



Virtual Machine Graphics Acceleration Deployment Guide

WHITE PAPER

Table of Contents

Introduction	4
Why 3D Matters for VMware Horizon View	4
Understanding the Differences Between Soft 3D/SVGA, vDGA and vSGA	5
Soft 3D and SVGA – The VMware Graphics Driver	5
vDGA – Virtual Dedicated Graphics Acceleration (Tech Preview)	5
vDGA Deployment	6
vDGA Does Not Support Live VMware vSphere vMotion Capabilities	6
vSGA – Virtual Shared Graphics Acceleration	6
Configure vSGA in VMware vSphere	6
Configure vSGA in Horizon View	7
Prerequisites	8
Host Hardware Requirements	8
Servers with Compatible Power and PCI Slot Capacity	8
Physical Host Size	8
PCIe x16	8
Host PSU (Power Supply Unit)	8
Virtual Technology for Directed I/O (VT-d)	8
Two-Display Adapters	8
Supported Graphics Cards	9
Software Requirements	9
End-User Clients	9
Application Requirements and Use Cases	11
DirectX 9.0c	11
OpenGL 2.1	11
Example Use Cases	11
Graphics Card Installation	13
Quadro Range	13
Tesla M2075	13
Kepler (Grid K1 and K2 Boards)	13
Confirm Successful Installation	13
vSGA Installation	15
NVIDIA Drivers	15
vSGA Post-Installation Checks	16
Xorg	16
gpvm	16
nvidia-smi	16
Log Files	17
vDGA Installation	18
Enable the Host for GPU Passthrough	18
Check VT-d or AMD IOMMU Is Enabled	18
Enable Device Passthrough	18
Enable the Virtual Machine for GPU Passthrough	18

Update to Hardware Version 9	18
Reserve All Configured Memory.	18
Adjust pciHole.start	18
Add the PCI Device	19
Install the NVIDIA Driver	19
Install the View Agent	19
Enable Proprietary NVIDIA Capture APIs.	19
VMware Horizon View Pool Configuration for vSGA.	20
Horizon View Pool Prerequisites	20
Video Memory (VRAM) Sizing.	20
Screen Resolution.	21
Horizon View Pool 3D Rendering Options	21
Manage Using vSphere Client.	21
Automatic.	21
Software	21
Hardware.	21
Disabled.	22
Best Practices for Configuring 3D Rendering.	22
Automatic.	22
Hardware.	22
Manage Using vSphere Client.	22
Software	22
Enable Horizon View Pools for vSGA 3D Hardware Rendering	22
Enable an Existing Horizon View Pool	22
Enable a New Horizon View Pool.	23
Performance Tuning Tips	24
Virtual Machine Resources	24
PCoIP	24
Relative Mouse	24
Enabling Relative Mouse	24
Virtual Machines Using VMXNET3	24
Workaround for CAD Performance Issue	25
Resource Monitoring	26
gpvm.	26
nvidia-smi.	26
Troubleshooting	27
Xorg 27	
Xorg Fails to Start	27
Verify that the NVIDIA VIB Bundle Is Installed	27
Verify that the NVIDIA Driver Loads.	27
Verify that Display Devices Are Present in the Host	27
Possible PCI Bus Slot Order Issue	28
Check Xorg Logs.	28
sched.mem.min Error.	28
About the Author and Contributors	29

Introduction

The purpose of this document is to explain the various types of virtual machine graphics acceleration available, how to implement and troubleshoot them, and to provide knowledge around the benefits offered by each technology.

Why 3D Matters for VMware Horizon View

The capability for 3D graphics and video in VMware® Horizon View™ further expands the use cases and target users that IT can deliver with virtual desktops. In addition to expanding the target use cases, 3D augments the virtual desktop user interface by enabling a more graphically rich experience.

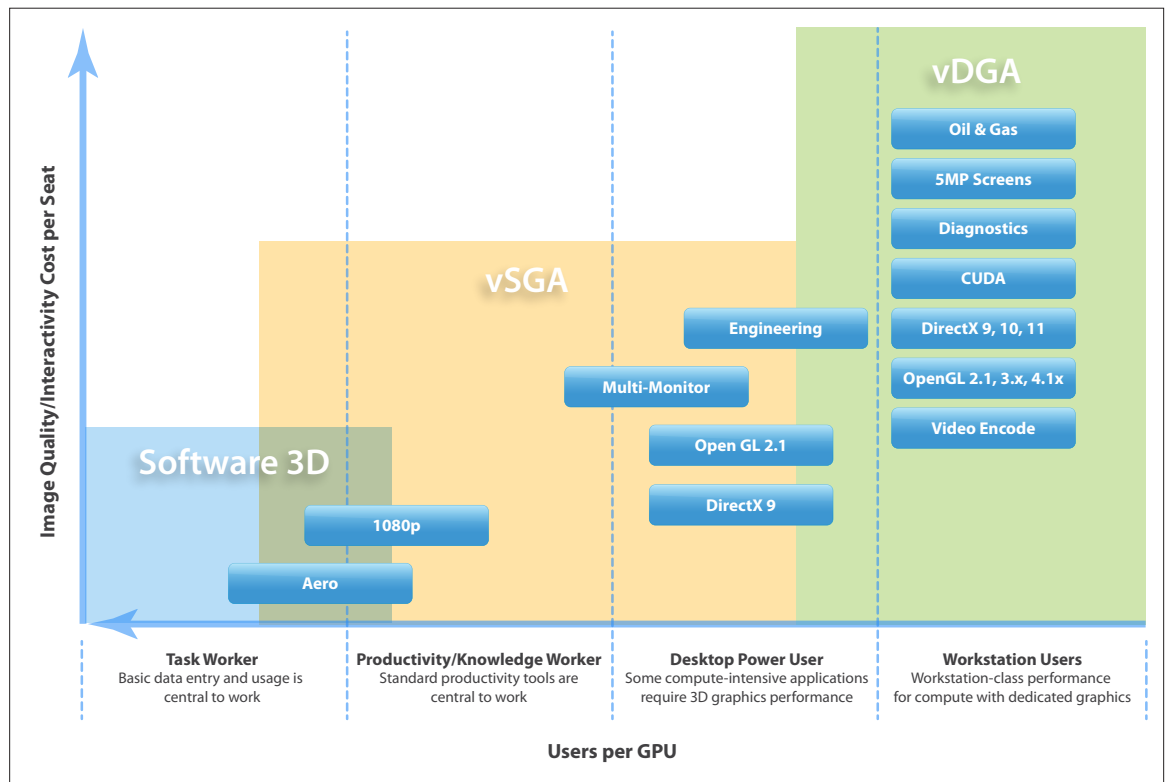


Figure 1: Virtual Desktop User Segmentation

Understanding the Differences Between Soft 3D/SVGA, vDGA and vSGA

This section will discuss the differences between Soft 3D/SVGA, vDGA, and vSGA. This is particularly important to understand, as the names are so similar.

