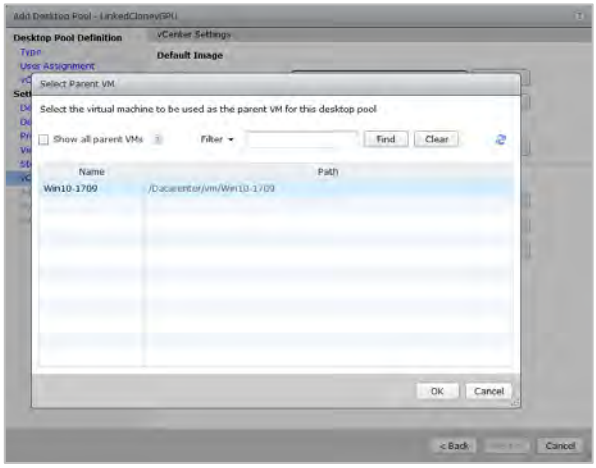
12. For simplicity, select Do not redirect disposable files, then click Next. The wizard displays the Storage Optimization screen:



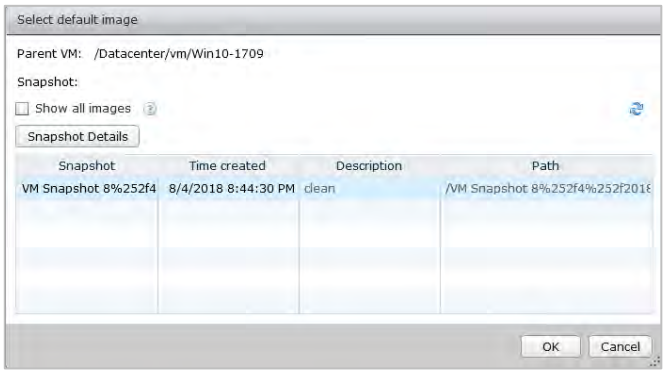
13. No changes are needed in this window. Click Next. The wizard displays the vCenter Settings screen:



14. Click the Parent VM > Browse button to display the Select Parent VM dialog:

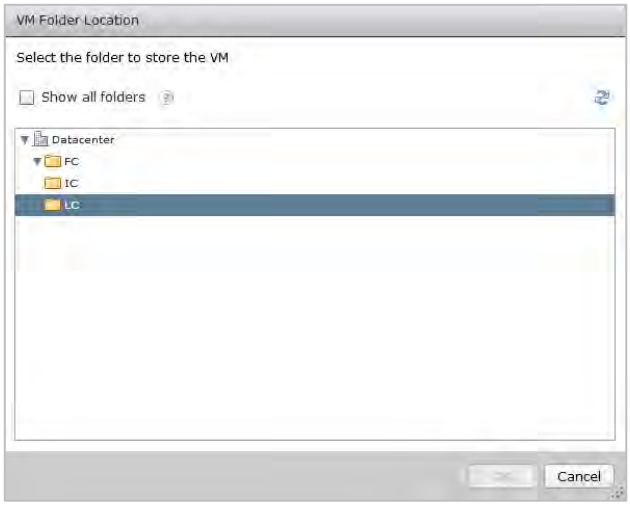


15. Select the gold master image VM. You can use the Filter option to condense the display if there are too many VMs to search easily. Click OK to close the dialog.
16. Click the Snapshot > Browse button to display the Select Default Image dialog:



17. Select a snapshot to provide the default image, then click OK to close the dialog.

18. Click the VM Folder > Browse button to display the VM folder location dialog:

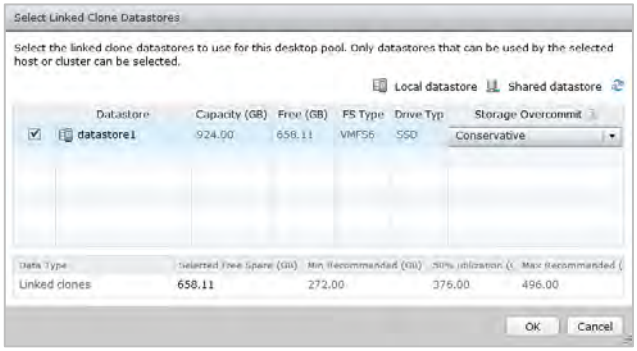


19. Navigate to VM Folder location and select a folder, then click OK to close the dialog.
20. Click Select a Host or Cluster > Browse to open the Host or Cluster dialog:

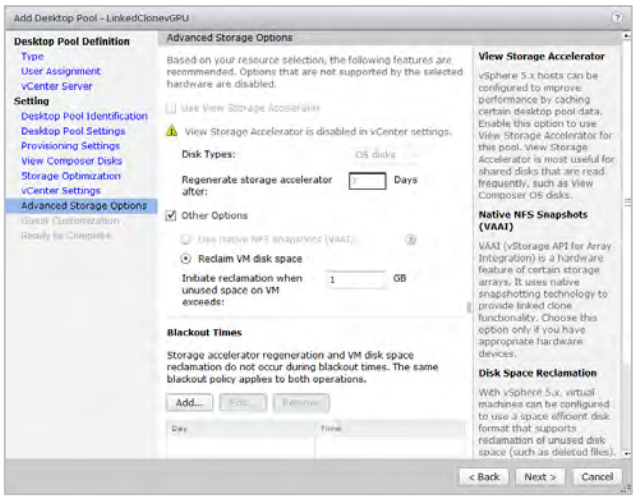
21. Navigate to Host or Cluster and select a host or cluster, then click OK to close the dialog.

22. Click Resource pool > Browse to open the Resource Pool dialog:

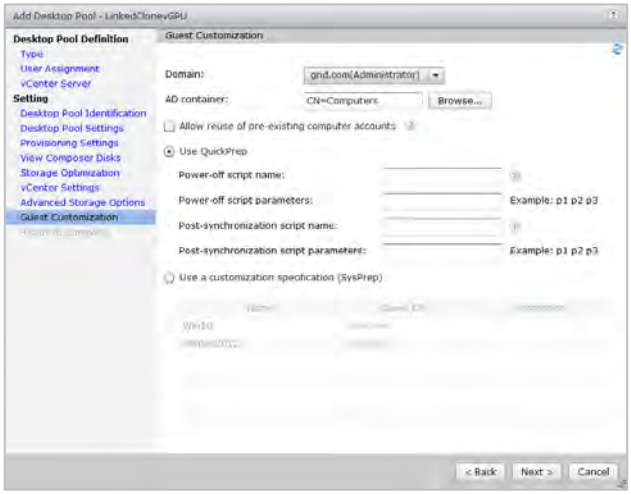
23. Navigate to and select a Resource Pool, then click OK to close the dialog.
24. Click Datastore > Browse to open the Select Linked Close Datastores dialog:



25. Select a datastore for the linked clone pool, then click OK to close the dialog.
26. Click Next to leave the vCenter Settings screen. The wizard displays the Add Desktop Pool screen:

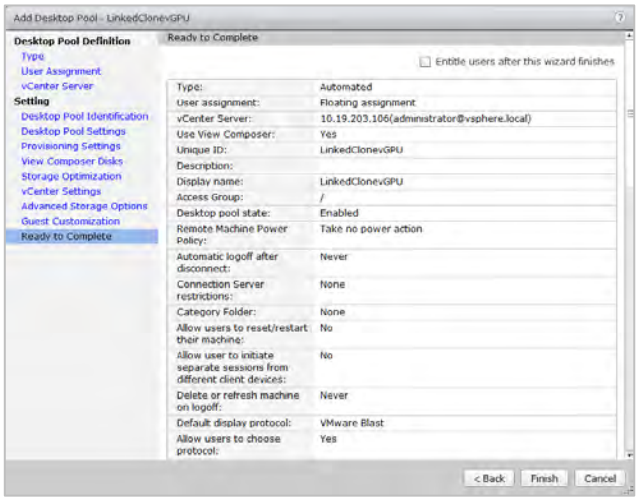


27. To improve performance and efficiency, check the Horizon Storage Accelerator checkbox and click the Reclaim VM disk space radio button if those controls are enabled. Then click Next. The wizard displays the Guest Customization screen:

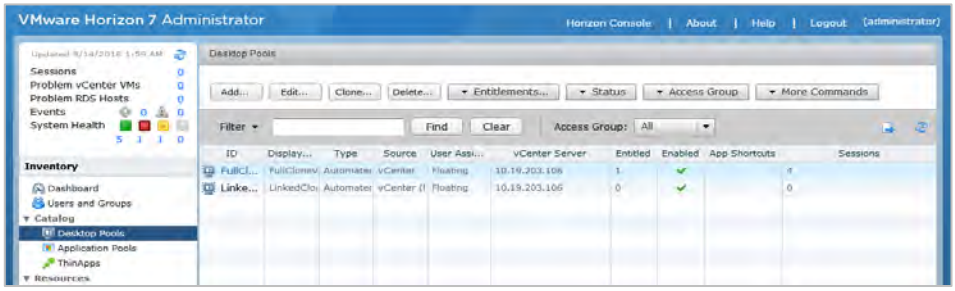


28. Click the Use QuickPrep radio button. (QuickPrep is a VMware system tool executed by Horizon Composer during a linked-clone desktop deployment. QuickPrep personalizes each desktop created from the Master Image.)

