Graphics Acceleration in View Virtual Desktops

VMware Horizon 6 with View

WHITE PAPER
# Graphics Acceleration in View Virtual Desktops

## Table of Contents

- **Introduction** .................................................. 5
- **Why 3D Matters for View** .................................. 5
- **Understanding the Differences Between Soft 3D, vSGA, and vDGA** ................. 6
  - **Soft 3D – Software-Based 3D Rendering** .................... 7
  - **vSGA – Virtual Shared Graphics Acceleration** ............... 7
    - Configuring vSGA in Pool Settings for View .................... 7
    - Configuring vSGA in VMware vSphere Client ..................... 8
  - **vDGA – Virtual Dedicated Graphics Acceleration** .......... 8
    - vDGA Deployment .................................................. 8
    - vDGA Does Not Support Live VMware vSphere vMotion Capabilities .......... 9
- **Prerequisites** ................................................ 9
  - **Host Hardware Requirements** ................................ 9
  - Servers with Compatible Power and PCI Slot Capacity ................. 9
  - Physical Host Size ............................................... 9
  - PCIe x16 .................................................................... 9
  - Host PSU (Power Supply Unit) ....................................... 9
  - Virtual Technology for Directed I/O (VT-d) ........................ 9
  - Two-Display Adapters .............................................. 10
  - ESXi Special Access Command .................................... 10
  - Supported Graphics Cards ........................................ 10
  - Software Requirements ............................................ 11
  - End-User Clients .................................................... 11
- **Application Requirements** ..................................... 12
  - DirectX 9.0c ................................................................ 12
  - OpenGL 2.1 .................................................................. 12
- **Confirm Graphics-Card Installation.** ............................. 12
- **vSGA Installation** ................................................ 14
  - Installing NVIDIA Drivers on ESXi 5.1 .......................... 14
  - Installing NVIDIA Drivers on ESXi 5.5 .......................... 15
  - Installing AMD Drivers on ESXi 5.5 .............................. 16
  - vSGA Post-Installation Checks ................................... 17
    - Xorg ..................................................................... 17
    - gpulumi ................................................................. 17
    - NVIDIA-Specific Tool: nvidia-smi ............................... 18
### vDGA Installation
- Enable the Host for GPU Pass-Through
- Check That VT-d or AMD IOMMU Is Enabled
- Enable Device Pass-Through Using the vSphere Web Client
- Enable the Virtual Machine for GPU Pass-Through

### View Pool Configuration for vSGA
- Pool Prerequisites for View
- Video-Memory (VRAM) Sizing
- Screen Resolution
- Pool 3D-Rendering Options in View
  - Manage Using vSphere Client
  - Automatic
  - Software
  - Hardware
  - Disabled
- Best Practices for Configuring 3D Rendering for Desktop Pools
  - When to Select the Automatic Option
  - When to Select the Hardware Option
  - Manage Using vSphere Client
  - When to Select the Software Option
- Enable View Pools for vSGA Hardware 3D Rendering
  - Enabling an Existing View Pool
  - Enabling a New View Pool

### Performance Tuning Tips
- Configuring Adequate Virtual Machine Resources
- Optimizing PCoIP
- Enabling Relative Mouse
- Virtual Machines Using VMXNET3
- Workaround for CAD Performance Issues

### Resource Monitoring
- gpuvm
- nvidia-smi
Troubleshooting ................................................................. 30
  Problem: Xorg Fails to Start ........................................... 30
    Solution: Warm-Reboot the Host .................................. 30
  Problem: Other Issues with vSGA or vDGA ....................... 30
    Solution: Verify That the GPU VIB Is Installed ................. 30
    Solution: Verify That the GPU Driver Loads .................... 30
    Solution: Verify That Display Devices Are Present in the Host 31
    Solution: Check the PCI Bus Slot Order ......................... 31
    Solution: Check Xorg Logs ........................................... 31
    Solution: Check sched.mem.min ..................................... 31
About the Authors and Contributors .................................... 32
Introduction

VMware® Horizon™ with View™ (formerly VMware Horizon View) enables you to access a virtual desktop from anywhere, anytime. Offering the highest-fidelity experience to users across the globe, View delivers applications, unified communications, 3D graphics, and more, with speed and efficiency.