NAME	DEFINITION	DESCRIPTION
SVGA Soft 3D VMware SVGA 3D	Super Video Graphics Array Software 3D Renderer	VMware WDDM (Windows Display Driver Model) 1.1-compliant driver
vDGA	Virtual Dedicated Graphics Acceleration	Graphics acceleration capability provided by VMware ESXi™ for high-end workstation graphics where a discrete GPU is needed
vSGA	Virtual Shared Graphics Acceleration	Multiple virtual machines leverage physical GPUs installed locally in the ESXi hosts to provide hardware-accelerated 3D graphics to multiple virtual desktops

Table 1: Graphics Driver Comparison

Soft 3D and SVGA – The VMware Graphics Driver

Software 3D Renderer, Soft 3D or SVGA, specifically VMware SVGA 3D, is the VMware WDDM (Windows Display Driver Model) 1.1-compliant driver. It is installed with VMware Tools™ onto Windows 7 virtual desktops. SVGA (super VGA, or super video graphics array) is easily confused with vSGA (Virtual Shared Graphics Acceleration).

The VMware SVGA 3D graphics driver provides support for DirectX 9.0c and OpenGL 2.1. This driver is supported on Windows 7 for 2D and 3D, and is used for both software 3D and vSGA. vDGA configurations do not use the VMware SVGA 3D driver; instead, they use the native graphics card driver installed directly in the guest OS.

One of the benefits of VMware SVGA 3D for both software 2D and 3D and vSGA implementations is that a virtual machine can dynamically switch between software or hardware acceleration, without you having to reconfigure it. It is also a major factor in why live VMware vSphere® vMotion® on the virtual machine is supported even when providing hardware-accelerated graphics using vSGA. Having a standard driver in your image greatly simplifies management and deployment.

Note: If you are dynamically moving from hardware 3D rendering to software 3D rendering, you might notice a performance drop in applications running in the virtual machine. However, if you are moving in the reverse direction (software to hardware), you should notice an improvement in performance.

vDGA – Virtual Dedicated Graphics Acceleration (Tech Preview)

vDGA is a graphics-acceleration capability provided by VMware ESXi for delivering high-end workstation graphics for use cases where a discrete GPU (graphics processing unit, also referred to as a PCI device or video card) is needed. vDGA dedicates a single GPU to a single virtual machine for high performance.

 **Important:** vDGA is currently a Tech Preview feature. A full release will be available soon.

If you are using vDGA, graphics adapters installed in the underlying host are assigned to virtual machines using VMware DirectPath I/O™. Assigning a discrete GPU to the virtual machine dedicates the entire GPU to it.

vDGA can be costly to implement, but should offer a reduction in cost compared to individual high-end workstations. The amount of 3D hardware-accelerated virtual machines per host is limited to the number of PCIe x16 slots in the server. Server hardware is available with up to four PCIe x16 slots and room in the chassis for high-end GPUs. Some blade enclosure hardware vendors also offer a sidecar-type expansion unit that can support up to eight GPUs.

Note: Both vSGA and vDGA can support a maximum of eight GPU cards per ESXi host.

vDGA Deployment

When you deploy vDGA, it uses the graphics driver from the GPU vendor rather than the virtual machine's SVGA 3D driver. vDGA uses an interface between the remoting protocol and graphics driver to provide frame buffer access.

Because of the nature of vDGA configuration, it is not a candidate for automated deployment using VMware View Composer™, as each individual virtual machine has a one-to-one relationship with a specific GPU. Users should configure each virtual machine manually, and carefully select the correct PCI device for each one.

vDGA Does Not Support Live VMware vSphere vMotion Capabilities

Live VMware vSphere vMotion is not supported with vDGA. vDGA uses VMware vSphere DirectPath I/O to allow direct access to the GPU card, bypassing the virtualization layer. By enabling direct passthrough from the virtual machine to the PCI device installed on the host, the virtual machine is effectively locked to that specific host.

If a user needs to move a vDGA-enabled virtual machine to a different host, they should power off the virtual machine, use vSphere vMotion to migrate it to another host that has a GPU card installed, and re-enable passthrough to the specific PCI device on that host. Only then should the user power on the virtual machine.

vSGA – Virtual Shared Graphics Acceleration

vSGA provides the ability for multiple virtual machines to leverage physical GPUs installed locally in the ESXi hosts to provide hardware-accelerated 3D graphics. vSGA allows multiple virtual machines to share hardware GPUs for 3D acceleration, rather than a one-to-one relationship like vDGA.

The maximum amount of video memory that can be assigned per virtual machine is 512MB. However, the video memory allocation is evenly divided. Half the video memory is reserved on the hardware GPU, while the other half is reserved via host RAM. (Take this into consideration when sizing your ESXi host RAM.) Use this rule to calculate basic consolidation ratios. For example, the NVIDIA Quadro 4000 card has 2GB of GPU RAM. If all virtual machines are configured with 512MB of video memory, half of which (256MB) is reserved on the GPU, you can calculate that a maximum of eight virtual machines can run on that specific GPU at any given time.

The ESXi host reserves GPU hardware resources on a first-come, first-served basis as virtual machines are powered on. If all GPU hardware resources are already reserved, additional virtual machines will be unable to power on if they are explicitly set to use hardware 3D rendering. If the virtual machines are set to Automatic, the virtual machines will be powered on using software 3D rendering.

Configure vSGA in VMware vSphere

To configure vSGA in the VMware vSphere 5.1 web interface, there are three 3D rendering options: Automatic (the default); Software; and Hardware.

