

DEPLOYING HARDWARE- ACCELERATED GRAPHICS WITH VMWARE HORIZON 7

Horizon 7 version 7.x

Table of Contents

Introduction	4
What Is Horizon 7?	4
Audience	4
Types of Graphics Acceleration	4
Virtual Shared Graphics Acceleration	5
Virtual Shared Pass-Through Graphics Acceleration	5
Virtual Dedicated Graphics Acceleration	6
Comparison of the Types of Graphics Acceleration	7
Hardware Requirements for Hardware-Accelerated Graphics	8
Use Cases for Hardware-Accelerated Graphics	8
Knowledge Workers	8
Power Users	8
Designers	9
Installation, Configuration, and Setup	10
ESXi 6.x Host	10
Installing and Configuring the ESXi Host for vSGA or vGPU	10
Installing and Configuring the ESXi Host for MxGPU	11
Installing and Configuring the ESXi Host for vDGA	12
Virtual Machine	13
General Settings for Virtual Machines	13
Virtual Machine Settings for vSGA	14
Virtual Machine Settings for vGPU	15
Virtual Machine Settings for MxGPU and vDGA	16
Guest Operating System	18
Windows Guest Operating System	18
Red Hat Enterprise Linux Operating System (vGPU and vDGA)	19
Horizon 7 version 7.x Pool and Farm Settings	21
License Server	22
Resource Monitoring	23
gpvm	23
nvidia-smi	23

Troubleshooting	24
General Troubleshooting for Graphics Acceleration	24
Verify That the GPU Driver Loads	24
Verify That Display Devices Are Present in the Host	25
Check the PCI Bus Slot Order	25
Check Xorg Logs	25
Troubleshooting Specific Issues in Graphics Acceleration	26
Summary	28
Additional Resources	29
About the Authors	29

Introduction

Engineers, designers, and scientists have traditionally relied on dedicated graphics workstations to perform the most demanding tasks, such as manipulating 3D models and visually analyzing large data sets. These standalone workstations carry high acquisition and maintenance costs. In addition, in industries such as oil and gas, space exploration, aerospace, engineering, scientific research, and manufacturing, users with these advanced requirements must be in the same physical location as the workstation.

Moving the graphics-acceleration hardware from the workstation to a server provides users with compute, memory, networking, and security advantages. With this data center architecture, users can access and manipulate complex models and very large data sets from virtually anywhere. With appropriate network bandwidth and suitable remote client devices, IT can offer the most advanced users an immersive 3D-graphics experience while freeing them from the limitations of fixed workstations and processors. Fewer physical resources are needed, the wait time to open complex models or run simulations is reduced, and users are no longer tied to a single physical location. In addition to handling the most demanding graphical workloads, hardware acceleration can also reduce CPU usage for less demanding basic desktop and published application usage, and for video encoding or decoding, which includes the default Blast Extreme remote display protocol.

This paper describes the three types of hardware-accelerated graphics in View virtual desktops in [VMware Horizon® 7](#) through typical use cases. It also includes installation and configuration instructions, best practices, and troubleshooting tips.

What Is Horizon 7?

[VMware Horizon 7](#) provides a virtual desktop solution and an enterprise-class application-publishing solution. Horizon 7 features and components, such as the Blast Extreme display protocol, instant-clone provisioning, VMware App Volumes™ application delivery, and VMware User Environment Manager™, are also integrated into Remote Desktop Session Host (RDSH) sessions to provide a seamless user experience and an easy-to-manage, scalable solution.

Audience

This white paper is for administrators deploying hardware-accelerated graphics in Horizon 7 or for anyone interested in the technology.

Types of Graphics Acceleration

Horizon 7 offers three types of graphics acceleration:

- [Virtual Shared Graphics](#)
- [Virtual Shared Pass-Through Graphics](#)
- [Virtual Dedicated Graphics](#)

Virtual Shared Graphics Acceleration

Virtual Shared Graphics Acceleration (vSGA) allows you to share a GPU across multiple virtual desktops. It is an attractive solution for users who require the GPU's full potential during brief periods. However, vSGA can create bottlenecks depending on which applications are used and the resources the applications require. This type of graphics acceleration is generally used for knowledge workers and occasionally for power users.

With vSGA, the host's physical GPUs are virtualized and shared across multiple guest virtual machines. You must install a vendor driver in the hypervisor. Each guest virtual machine uses a proprietary VMware vSGA 3D driver that communicates with the vendor driver in VMware vSphere®. Drawbacks of vSGA are that applications might need to be recertified to be supported, API support is limited, and support is restricted for OpenGL and DirectX.

Supported vSGA cards for Horizon 7 version 7.x and vSphere 6.5 include

- Intel Iris Pro Graphics P580
- NVIDIA Tesla M10/M60/P40

For a list of compatible vSGA cards, see the [VMware Virtual Shared Graphics Acceleration Guide](#).

Virtual Shared Pass-Through Graphics Acceleration

Virtual Shared Pass-Through Graphics Acceleration allows you to share a GPU across multiple virtual desktops instead of focusing on only one user. Unlike vSGA, it does not use the proprietary VMware 3D driver, and most graphics card features are supported.

You must install the appropriate vendor driver on the guest virtual machine. All graphics commands are passed directly to the GPU without having to be translated by the hypervisor. On the hypervisor, a vSphere Installation Bundle (VIB) is installed, which aids or performs the scheduling. Depending on the card, up to 24 virtual machines can share a GPU, and some cards have multiple GPUs. Calculating the number of desktops or users per GPU depends on the type of card, application requirements, screen resolution, number of displays, and frame rate, measured in frames per second (fps).

The amount of frame buffer (VRAM) per virtual machine (VM) is fixed, and the GPU engines are shared between VMs. AMD has an option to have a fixed amount of compute, which is called *predictable performance*.

Virtual shared pass-through technology provides better performance than vSGA and higher consolidation ratios than Virtual Dedicated Graphics Acceleration (vDGA). It is a good technology for low-, mid-, and advanced-level engineers and designers and power users with 3D application requirements. Its drawbacks are the lack of VMware vSphere vMotion® support and that the technology might require applications to be recertified to be supported.