This document defines the various types of 3D-graphics acceleration available in View virtual desktops, describes how to implement and troubleshoot each type, and provides knowledge around the benefits of each technology.

Why 3D Matters for View

The recently introduced 3D-graphics capabilities in View further expand the target user base and potential use cases IT organizations can deliver with virtual desktops. In addition, 3D augments the virtual desktop user interface by enabling a more graphically rich user experience.

Figure 1: Virtual Desktop User Segmentation
Understanding the Differences Between Soft 3D, vSGA, and vDGA

There are three types of 3D-graphics acceleration available in View virtual desktops: Soft 3D, vDGA, and vSGA. This section details the differences between the three.

<table>
<thead>
<tr>
<th>NAME</th>
<th>DEFINITION</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>Soft 3D</td>
<td>Software 3D Renderer</td>
<td>Support for software-accelerated 3D graphics is provided via a Soft 3D graphics driver without any physical GPUs installed in the ESXi virtual-machine host.</td>
</tr>
<tr>
<td>vSGA</td>
<td>Virtual Shared Graphics Acceleration</td>
<td>Multiple virtual machines leverage physical GPUs installed locally in the ESXi virtual-machine host(s) to provide hardware-accelerated 3D graphics to multiple virtual desktops.</td>
</tr>
<tr>
<td>vDGA</td>
<td>Virtual Dedicated Graphics Acceleration</td>
<td>Only one virtual machine is mapped to a single physical GPU installed in the ESXi virtual-machine host to provide high-end, hardware-accelerated workstation graphics where a discrete GPU is needed.</td>
</tr>
</tbody>
</table>

Table 1: Comparison of Graphics-Acceleration Methods

Each type of 3D-graphics acceleration requires a graphics driver on the virtual desktop. The Software 3D Renderer (Soft 3D) uses a Soft 3D graphics driver, which is automatically installed on Windows 7 virtual desktops with VMware Tools™. The Soft 3D graphics driver provides support for DirectX 9.0c and OpenGL 2.1 and supports both software 2D and 3D rendering in Windows 7 virtual desktops.

Virtual Shared Graphics Acceleration (vSGA) uses the same Soft 3D graphics driver installed with VMware Tools.

Virtual Dedicated Graphics Acceleration (vDGA) configurations do not use the Soft 3D graphics driver; instead, they use the native graphics-card driver provided by the graphics-card vendor, which must be installed in the guest OS.

**Note:** View virtual desktops utilizing vDGA support only NVIDIA graphics cards at this time.

As a benefit, the Soft 3D graphics driver allows vSGA-enabled virtual machines to dynamically switch between software and hardware 3D rendering, without your having to reconfigure. This capability leads to higher availability for graphics acceleration, as it allows administrators to set a virtual machine to **Automatic**, which enables the virtual machine to automatically switch to hardware graphics acceleration for high-end graphics applications. Additionally, this driver allows the use of VMware high-availability technologies such as VMware vSphere® vMotion®. Finally, having a single driver for 2D, 3D, and vSGA graphics acceleration greatly simplifies image management and deployment.

**Note:** If a virtual desktop is dynamically moving from hardware 3D rendering to software 3D rendering, the user may notice a performance drop in applications running in the virtual machine. However, if the virtual desktop is moving in the reverse direction (software to hardware), the user should notice an improvement in performance. This is because hardware 3D rendering is a more effective graphics-rendering solution than software 3D rendering.
Soft 3D – Software-Based 3D Rendering

Introduced with View 5.0, Soft 3D is differentiated from vSGA and vDGA in that it does not require any physical GPUs to be installed in the ESXi host. This solution requires only the Soft 3D graphics driver automatically installed on View virtual desktops with VMware Tools.

vSGA – Virtual Shared Graphics Acceleration

Introduced with View 5.2, Virtual Shared Graphics Acceleration (vSGA) provides hardware-accelerated 3D graphics by allowing multiple virtual machines to share physical GPUs installed locally in the ESXi hosts.

The maximum amount of video memory that can be assigned per View virtual desktop is 512MB. However, 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.)

