Deploying Hardware-Accelerated Graphics with View Virtual Desktops in Horizon 6

VMware Horizon 6.2

TECHNICAL WHITE PAPER

vmware®

Table of Contents

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 carried high acquisition and maintenance costs, and they required the user to be present in the lab or office. In areas such as oil and gas, space exploration, aerospace, engineering, science, and manufacturing this meant that an organization's individuals had to be located in the same physical location as the workstation.

Virtualizing the graphics workstation provides immediate benefits by enabling workers to collaborate and be productive from any location, using any endpoint device. This enables organizations to employ the best global talent available and work more easily with valued partners and suppliers. Virtualization also offers both CapEx and OpEx savings, driving down acquisition and maintenance costs.

Innovations in hardware-based graphics acceleration and virtualization technology have made graphics workstation virtualization a reality. With the latest generation of virtualization software, display protocols, and advanced graphics cards, virtual desktops can meet and even exceed the performance expectations of designers and researchers accustomed to working on high-end standalone graphics workstations. The ability of a single physical GPU to support multiple users also has important implications for IT as well as for users who are increasingly accustomed to mobility.

Architectural Innovations

Moving the graphics acceleration hardware from the workstation to a server is a key architectural innovation. This shift changes the computing metaphor for graphics processing, putting the additional compute, memory, networking, and security advantages of the data center at the disposal of the user, so that complex models and very large data sets can be accessed and manipulated from virtually anywhere. With appropriate network bandwidth and suitable remote client devices, IT can now offer the most advanced users an immersive 3D graphics experience while freeing them from the limitations of the old computing metaphor. Fewer physical resources are needed, the wait time to open complex models or run simulations is greatly reduced, and users are no longer tied to a single physical location.

With vGPU, several advanced graphics users can typically be hosted on a single physical GPU without diminishing performance. A vSGA solution enables a larger number of users who run less intensive workloads or who require only occasional access to the shared GPU. With vDGA, a GPU is reserved for the exclusive use of a single user. A vDGA solution can be used to meet specific compatibility requirements, but it provides a low consolidation benefit.

Centralizing graphics workloads in the data center also simplifies the management, provisioning, and maintenance of desktops, applications, and data.

Technological Foundation

The key enabling technologies for these solutions are:

- Graphics cards that support GPU virtualization
  - NVIDIA GRID
  - NVIDIA Tesla
  - AMD FirePro
  - Intel HD Graphics P4700
- VMware vSphere® 5.1 or later
- VMware View® 5.3 or later (now called View in VMware Horizon® 6)
- Protocols
  - PCoIP
  - Secure WebSocket protocol for remote display
- VMware vRealize® Operations Manager™

About This Paper

This paper begins by explaining the differences between the four types of graphics acceleration, matching them with typical use cases. Later sections provide installation and configuration instructions as well as best practices and troubleshooting. For a more detailed discussion of use cases, sizing, and test results, see the VMware Horizon 6 3D Engineering Workloads Reference Architecture.

Types of Graphics Acceleration

Horizon 6 offers four types of 3D graphics acceleration, which map fairly closely to four common use cases.