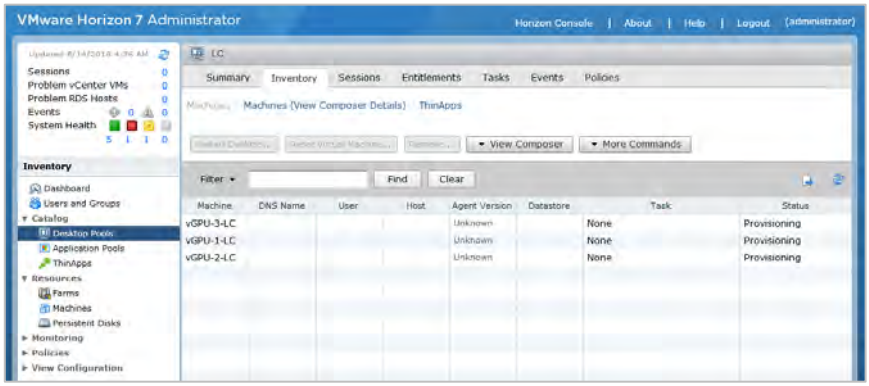
    Click Next. The wizard displays the Ready to Complete screen, which shows the settings you have entered for the new linked pool:



29. Review the settings. If any are incorrect, click Back to correct them. When all of them are correct, click Finish. The wizard creates the linked pool and displays the Desktop Pools window, which displays all of the desktop pools that have been defined:

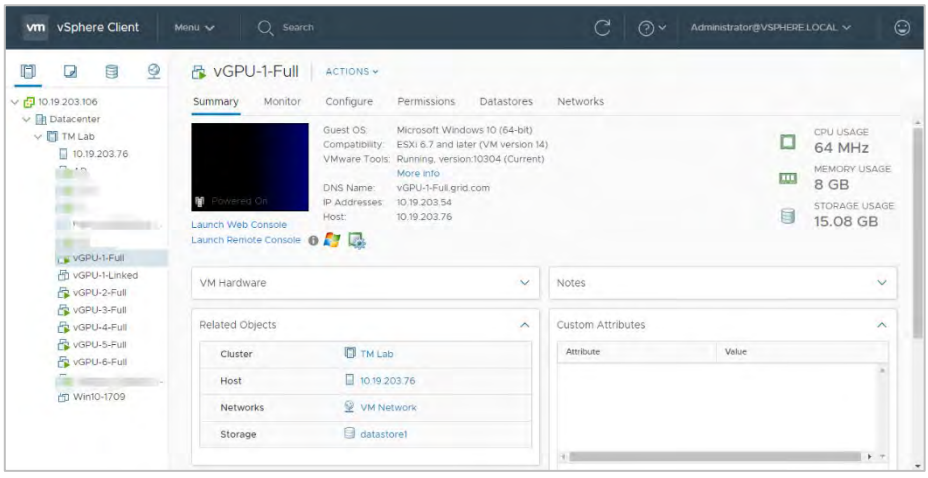


30. Monitor linked pool creation status. Refresh the window by clicking the recycle icon at the top of the left-hand pane.
31. Click Machines in the left-hand pane, under the heading “Inventory,” then select the Inventory tab, which displays the status of the virtual machines that have been defined:



32. When the Status column shows that all VMs are Connected, the VDI desktops are ready.

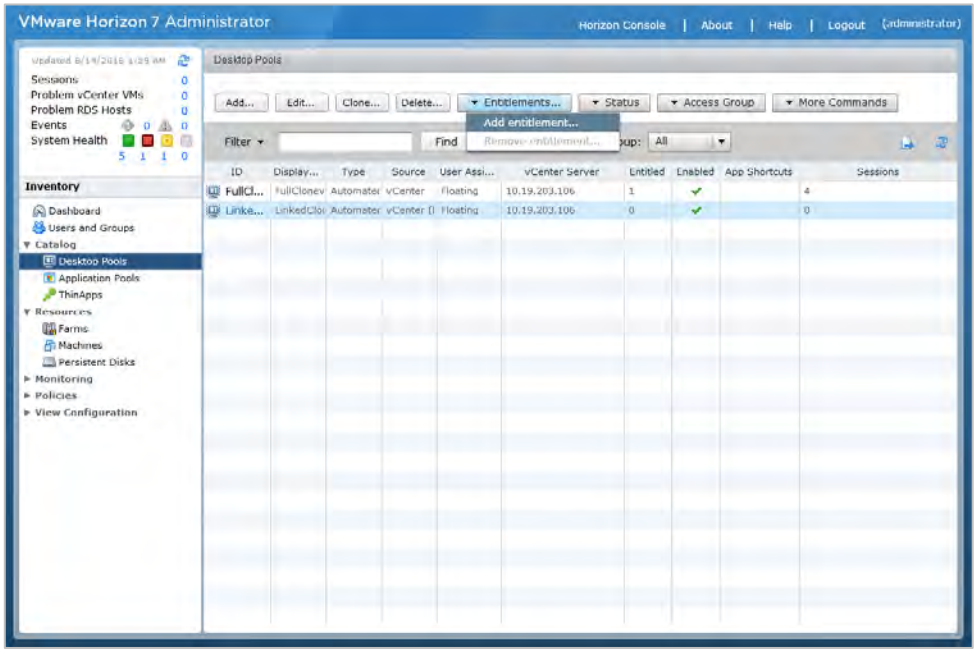
    At this point the desktop VMs appear in vSphere Web Client:



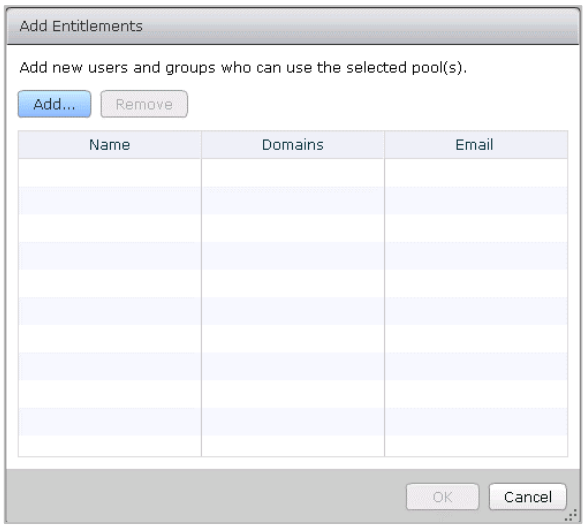
10.6 Enabling User Access to Desktop Pools

To entitle users to access VMs in a desktop pool:

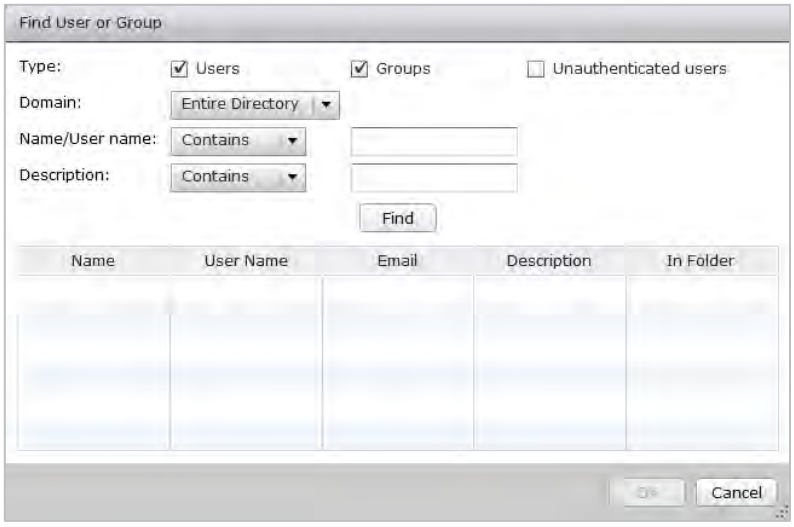
1. Start or return to VMware Horizon Administrator and click Desktop Pools in the left-hand pane:



2. Click the Entitlements button, then select Add entitlement from the resulting dropdown menu. VHA opens the Add Entitlements dialog:



3. Click the Add button. VHA displays the Find User or Group dialog:



4. Check the Users checkbox or the Groups checkbox (or both) to display the defined users or groups (or both),

   Enter a string in the Name/Username field or the Description field (or both) to limit the display to users and/or groups whose name or description (or both) match that string.