You can use this rule to calculate basic consolidation ratios. For example, if a graphics card has 16GB of GPU RAM, and 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 64 virtual machines can run on that specific GPU at any given time:

- 16GB GPU RAM on the graphics card (approximately equal to 16,384MB)
- 256MB video memory reserved on the GPU per virtual machine

16,384 / 256MB = 64 virtual machines can run on that GPU

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 additional virtual machines are set to Automatic, they will be powered on using software 3D rendering.

Virtual Shared Graphics Acceleration (vSGA) can be configured either via the VMware vSphere Client™ or vSphere Web Client, or in Pool Settings for View.

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

On the other hand, vSGA is limited by the amount of memory on the installed boards. VMware ESXi assigns a virtual machine to a particular graphics device during power-on. The assignment is based on graphics-memory reservation that occurs in a round-robin fashion. The current policy is to reserve one-half of the virtual machine’s VRAM size, with a minimum of 128MB. This means a graphics device with 4GB memory can accept at most 32 virtual machines with minimum reservation (4GB/128MB is approximately equal to 32). After a graphics device reaches its reservation maximum, no more virtual machines can be assigned to it until another virtual machine leaves the GPU. This can occur when a virtual machine is powered off, suspended, or moved to another host by vSphere vMotion.

Configuring vSGA in Pool Settings for View

When configuring vSGA in Pool Settings for View, there are five 3D-rendering options:

• Manage using vSphere Client does not make any changes to the 3D settings of the virtual machines in that pool. Instead, this option allows virtual machines to be configured individually through the vSphere Client or vSphere Web Client. This setting is most useful during testing or for manual desktop pools.

• Automatic (the default) uses hardware acceleration if there is a capable and available hardware GPU in the host in which the virtual machine is running. However, if a hardware GPU is not available, the virtual machine uses software 3D rendering for any 3D tasks. This allows the virtual machine to be started on, or migrated to (via vSphere vMotion), any host (vSphere 5.0 or later) and use the best solution available on that host.

• Software uses only software 3D rendering, even if there is an available hardware GPU in the host in which the virtual machine is running. This setting does not provide the performance benefits of hardware 3D acceleration. However, it both allows the virtual machine to run on any host (vSphere 5.0 or later) and allows you to block virtual machines from using a hardware GPU in a host.
• **Hardware** uses only hardware GPUs. If a hardware GPU is not present in a host, the virtual machine will either not start, or you will not be able to live-migrate it via vSphere vMotion to any host without a hardware GPU. As long as the host that the virtual machine is being migrated to has a capable and available hardware GPU, vSphere vMotion is possible with the **Hardware** specification. 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). Use this setting to ensure that View desktop pools with non-graphical workloads do not use unnecessary resources. For example, the **Disabled** setting prevents a desktop pool from accessing a hardware GPU that is available in a cluster shared with View desktops that have heavy graphical workloads.

**Configuring vSGA in VMware vSphere Client**

When configuring vSGA via the vSphere Client or vSphere Web Client, there are four 3D-rendering options.

**Note:** To configure vSGA via the vSphere Client or vSphere Web Client, **Manage using vSphere Client** must be enabled in Pool Settings for View.

• **Automatic** (the default)
• **Software**
• **Hardware**
• **Disabled**

**vDGA – Virtual Dedicated Graphics Acceleration**

Introduced with View 5.3, Virtual Dedicated Graphics Acceleration (vDGA) is a graphics-acceleration capability provided by VMware ESXi that delivers high-end workstation graphics for use cases where a discrete GPU is needed. This graphics-acceleration method dedicates a single GPU to a single virtual machine for high performance.

**Note:** Some graphics cards can have multiple GPUs on them.

If you are using vDGA, graphics adapters installed in the underlying host are assigned to virtual machines using VMware vSphere DirectPath I/O™. Assigning a discrete GPU to a virtual machine dedicates the entire GPU to it. Server hardware is available with up to four PCIe x16 graphics-card slots. Some blade-enclosure hardware vendors also offer a sidecar-type expansion unit that can support up to eight graphics cards.

The number of 3D-hardware-accelerated virtual machines per host is limited to the number of graphics card (PCIe x16) slots in the server (and the available space per slot), which, in turn, limits the number of GPUs and amount of video RAM that can be installed in a server. The number of 3D-enabled virtual machines is limited by the number of GPUs in the server.