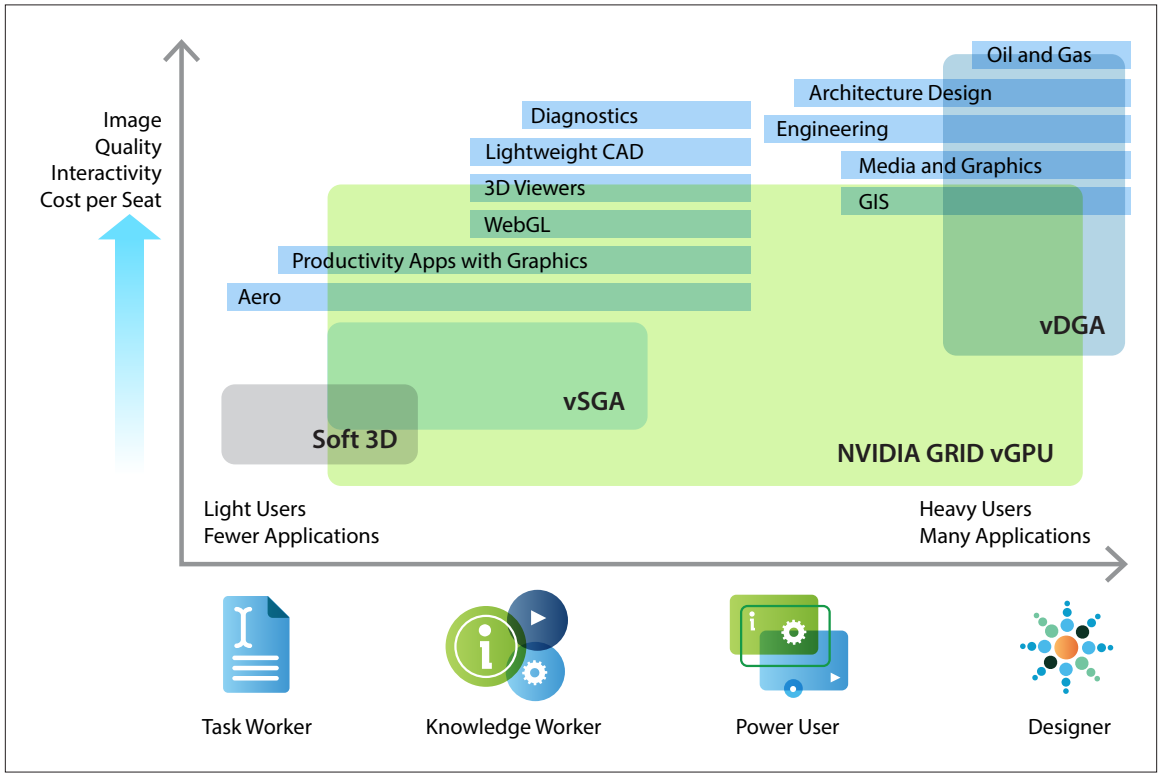

Figure 1: 3D Technologies for Different Use Cases

Soft 3D

Software-based Soft 3D is included in all editions of Horizon 6 to provide improved performance for users who can be characterized as task workers. Those who perform routine office tasks, or who spend most of their time coding, or even writing about graphics, can benefit from this technology without requiring a hardware-based solution.

vSGA

Virtual Shared Graphics Acceleration (vSGA) provides a significant level of enhancement for office workers and executives—people who create or manipulate 2D or 3D graphics using Microsoft Office, Adobe Photoshop, and other non-specialized end-user applications. A vSGA solution can improve performance for this use case and provide high levels of consolidation of users across GPUs. One graphics acceleration card installed on the vSphere host can support a large number of users who require only occasional access, and many physical servers can support multiple graphics acceleration cards. Virtual Shared Graphics Acceleration does not support all graphics API versions and may not be supported by certain ISVs. The vSGA solution uses a proprietary VMware graphics driver that provides an abstraction layer on top of the physical GPU, so that virtual desktops benefit from hardware acceleration, but they do not have direct physical access to the GPU. This arrangement allows functionality such as VMware vSphere vMotion® but means that not all graphics APIs are supported.

vGPU

Virtual Graphics Processing Unit (vGPU) is similar in concept and design to vSGA, but it provides mediated, direct access to the physical GPU—so multiple users can share a single GPU—using native graphics card drivers. NVIDIA GRID cards can support several engineers, data scientists, and designers who rely on advanced graphics software such as CAx applications, with performance that can meet or exceed that of a traditional dedicated workstation. This is usually the most cost-effective solution both for large enterprises and for small-to-medium-sized design studios, largely because of the cost savings that result from supporting several designers per graphics card instead of one. A vGPU solution supports OpenGL and DirectX graphic APIs. With NVIDIA GRID 2.0, vGPU can also support OpenCL and NVIDIA CUDA.

GRAPHICS CARD	PHYSICAL GPUs	VIRTUAL GPU	USE CASE	FRAME BUFFER	MAXIMUM NUMBER OF DISPLAYS PER USER AT MAXIMUM RESOLUTION OF 2560x1600	MAXIMUM vGPUs	
						PER GPU	PER GRID CARD
GRID K2	2	K280Q	Designer	4 GB	4	1	2
		K260Q	Designer	2 GB	4	2	4
		K240Q	Designer or Power User	1 GB	2	4	8
		K220Q	Designer or Power User	512 MB	2	8	16
GRID K1	4	K180Q	Entry Designer	4 GB	4	1	4
		K160Q	Power User	2 GB	4	2	8
		K140Q	Power User	1 GB	2	4	16
		K120Q	Knowledge Worker	512 MB	2	8	32

Table 1: ESXi Sizing for Rich Graphics with NVIDIA GRID vGPU

vDGA

Virtual Dedicated Graphics Acceleration (vDGA) provides direct pass-through to a physical GPU, providing a user with unrestricted, fully dedicated access to a single vGPU. Although there are consolidation and management trade-offs associated with dedicated access, vDGA offers the highest level of performance for users with the most intensive applications, such as 3D design, molecular modeling, and medical diagnostics software. It enables the use of applications that run OpenGL, DirectX, OpenCL, and NVIDIA CUDA.

Prerequisites

This section describes the hardware, software, client device, and application requirements for vSGA, vGPU, and vDGA solutions.

Hardware Requirements

The hardware requirements for the vDGA and vGPU hardware-based 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, most 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 chosen GPU card in the appropriate PCIe slot.
Host power supply unit (PSU)	Check the power requirements of the GPU to make sure that the PSU is powerful enough and contains the proper power cables to power the GPU. For example, a single NVIDIA K2 GPU can use as much as 225 Watts of power and requires either an 8-pin PCIe power cord or a 6-pin PCIe power cord.
Virtualization Technology for Directed I/O (VT-d)	Verify that the host supports either Intel VT-d or AMD IOMMU (input-output memory management unit). Without this, GPU pass-through cannot be enabled. To see if VT-d or AMD IOMMU is enabled on the host, check the server BIOS. To locate this setting in the server BIOS, contact the hardware vendor.
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 because the VMware ESXi™ console display adapter is not available to Xorg. If the high-end AMD or NVIDIA GPU card 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 vDGA and vGPU

Note: GPU support is dictated by the graphics-card vendor, not by VMware. For more information, see NVIDIA GRID Partners or AMD Professional GPUs for Servers.

Software Requirements

The software requirements for the three hardware-based graphics acceleration solutions are listed in Table 3.