   By default, “matching the string” means “containing the string.” Possible matching criteria are “Contains,” “Begins with,” etc. You can change the criteria for a match on either field by selecting a different entry in the dropdown between the field’s label and the field’s matching string.

5. When the Find User or Group contains the exact set of users and/or groups that you want to entitle, click OK to entitle them and close the dialog.

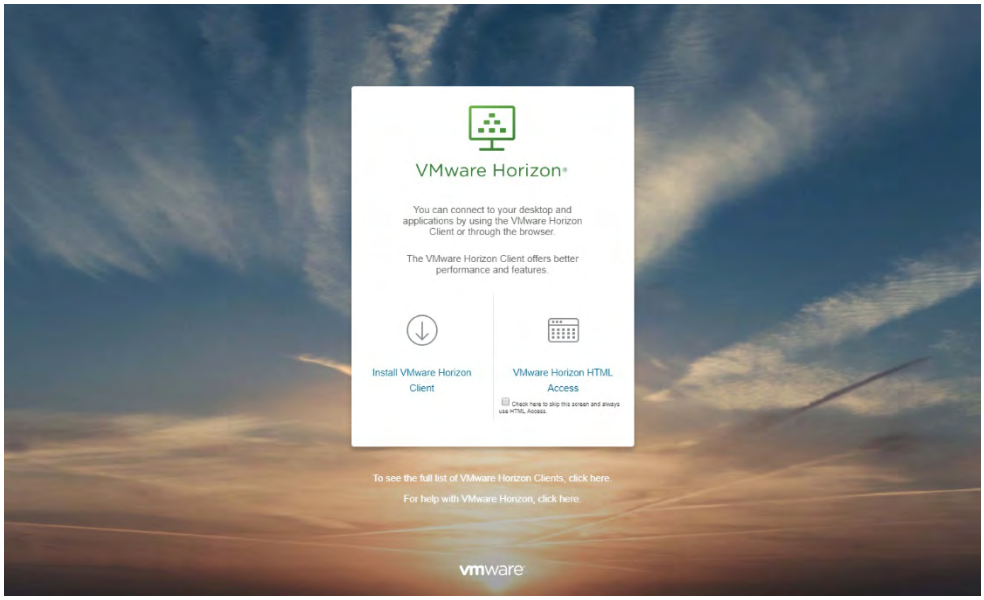
Chapter 11. VMware Horizon Client

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

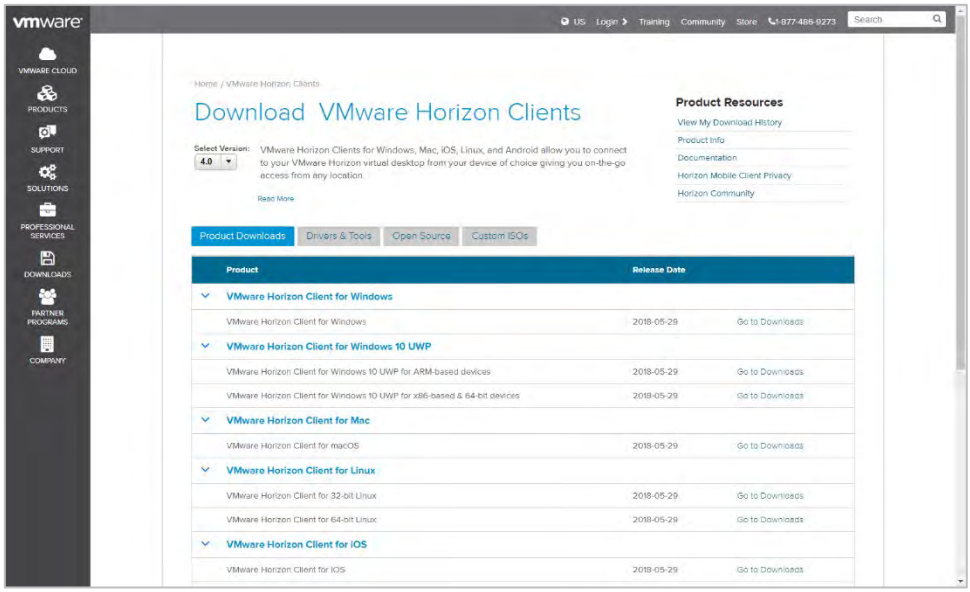
11.1 Installing VMware Horizon Client

To install VMware Horizon Client:

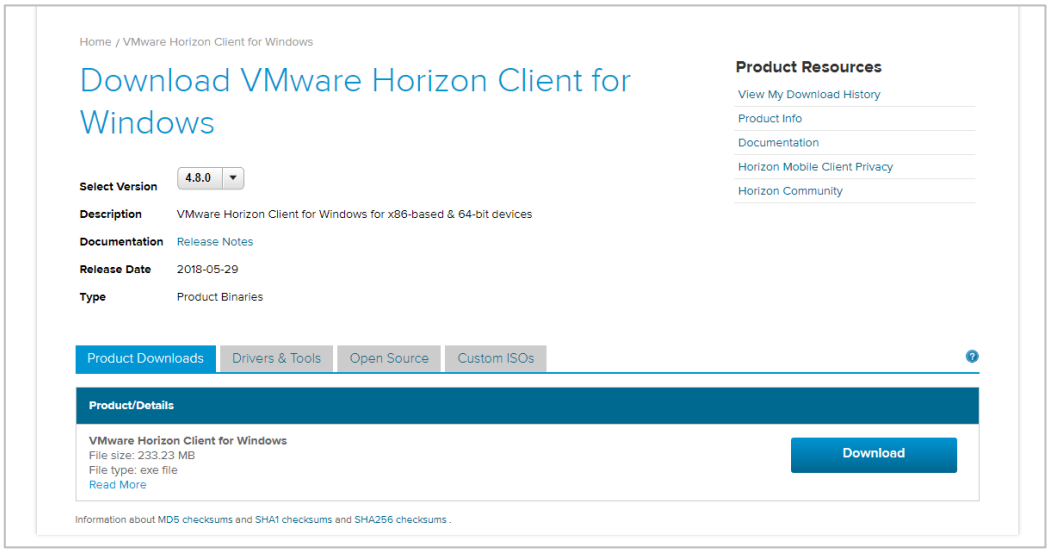
1. Log into the physical device, then open an internet browser. Navigate to the URL of the Horizon Connection Server installed earlier in this section. Click Install VMware Horizon Client:



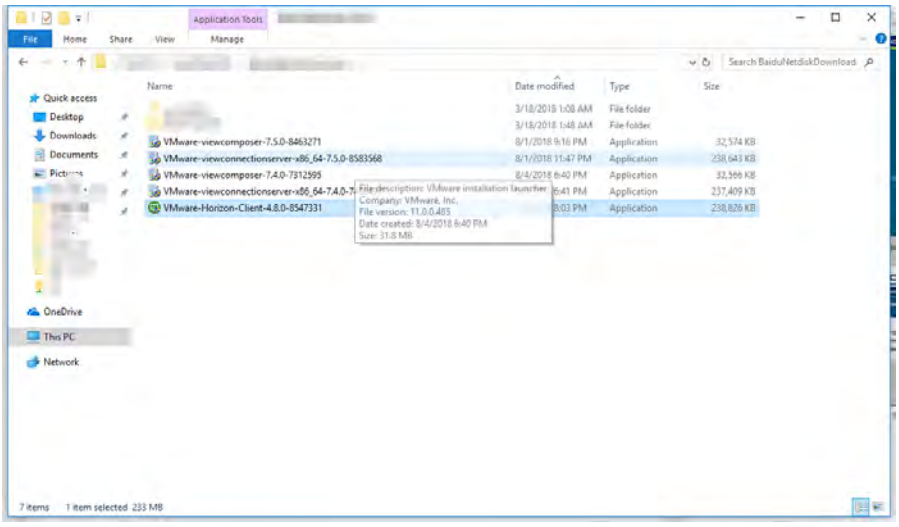
2. The browser opens the Download VMware Horizon Clients page:



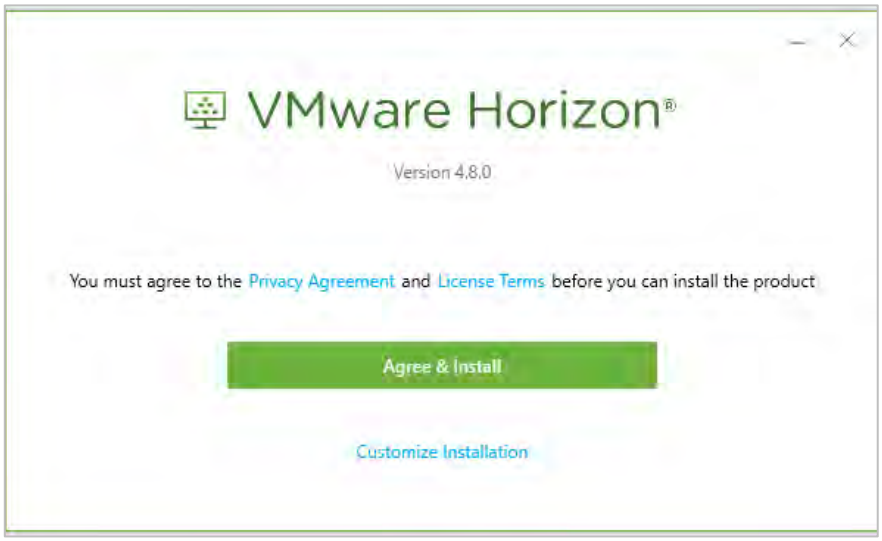
3. Locate the download for the appropriate OS and click Go to Downloads to its right. The browser opens the Download VMware Horizon Client for… page:



4. Click Download to begin downloading:

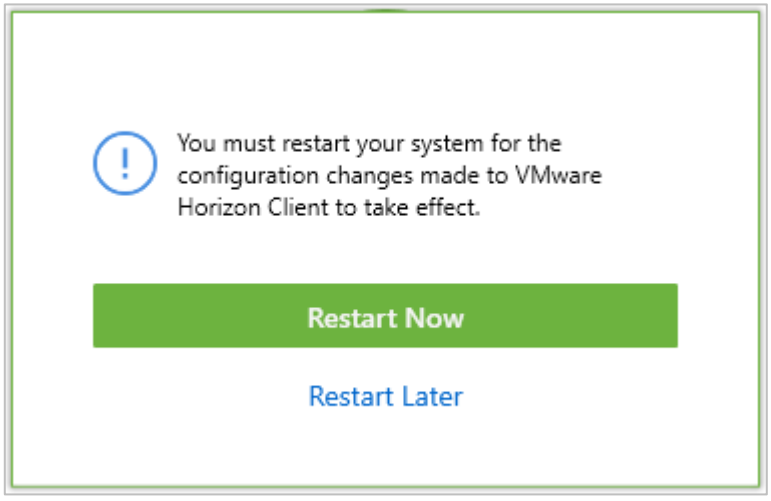


When the download is complete, locate the downloaded installer and double click it to begin installation. The installer prompts you to accept the privacy agreement and license:



5. Click Agree & Install to continue installation. When the installation is done, the installer displays the Success dialog:

6. Click Finish to complete installation. The installer displays the Restart dialog:

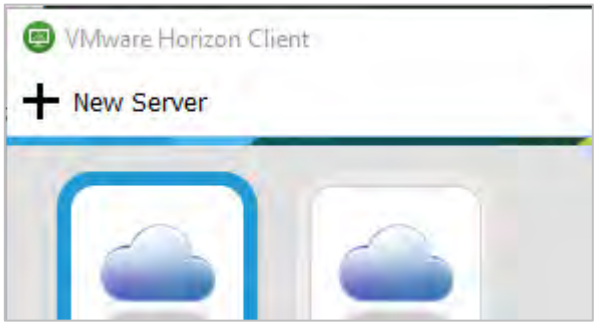


7. Click Restart Now to restart the physical desktop. When the restart is complete, log in to see Horizon Client installed.

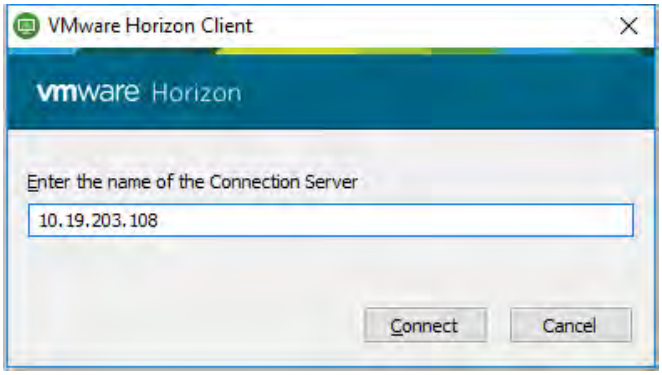
11.2 Configuring VMware Horizon Client Connection