For instance, a Dell R720 server can accommodate two dual-wide cards or three single-wide cards. Example: The NVIDIA GRID K2 card has two GPUs in a dual-wide form factor. If an administrator installs a GRID K2 card into each available slot supporting dual-wide cards on the Dell R720, there is a total of 4 GPUs in the system. This is the total number of vDGA-enabled virtual machines that the server can support.

**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, which you install in the guest OS, rather than the virtual machine’s Soft 3D graphics driver, which is included in VMware Tools. To provide framebuffer access, vDGA uses an interface between the remoting protocol and the graphics driver.
vDGA Does Not Support Live VMware vSphere vMotion Capabilities
Live VMware vSphere vMotion is not supported with vDGA. Bypassing the virtualization layer, vDGA uses VMware vSphere DirectPath I/O to allow direct access to the GPU card. By enabling direct pass-through from the virtual machine to the PCI device installed on the host, the virtual machine is effectively locked to that specific host.

If an administrator 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 pass-through to the specific PCI device on that host. Only then should the user power on the virtual machine.

Prerequisites
This section lists the hardware and software requirements of vSGA and vDGA.

Host Hardware Requirements
Host hardware requirements for both vSGA and vDGA are documented below.

Servers with Compatible Power and PCI Slot Capacity
For a list of supported hardware proven to work with vSGA, visit the VMware Compatibility Guide.

Virtual Dedicated Graphics Acceleration (vDGA) is certified on a per-ESXi-host basis. To see a list of certified systems, visit the VMware Server Compatibility Guide and choose Virtual Dedicated Graphics Acceleration from the Features menu in the middle of the window.

Physical Host Size
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.

PCIe x16
PCIe x16 is required for all supported highest-end GPU cards.

Host PSU (Power Supply Unit)
Check the power requirements of the GPU to make sure the PSU both is 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 older Quadro series cards. The trade-off is that GRID K1 and K2 cards are passively cooled, relying on internal chassis fans to cool them. It is important to note that at this time NVIDIA sells only GRID K1 and K2 cards as part of a certified VMware-compatible system. AMD FirePro S series cards may also support lower power than standard AMD FirePro workstation cards and are validated by AMD to work in supported systems with passive cooling. Moreover, even though a system may be certified by NVIDIA or AMD, it must still be validated by VMware and listed on the hardware-compatibility list.

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

To check if VT-d or AMD IOMMU is enabled on the host, check the server BIOS. If there are any questions about finding 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 your primary display adapter. This is because the 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 will not be able to use the GPU for rendering.

If you have two GPUs installed, the server BIOS may 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 have only a single GPU, there is a command that will enable that GPU to be used with vSGA. However, VMware does not recommend this configuration for production use because 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:

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

You will see something similar to this:

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

The PCI ID of the graphics device is in bold.