COMPONENT	DESCRIPTION
VMware vSphere hypervisor	• vSGA and vDGA – ESXi 5.1 U1 or ESXi 5.5 (ESXi 5.5 recommended) • vGPU – ESXi 6.0
View in VMware Horizon 6	• vSGA – View 5.2 or later (View in Horizon 6 version 6.1 recommended) • vDGA – View 5.3 or later (View in Horizon 6 version 6.1 recommended) • vGPU – Horizon 6 version 6.1 or later • Linux – Horizon 6 version 6.1.1 (with Horizon Client 3.4)
Display protocol	• vSGA, vGPU, and vDGA – PCoIP with a maximum of two display monitors • vDGA – Secure (HTTPS) WebSocket connection from the Horizon Client (Horizon 6 version 6.1.1 required), up to four display monitors
NVIDIA drivers	• vSGA – NVIDIA drivers for vSphere ESXi 5.5 version (latest version) • vDGA – Tesla/GRID desktop driver version (latest version) • vGPU – NVIDIA GRID drivers for VMware (latest version) **Note:** These drivers are supplied and supported by NVIDIA and can be downloaded from the NVIDIA Driver Downloads page.
Guest OS	• vSGA – Windows 7, 32- or 64-bit • vDGA – Windows 7, 64-bit • vGPU – Windows 7, 64-bit

Table 3: Software Requirements for vSGA, vDGA, and vGPU

Client Device Requirements

3D applications require network bandwidth sufficient to handle traffic between the GPU and client devices, and the client devices themselves must have sufficient processing power to display constantly changing images in real time.

Some low-end thin clients do not have the CPU processing power they need to decode PCoIP data fast enough to render a smooth and uninterrupted end-user experience. However, this is not always the case for every environment and end-user 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 PCoIP Tera2-based chip or a modern Core i3-, i5-, or i7-based Windows PC to achieve best performance with multiple high-resolution displays.

Note: Prior-generation Teradici zero clients with Tera1 chips can support a maximum rate of 30 fps, whereas the current Tera2 chip can achieve up to 60 fps. High frame rates can be important to the usability of certain engineering applications as well.

Application Requirements

Both vGPU and vDGA support the versions of DirectX and OpenGL that their respective graphics drivers support, typically the latest version of these technologies. For more information, see the NVIDIA Driver Downloads page.

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

DirectX 9.0c
Currently, vSGA supports only up to DirectX 9.0c. Applications that require a later version of DirectX may not perform correctly with vSGA.

OpenGL 2.1
Currently, vSGA supports only up to OpenGL 2.1. Applications that require a later version of OpenGL may not perform correctly with vSGA.

CUDA
CUDA is currently supported only with vDGA, or vGPU with NVIDIA GRID 2.0. For further details, see the CUDA FAQ.

ESXi Installation and Configuration

Brief installation instructions for graphics cards are provided in the following sections. For further installation and configuration instructions see the NVIDIA GRID vGPU Deployment Guide for VMware Horizon 6.1 or later or the AMD driver download page.

Confirm Graphics-Card Installation

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
            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
```

Install the GPU vSphere Installation Bundle (VIB)

Before using any of the supported hardware-based graphics acceleration modes (vSGA, vGPU, or vDGA) in vSphere, install and configure the AMD or NVIDIA GPU vSphere Installation Bundle (VIB) on each ESXi host.

To use ESXi Image Builder to create a bootable image that contains the NVIDIA VIB, see the vSphere Installation and Setup Guide. For more information on deployment, see the NVIDIA GRID vGPU Deployment Guide for VMware Horizon 6.1 or later.

To install the GPU VIB:

1. Download either the NVIDIA or the AMD VIB from the appropriate download page.
   - For NVIDIA GPUs, see:
     - ESXi 5.1
     - ESXi 5.5
     - ESXi 6.0 vGPU
     - ESXi 6.0 vSGA
   - For AMD GPUs, see the AMD driver downloads page.
2. Place the ESXi host in maintenance mode.

```
# vim-cmd hostsvc/maintenance_mode_enter
```

3. Open a command prompt on the ESXi host and run one of the following commands:

```
# esxcli software vib install -v /<path_to_vib>/nvidia_vib
```

or

```
# esxcli software vib install -v /<path_to_vib>/amd_vib
```

Important: The full path to the VIB file is required, even when running from the directory it is stored in. If the VIB file is located on a VMware Virtual SAN™ datastore, navigate to that directory before running the VIB.

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: <path_to_vib>
      VIBs Removed:
      VIBs Skipped:
```

Note: Although the output states that a reboot is not required (`Reboot Required: false`), it is a best practice to reboot the ESXi host at this point.

Start the Xorg Service
Before you start any virtual machines, make sure that the Xorg service is running. You can use the vSphere Client, the vSphere Web Client, or the CLI.

To use the vSphere Client to verify that the Xorg service is running:

1. Connect to the ESXi host.
   a. On the Configuration tab under Software, click **Security Profile**.
   b. Click the **Properties** link for Services.
   c. Verify that Xorg is `Running` and has the desired startup policy.
2. If Xorg is `Stopped`:
   a. Select **xorg** and click **Options**.
   b. Click **Start**.
   c. Select the **Startup Policy** and click **OK**.

To use the vSphere Web Client to verify that the Xorg service is running:

1. Connect to vCenter Server.
   From the Inventory, select the ESXi host.
   On the Manage tab under System, click **Security Profile**.
   Click the **Edit** button for Services.
   Verify that Xorg is `Running` and has the desired startup policy.
2. If Xorg is Stopped:
   a. Select **xorg** and click **Start**.
   b. Select the **Startup Policy** and click **OK**.

To use the CLI to verify that the Xorg service is running, run the following command:

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

Uninstall and Update the GPU VIB

To update the GPU VIB, you must uninstall the currently installed VIB and install the new VIB.