- **Automatic** uses hardware acceleration if there is a capable, and available, hardware GPU in the host that the virtual machine is starting in. However, if a hardware GPU is not available it uses software 3D rendering for any 3D tasks. This allows the virtual machine to be started on, or migrated (via vSphere vMotion) to *any* host (VMware vSphere version 5.0 and higher); and to use the best solution available on that host.
- **Software** *only* uses vSphere software 3D rendering, even if there is an available hardware GPU in the host the virtual machine is running on. This will not provide the performance benefits that hardware 3D acceleration offers. However, it both allows the virtual machine to run on any host (vSphere 5.0 and higher), and allows you to block virtual machines from using a hardware GPU in a host, if that level of performance is not required (like the 3D aspects of Microsoft Office).

- **Hardware *only*** uses hardware-accelerated GPUs. If a hardware GPU is not present in a host, the virtual machine will not start; or you will not be able to perform a live vSphere vMotion migration to that host. vSphere vMotion is possible with this specification as long as the host the virtual machine is being moved to has a capable and available hardware GPU. This setting can be used to guarantee that a virtual machine will always use hardware 3D rendering when a GPU is available; but that in turn limits the virtual machine to hosts with hardware GPUs.

Configure vSGA in Horizon View

To configure vSGA in Horizon View 5.2 Pool Settings, there are five 3D rendering options: Manage Using VMware vSphere Client™; Automatic; Software; Hardware; and Disabled.

- **Manage Using vSphere Client** will not make any changes to the 3D settings of the individual virtual machines in that pool. This allows individual virtual machines to have different settings set through vSphere. The most likely use for this setting is during testing or for manual desktop pools.
- **Automatic** uses hardware acceleration if there is a capable, and available, hardware GPU in the host that the virtual machine is starting in. However, if a hardware GPU is not available it uses software 3D rendering for any 3D tasks. This allows the virtual machine to be started on, or migrated (via vSphere vMotion) to *any* host (vSphere version 5.0 and higher); and to use the best solution available on that host.
- **Software *only*** uses vSphere software 3D rendering, even if there is an available hardware GPU in the host the virtual machine is running on. This will not provide the performance benefits that hardware 3D acceleration offers. However, it both allows the virtual machine to run on any host (vSphere 5.0 and higher), and allows you to block virtual machines from using a hardware GPU in a host, if that level of performance is not required (like the 3D aspects of Microsoft Office).
- **Hardware *only*** uses hardware-accelerated GPUs. If a hardware GPU is not present in a host, the virtual machine will not start, or you will not be able to perform a live vSphere vMotion migration to that host. vSphere vMotion is possible with this specification as long as the host the virtual machine is being moved to has a capable and available hardware GPU. This setting can be used to guarantee that a virtual machine will always use hardware 3D rendering when a GPU is available; but that in turn limits the virtual machine to hosts with hardware GPUs.
- **Disabled** does not use 3D rendering at all (software or hardware), and overrides vSphere 3D settings to disable 3D. Use the Disabled setting to ensure that non-graphical workload Horizon View desktop pools do not use unnecessary resources, like sharing a hardware GPU, when running on the same cluster as heavier graphics workload Horizon View desktops.

Prerequisites

This section lists both the hardware and software required to support the use of vSGA and vDGA.

Host Hardware Requirements

Hardware requirements for both vSGA and vDGA are documented below.

Servers with Compatible Power and PCI Slot Capacity

For a list of supported hardware that has been tested and proven to work with vSGA, please visit the [VMware Compatibility Guide](#).

Physical Host Size

Many high-end GPU cards are full-height, full-length and double-width (taking up two slots on the motherboard, but using only a single PCIe x16 slot). Verify that the host has enough room internally to hold the GPU card you have chosen in the appropriate PCIe slot.

PCIe x16

PCIe x16 is required for all supported most-high-end GPU Cards.

Host PSU (Power Supply Unit)

Verify the power requirements of the GPU to make sure the PSU is both powerful enough and contains the proper power cables to power the GPU. As an example, a single NVIDIA Quadro 6000 GPU can use as much as 204W of power, and requires either a single 8-pin PCIe power cord or dual 6-pin PCIe power cords.

A major advantage of GRID boards (K1 or K2) is their lower power requirements. The K1 card operates near 130W, which is considerably less than the current Quadro series cards. The tradeoff is that K1 and K2 cards are passively cooled, relying on internal chassis fans to cool them. This has presented challenges to server manufacturers. If you plan to use GRID boards in your servers, ensure that there is ample cooling for the cards, or you could experience card failure due to overheating.

Virtual Technology for Directed I/O (VT-d)

To use vDGA, verify that the host either supports Intel VT-d (Virtualization Technology for Directed I/O) or AMD IOMMU (input/output memory management unit). Without this, GPU passthrough cannot be enabled.

To check if VT-d or AMD IOMMU is enabled on the host, run the following command via SSH or on the host console. (**Note:** replace <module_name> with the name of the module, **vtddmar** for Intel, **AMDiommu** for AMD).

```
# esxcfg-module -l | grep <module_name>
```

If the appropriate module is not present, you might have to enable it in the BIOS or your hardware might not be capable of providing PCI passthrough.

Two-Display Adapters

If the host does not have an onboard graphics adapter, VMware recommends that you install an additional low-end display adapter to act as the primary display adapter. This is because the ESXi console display adapter is not available to Xorg. If the high-end NVIDIA card is set as the primary adapter, Xorg will not be able to use the GPU for rendering.

If you have two GPUs installed, the server BIOS might give you the option to select which GPU should be the Primary and which should be the Secondary. If this option is available, make sure the standard GPU is set as Primary and the high-end GPU is set as Secondary.

ESXi Special Access Command

If you only have a single GPU, there is a command that *will* enable said single GPU to be used with vSGA; however, VMware does not recommend that the single GPU be used regularly as the setting is not persistent.

To configure an ESXi host with only a single GPU, first find the PCI ID of the graphics device by running the following command:


```
# lspci | grep -i display
```

You'll see something similar to this:

```
000:128:00.0 Display controller: NVIDIA Corporation GT200b  
[GeForce GTX 275]
```

Then you must reset the ownership flags referencing the PCI ID above (in bold):

```
# vmkchdev -v 0:128:0:0
```

 **Important:** This setting is not persistent, and must be re-run each time ESXi reboots.

Supported Graphics Cards

For a list of supported hardware that has been tested and proven to work with vSGA, please visit the [VMware Compatibility Guide](#).

Note: GPU support is dictated by the graphics card vendor, not by VMware.