Supported shared pass-through cards for Horizon 7.x and vSphere 6.5 include

- AMD FirePro S7100X/S7150/S7150X2 (multi-user GPU, or MxGPU)
- NVIDIA Tesla M10/M60/P40 (virtual GPU, or vGPU)

For a full list of compatible shared pass-through GPU cards, see the [VMware Shared Pass-Through Graphics Guide](#).

Virtual Dedicated Graphics Acceleration

Virtual Dedicated Graphics Acceleration (vDGA) technology, also known as *GPU pass-through*, provides each user with unrestricted, fully dedicated access to one of the host's GPUs. Although dedicated access has some consolidation and management trade-offs, vDGA offers the highest level of performance for users with the most intensive graphics computing needs.

The hypervisor passes the GPUs directly to individual guest virtual machines. No special drivers are required in the hypervisor. However, to enable graphics acceleration, you must install the appropriate vendor driver on each guest virtual machine. The installation procedures are the same as for physical machines. One drawback of vDGA is its lack of vMotion support.

Supported vDGA cards in Horizon 7 version 7.x and vSphere 6.5 include

- AMD FirePro S7100X/S7150/S7150X2
- Intel Iris Pro Graphics P580/P6300
- NVIDIA Quadro M5000/P6000, Tesla M10/M60/P40

For a list of partner servers that are compatible with specific vDGA devices, see the [VMware Virtual Dedicated Graphics Acceleration \(vDGA\) Guide](#).

Comparison of the Types of Graphics Acceleration

Table 1 compares the features of the three types of graphics acceleration..

TYPE	VIRTUAL SHARED GRAPHICS ACCELERATION	VIRTUAL SHARED PASS-THROUGH GRAPHICS ACCELERATION	VIRTUAL DEDICATED GRAPHICS ACCELERATION
Abbreviation	vSGA	vGPU or MxGPU	vDGA
Consolidation	High (limited by video memory)	Up to 1:32	None (1:1)
Performance level	Lightweight	Lightweight or Workstation	Workstation
Compatibility	Limited	Full, but not all applications are certified	Maximum
DirectX level	9.0c SM3 only	All supported versions	All supported versions
OpenGL version	2.1 only	All supported versions	All supported versions
Video encoding and decoding	Software	Hardware	Hardware
OpenCL or CUDA compute	No	MxGPU: OpenCL only GRID 1: No GRID 2: 1:1 only	Yes
vSphere vMotion support	Yes	No	No

Table 1: Feature Comparison for the Types of Graphics Acceleration

Hardware Requirements for Hardware-Accelerated Graphics

The hardware requirements for graphics acceleration solutions are listed in Table 2.

COMPONENT	DESCRIPTION
Physical space for graphics cards	Many high-end GPU cards are full height, full length, and double width and take up two slots on the motherboard, but use only a single PCIe x16 slot. Verify that the host has enough room to hold the GPU card in the appropriate PCIe slot.
Host power supply unit (PSU)	Make sure that the PSU meets the GPU's power requirements and contains the proper power cables. For example, a single NVIDIA K2 GPU can use as much as 225 watts of power and requires either a 6-pin or 8-pin PCIe power cord.
BIOS	For MxGPU, single-root IO virtualization (SR-IOV) must be enabled. For vGPU, Intel Virtualization Technology support for Direct I/O (Intel VT-d) or AMD IO memory management unit (IOMMU) must be enabled. To locate these settings in the server BIOS, contact the hardware vendor.
Display adapters	If the host does not have an extra graphics adapter, VMware recommends that you install an additional low-end display adapter to act as the primary display adapter, because the VMware ESXi™ console display adapter is not available to Xorg. If the GPU is set as the primary adapter, Xorg cannot use the GPU for rendering. If two GPUs are installed, the server BIOS might have an option to select which GPU is primary and which is secondary.

Table 2: Hardware Requirements for Hardware-Accelerated Graphics

Use Cases for Hardware-Accelerated Graphics

The following typical use cases offer guidelines for the different types of graphics acceleration.

Knowledge Workers

Office workers and executives fall into the knowledge-worker category, typically using applications such as Microsoft Office, Adobe Photoshop, and other non-specialized end-user applications.

Because the graphical load of these users is expected to be low, consolidation is important. These users are best matched with one of the following types of graphics acceleration.

- **Virtual Shared Pass-Through Graphics Acceleration (MxGPU or vGPU)** – When performance and features, such as hardware video encoding and decoding, or DirectX/OpenGL levels, matter most.
- **vSGA** – When consolidation matters most.

Power Users

Power users consume more complex visual data, but their requirements for manipulations of large datasets and specialized software are less intense than for designers, or they use only viewers, like Autodesk DWG TrueView.

Power users are best matched with Virtual Shared Pass-Through Graphics Acceleration (MxGPU or vGPU).

Designers

Designers and advanced engineering and scientific users often create and work with large, complex datasets. They require graphics-intensive applications, such as 3D design, molecular modeling, and medical diagnostics software from companies such as Dassault Systèmes, Enovia, Siemens NX, and Autodesk.

Designers are best matched with one of the following.

- **Virtual Shared Pass-Through Graphics Acceleration (MxGPU or vGPU)** – When availability or consolidation matters most.
- **vDGA** – When every bit of performance counts.

Figure 1 summarizes the performance and consolidation profiles of the three types of graphics acceleration.

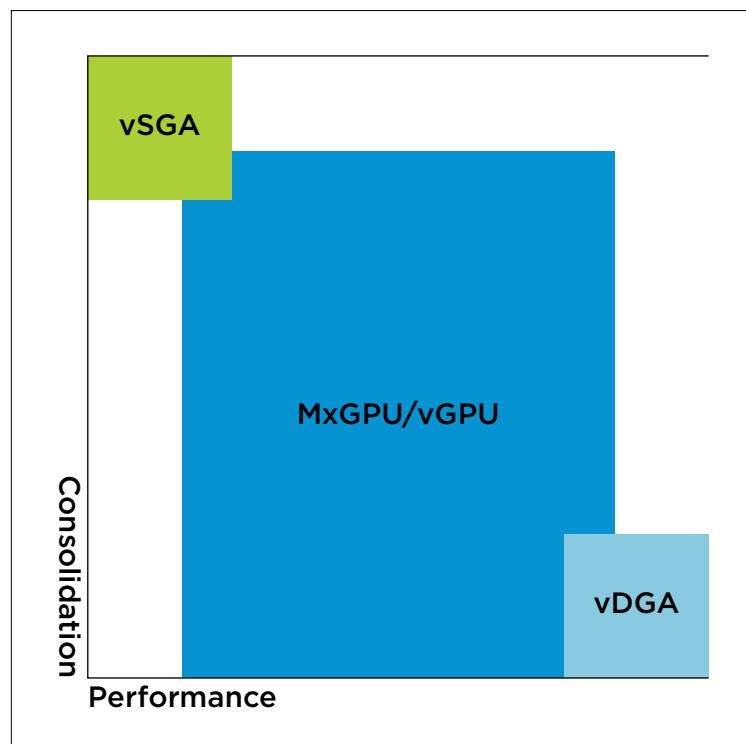


Figure 1: Consolidation and Performance Overview

Installation, Configuration, and Setup

For graphics acceleration, you need to install and configure the following components:

- ESXi 6.x host
- Virtual machine
- Guest operating system
- Horizon 7 version 7.x desktop pool settings
- License server

ESXi 6.x Host

Installing the graphics card and configuring the ESXi host vary based on the type of graphics acceleration.

Installing and Configuring the ESXi Host for vSGA or vGPU

1. Install the graphics card on the ESXi host.
2. Put the host in maintenance mode.
3. If you are using an NVIDIA Tesla P card, disable ECC.
4. If you are using an NVIDIA Tesla M card, set the card to `graphics` mode (the default is `compute`) using `GpuModeSwitch`, which comes as a bootable ISO or a VIB.

- a. Install `GpuModeSwitch` without an NVIDIA driver installed:

```
esxcli software vib install --no-sig-check -v
  /<path_to_vib>/NVIDIA-GpuModeSwitch-
  1OEM.xxx.0.0.xxxxxxx.x86_64.vib
```

- b. Reboot the host.

- c. Change all GPUs to graphics mode:

```
gpumodeswitch --gpumode graphics
```

- d. Remove `GpuModeSwitch`:

```
esxcli software vib remove -n NVIDIA-
  VMware_ESXi_xxx_GpuModeSwitch_Driver
```

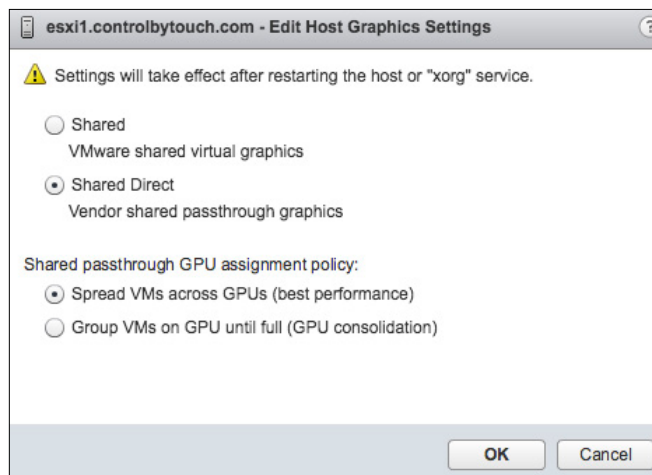
5. Install the GPU VIB:

```
esxcli software vib install -v /<path_to_vib>\NVIDIA-
  VMware_ESXi_xxx_Host_Driver_xxx.xx-1OEM.xxx.0.0.xxxxxxx.vib
```

If you are using ESXi 6.0, vSGA and vGPU have different VIB files. If you are using ESXi 6.5, both vSGA and vGPU use the same VIB file.

6. Reboot, and take the host out of maintenance mode.

7. If you are using an NVIDIA card and vSphere 6.5 or later, in the vSphere Web Client, navigate to **Host > Configure > Hardware > Graphics > Host Graphics > Edit** to open the Edit Host Graphics Settings window.



- a. For vGPU, select **Shared Direct**. For vSGA, select **Shared**.
- b. If you are using vGPU with different profiles per GPU, select **Group VMs on GPU until full (GPU consolidation)**. In this case, different profiles are placed on different GPUs, and same profiles are placed on the same GPU until it is full. This method prevents you from running out of free GPUs for different profiles.

Example:

The host has a single M60 card, which has two GPUs. Each GPU has 8 GB of memory. Two VMs with 4 GB of frame buffer and four VMs with 2 GB are trying to run. If the first two machines started have the same profile, they are placed on different GPUs. As a result, no GPU is available for the other profile. With **Group VMs on GPU until full (GPU consolidation)**, virtual machines with the same profile start on the same GPU.

Installing and Configuring the ESXi Host for MxGPU

1. Install the graphics card on the ESXi host.
2. Put the host in maintenance mode.
3. In the BIOS of the ESXi host, verify that single-root IO virtualization (SR-IOV) is enabled and that one of the following is also enabled.
 - Intel Virtualization Technology support for Direct I/O (Intel VT-d)
 - AMD IO memory management unit (IOMMU)
4. Browse to the location of the AMD FirePro VIB driver and AMD VIB install utility:


```
cd /<path_to_vib>
```
5. Make the VIB install utility executable, and execute it:

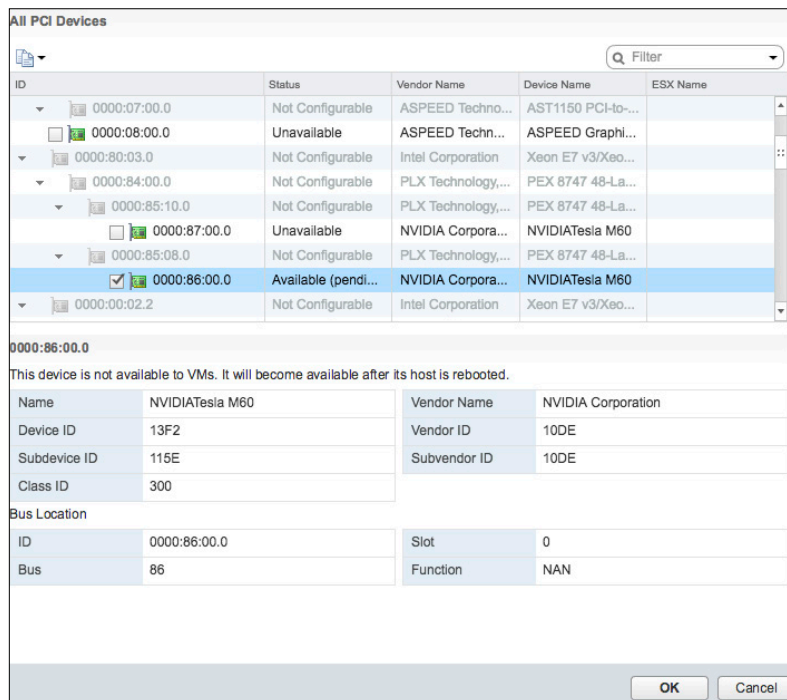

```
chmod +x mxgpu-install.sh && sh mxgpu-install.sh -i
```
6. In the script, select the option that suits your environment:


```
Enter the configuration mode ( [A]uto / [H]ybrid / [M]anual, default:A) A
```