To uninstall the currently installed VIB:

1. Stop all virtual machines that use 3D acceleration.
   a. Place the ESXi host into maintenance mode.
   b. Open a command prompt on the ESXi host.
   c. Stop the Xorg service by running the following command:

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

2. Remove the NVIDIA VMkernel driver by running the following command:

```
# vmkload_mod -u nvidia
```

3. Identify the NVIDIA VIB name by running the following command:

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

4. Remove the VIB by running the following command:

```
# esxcli software vib remove -n <nameofNVIDIAVIB>
```

You can now install a newer GPU VIB.

Post-Installation Checks

This section contains various commands that can be used to install the GPU card and its respective drivers correctly.

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. To check the status of Xorg, run 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.

gpuvm

The `gpuvm` command gives a list of working GPUs, with information on which virtual machine is using which GPU and the amount of video memory reserved for each GPU.

Issue the gpuvm command through an ESXi SSH session:

```
# gpuvm
```

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

The NVIDIA-specific nvidia-smi tool shows how much of each GPU is in use when the NVIDIA driver is used. To use **nvidia-smi**, issue the following command in an SSH session:

```
# nvidia-smi
```

This shows several details of GPU usage at the time you issue 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 issues the **nvidia-smi** command every second to provide a refresh of that point-in-time information. Sample output is shown in Figure 2.

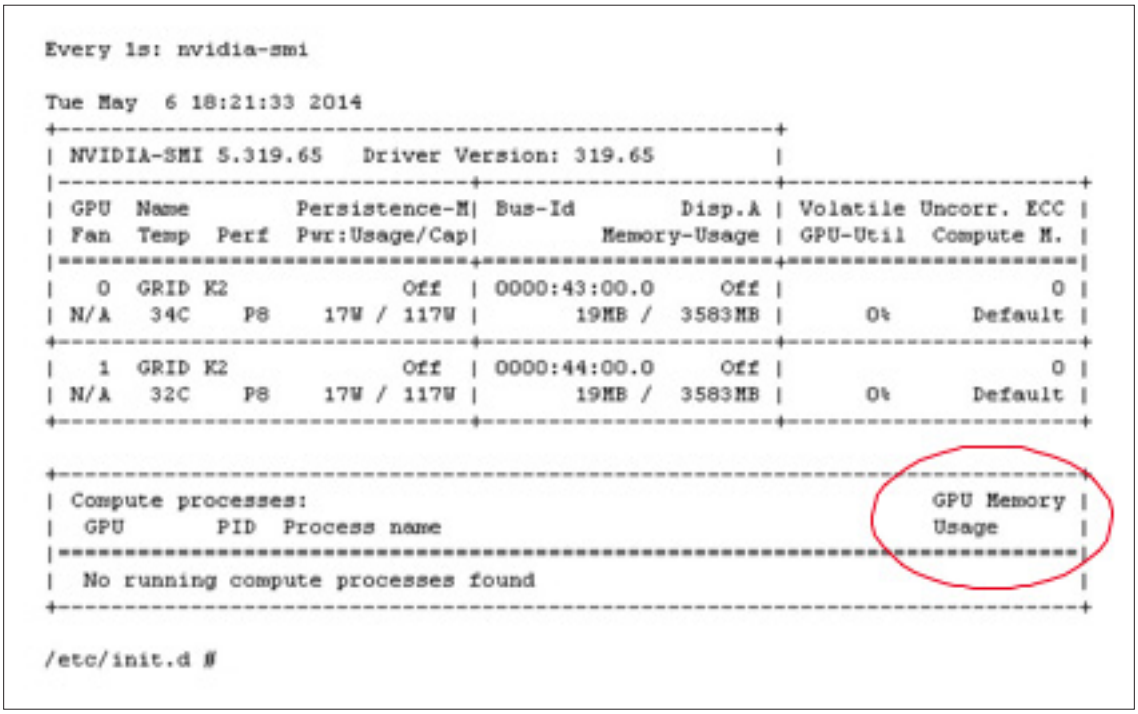

Figure 2: The nvidia-smi Display

Note: The most meaningful metric in the **nvidia-smi** display is GPU memory usage. It is located at the right of the lower section, as shown in Figure 2. It shows the percentage of each GPU's processing cores in use when you issue the command. This metric can be helpful in troubleshooting poor performance.