Then reset the ownership flags referencing the PCI ID you found:

```bash
# 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 graphics cards proven to work with vSGA, 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.

<table>
<thead>
<tr>
<th></th>
<th>vSGA</th>
<th>vDGA</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>VMware vSphere Hypervisor</strong></td>
<td>ESXi 5.1 U1 or ESXi 5.5 (5.5 is recommended)</td>
<td>ESXi 5.1 U1 or ESXi 5.5 (5.5 is recommended)</td>
</tr>
<tr>
<td><strong>VMware Horizon with View</strong></td>
<td>View 5.2 or later (5.3 or later is recommended)</td>
<td>View 5.3 or later</td>
</tr>
<tr>
<td><strong>Display Protocol</strong></td>
<td>PCoIP display protocol with a maximum of two display monitors</td>
<td>PCoIP display protocol with a maximum of two display monitors</td>
</tr>
<tr>
<td><strong>NVIDIA Graphics Driver</strong></td>
<td>VMware vSphere NVIDIA driver version 304.76</td>
<td>VMware vSphere NVIDIA Quadro/NVS/Tesla/GRID desktop driver version 320.86</td>
</tr>
<tr>
<td></td>
<td><em>Note:</em> The driver is supplied and supported by NVIDIA; download from NVIDIA Driver Download</td>
<td><em>Note:</em> The driver is supplied and supported by NVIDIA; download from NVIDIA Driver Download</td>
</tr>
<tr>
<td><strong>AMD Graphics Drivers</strong></td>
<td>ESXi 5.5 drivers for AMD FirePro</td>
<td>NA</td>
</tr>
<tr>
<td></td>
<td><em>Note:</em> Download the drivers for vSGA on AMD FirePro products from AMD Driver Download</td>
<td></td>
</tr>
<tr>
<td><strong>Guest Operating System</strong></td>
<td>Windows 7 32- or 64-bit</td>
<td>Windows 7 64-bit</td>
</tr>
</tbody>
</table>

Table 2: Software Requirements for vSGA and vDGA

End-User Clients

With both graphics and compute processing handled by the 3D-enabled ESXi hosts, IT might overlook end-user clients by assuming all major processing is handled inside the data center.

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

In some 3D use cases, as much as 70Mbps of PCoIP traffic per virtual desktop has been observed during peak loads. This high bandwidth requirement 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 and ensure that the display on the screen is accurate and current. This large flow of PCoIP traffic sent to end-user clients can often lead to significant performance problems.

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 using Tera1 chips can support a maximum rate of 30fps, whereas the current Tera2 chip can achieve up to 60fps. Achieving high frame rates can be important to the usability of certain engineering applications as well.
Application Requirements

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 when using 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 when using vSGA.

**Note:** Virtual Dedicated Graphics Acceleration (vDGA) supports the versions of DirectX and OpenGL that the GPU manufacturer’s graphics driver supports. This is generally the latest version of these technologies.

Confirm Graphics-Card Installation

If your system did not ship with a graphics card installed, consult the AMD or NVIDIA user guides for your adapter to ensure proper installation.

To ensure that 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 AMD or NVIDIA driver (VIB) for vSGA on an ESXi host. Installation instructions differ for ESXi 5.1 and ESXi 5.5 for the NVIDIA driver. AMD supports only ESXi 5.5.

Installing NVIDIA Drivers on ESXi 5.1

1. The NVIDIA vSphere VIBs for vSGA can be downloaded from the NVIDIA Driver Downloads page.
   
   Note: The downloaded .zip file actually contains a readme file and another .zip file. Extract this second .zip file and use it in the following step.

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 such as FastSCP or WinSCP

3. Run the following esxcli command through an ESXi SSH session to install the VIB onto the host:
   
   Note: The system should be in maintenance mode. If it is not, you need to use the maintenance mode switch. For example:

   ```
   # 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: You can either put the host into maintenance mode or add the following command option to the esxcli command:

   ```
   --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
   ```

   You may receive the following error:

   Could not find a trusted signer

   This error indicates that the VIB is not signed. Use the following option in the esxcli command to remove the signature check:

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

   Note: If the driver was retrieved from a trusted source, such as NVIDIA, removing the signature check poses only minimal security risks. However, always be sure to check the MD checksum against the downloaded file.

   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 rebooting the ESXi host to verify that the Xorg service will start correctly when restarting the
host in the future.

If you do not reboot the host, you will later have to manually start the Xorg service. You can do this by
issuing the following command:

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

If you need to remove the installed VIB from a host, run the following command:

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

### Installing NVIDIA Drivers on ESXi 5.5

This section outlines the steps required to install the NVIDIA driver (VIB) for vSGA on an ESXi 5.5 host.

1. The NVIDIA vSphere VIBs for vSGA can be downloaded from the NVIDIA Driver Downloads page.

   **Note:** The downloaded .zip file actually contains a readme file and another .zip file. Extract this
   second .zip file and use it in the following step.

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 such as FastSCP or WinSCP

3. Run the following command through an ESXi SSH session to install the VIB onto the host:

   ```
   # vim-cmd hostsvc/maintenance_mode_enter
   # localcli software vib install --no-sig-check -v /<path-to-vib>/NVIDIA-VMware-319.65-1OEM.550.0.0.1331820.x86_64.vib
   # vim-cmd hostsvc/maintenance_mode_exit
   ```

   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 rebooting the ESXi host to verify that the Xorg service will start correctly when restarting the host in the future.