Software Requirements

Software requirements for both vSGA and vDGA are documented below.

PRODUCT	DESCRIPTION
Hypervisor	ESXi 5.1 or higher with latest patches. (ESXi 5.1 and ESXi510-201210001 patch recommended at the time of this writing.)
VMware Horizon View	VMware Horizon View 5.2 or higher.
Display Protocol	vSGA and vDGA support only PCoIP with a maximum of two display monitors.
NVIDIA Drivers	The ESXi VIBs are written, maintained, and supported by NVIDIA, not VMware. The NVIDIA vSphere VIBs for vSGA and the NVIDIA Windows 7 driver for vDGA can be downloaded from the NVIDIA Download Drivers page.
Guest Operating System	The virtual machines must be running Windows 7 or later. vSGA supports both 32-bit and 64-bit Windows. vDGA requires the Windows 7 64-bit edition.

Table 2: Software Requirements

End-User Clients

With both graphical and compute processing handled by the ESXi hosts that run the 3D virtual desktops, IT might overlook end clients because they often perceive that major processing is handled inside the datacenter. This is not always the case for high-end graphics applications or games running on virtual desktops.

Using 3D applications with fast-changing graphics often results in a massive bandwidth requirement to support the PCoIP traffic that flows between the virtual desktop and the end clients. This is often the cause of a poor user experience.

In some 3D use cases, as much as 70Mbit/s of PCoIP traffic per virtual desktop has been observed during peak loads. This high bandwidth is caused by constant changes to images on the virtual desktop screen. This requires PCoIP to continually send data to keep up with the changes, ensuring that the display on the screen is accurate and current. This large flow of PCoIP traffic sent to end clients can lead to performance problems.

Some low-end thin clients do not have the CPU processing power they need to decode PCoIP data fast enough to make the end-user experience smooth and uninterrupted. However, this is not always the case for every environment or end client. It depends on which applications users are running on their virtual desktops.

For high-end 3D and video workloads, use a high-performance Zero Client with a Teradici Tera2 chip, or a modern Core i5- or Core i7-based Windows PC, for best performance with multiple high-resolution displays.

Note: The Tera1 chip can support a maximum frames-per-second rate of 30FPS, whereas the new Tera2 chip can achieve up to 60FPS. Achieving high frame rates can be important to the usability of certain engineering applications.

Application Requirements and Use Cases

If an application does not run or is underperforming, check the software vendor's system requirements for hardware and graphics acceleration.

DirectX 9.0c

vSGA currently supports only up to DirectX 9.0c. Applications that require a newer version of DirectX might not function or perform correctly when using vSGA.

OpenGL 2.1

vSGA currently supports only up to OpenGL 2.1. Applications that require a newer version of OpenGL might not function or perform correctly when using vSGA.

Note: vDGA will support the versions of DirectX and OpenGL that the GPU manufacturer's graphics driver supports. This is generally the latest version of these technologies.

Example Use Cases

The following tables summarize the performance of a sample selection of office and power user applications, 3D and video design applications, and engineering and compute applications.

APPLICATION	SOFT 3D	VSGA	VDGA
Windows Aero	✓	✓	✓
Microsoft Office	✓	✓	✓
Microsoft Visio	✓	✓	✓
Google Earth	✓	✓	✓
HTML 5/Web 3D	✓	✓	✓
Adobe Photoshop	✓	✓	✓
Epic	✓	✓	✓
SolidWorks View	✓	✓	✓
Team Center Vis	✓	✓	✓
PTC Creo View	✓	✓	✓
Siemens NX Viewer	✓	✓	✓
✓ Works ✓ Not appropriate			

Table 3: Example Application Use Cases – Office Applications and Power Users

APPLICATION	SOFT 3D	VSGA	VDGA
Autodesk AutoCAD	✓	✓	✓
Autodesk Inventor	✓	✓	✓
Autodesk 3DS Max	✓	✓	✓
Autodesk Maya	✓	✓	✓
CATIA	✓	✓	✓
SolidWorks 2012	✓	✓	✓
SolidWorks 2013	✓	✓	✓
Enovia	✓	✓	✓
Siemens NX	✓	✓	✓
Adobe Premiere	✓	✓	✓
Siemens NX Viewer	✓	✓	✓
<p>✓ Works ✓ Useful for reviewers and lightweight use cases, but not NVIDIA Graphics Driver Vendor Certified ✓ NVIDIA Graphics Driver Vendor Certified ✓ Not appropriate</p>			

Table 4: Graphics Application Use Cases – 3D and Video Design Applications

APPLICATION	SOFT 3D	VSGA	VDGA
Schumberger	✓	✓	✓
Petrel	✓	✓	✓
CUDA Applications	✓	✓	✓
OpenCL Applications	✓	✓	✓
Custom 3D Applications	✓	✓	✓
<p>✓ NVIDIA Graphics Driver Vendor Certified ✓ Not appropriate</p>			

Table 5: Graphics Application Use Cases – Engineering and Compute Applications

Graphics Card Installation

See the NVIDIA User Guides to make sure that the graphics card is installed correctly in your server.

Quadro Range

The [Quadro 4000/5000/6000 SDI User's Guide](#) can be downloaded from the NVIDIA Web site.

Tesla M2075

The [Tesla M2075 Dual-Slot Computing Processor Module Board Specification](#) document can be downloaded from the NVIDIA Web site.

Kepler (GRID K1 and K2 Boards)

Installation guides for K1 and K2 boards are supplied with the hardware.