Log Files

Verify that the virtual machine has graphics acceleration by searching for OpenGL in the virtual machine's vmware.log file. 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"
```

However, if the virtual machine is using Soft 3D, the vmware.log file will contain:

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

Enabling GPU Pass-Through for vDGA and vGPU

This section describes how to enable GPU pass-through at the host level and how to prepare virtual machines for vDGA or vGPU.

Enable the Host for GPU Pass-Through

To enable an ESXi host for GPU pass-through from the host to the virtual desktop, follow the checks and steps in the following sections.

Verify That VT-d or AMD IOMMU Is Enabled

Before pass-through can be enabled, check to make sure VT-d or AMD IOMMU is enabled on the ESXi host by consulting the server BIOS. If there are any questions about finding this setting in the server BIOS, contact the hardware vendor.

Enable Device Pass-Through Using the vSphere Web Client

To enable GPU device pass-through on the ESXi host, perform the following steps:

1. Using the vSphere Web Client, connect to VMware vCenter and select the host that has the GPU card installed.
   a. Click the **Manage** tab for this host.
   b. If the Hardware group is not expanded, click the down arrow next to it.
   c. Click PCI Devices.
   d. Right-click one of the GPUs installed in the system and select **Edit**.
   e. In the Edit PCI Device Availability window, select the check box or boxes that correspond to the GPU adapters you wish to use for pass-through.
2. Click **OK**.

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

Note: If the PCI devices are not shown as `Available`, the host must be restarted to enable them.

Virtual Machine Configuration and Setup

This section provides instructions for configuring and enabling virtual machines for 3D graphics acceleration.

Configure Virtual Machine Video Card 3D Capabilities

To configure virtual machine video card 3D capabilities using the vSphere Client:

1. Connect to the ESXi host.

   a. Select the virtual machine.

   b. On the Summary tab under Commands, click Edit Settings.

   c. Under Hardware, click Video card.

   d. Under Displays and video memory, set the video card 3D capabilities:

   • Set the Total video memory to a value between 64 MB and 128 MB.

   Most applications should work with 128 MB. Video memory values larger than 128 MB are available only with virtual machines with hardware version 9 or 10.

   • Under 3D graphics, select Enable 3D Support.

2. Click OK.

   Note: You cannot set the 3D renderer from the vSphere Client. Use the vSphere Web Client to configure video card 3D capabilities.

To set the 3D renderer from the vSphere Web Client:

1. Connect to vCenter Server.

   a. Select the virtual machine.

   b. Click a data center, folder, cluster, resource pool, or host.

   c. Click the **Related Objects** tab, and click **Virtual Machines**.

   d. On the Manage tab, click **Settings**.

   e. Click **VM Hardware**.

   f. Click **Edit**.

2. Expand the Video card setting.

   a. Set the Total video memory to a value from 64 MB to 128 MB.

   Most applications should work with 128 MB video memory. Values larger than 128 MB are available only on virtual machines with hardware version 9 or 10.

   • For 3D Graphics, select Enable 3D Support.

   • For 3D Renderer, select Hardware or Automatic.

For more information, see Best Practices for Configuring 3D Rendering in Horizon 6.

   b. Click **OK**.

To configure virtual machine video card 3D capabilities from View Administrator:

1. Connect to View Administrator.
   a. Add or edit a desktop pool. For more information, see View Administration.
   b. On the Pool Settings page (or tab), set the 3D Renderer to Automatic or Hardware.
2. Complete the Add Pool wizard or click **OK** in the Edit Pool dialog.

Important: If you use the **Manage using vSphere Client** option in View Administrator, you can configure 3D rendering from the vSphere Web Client on a per-virtual-machine basis. If **Manage using vSphere Client** is not selected in View Administrator, the 3D rendering settings made in View Administrator apply to the entire pool and overwrite any 3D rendering settings made in the vSphere Web Client.

Enable the Virtual Machine for vDGA or vGPU

To enable a virtual machine for GPU pass-through, use the following checks and steps.

1. Update the virtual machine to the latest hardware version, or at least to version 9.

   You must upgrade any virtual machine that will use 3D graphics to the latest hardware version (for example, HWv9 shows as vmx-09) to ensure maximum compatibility with GPU pass-through.

   In the vCenter client, right-click the virtual machine and select **Upgrade Virtual Hardware**.

   Note: You can manage virtual machines with a hardware version of 9 or later only with the vSphere Web Client.

   Note: With ESXi 5.5, the virtual machine will be upgraded to version 10 (vmx-10).

   If you are using ESXi 5.x, adjust pciHole.start.

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

   For virtual machines that have more than 2 GB 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"
```

2. Add the PCI device.

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

   a. Using the vSphere Web Client, connect directly to the ESXi host that has the GPU card installed, or select the host in vCenter.
   b. Right-click the virtual machine and select **Edit Settings**.
   c. From the New Device drop-down menu at the bottom of the window, select **PCI Device**.
   d. Click **Add**.
   e. Select the appropriate PCI device from the drop-down menu.
      - For vDGA, select the available GPU.
      - For vGPU, select **NVIDIA GRID vGPU**.
   f. For vGPU, you are prompted to select a vGPU Profile (see Table 1).
   g. If prompted, select **Reserve All Memory** and click **OK**.

3. Install the NVIDIA driver.

   a. Download the latest NVIDIA Windows desktop driver and install it on the virtual machine. Ensure you have downloaded the correct NVIDIA GRID vGPU driver for Windows if using vGPU.

      For a list of available NVIDIA drivers, see the NVIDIA Driver Downloads page.

   b. After the driver is installed, reboot the virtual machine.

4. Install the View Agent.

   a. After the NVIDIA driver is installed correctly, install the View Agent on the virtual machine.

   b. Reboot when requested.

   c. Connect to the virtual machine for the first time.

      To activate the NVIDIA display adapter, you must connect to the virtual machine for the first time over PCoIP in full-screen mode from the endpoint at native resolution, otherwise the virtual machine will use the Soft 3D display adapter. Virtual Dedicated Graphics Acceleration (vDGA) does not work through the vSphere console session.

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

   d. Click the **Start** menu.

   e. Click the DXDiag icon or type dxdiag and click **Enter** after DxDiag appears in the list, or click it in the list.

   f. After DxDiag launches, check the Display tab to verify that the virtual machine is using the NVIDIA GPU and NVIDIA driver.

Note: If the GPU is not enabled, make sure that the NVIDIA control panel shows active displays on the GPU and that no active displays are attached to the VMware Soft 3D driver.

View Setup and Configuration

This section outlines the steps required to enable vSGA or vGPU for pools of virtual desktops in a View environment.