7. For the number of virtual functions, enter the number of users you want to run on a GPU:
Please enter number of VFs: (default:4): 8
8. Choose whether you want keep performance fixed and independent of the number of active VMs:
Do you want to enable Predictable Performance? ([Y]es/[N]o,default:N)N
...
Done
The configuration needs a reboot to take effect
9. Reboot and take the host out of maintenance mode.

Installing and Configuring the ESXi Host for vDGA

1. Install the graphics card on the ESXi host.
2. In the BIOS of the ESXi host, verify that Intel VT-d or AMD IOMMU is enabled.
3. To enable pass-through for the GPU in the vSphere Web Client, navigate to **Host > Configure > Hardware > PCI Devices > Edit**.
4. In the All PCI Devices window, select the GPU, and reboot.



Virtual Machine

Configure the general settings for the virtual machine, and then configure it according to the type of graphics acceleration you are using.

General Settings for Virtual Machines

Hardware level – The recommended hardware level is the highest that all hosts support. The minimum is hardware level version 11.

CPU – The number of CPUs required depends on usage and is determined by actual workload. As a starting point, consider these numbers:

Knowledge workers: 2

Power users: 4

Designers: 6

Memory – The amount of memory required depends on usage and is determined by actual workload. As a starting point, consider these amounts:

Knowledge workers: 2 GB

Power users: 4 GB

Designers: 8 GB

Virtual network adapter – The recommended virtual network adapter is VMXNET3.

Virtual storage controller – The recommended virtual disk is LSI Logic SAS, but demanding workloads using local flash-based storage might benefit from using VMware Paravirtual.

Other devices – We recommend removing devices that are not used, such as a COM port, a printer port, DVD, or floppy.

Now that you have configured the general settings for the virtual machine, configure the settings for the type of graphics acceleration.

Virtual Machine Settings for vSGA

Configure the virtual machine as follows if you are using vSGA.

1. Enable 3D graphics by selecting **Enable 3D Support**.
2. Set the 3D Renderer to **Automatic** or **Hardware**.

Automatic uses hardware acceleration if the host that the virtual machine is starting in has a capable and available hardware GPU. If a hardware GPU is not available, the virtual machine uses software 3D rendering for 3D tasks. The Automatic option allows the virtual machine to be started on or migrated to (via vSphere vMotion) any host (vSphere version 5.0 or later) and to use the best solution available on that host.

Hardware uses only hardware-accelerated GPUs. If a hardware GPU is not present in a host, the virtual machine does not start, or you cannot perform a live vSphere vMotion migration to that host. Migration is possible as long as the host that the virtual machine is being moved to has a capable and available hardware GPU. The Hardware option guarantees that a virtual machine always uses hardware 3D rendering when a GPU is available, but it limits the virtual machine to using hosts that have hardware GPUs.

3. Select the amount of video memory (3D Memory).

3D Memory has a default of 96 MB, a minimum of 64 MB, and a maximum of 512 MB.

3D Graphics (*)	<input checked="" type="checkbox"/> Enable 3D Support
3D Renderer	Automatic
3D Memory	256 MB

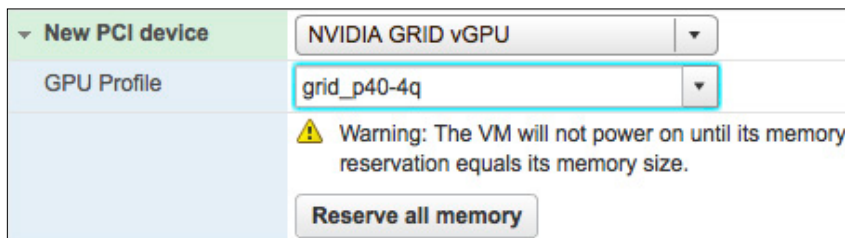
Virtual Machine Settings for vGPU

Configure the virtual machine as follows if you are using vGPU.

1. On the vSphere console, select your virtual machine, and navigate to **Edit Settings**.
2. Add a shared PCI device to the virtual machine, and select the appropriate PCI device to enable GPU pass-through on the virtual machine. In this case, select NVIDIA GRID vGPU.



3. From the GPU Profile drop-down menu, select the correct profile.



The last part of the GPU Profile string (4q in this example) indicates the size of the frame buffer (VRAM) in gigabytes and the required GRID license. For the VRAM, 0 means 512 MB, 1 means 1024 MB, and so on. So for this profile, the size is 4 GB. The possible GRID license types are:

b - GRID Virtual PC virtual GPUs for business desktop computing

a - GRID Virtual Application virtual GPUs for Remote Desktop Session Hosts

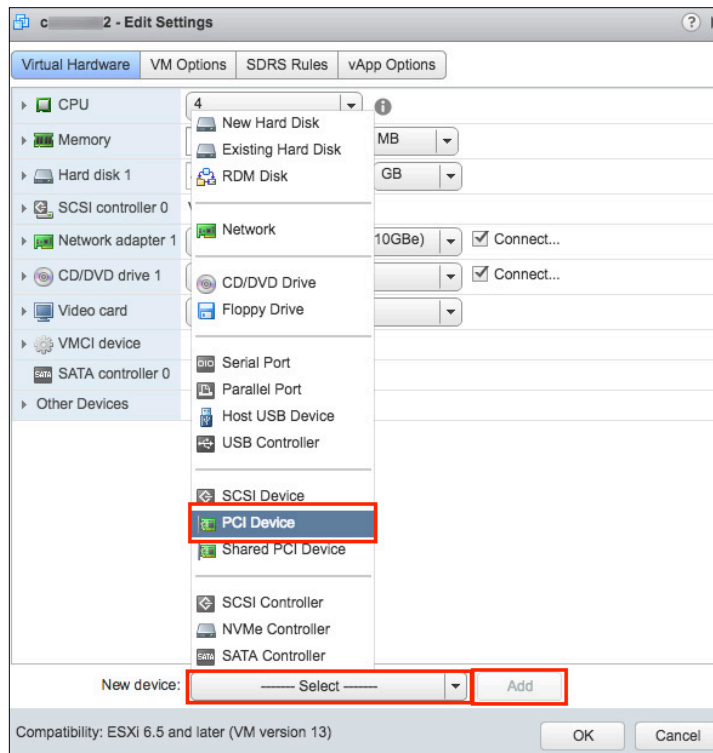
q - Quadro Virtual Datacenter Workstation (vDWS) for workstation-specific graphics features and accelerations, such as up to four 4K monitors and certified drivers for professional applications

4. Click **Reserve all memory**, which reserves all memory when creating the virtual machine.

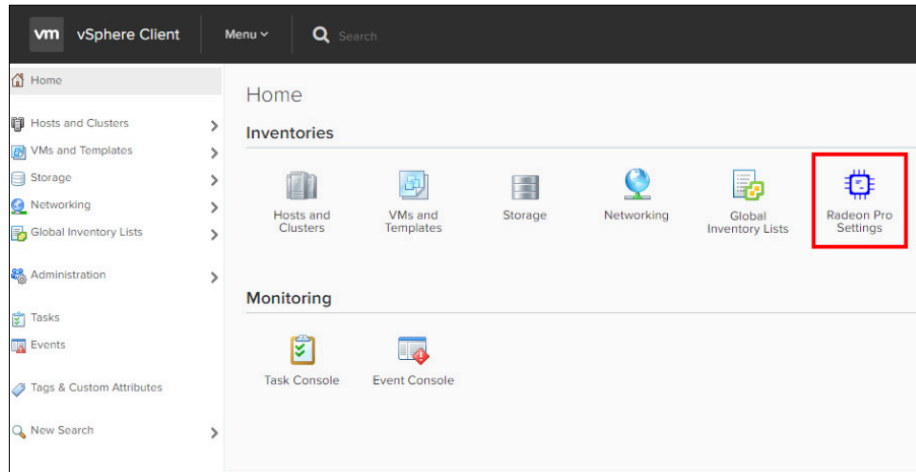
Virtual Machine Settings for MxGPU or vDGA

Configure the virtual machine as follows if you are using MxGPU or vDGA.

1. For devices with a large BAR size (for example, Tesla P40), you must use vSphere 6.5 and set the following advanced configuration parameters on the VM:
 - `firmware="efi"`
 - `pciPassthru.use64bitMMIO="TRUE"`
 - `pciPassthru.64bitMMIOSizeGB="64"`
2. Add a PCI device (virtual functions are presented as PCI devices) to the virtual machine, and select the appropriate PCI device to enable GPU pass-through.



With MxGPU, you can also do this by installing the Radeon Pro Settings for the VMware vSphere Client Plug-in.



To add a PCI device for multiple machines at once, from `ssh`:

- a. Browse to the AMD FirePro VIB driver and AMD VIB install utility:

```
cd /<path_to_vib
```

- b. Edit `vms.cfg`:

```
vi vms.cfg
```

Press `I` and change the instances of `.*` to match the names of your VMs that require a GPU. For example, to match `*MxGPU*` to VM names that include `MxGPU`, such as `WIN10-MxGPU-001` or `WIN8.1-MxGPU-002`:

```
.*MxGPU.*
```

To save and quit, press `Esc`, type `:wq`, and press `Enter`.

- c. Assign the virtual functions to the VMs:

```
sh mxgpu-install.sh -a assign
```

Eligible VMs:

```
WIN10-MxGPU-001
```

```
WIN10-MxGPU-002
```

```
WIN8.1-MxGPU-001
```

```
WIN8.1-MxGPU-002
```

These VMs will be assigned a VF, is it OK? [Y/N]y

- d. Press `Enter`.

3. Select **Reserve all guest memory (All locked)**.

Memory	4096	MB
Reservation	4096	MB
<input checked="" type="checkbox"/> Reserve all guest memory (All locked)		

Guest Operating System

Install and configure the guest operating system.

Windows Guest Operating System

For a Windows guest operating system, install and configure as follows.

1. Install Windows 7, 10, or 2012 R2, and install all updates.
2. The following installations are also recommended.
 - a. Install common Microsoft runtimes and features.

Before updating Windows in the VM, install the required versions of Microsoft runtimes that are patched by Windows Update and that can run side by side in the image. For example, install:

 - .NET Framework (3.5, 4.5, and so on)
 - Visual C++ Redistributables x86 / x64 (2005 SP1, 2008, 2012, and so on)
 - b. Install Microsoft updates.

Install the updates to Microsoft Windows and other Microsoft products with Windows Update or Windows Server Update Service. You might need to first manually install [Windows Update Client for Windows 8.1 and Windows Server 2012 R2: March 2016](#).
 - c. Tune Windows with the [VMware OS Optimization Tool](#) using the default options
3. If you are not using vSGA:
 - a. Obtain the GPU drivers from the GPU vendor (with vGPU, this is a matched pair with the VIB file).
 - b. Install the GPU device drivers in the guest operating system of the virtual machine. For MxGPU, make sure that the **GPU Server** option is selected.
4. Install VMware Tools™ and Horizon Agent (select **3D RDSH feature for Windows 2012 R2 Remote Desktop Session Hosts**) in the guest operating system.
5. Reboot the system.

Red Hat Enterprise Linux Operating System (vGPU and vDGA)

For a Red Hat Enterprise Linux guest operating system, install and configure as follows.