Confirm Successful Installation

To check if the Graphics Adapter has been installed correctly, run the following command on the ESXi host:

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

You should see an output similar to the following:

```
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
Spawned Bus: 0x00
Flags: 0x0201
Module ID: 71
Module Name: nvidia
Chassis: 0
Physical Slot: 1
Slot Description:
Passthru Capable: true
Parent Device: PCI 0:0:1:0
Dependent Device: PCI 0:0:1:0
Reset Method: Bridge reset
FPT Sharable: true

vSGA Installation

This section takes you through the steps required to install the NVIDIA driver (VIB) on an ESXi host.

NVIDIA Drivers

1. The NVIDIA vSphere VIBs for vSGA can be downloaded from the [NVIDIA Download Drivers](#) page.
2. Upload the bundle (.zip) to a datastore on the host. You can do this in two ways:
 - Upload the bundle by browsing the datastore using the vSphere Client.
 - Upload the bundle to the host datastore using an SCP tool (e.g., FastSCP or WinSCP).
3. Run the following command through an ESXi SSH session to install the VIB onto the host:

```
# esxcli software vib install -d /xxx-path-to-vib/vib-name.zip
```

Here is an example of the complete command:

```
# esxcli software vib install -d /vmfs/volumes/509aa90d-69ee45eb-c96b-4567b3d/NVIDIA-VMware-x86_64-304.59-bundle.zip
```

During the installation, if your host is not in Maintenance Mode you will receive the following error:

```
[MaintenanceMode Error].
```

You have two options: either put the host into Maintenance Mode; or add the following command option in the esxcli command above:

```
--maintenance-mode
```

Here is an example of the complete command:

```
# esxcli software vib install --maintenance-mode -d /vmfs/volumes/509aa90d-69ee45eb-c96b-4567b3d/NVIDIA-VMware-x86_64-304.59-bundle.zip
```

If you received the error:

```
Could not find a trusted signer
```

indicating that the VIB bundle is not signed, use the following option in the esxcli to remove the signature check:

```
--no-sig-check
```

Here is an example of the complete command:

```
# esxcli software vib install --no-sig-check -d /vmfs/volumes/509aa90d-69ee45eb-c96b-4567b3d/NVIDIA-VMware-x86_64-304.59-bundle.zip
```

Installation can take a few minutes. After it is complete you should see the following output in the SSH console:

Installation Result

```
Message: Operation finished successfully.
```

```
Reboot Required: false
```

```
VIBs Installed: <VIB NAME HERE>
```

```
VIBs Removed:
```

```
VIBs Skipped:
```

4. Although the output states that a reboot is not required (**Reboot Required: false**), VMware recommends the ESXi host is rebooted to verify that Xorg will start correctly on future restarts of the host. If you do not reboot the host, you will have to manually start the Xorg service. You can do this by issuing the following command:

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

To remove the installed VIB from a host, run the following command:

```
# esxcli software vib remove --vibName=name
```

vSGA Post-Installation Checks

This section contains various commands that can be used to ensure that the GPU card and its respective drivers have been installed correctly. VMware recommends that you learn these commands, which are useful if you have to troubleshoot issues.

Xorg

Xorg is a full featured X server that was originally designed for UNIX and UNIX-like operating systems running on Intel x86 hardware. It now runs on a wider range of hardware and OS platforms including ESXi. The status of Xorg can be checked using the following command in an SSH session:

```
# /etc/init.d/xorg status
```

If Xorg is not started, run the following command to start it:

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

If Xorg fails to start, go to the [Troubleshooting](#) section.

gpvm

Issue the following command through an ESXi SSH session:

```
# gpvm
```

This results in a list of working GPUs that shows the virtual machines using each GPU, and the amount of video memory reserved for each one.

If this command has no output at all, then the Xorg service is most likely not running. Run the following command in an SSH session to show the status of Xorg:

```
# /etc/init.d/xorg status
```

If Xorg is not started, run the following command to start it:

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

If Xorg fails to start, go to the [Troubleshooting](#) section.

nvidia-smi

To see how much of each GPU is in use, issue the following command in an SSH session:

```
# nvidia-smi
```

This will show several details of GPU usage at the point in time when you issued the command (this display is not dynamic and must be reissued to update the information). You can also issue the following command:

```
# watch -n 1 nvidia-smi
```

This command will issue the nvidia-smi command every second to provide a refresh of that point-in-time information.

Note: The most meaningful metric in the nvidia-smi display is at the right of the middle section. It shows the percentage of each GPU's processing cores in use at that point in time. This can be helpful if you have to troubleshoot poor performance—verify if the GPU processing cores are being overtaxed, and the cause of their poor performance.

Log Files

Verify that the virtual machine has graphics acceleration by searching for OpenGL in the virtual machine's vmware.log. You should see something like:

```
mks| I120: OpenGL Version: "3.2.0 NVIDIA 304.59" (3.2.0)
mks| I120: GLSL Version: "1.50 NVIDIA via Cg compiler" (1.50.0)
mks| I120: OpenGL Vendor: "NVIDIA Corporation"
mks| I120: OpenGL Renderer: "Quadro 6000/PCIe/SSE2"
```

If the virtual machine is using the VMware software renderer, however, the vmware.log will contain:

```
mks| I120: VMiopLog notice: SVGA2 vmiop started - llvmpipe
```

vDGA Installation

This section takes you through enabling GPU passthrough at the host level and preparing the virtual machines for 3D rendering.

Enable the Host for GPU Passthrough

To enable an ESXi host for GPU passthrough, follow the documented checks and steps in the following section.

Check VT-d or AMD IOMMU Is Enabled

Before passthrough can be enabled, check if VT-d or AMD IOMMU is enabled on the host by running the following command, either via SSH or on the console. (**Note:** replace <module_name> with the name of the module: `vtddmar` for Intel, `AMDiommu` for AMD).

```
# esxcfg-module -l | grep <module_name>
```

If the appropriate module is not present, you might have to enable it in the BIOS, or your hardware might not be capable of providing PCI passthrough.

Enable Device Passthrough

1. Using the vSphere Client, connect to VMware vCenter™ and select the host with the GPU card installed.
2. Select the **Configure** tab for the host, and click **Advanced Settings** (in the top left section). If the host already has devices enabled for passthrough, these devices will be listed here.
3. To configure passthrough for the GPU, click **Configure Passthrough**.
4. In the Mark Devices for Passthrough window, check the box that corresponds to the GPU adapter installed in the host.
5. Click **OK**.

The GPU should now be listed in the Window on the Advanced Settings page.

Note: If the device has an **orange** arrow displayed on the icon, the host needs to be rebooted before passthrough will function.

If the device icon is **green**, passthrough is enabled.

Enable the Virtual Machine for GPU Passthrough

To enable a virtual machine for GPU passthrough, follow the documented checks and steps in the following section.

Update to Hardware Version 9

You must upgrade all 3D virtual machines to Hardware version 9 (HWv9 shows as `vmx-09`) to ensure maximum compatibility.

Reserve All Configured Memory

1. For vDGA to function, all the virtual machine configured memory must be reserved. If each virtual machine has 2GB of memory allocated, you should reserve all 2GB. To do this, select the **Reserve all guest memory** option when you view the **Memory** option under the **Resources** tab in a virtual machine's settings window.

Adjust pciHole.start

Note: This is required only if the virtual machine has more than 2GB of configured memory.

2. For virtual machines that have more than 2GB of configured memory, add the following parameter to the `.vmx` file of the virtual machine (you can add this at the end of the file):