Pool Prerequisites for View in Horizon 6

To enable 3D-graphics rendering to the GPU, the 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-machine hardware version 9 or later.

Enabling View Pools for Hardware 3D Rendering

If all the prerequisites discussed previously are met, both existing and new View pools can use hardware 3D rendering.

Enable an Existing View Pool

To enable an existing View pool to use hardware 3D rendering:

1. In View Administrator, navigate to the View pool that you wish to enable 3D rendering in, and click **Edit**.
2. Navigate to the Pool Settings tab.
3. Scroll down the page until you reach the Remote Display Protocol section. In this section, you see the 3D Renderer option.
4. Select either **Hardware** or **Automatic** as the 3D rendering option from the drop-down menu and click **Configure**.
5. Configure the amount of VRAM you want each virtual desktop to have.

Note: If the 3D Renderer section is dimmed, make sure that you have **PCoIP** selected as the **Default Display Protocol**, and that **Allow users to choose protocol** is set to **No**.

Important: The 3D Renderer setting does not take effect until you power existing virtual desktops off and then power them on again. Restarting or rebooting a virtual desktop does not cause the setting to take effect.

Enabling a New View Pool

During the creation of a new 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. In this section, you see the 3D Renderer option.
2. Select either **Hardware** or **Automatic** as the 3D rendering option from the drop-down menu and click **Configure**.
3. Configure the amount of VRAM you want each virtual desktop to have.

Note: If the 3D Renderer section is grayed out, make sure that you have **PCoIP** selected as the **Default Display Protocol**, and that **Allow users to choose protocol** is set to **No**.

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

	SOFT 3D (SOFTWARE 3D)	HARDWARE 3D
Minimum	1.18 MB	64 MB
Default	64 MB	96 MB
Maximum	512 MB*	512 MB

Table 4: Video Memory (VRAM) Sizing

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

VRAM settings that you configure in View Administrator take precedence over VRAM settings that are configured for the virtual machines in vSphere Client or vSphere Web Client. The preferred option is to select **Manage using vSphere Client**.

Note: If you are using the **Manage using vSphere Client** option, VMware recommends that you use the vSphere Web Client, rather than the traditional vSphere Client, to configure the virtual machines. The traditional vSphere Client does not display the full range of rendering options; it displays only the option to enable or disable 3D support.

Note: After you make VRAM changes to the View desktop pool, there may be a short delay (sometimes a couple of minutes) before the `Reconfiguring virtual machine settings` message appears in the vCenter console. 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 maximum number of monitors setting for one or two monitors. You cannot select more than two monitors. The maximum resolution of any one monitor setting is 1920 x 1200 pixels. You cannot configure this value to be higher.

* If you are still using virtual-machine hardware version 8, the maximum VRAM is 128 MB, and only software rendering is allowed.

Pool 3D-Rendering Options in View

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

To access 3D-rendering options in View Pool Settings:

1. Open the View Administrator dashboard, click your desired pool, and select **Edit**.
2. Open Pool Settings on the second tab, and click the drop-down menu for **3D Renderer** to view the 3D-rendering options.