1. Install Red Hat Enterprise Linux 6.9 or 7.4 x64, install all updates, and reboot.
2. Install gcc, kernel makefiles, and headers:


```
sudo yum install gcc-c++ kernel-devel-$(uname -r) kernel-headers-$(uname -r) -y
```
3. Disable libvirt:


```
sudo systemctl disable libvirtd.service
```
4. Disable the open-source nouveau driver.
 - a. Open the following configuration file using vi:


```
sudo vi /etc/default/grub
```

 If you are using RHEL 6.x:


```
sudo vi /boot/grub/grub.conf
```
 - b. Find the line for GRUB_CMDLINE_LINUX, and add `blacklist=nouveau` to the line.
 - c. Add the line `blacklist=nouveau` anywhere in the following configuration file:


```
sudo vi /etc/modprobe.d/blacklist.conf
```
5. Generate new `grub.cfg` and `initramfs` files:


```
sudo grub2-mkconfig -o /boot/grub2/grub.cfg
sudo dracut /boot/initramfs-$(uname -r).img $(uname -r) -f
```
6. Reboot.
7. Install the NVIDIA driver, and acknowledge all questions:

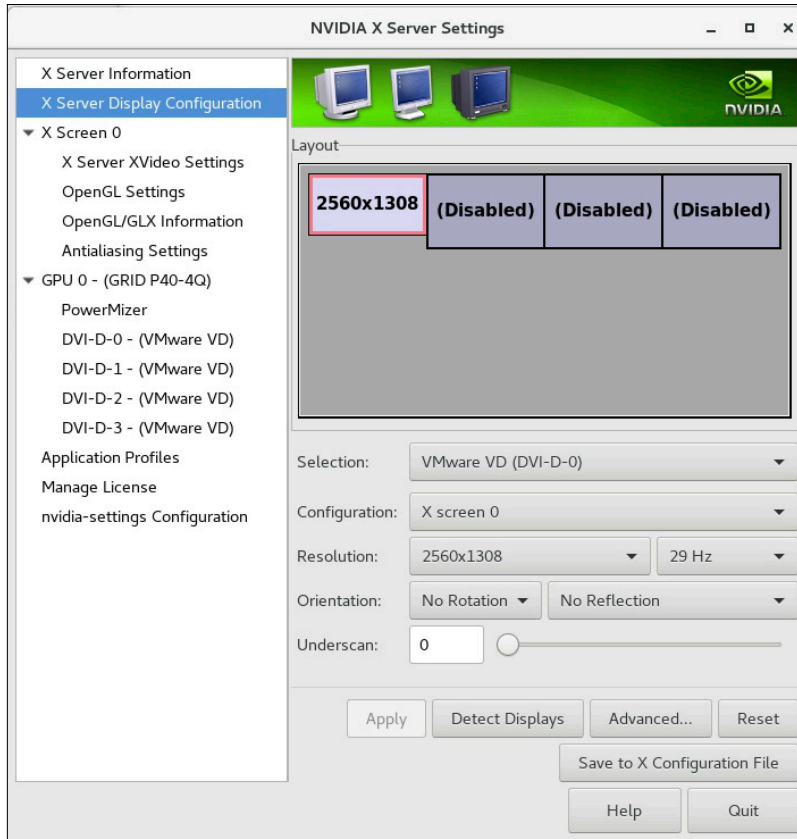

```
init 3
chmod +x NVIDIA-Linux-x86_64-xxx.xx-grid.run
sudo ./NVIDIA-Linux-x86_64-xxx.xx-grid.run
```
8. (Optional) Install the CUDA Toolkit (run file method recommended), but do not install the included driver.
9. Add license server information:


```
sudo cp /etc/nvidia/gridd.conf.template /etc/nvidia/gridd.conf
sudo vi /etc/nvidia/gridd.conf
```

 Set `ServerAddress` and `BackupServerAddress` to the DNS names or IPs of your license servers, and `FeatureType` to 1 for vGPU and 2 for vDGA.
10. Install the Horizon Agent:


```
tar -zxvf VMware-horizonagent-linux-x86_64-7.3.0-6604962.tar.gz
cd VMware-horizonagent-linux-x86_64-7.3.0-6604962
sudo ./install_viewagent.sh
```

Following is a screenshot of the NVIDIA X Server Settings window showing the results of installation and configuration for a Red Hat Enterprise Linux guest operating system.



Horizon 7 version 7.x Pool and Farm Settings

During the creation of a new farm in Horizon 7, configuring a 3D farm is the same as a normal farm. During the creation of a new View desktop pool in Horizon 7, configure the pool as normal until you reach the Desktop Pool Settings section.

1. In the Add Desktop Pool window scroll to the Remote Display Protocol section.
2. For the **3D Renderer** option, do one of the following.
 - For vSGA, select either **Hardware** or **Automatic**.
 - For vDGA or MxGPU, select **Hardware**.
 - For vGPU, select **NVIDIA GRID VGPU**.

The screenshot shows the 'Add Desktop Pool - test' window with the 'Desktop Pool Settings' section selected in the left-hand navigation pane. The main content area is divided into several sections:

- General:** State is set to 'Enabled'. Connection Server restrictions and Category Folder are both set to 'None'.
- Remote Settings:** Remote Machine Power Policy is 'Take no power action'. Automatically logoff after disconnect is 'Never'. Allow users to reset/restart their machines, allow user to initiate separate sessions from different client devices, and delete machine after logoff are all set to 'No'.
- Remote Display Protocol:** Default display protocol is 'VMware Blast'. Allow users to choose protocol is 'Yes'. The 3D Renderer dropdown menu is open, showing options: Automatic, Software, Hardware, **NVIDIA GRID VGPU** (selected), and Disabled. Max number of monitors is set to 'Automatic'. Max resolution of any one monitor is set to 'NVIDIA GRID VGPU'. HTML Access is checked and set to 'Enabled'.
- Adobe Flash Settings for Sessions:** Adobe Flash quality is 'Do not control' and Adobe Flash throttling is 'Disabled'.
- Mirage Settings:** The 'Override global Mirage settings' checkbox is unchecked. The Mirage Server configuration field is empty.

At the bottom of the window, there are navigation buttons: '< Back', 'Next >', and 'Cancel'.

Automatic uses hardware acceleration if the host that the virtual machine is starting in has a capable and available hardware GPU. If a hardware GPU is not available, the virtual machine uses software 3D rendering for any 3D tasks. The Automatic option allows the virtual machine to be started on, or migrated (via vSphere vMotion) to any host (VMware vSphere version 5.0 or later), and to use the best solution available on that host.