```
pciHole.start = "2048"
```

Add the PCI Device

To enable vDGA for a virtual machine, the PCI device needs to be added to the virtual machine's hardware.

3. Using the vSphere Client, connect directly to the ESXi host with the GPU card installed, or select the host in vCenter.
4. Right-click the virtual machine and select **Edit Settings**.
5. **Add** a new device by selecting **PCI Device** from the list, and click **Next**.
6. Select the GPU as the passthrough device to connect to the virtual machine from the drop-down list, and click **Next**.
7. Click **Finish**.

Install the NVIDIA Driver

8. Download and install the latest NVIDIA Windows 7 desktop driver on the virtual machine. All NVIDIA drivers can be downloaded from the [NVIDIA Download Drivers](#) page.
9. After the driver is installed, reboot the virtual machine.

Install the View Agent

10. After the NVIDIA drivers are installed correctly, install the View Agent on the virtual machine.
11. Reboot when requested.

Enable Proprietary NVIDIA Capture APIs

Note: This is required only if the virtual machine has more than 2GB of configured memory.

12. After the virtual machine has rebooted, enable the proprietary NVIDIA capture APIs by running:

```
"C:\Program Files\Common Files\VMware\Teradici PCoIP Server\MontereyEnable.exe" -enable
```

Note: If `MontereyEnable.exe` is not found, use `NvFBCEnable.exe`. In the new SDK, `MontereyEnable` is replaced with `NvFBCEnable`.

13. After the process is complete, **Restart** the virtual machine.
14. In order to activate the NVIDIA display adapter, you must connect via PCoIP, and at *full screen* from the endpoint (at native resolution), or the virtual machine will use the SVGA 3D display adapter. vDGA will not work through the vSphere console session.

After the virtual machine has rebooted and you have connected via PCoIP in full screen, check to ensure that the GPU is active by viewing the display information in `DXDiag.exe`.

15. Click the **Start** menu.
16. Type `dxdiag` and click **Enter** after DXDiag shows up in the list, or **click** on it in the list.
17. After DxDiag launches, check the **Display** tab to verify that it is using the NVIDIA GPU/Driver.

VMware Horizon View Pool Configuration for vSGA

This section provides the steps required to enable vSGA for pools of virtual desktops within a VMware Horizon View environment.

Horizon View Pool Prerequisites

To enable 3D graphics rendering to the GPU, the Horizon View desktop and pool settings must meet the following criteria:

- The desktops must be Windows 7 (32-bit or 64-bit) or later
- The pool must use PCoIP as the default display protocol
- Users must *not* be allowed to choose their own protocol
- The desktop virtual machines must be Virtual Hardware version 9 or later

Video Memory (VRAM) Sizing

If you enable the 3D Renderer setting, configure the amount of VRAM that is assigned to the desktops in the pool by moving the slider in the Configure VRAM for 3D guests dialog box.

Table 6 documents the minimum and maximum VRAM for both vSGA and Software 3D rendering.

	SOFT 3D (SOFTWARE 3D)	VSGA (HARDWARE 3D)
Minimum	118MB	64MB
Default	64MB	96MB
Maximum	512MB*	512MB
* If you are still using Virtual Hardware version 8, the maximum VRAM is still 128MB and software rendering only.		

Table 6: Video Memory (VRAM) Sizing

Note: Whenever you change the 3D Renderer setting, it reverts the amount of video memory back to the default of 96MB. Make sure you change the video memory to the appropriate number after you change this setting.

VRAM settings that you configure in Horizon View Administrator take precedence over the VRAM settings that are configured for the virtual machines in vSphere Client or vSphere Web Client. Select the **Manage using vSphere Client** option to prevent this.

If you are using **Manage using vSphere Client**, VMware recommends that you use the Web Client to configure the virtual machines, rather than the software vSphere Client. This is because the software vSphere Client will not display the various rendering options; it will only display Enable/Disable 3D support.




Important: You must power off and on existing virtual machines for the 3D Renderer setting to take effect. Restarting or Rebooting a virtual machine does not cause the setting to take effect.

Note: After making VRAM changes to Horizon View Pool, there might be a short delay (sometimes a couple of minutes) before the message **Reconfiguring virtual machine settings** appears in vCenter. It is important to wait for this process to complete before power cycling the virtual machines.

Screen Resolution

If you enable the **3D Renderer** setting, configure the **Max number of monitors** setting for one or two monitors. You cannot select more than two monitors. The **Max resolution of any one monitor** setting is 1920 x 1200 pixels. You cannot configure this value higher.

 **Important:** You must power off and on existing virtual machines for the **3D Renderer** setting to take effect. Restarting or Rebooting a virtual machine does not cause the setting to take effect.

Horizon View Pool 3D Rendering Options

The 3D Renderer setting for desktop pools provides options to configure graphics rendering in various ways.

Manage Using vSphere Client

The 3D Renderer option set in vSphere Software or Web Client for a virtual machine determines the type of 3D graphics rendering that takes place. Horizon View does not control 3D rendering (vSphere Software Client will always set it as **Automatic**).

In vSphere Web Client, configure the Automatic, Software, or Hardware options. These options have the same effect as they do if you set them in Horizon View Administrator.

If you select the **Manage using vSphere Client** option, the **Configure VRAM for 3D Guests**, **Max number of monitors**, and **Max resolution of any one monitor** settings are inactive in Horizon View Administrator. Configure these settings for a virtual machine in vSphere Web Client.

Automatic

3D rendering is enabled. The ESXi host controls the type of 3D rendering that takes place. For example, the ESXi host reserves GPU hardware resources on a first-come, first-served basis as virtual machines are powered on. If all GPU hardware resources are already reserved when a virtual machine is powered on, ESXi uses the software renderer for that virtual machine. If you configure hardware-based 3D rendering, you can examine the GPU resources that are allocated to each virtual machine on an ESXi host.


Software

Software 3D rendering is enabled. The ESXi host uses software 3D graphics rendering only. If a GPU graphics card is installed on the ESXi host, it is ignored. If software rendering is configured, the default VRAM size is 64MB, the minimum size. In the Configure VRAM for 3D Guests dialog box, use the slider to increase the amount of VRAM that is reserved. With software rendering, the ESXi host allocates up to a maximum of 512MB per virtual machine (with Hardware version 9, and a maximum of 128MB if using Hardware version 8). If you set a higher VRAM size, it is ignored.

Hardware

Hardware 3D rendering is enabled. The ESXi host reserves GPU hardware resources on a first-come, first-served basis as virtual machines are powered on. If hardware GPU resources are not available, the virtual machine will not power on.

The ESXi host allocates GPU VRAM to a virtual machine based on the value you set in the Configure VRAM for 3D Guests dialog box. The minimum VRAM size is 64MB. The default size is 96MB. You can set a maximum VRAM size of 512MB.

 **Important:** If you configure the Hardware option, consider these potential constraints:

- If a user tries to connect to a desktop when all GPU hardware resources are reserved, the virtual machine will not power on, and the user will receive an error message.
- A desktop cannot be migrated by vSphere vMotion to an ESXi host that does not have GPU hardware configured.
- All ESXi hosts in the cluster must be version 5.1 or later. If a desktop is created on an ESXi 5.0 host in a mixed cluster, the virtual machine will not power on.