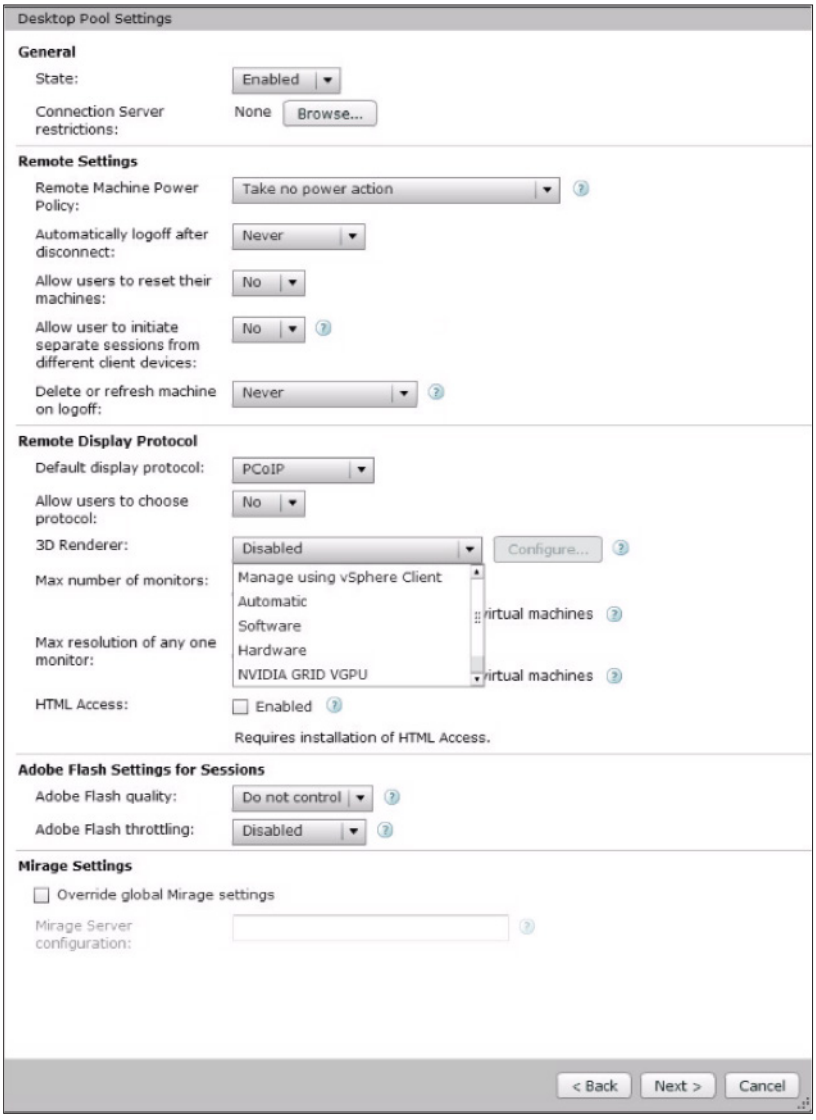

Figure 3: 3D Rendering Options in View Pool Settings

Manage Using vSphere Client
You can use the vSphere Client or the vSphere Web Client to set the 3D Renderer option. In this case, View does not control 3D rendering. The vSphere Client always sets this option to Automatic. The vSphere Web Client lets you choose the Automatic, Software, or Hardware option. If you use the vSphere Web Client, these options have the same effect as they do when you set them in View Administrator.

When you select the Manage using vSphere Client option in the properties of the virtual machine, the Configure VRAM for 3D Guests, maximum number of monitors, and maximum resolution of any one monitor settings become dimmed in View Administrator. Use the vSphere Web Client to configure these settings.

Automatic
With the Automatic setting, 3D rendering is enabled, and 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 3D renderer for that virtual machine. When you configure hardware-based 3D rendering, you can examine the GPU resources that are allocated to each virtual machine on an ESXi host.

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

Hardware
With the Hardware setting, 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 that you set in the Configure VRAM for 3D Guests dialog box. The minimum VRAM size is 64 MB. The default size is 96 MB. You can set a maximum VRAM size of 512 MB.

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

- If a user tries to connect to a desktop when all GPU hardware resources are reserved, the virtual machine does not power on, and the user receives 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) or later in order to use hardware 3D rendering. Hardware version 8 allows only software 3D rendering.

Disabled
With the Disabled setting, all types of 3D rendering are inactive.

NVIDIA GRID vGPU
Select the NVIDIA GRID vGPU setting if you are using NVIDIA GRID vGPU and want to force the VMs to use vGPU only.

Best Practices for Configuring 3D Rendering in Horizon 6

The 3D-rendering options in View Pool Settings offer various advantages and drawbacks. Select the option that best supports your vSphere hardware infrastructure and your users' requirements for graphics rendering.

When to Select the Automatic Option

The Automatic option is the best choice for many 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 discern 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 (for example, some virtual machines require only software 3D rendering for Microsoft Office applications, while other virtual machines require hardware 3D rendering for CAD applications).

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

When to Select the Hardware Option

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 may 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, 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 make sure that resources are available for your desktops. This option works well if pools and hardware resources are sized and configured appropriately for the given use case. For example, create a vSphere cluster where all hosts within the cluster have the same hardware GPUs and restrict these to running only the desktop pools that require hardware 3D rendering.

When to Select the Manage Using vSphere Client Option

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

When to Select the Software Option

Select the Software option for any of the following use cases:

- ESXi 5.0 hosts only
- Some ESXi 5.1 hosts do not have GPU graphics cards
- Users require only software 3D rendering

This setting can be used on specific pools that run in a cluster in which some hosts have hardware GPUs, but the desktop pool does not require hardware 3D rendering, and you want to make sure those resources are available for virtual machines that do require it.

Performance Tuning Tips and Best Practices

This section offers tips to improve the performance of vSGA, vGPU, and vDGA.

COMPONENT	BEST PRACTICES
CPU	• During testing of 3D applications, we found that the most common CAD applications are single-threaded. CPUs with higher frequency lead to better end-user experience, so choose frequency over core count. • vCPUs assigned to a VM: Assign at least 2 vCPUs (we recommend 4) to a vGPU-enabled VM because the remote protocol requires CPU.
Host hardware	• Set the Power Profile to Maximum Performance to make sure CPUs can reach their highest clock speeds. • Load the server with enough memory for guests. • NVIDIA GRID vGPU-enabled guest operating systems require memory reservation. • Memory overcommit does not work with vGPU.
Storage	• Configure storage with sufficient IOPS. • Power users and designers expect SSD performance because they use it locally.

Table 5: Performance Tips for vSGA, vGPU, and vDGA

Some other best practices for sizing virtual machines for rich graphics workloads are summarized in Table 6.

VIRTUAL MACHINE	VCPU	VRAM	OS	VMWARE OS OPTIMIZATION TOOL
Hardware v11 VMXNET3 NIC • For vGPU: Configure Profile • For vDGA: Enable Pass-through	2 vCPUs • 1 vCPU, OS • 1 vCPU, PCoIP At least 4 vCPUs for power users	• 4 GB per power user or entry-level engineer • 8 GB per mid-level engineer or video designer • 16 GB per advanced engineer • 32 GB for CAD/CAM applications • 64 GB for digital mock-ups	vDGA: 64-bit Windows vGPU: 32-bit or 64-bit Windows Install: • VMware Tools • View Agent • NVIDIA driver (vGPU)	40% savings Download from VMware OS Optimization Tool

Table 6: Virtual Machine Sizing Guidelines

Virtual Machine Configuration

The hardware versions for each type of hardware-based graphics acceleration are listed in Table 7.