Hardware uses only hardware-accelerated GPUs. If a hardware GPU is not present in a host, the virtual machine will not start, or you cannot perform a live vSphere vMotion migration to that host. Migration is possible as long as the host the virtual machine is being moved to has a capable and available hardware GPU. The Hardware option guarantees that a virtual machine always uses hardware 3D rendering when a GPU is available, but it limits the virtual machine to using hosts that have hardware GPUs.

For Horizon 7 version 7.0 or 7.1, configure the amount of VRAM you want each virtual desktop to have. If you are using vGPU, also select the profile to use. With Horizon 7 version 7.1, you can use vGPU with instant clones, but the profile must match the profile set on the parent VM with the vSphere Web Client.

3D Memory has a default of 96 MB, a minimum of 64 MB, and a maximum of 512 MB.

With Horizon 7 version 7.2 and later, the video memory and vGPU profile are inherited from the VM or VM snapshot.

Clean	29-05-17 14:29:39		/Snapshot/Snapshot 2/Clean
SVGA settings for Instant Clone Pool (Inherited from Master VM)			
Number of monitors:	1	VRAM Size:	8.0 MB
		Resolution:	1600x1200
			OK Cancel

License Server

For vGPU with GRID 2.0, you must install a license server. See the GRID Virtual GPU User Guide included with your NVIDIA driver download.

Resource Monitoring

Various tools are available for monitoring resources when using graphics acceleration.

gpupvm

To better manage the GPU resources available on an ESXi host, examine the current GPU resource allocation. The ESXi command-line query utility `gpupvm` lists the GPUs installed on an ESXi host and displays the amount of GPU memory that is allocated to each virtual machine on that host.

```
gpupvm
```

```
Xserver unix:0, GPU maximum memory 2076672KB
```

```
pid 118561, VM "Test-VM-001", reserved 131072KB of GPU memory pid 664081,
VM "Test-VM-002", reserved 261120KB of GPU memory GPU memory left 1684480KB
```

nvidia-smi

To get a summary of the vGPUs currently running on each physical GPU in the system, run `nvidia-smi` without arguments.

```
Thu Oct 5 09:28:05 2017
```

```
+-----+
| NVIDIA-SMI 384.73                 Driver Version: 384.73           |
+-----+-----+-----+-----+-----+-----+
| GPU  Name            Persistence-M| Bus-Id        Disp.A | Volatile Uncorr. ECC |
| Fan  Temp  Perf    Pwr:Usage/Cap|      Memory-Usage | GPU-Util  Compute M. |
|====+=====+====+=====+=====+=====+=====+
|   0   Tesla P40                On          | 00000000:84:00:0 Off  |                    |
| N/A   38C    P0     60W / 250W | 12305MiB / 24575MiB |      0%      Default  |
+-----+-----+-----+-----+-----+-----+

```

```
+-----+
| Processes:                               GPU Memory |
| GPU      PID    Type    Process name      Usage          |
|====+=====+====+=====+=====+=====+
|   0      135930  M+C+G   manual              4084MiB |
|   0      223606  M+C+G   centos3D004        4084MiB |
|   0      223804  M+C+G   centos3D003        4084MiB |
+-----+

```

To monitor vGPU engine usage across multiple vGPUs, run `nvidia-smi vgpu` with the `-u` or `--utilization` option:

```
nvidia-smi vgpu -u
```

The following usage statistics are reported once every second for each vGPU.

#GPU	VGPU	SM	MEM	ENC	DEC
#IDX	ID	%	%	%	%
0	11924	6	3	0	0
1	11903	8	3	0	0
2	11908	10	4	0	0

Key:

gpu - GPU ID
 vgpu - vGPU ID
 sm - Compute
 mem - Memory controller bandwidth
 enc - Video encoder
 dec - Video decoder

Troubleshooting

Try these troubleshooting techniques to address general problems or a specific symptom.

General Troubleshooting for Graphics Acceleration

If an issue arises with vSGA, vGPU, or vDGA, or if Xorg fails to start, try one or more of the following solutions in any order.

Verify That the GPU Driver Loads

To verify that the GPU VIB is installed, run one of the following commands:

- For AMD-based GPUs:

```
#esxcli software vib list | grep fglrx
```
- For NVIDIA-based GPUs:

```
#esxcli software vib list | grep NVIDIA
```

If the VIB is installed correctly, the output resembles the following:

```
NVIDIA-VMware 304.59-1-OEM.510.0.0.799733 NVIDIA
VMwareAccepted 2012-11-14
```

To verify that the GPU driver loads, run the following command:

- For AMD-based GPUs:

```
#esxcli system module load -m fglrx
```
- For NVIDIA-based GPUs:

```
#esxcli system module load -m nvidia
```


If the driver loads correctly, the output resembles the following:

```
Unable to load module /usr/lib/vmware/vmkernel/nvidia: Busy
```

If the GPU driver does not load, check the `vmkernel.log`:

```
# vi /var/log/vmkernel.log
```

On AMD hardware, search for `FGLRX`. On NVIDIA hardware, search for `NVRM`. Often, an issue with the GPU is identified in the `vmkernel.log`.

Verify That Display Devices Are Present in the Host

To make sure that the graphics adapter is installed correctly, run the following command on the ESXi host:

```
# esxcli hardware pci list -c 0x0300 -m 0xff
```

The output should resemble the following example, even if some of the particulars differ:

```
000:001:00.0
Address: 000:001:00.0
Segment: 0x0000
Bus: 0x01
Slot: 0x00
Function: 0x00
VMkernel Name:
Vendor Name: NVIDIA Corporation
Device Name: NVIDIA Quadro 6000
Configured Owner: Unknown
Current Owner: VMkernel
Vendor ID: 0x10de
Device ID: 0x0df8
SubVendor ID: 0x103c
SubDevice ID: 0x0835
Device Class: 0x0300
Device Class Name: VGA compatible controller
Programming Interface: 0x00
Revision ID: 0xa1
Interrupt Line: 0x0b
IRQ: 11
Interrupt Vector: 0x78
PCI Pin: 0x69
```

Check the PCI Bus Slot Order

If you installed a second lower-end GPU in the server, the ESXi console session chooses the higher-end card. If this occurs, swap the two GPUs between PCIe slots, or change the primary GPU settings in the server BIOS. Then the card for the console (low-end) will come first.

Check Xorg Logs

If the correct devices are present in the previous troubleshooting methods, view the Xorg log file to see if there is an obvious issue:

```
# vi /var/log/Xorg.log
```

Troubleshooting Specific Issues in Graphics Acceleration

This section describes solutions to specific issues that could arise in graphics acceleration deployments.