To configure VMware Horizon Client:

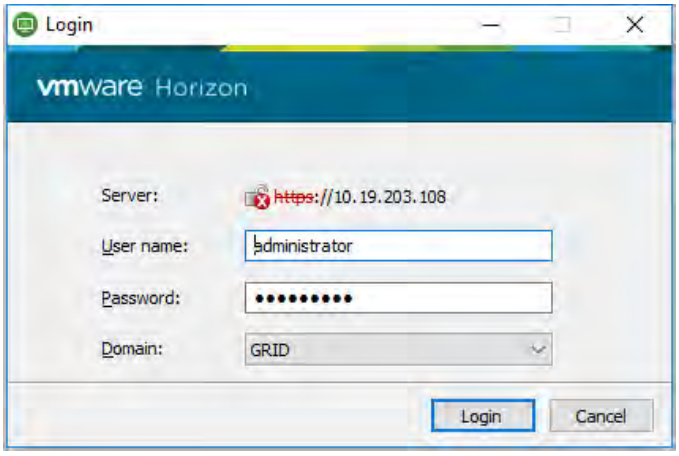
1. Start VMware Horizon Client on the desktop from which you want to connect. The application displays a splash screen, then its home window:



2. Click Add New Server to register a new virtual machine connection. VMware Horizon Client prompts you to enter the name of the server:



3. Enter the IP address of the VDI desktop to which you want to connect, then click Continue. (You are connecting directly to a desktop, so don't enter the Horizon Connection Broker address at this time.) VMware Horizon Client displays a log-in dialog:



4. Enter the local username and password (or the domain user's username and password if the virtual machine is a domain member) and click Login.

Chapter 12. Troubleshooting

This chapter provides links and examples for learning more about the VMware products described in this deployment guide and for finding solutions or assistance.

Before troubleshooting or filing a bug report, review the release notes for information about known issues and potential workarounds in the current release.

12.1 Forums

NVIDIA forums are a very extensive source of solutions to problems that you may face when deploying a virtualized environment. NVIDIA recommends that you search the NVIDIA Virtual GPU Forums first.

You may also want to look through the NVIDIA Enterprise Services Knowledgebase for articles and links that provide further support.

Some issues with your deployment may not be answered in the NVIDIA vGPU forums. You may also need to reference forums from your hardware supplier, VMware, or your users' applications. Here are some examples of key forums to consult:

- 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 you must file a bug or request support assistance, it is critical to include information about your computing environment so the technical staff can help you resolve the issue. NVIDIA includes the `nvidia-bug-report.sh` script in the `.vib` installation package to collect and transmit 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
- Identities of virtual machines that have vGPU or vSGA configured
- NSMI output

When you run this script you may specify the output location for the bug report using either the `-o` or `--output` switch, followed by the output file's name. If you do not specify an output directory, the script writes the bug report to the current directory.

If you do not specify a filename, the script uses the default name `nvidia-bug-report.log.gz`.

If the selected directory already contains a bug report file, the script changes that report file's name to `nvidia-bug-report.log.old.gz` before generating a new `nvidia-bug-report.log.gz` file.

To collect a bug report, enter 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 a problem to NVIDIA.

Appendix A. About This Document

A.1. Related Documentation

NVIDIA publishes several other documents that are helpful to users of VMware Hypervisor with NVIDIA vSPU software. See the NVIDIA Virtual GPU (vGPU) resources page for additional information about NVIDIA vGPU technology, including:

- NVIDIA Virtual GPU Technology
- Purchasing Guide for NVIDIA vGPU Solutions

  NVIDIA GRID GPU Datasheets
- Application Deployment Guides and Solution Overviews
- *Customer success stories*
- *Relevant White Papers*
- *Relevant videos*

A.2. Support Contact Information

NVIDIA and other vendors provide several technical support resources to assist you:

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

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

You can also reach out to your local VMware Horizon and NVIDIA vGPU teams for guidance.

Should you find a bug, please follow the instructions in section 12.2 to create a bug report and submit it to the VMware vGPU Community site.

If you want support when architecting your solution, your VMware Horizon and NVIDIA vGPU teams are available to assist. After you contact them, make sure you keep them up to date with your progress. If you do not know your account management teams, send email to the appropriate group:

- NVIDIA vGPU team: gridteam@nvidia.com

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