GRAPHICS ACCELERATION CONFIGURATION	VIRTUAL MACHINE HARDWARE VERSION
NVIDIA GRID vGPU	Version 11 or later
vDGA	Version 10 or later
vSGA	Version 9 or later

Table 7: Virtual Machine Configuration

Virtual Machines Using VMXNET3

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

1. Start the 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 add the following registry value:

```
Value Name        FastSendDatagramThreshold
Data Type         REG_DWORD
Value             1500
```

4. Quit the Registry Editor.

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

Further information on this setting can be found on the Microsoft Support Web site.

Configuring Adequate Virtual Machine Resources, vCPU, and vRAM

Unlike traditional VDI desktops, desktops using high-end 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 heavy 3D workloads are 4 vCPUs and 8 GB of RAM. Some CAD/CAM applications can demand as much as 64 GB of RAM.

Optimizing PCoIP

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

With PCoIP, another option is to enable **Disable Build-To-Lossless**. This reduces the overall amount of PCoIP traffic, which, in turn, reduces the load placed on both the virtual machine and endpoint. For more information, see VMware Horizon 6 with View Performance and Best Practices and PCoIP Happiness for View Virtual Desktops in Horizon 6.

Enabling Relative Mouse

If you are using an application or game in which the cursor is moving uncontrollably, enabling the relative-mouse feature may improve mouse control.

Relative Mouse is a Windows Horizon Client 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 the absolute mouse. Two notable classes of applications, CAD applications and 3D games, rely on relative mouse events to function correctly.

VMware expects the requirements for relative mouse to increase rapidly as CAD and 3D games become more heavily used in View environments.

The Horizon Windows Client is required to enable relative mouse. As of now, this feature is not available through any other Horizon Clients or zero clients.

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 Horizon Client window and select **Relative Mouse**. A check mark appears next to Relative Mouse.

Note: Relative Mouse must be selected on each and every connection. As of now, there is no option to enable this by default or as a persistent feature.

Workaround for CAD Performance Issues

Occasionally, when working with CAD models (when turning and spinning objects), you may find that objects move irregularly and with a delay. 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, it defaults to 30.

Caution: This change can negatively affect other applications. Use it only if you are experiencing the symptoms mentioned above.

Resource Monitoring

This section outlines ways to monitor the GPU resources on each ESXi host.

gpuvm

To better manage the GPU resources available on an ESXi host, examine the current GPU resource allocation. The ESXi command-line query utility gpuvm 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.

```
# gpuvm
```

For example, the utility might display the following output:

```
# gpuvm
     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

The NVIDIA-specific **nvidia-smi** utility shows several details of GPU usage at the point in time in which you issued the command. This display is not dynamic and must be reissued to update the information.

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

```
# nvidia-smi
```

You can also issue the following command:

```
# watch –n 1 nvidia-smi
```

This command issues 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, shown in Figure 2, is at the right of the middle section. It shows the percentage of each GPU's processing cores in use at that time. This metric can be helpful in troubleshooting poor performance.

For more details on how to use the **nvidia-smi tool**, refer to the nvidia-smi main page.

Troubleshooting

This section provides troubleshooting steps to follow if you have issues with either vSGA or vDGA 3D rendering. These solutions can be used to solve a variety of issues, including the problem of Xorg failing to start.

Problem: Xorg Fails to Start

If your attempt to start Xorg fails, it is most likely due to the NVIDIA VIB module not loading properly.

Solution: Warm-Reboot the Host

Often, the improper loading of the NVIDIA VIB module can be resolved by warm-rebooting the host (in some instances it appears 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 troubleshooting options.

Problem: Other Issues with vSGA, vGPU, or vDGA

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

Solution: Verify That the GPU VIB Is Installed

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 should resemble the following example:

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

Solution: Verify That the GPU Driver Loads

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 should resemble the following example:

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

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

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

Search for FGLRX on AMD hardware or NVRM on NVIDIA hardware. Often, an issue with the GPU will be identified in the `vmkernel.log`.

Solution: Verify That Display Devices Are Present in the Host

To verify that display devices are present in the host, run the `esxcli` command as shown in the Confirm Graphics-Card Installation section of this paper.

Solution: Check the PCI Bus Slot Order

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 the two GPUs between PCIe slots or change the Primary GPU settings in the server BIOS.

Solution: 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
```

Solution: Check `sched.mem.min`

If you get a vSphere error about the `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 here, must match the amount of configured virtual machine memory. The example is for a virtual machine with 4 GB of RAM.

About the Authors and Contributors

This paper was updated for vGPU solutions by Stephane Asselin, who also wrote the original version, and Gary Sloane. Stephane is a VMware End-User-Computing Architect. Gary is a Consulting Editor and Writer for VMware. They wish to thank the following people for their contributions:

- Matt Coppinger, Director, Technical Marketing and Enablement, VMware End-User Computing
- Scott Guscar, Staff Systems Engineer, VMware End-User Computing
- Jason Ruby, Senior Systems Engineer, VMware End-User Computing
- Jeff Weiss, former Staff Engineer, Strategic Partners, VMware End-User Computing, currently GRID Solutions Architect Manager, NVIDIA
- Alex West, former Technical Writer, VMware End-User Computing

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

References

AMD Professional GPUs for Servers

Horizon 6 with PCoIP—Up to 30% Bandwidth Savings out of the Box

Installing and configuring the NVIDIA VIB on ESXi

NVIDIA CUDA FAQ

NVIDIA Driver Downloads

NVIDIA GRID Partners

VMware Horizon 6 3D Engineering Workloads Reference Architecture

VMware Horizon 6 with View Performance and Best Practices

vmware®

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