Problem:

`sched.mem.min` error when starting virtual machine.

Solution:

Check `sched.mem.min`.

If you get a vSphere error about `sched.mem.min`, add the following parameter to the `VMX` file of the virtual machine:

```
sched.mem.min = "4096"
```

Note: The number in quotes, `4096` in the previous example, must match the amount of configured virtual machine memory. The example is for a virtual machine with 4 GB of RAM.

Problem:

Only able to use one display in Windows 10 with vGPU -OB or -OQ profiles.

Solution:

Use a profile that supports more than one virtual display head and has at least 1 GB of frame buffer.

To reduce the possibility of memory exhaustion, vGPU profiles with 512 MB or less of frame buffer support only one virtual display head on a Windows 10 guest OS.

Problem:

Unable to use NVENC with vGPU -OB or -OQ profiles.

Solution:

If you require NVENC to be enabled, use a profile that has at least 1 GB of frame buffer.

Using the frame buffer for the NVIDIA hardware-based H.264 / HEVC video encoder (NVENC) might cause memory exhaustion with vGPU profiles that have 512 MB or less of frame buffer. To reduce the possibility of memory exhaustion, NVENC is disabled on profiles that have 512 MB or less of frame buffer.

Problem:

Unable to load vGPU driver in guest operating system.

Depending on the versions of drivers in use, the vSphere VM's log file reports one of the following errors.

- A version mismatch between guest and host drivers:

```
vthread-10| E105: vmiop_log: Guest VGX version(2.0) and Host VGX
version(2.1) do not match
```

- A signature mismatch:

```
vthread-10| E105: vmiop_log: VGPU message signature mismatch
```

Solution:

Install the latest NVIDIA vGPU release driver matching the installed VIB on ESXi in the VM.

Problem:

Tesla-based vGPU fails to start.

Solution:

Disable error-correcting code (ECC) on all GPUs.

Tesla GPUs support ECC, but the NVIDIA GRID vGPU does not support ECC memory. If ECC memory is enabled, the NVIDIA GRID vGPU fails to start. The following error is logged in the VMware vSphere VM's log file:

```
vthread10|E105: Initialization: VGX not supported with ECC Enabled.
```

1. Use `nvidia-smi` to list the status of all GPUs.
2. Check whether ECC is enabled on the GPUs.
3. Change the ECC status to **Off** on each GPU for which ECC is enabled by executing the following command:

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

(`id` is the index of the GPU as reported by `nvidia-smi`)
4. Reboot the host.

Problem:

Single vGPU benchmark scores are lower than the pass-through GPU.

Solution:

Disable the Frame Rate Limiter (FRL) by adding the configuration parameter `pciPassthru0.cfg.frame_rate_limiter` with a value of `0` in the VM's advanced configuration options.

FRL is enabled on all vGPUs to ensure balanced performance across multiple vGPUs that are resident on the same physical GPU. FRL is designed to provide a good interactive remote graphics experience, but it can reduce scores in benchmarks that depend on measuring frame-rendering rates as compared to the same benchmarks running on a pass-through GPU.

Problem:

VMs configured with large memory fail to initialize the vGPU when booted.

When starting multiple VMs configured with large amounts of RAM (typically more than 32 GB per VM), a VM might fail to initialize the vGPU. The NVIDIA GRID GPU is present in Windows Device Manager but displays a warning sign and the following device status:

Windows has stopped this device because it has reported problems. (Code 43)

The vSphere VM's log file contains these error messages:

```
vthread10|E105: NVOS status 0x29
vthread10|E105: Assertion Failed at 0x7620fd4b:179
vthread10|E105: 8 frames returned by backtrace
...
vthread10|E105: VGPU message 12 failed, result code: 0x29
...
vthread10|E105: NVOS status 0x8
vthread10|E105: Assertion Failed at 0x7620c8df:280
vthread10|E105: 8 frames returned by backtrace
...
vthread10|E105: VGPU message 26 failed, result code: 0x8
```

Solution:

A vGPU reserves a portion of the VM's frame buffer for use in GPU mapping of VM system memory. The default reservation is sufficient to support up to 32 GB of system memory. You can accommodate up to 64 GB by adding this configuration parameter:

```
pciPassthru0.cfg.enable_large_sys_mem
```

with a value of `1` in the VM's advanced configuration options.

Summary

VMware Horizon 7 offers three technologies for hardware-accelerated graphics, each with its own advantages.

- **Virtual Shared Pass-Through Graphics Acceleration (MxGPU or vGPU)** – Best match for nearly all use cases.
- **Virtual Shared Graphics Acceleration (vSGA)** – For light graphical workloads that use only DirectX9 or OpenGL 2.1 and require the maximum level of consolidation.
- **Virtual Dedicated Graphics Acceleration (vDGA)** – For heavy graphical workloads that require the maximum level of performance.

With the information in this paper, you can install, configure, and manage your 3D workloads for Horizon 7 version 7.x on vSphere 6.x.

Additional Resources

Setting Up Graphics for Linux Desktops in [Setting Up Horizon 7 for Linux Desktops](#)

Configuring Desktop Pools > Configuring 3D Rendering for Desktops in [Setting Up Virtual Desktops in Horizon 7](#)

About the Authors

This paper was updated by Hilko Lantinga, who is an End-User-Computing Architect in VMware Technical Marketing with a focus on 3D, Horizon 7 Windows desktops, and RDSH, Linux, and applications. Previously, he was a senior consultant in VMware Professional Services, leading large-scale EUC deployments in EMEA. Hilko has 18 years of experience in end-user computing.

The original version of this paper was written by Stéphane Asselin and Gary Sloane.

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



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

Copyright © 2017 VMware, Inc. All rights reserved. This product is protected by U.S. and international copyright and intellectual property laws. VMware products are covered by one or more patents listed at <http://www.vmware.com/go/patents>. VMware is a registered trademark or trademark of VMware, Inc. in the United States and/or other jurisdictions. All other marks and names mentioned herein may be trademarks of their respective companies. Item No: 5199-VMW-WP-HORIZON7-DEPLOYING-HARDWARE-ACCELERATED-GRAPHICS-7_3_1-USLET-20171201
12/17