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:

```
# kill -HUP $(cat /var/run/vmware/vmkdevmgr.pid)
```

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

```
# /etc/init.d/xorg stop                        // stop Xserver
# vmkload_mod -u nvidia                       // unload running module
# localcli software vib remove -n <vib name> // vib name from “esxcli software vib list” — usually NVIDIA-VMware or gpu-nvidia
```

### Installing AMD Drivers on ESXi 5.5

This section outlines the steps required to install the AMD driver (VIB) for vSGA on an ESXi 5.5 host.

**Note:** AMD does not support ESXi 5.1.

1. The AMD vSphere ESXi 5.5 VIBs for vSGA can be downloaded from the AMD driver-downloads page.

   **Note:** The downloaded .zip file actually contains a VIB and a .doc directory with a readme file and another .zip file for an upload bundle. Extract this second .zip file and use it in the following step.

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 such as FastSCP or WinSCP

3. Run the following command through an ESXi SSH session to install the VIB onto the host:

   ```
   # vim-cmd hostsvc/maintenance_mode_enter
   # localcli software vib install --no-sig-check -v /<path-to-vib>/fglr-x12.10.3.985289-1OEM.550.0.0.1331820.x86_64.vib
   # vim-cmd hostsvc/maintenance_mode_exit
   ```

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 rebooting the ESXi host to verify that the Xorg service will start correctly when restarting the host in the future.

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:

```bash
# kill -HUP $(cat /var/run/vmware/vmkdevmgr.pid)
```

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

```bash
# /etc/init.d/xorg stop // stop Xserver
# vmkload_mod -u fglrx // unload running module
# localcli software vib remove -n <vib name> // vib name from “esxcli software vib list” - usually fglrx for AMD VIB
```

**vSGA Post-Installation Checks**

This section contains various commands that can be used to ensure correct installation of the GPU card and its respective drivers. VMware recommends that you learn these commands, which are useful in troubleshooting 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:

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

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

```bash
# /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:

```bash
# 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:

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

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

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

If Xorg fails to start, go to the Troubleshooting section.
NVIDIA-Specific Tool: nvidia-smi
This is an NVIDIA-specific tool to see how much of each GPU is in use when using the NVIDIA driver, issue the following command in an SSH session:

```bash
# 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:

```bash
# 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 is at the right of the lower section. It shows the percentage of each GPU's processing cores in use at that point in time. This metric can be helpful in troubleshooting poor performance.

---

The nvidia-smi Display

```
---
| 0 GRID K2 | Off | 0000:43:00.0 | Off | 0 |
| N/A 34C | P8 | 17W / 117W | 19MB / 3583MB | 0% Default |
| 1 GRID K2 | Off | 0000:44:00.0 | Off | 0 |
| N/A 32C | P8 | 17W / 117W | 19MB / 3583MB | 0% Default |
---
```