- Virtual machines must be configured for Hardware version 9 (vmx-09) in order to use hardware 3D. Hardware version 8 will allow only software 3D.

Disabled

3D rendering of any kind is inactive.

Best Practices for Configuring 3D Rendering

The 3D rendering options and other pool settings offer various advantages and drawbacks. Select the option that best supports your vSphere hardware infrastructure and your users' requirements for graphics rendering.

Automatic

The *Automatic* option is the best choice for many Horizon View deployments that require 3D rendering. This option ensures that some type of 3D rendering takes place even if GPU resources are completely reserved. In a mixed cluster of ESXi 5.1 and ESXi 5.0 hosts, the Automatic option ensures that a virtual machine is powered on successfully and uses 3D rendering—even if, for example, vSphere vMotion migrated the virtual machine to an ESXi 5.0 host.

A drawback with the Automatic option is that you cannot easily tell whether a virtual machine is using hardware or software 3D rendering. You also have no control over whether the virtual machine uses hardware or software to dictate any type of performance level for various use case requirements (e.g., some virtual machines require only software 3D for Office applications while other virtual machines require hardware 3D for CAD applications).

Note: To see if a virtual machine is using hardware 3D rendering, run the `gpuvm` command.

Hardware

The *Hardware* option guarantees that every virtual machine in the pool uses hardware 3D rendering, provided that GPU resources are available on the ESXi hosts. This option might be the best choice if all your users run applications that require intensive graphics resources.

With the Hardware option, you must strictly control your vSphere environment. All ESXi hosts must be version 5.1 or later, and must have GPU graphics cards installed. If all GPU resources on an ESXi host are reserved, Horizon View cannot power on a virtual machine for the next user who tries to log in to a desktop. You must manage the allocation of GPU resources and the use of vSphere vMotion to ensure that resources are available for your desktops. This option will work well if pools and hardware resources are sized and configured appropriately for the given use case. An example of this is to create a vSphere cluster where all hosts within the cluster have the same hardware GPUs, and you restrict these to running only the desktop pool(s) that requires hardware 3D rendering.

Manage Using vSphere Client

Select the *Manage using vSphere Client* option to support a mixed configuration of 3D rendering and VRAM sizes for virtual machines in a pool. In vSphere Web Client, you can configure individual virtual machines with different options and VRAM values.

Software

Select the *Software* option if you have ESXi 5.0 hosts only; some of the ESXi 5.1 hosts do not have GPU graphics cards; or your users require only software 3D rendering. This setting can be used on specific pools that will run in a cluster where some hosts have hardware GPUs, but the desktop pool does not require hardware 3D rendering, and you want to ensure those resources are available for virtual machines that do require it.


Enable Horizon View Pools for vSGA 3D Hardware Rendering

If all the prerequisites discussed above are met, both existing and new Horizon View pools can be enabled to use hardware 3D rendering.

Enable an Existing Horizon View Pool

1. In Horizon View Manager, navigate to the Horizon View pool that you wish to enable 3D rendering in and click **Edit**.
2. Go to the **Pool Settings** tab.


3. Scroll down the page until you reach the **Remote Display Protocol** section. In this section, you will see the **3D Renderer** option.
4. Select either **Hardware** or **Automatic** as the 3D rendering option from the dropdown list and click **Configure..** to configure the amount of VRAM you want each virtual desktop to have.
5. If the **3D Renderer** section is grayed out, ensure that you have **PCoIP** selected as your **Default Display Protocol**, and that **Allow users to choose protocol** is set to **No**.

 **Important:** You must power off and on existing virtual desktops for the 3D Renderer setting to take effect. Restarting or Rebooting a virtual desktop does not cause the setting to take effect.

Enable a New Horizon View Pool

During the creation of a new Horizon View pool, configure the pool to Normal until you reach the **Pool Settings** section.

1. Scroll down the page until you reach the **Remote Display Protocol** section.
2. In this section, you will see the **3D Renderer** option.
3. Select either **Hardware** or **Automatic** as the 3D rendering option from the dropdown list and click **Configure..** to configure the amount of VRAM you want each virtual desktop to have.
4. If the **3D Renderer** section is grayed out, ensure that you have **PCoIP** selected as your Default Display Protocol, and that **Allow users to choose protocol** is set to **No**.

 **Important:** You must power off and on existing virtual desktops for the 3D Renderer setting to take effect. Restarting or Rebooting a virtual desktop does not cause the setting to take effect.

Performance Tuning Tips

This section offers some tips to help improve the performance of both vSGA and vDGA.

Virtual Machine Resources

Unlike many traditional VDI desktops, desktops using high 3D capabilities must be provisioned with more vCPUs and memory. Make sure your desktop virtual machines meet the memory and CPU requirements for the applications you use. The minimum requirements VMware recommends for 3D workloads are two vCPUs and 4GB of RAM.

PCoIP

Occasionally PCoIP custom configurations can contribute to poor performance. By default, PCoIP is set to allow a maximum of 30 frames per second. Some applications require significantly more than 30FPS. If you notice that the frame rate of an application is lower than expected, reconfigure the PCoIP GPO to allow a maximum of 120 frames per second.

Another option with PCoIP is to enable **Disable Build-To-Lossless**. This will reduce the overall amount of PCoIP traffic, which will in turn reduce the load placed on both the virtual machine and endpoint.

Relative Mouse

If you are using an application or game and the cursor is moving uncontrollably, enabling the relative mouse feature might improve mouse control.

Relative mouse is a new Windows Horizon View Client (v5.3) feature that changes the way client mouse movement is tracked and sent to the server via PCoIP. Traditionally PCoIP uses absolute coordinates. Absolute mouse events allow the client to render the cursor locally, which is a significant optimization for high-latency environments. However, not all applications work well when using absolute mouse. Two notable classes of applications, CAD applications and 3D games, rely on relative mouse events to function correctly.

With the introduction of vSGA and vDGA in Horizon View 5.2, VMware expects the requirement for relative mouse to increase rapidly as CAD and 3D games become more heavily used in Horizon View environments.

The Horizon View Windows client v5.3 is required to enable relative mouse. At the time of writing, this feature is not available through any other software clients or Zero Clients.

Enabling Relative Mouse

The end user can enable relative mouse manually.

To manually enable relative mouse, right-click the **Horizon View Client Shade** at the top of the screen and select **Relative Mouse**. You should see a tick sign (✓) next to Relative Mouse.

Note: Relative Mouse must be selected on each and every connection. There is no option to enable this by default at the time of this writing.

Virtual Machines Using VMXNET3

For desktop virtual machines using VMXNET3 Ethernet adapters, you can significantly improve peak video playback performance of your Horizon View desktop by following these steps, which are recommended by Microsoft for virtual machines:

1. Start Registry Editor (**Regedt32.exe**).
2. Locate the following key in the registry:

```
HKLM\System\CurrentControlSet\Services\Afd\Parameters
```


3. In the **Edit** menu, click **Add Value**, and then add the following registry value:

Value Name: FastSendDatagramThreshold

Data Type: REG_DWORD

Value: 1500

4. Quit Registry Editor.

Note: A reboot of the desktop virtual machine is required after changing this registry setting. If this setting does not exist, create it as a **DWORD** value.

Further information on what this change does can be found on the [Microsoft Support](#) Web site.

Workaround for CAD Performance Issue

VMware has experienced a performance issue when users deploy CATIA.

Sometimes when you are working with CAD models (turning and spinning), you might find that objects move with a delay and their movement is irregular. However, the objects themselves are displayed clearly, without blurring.


The workaround in this case is to disable the **MaxAppFrameRate** registry entry. The registry key can be found at:

HKLM\Software\VMware, Inc.\VMware SVGA DevTap\MaxAppFrameRate

Change this registry setting to:

dword:00000000

Note: If this registry key does not exist, the setting defaults to 30.

 **Important:** This change can negatively effect other applications, so use with caution and make this change only if you are experiencing the symptoms mentioned above.

Resource Monitoring

The following section documents ways to monitor the GPU resources on each ESXi host.

gpuvmm

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

To run the utility, run the following command from a console on the host or an SSH connection:

```
# gpuvmm
```

For example, the utility might display the following output:

```
# gpuvmm
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 run the utility, run the following command from a console on the host or an SSH connection.

```
# nvidia-smi
```

This will show several details of GPU usage at the point in time when you issued the command (this display is *not* dynamic and must be reissued to update the information). You can also issue the following command:

```
# watch -n 1 nvidia-smi
```

This command will issue the `nvidia-smi` command every second to provide a refresh of that point-in-time information.

Note: The most meaningful metric in the `nvidia-smi` display is at the right of the middle section. It shows the percentage of each GPU's processing cores in use at that point in time. This can be helpful if you have to troubleshoot poor performance—verify if the GPU processing cores are being overtaxed, and the cause of their poor performance.

For more details on how to use the `nvidia-smi` tool, please refer to the [nvidia-smi documentation](#).

Troubleshooting

This section provides troubleshooting steps to follow if you have issues with 3D rendering with both vSGA and vDGA.

Xorg

Xorg Fails to Start

If you attempt to start Xorg and it fails, this is most likely due to the NVIDIA VIB module not loading properly. Often, this can be resolved by warm rebooting the host (it appears in some instances that the GPU is not fully initialized by the time the VIB module tries to load).

If Xorg still fails to start, try some of the following steps.

Verify that the NVIDIA VIB Bundle Is Installed

To verify that the NVIDIA VIB bundle is installed, run the following command:

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

If the VIB is installed correctly, you should see a similar output to the example below:

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

Verify that the NVIDIA Driver Loads

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

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

If the driver loads correctly, you should see a similar output to the example below:

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

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

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

Search for **NVRM**.

Often, an issue with the GPU will be identified in the vmkernel.log.

Verify that Display Devices Are Present in the Host

To verify that display devices are present in the host, run the following command:

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

You should see a similar output to the following:

```
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: NVIDIAQuadro 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
Spawned Bus: 0x00
Flags: 0x0201
Module ID: 71
Module Name: nvidia
Chassis: 0
Physical Slot: 1
Slot Description:
Passthru Capable: true
Parent Device: PCI 0:0:1:0
Dependent Device: PCI 0:0:1:0
Reset Method: Bridge reset
FPT Sharable: true
```

Possible PCI Bus Slot Order Issue

If you installed a second lower-end GPU in the server, it is possible that the order of the cards in the PCIe slots will choose the higher-end card for the ESXi console session. If this occurs, swap PCIe slots between the two GPUs; or change the Primary GPU settings in the server BIOS.

Check Xorg Logs

If the correct devices are present, view the Xorg log file to see if there is an obvious issue.

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

sched.mem.min Error

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

```
sched.mem.min = "4098"
```

Note: "4098" must match the amount of configured virtual machine memory. The example above is for a virtual machine with 4GB of RAM.

About the Author and Contributors

Simon Long, Senior Consultant in Global Professional Services Engineering (End-User Computing) at VMware wrote this document.

Contributors to this document include:

- Aaron Blasius
- Pat Lee
- Warren Ponder
- Joel Lindberg
- Rasmus Jensen
- Josh Spencer
- Tommy Walker
- Vincent Wu

All contributors are internal to VMware, Inc.