Figure 2: The nvidia-smi Display
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
```

vDGA Installation

This section describes enabling GPU pass-through at the host level, and preparing virtual machines for vDGA 3D rendering.

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 section.

Check 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.
2. Select the Manage tab for this host.
3. If the Hardware group is not expanded, click the down arrow next to it.
4. Click PCI Devices.
5. Right-click one of the GPUs installed in the system and select Edit.
6. 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.
7. 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.
Enable the Virtual Machine for GPU Pass-Through

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

1. Update the virtual machine to hardware version 9 or hardware version 10.
   You must upgrade any virtual machine that will use 3D graphics to hardware version 9 or 10 (for example, HWv9 shows as \vmx-09\) to ensure maximum compatibility with GPU pass-through.
   To do this, right-click the virtual machine in the vCenter client and choose Upgrade Virtual Hardware.
   **Note:** Virtual machines with a hardware version of 9 or later can have their settings managed only via the vSphere Web Client.
   **Note:** With ESXi 5.5, the virtual machine will be upgraded to version 10 (\vmx-10\).

2. Adjust \pciHole.start\.
   **Note:** This is required only if the virtual machine has more than 2GB of configured memory.
   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”

3. Add the PCI device.
   To enable vDGA 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.
   f. If prompted, click Reserve All Memory.
   g. Click OK.

4. Install the NVIDIA driver.
   a. Download the latest NVIDIA Windows 7 desktop driver and install it on the virtual machine. (NVIDIA drivers can be downloaded from the NVIDIA Driver Download page.)
   b. After the driver is installed, reboot the virtual machine.

5. Install the View Agent.
   a. After the NVIDIA driver is installed correctly, install the View Agent on the virtual machine.
   b. Reboot when requested.

6. Enable proprietary NVIDIA capture APIs.
   a. 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
   b. After the process is complete, restart the virtual machine.
   **Note:** Currently only proprietary capture APIs are available for NVIDIA drivers.
7. Connect to the virtual machine for the first time.

In order to activate the NVIDIA display adapter, you must connect to the virtual machine for the first time via PCoIP in full screen from the endpoint (at native resolution), or 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 ensure that the GPU is active by viewing the display information in `DXDiag.exe`:

a. Click the Start menu.

b. Type `dxdiag` and click Enter after DxDiag shows up in the list, or click on it in the list.

c. 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, ensure 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 Pool Configuration for vSGA**

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

**Pool Prerequisites for View**

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

<table>
<thead>
<tr>
<th></th>
<th>SOFT 3D (SOFTWARE 3D)</th>
<th>vSGA (HARDWARE 3D)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Minimum</td>
<td>1.18MB</td>
<td>64MB</td>
</tr>
<tr>
<td>Default</td>
<td>64MB</td>
<td>96MB</td>
</tr>
<tr>
<td>Maximum</td>
<td>512MB*</td>
<td>512MB</td>
</tr>
</tbody>
</table>

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

Table 3: Video-Memory (VRAM) Sizing

Note: Whenever you change the 3D Renderer setting, it reverts the amount of video memory to the 96MB 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. Select the Manage using vSphere Client option to prevent this. This is the preferred option, as it will avoid complexity and confusion in managing desktops using this functionality.

Note: If you are using the Manage using vSphere Client option, VMware recommends that you use the vSphere Web Client to configure the virtual machines, rather than the traditional vSphere Client. This is because the traditional vSphere Client does not display the various rendering options; it will display only 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 the View desktop pool, there may be a short delay (sometimes a couple of minutes) before the message Reconfiguring virtual machine settings 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 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 to be 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.
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, open the View Administrator dashboard, click your desired pool, and select **Edit**. Then open **Pool Settings** on the second tab, and click the drop-down menu for **3D Renderer** to view your 3D-rendering options.

![Figure 3: 3D-Rendering Options in View Pool Settings](image_url)

**Manage Using vSphere Client**

The **3D Renderer** option that is set in either the traditional vSphere Client or the vSphere Web Client for a virtual machine determines the type of 3D-graphics rendering that takes place. View does not control 3D rendering (the vSphere Client always sets this as **Automatic**).

**Note:** VMware currently encourages administrators to use the vSphere Web Client instead of the traditional vSphere Client. This is because the traditional vSphere Client does not display the various rendering options; it displays only **Enable/Disable 3D support**.

In the vSphere Web Client, you can select from the **Automatic**, **Software**, or **Hardware** options. 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**, **Max number of monitors**, and **Max resolution of any one monitor** settings become inactive in View Administrator. You can configure these settings for a virtual machine in the vSphere Web Client.
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 64MB, 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 512MB per virtual machine with hardware version 9, and a maximum of 128MB 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 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 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, 3D rendering of any kind is inactive.
Best Practices for Configuring 3D Rendering for Desktop Pools

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 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 (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 ensure 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.

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. Alternatively, in vSphere Web Client, you can configure individual virtual machines to have different options and VRAM values.

**Note:** The vSphere Client and the vSphere Web Client are different options, but they can be used to achieve the same goal. Again, VMware recommends vSphere Web Client over the traditional software vSphere Client, as the traditional vSphere Client does not display the various rendering options; it displays only **Enable/Disable 3D support**.

When to Select the Software Option

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

- You have ESXi 5.0 hosts only
- Some of your ESXi 5.1 hosts do not have GPU graphics cards
- Your 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 ensure those resources are available for virtual machines that do require it.
Enable View Pools for vSGA Hardware 3D Rendering

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

Enabling 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**. Configure the amount of VRAM you want each virtual desktop to have.

**Note:** 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.

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**. Configure the amount of VRAM you want each virtual desktop to have.

**Note:** If the **3D Renderer** section is grayed out, ensure that you have **PCoIP** selected as your **Default Display Protocol**, and **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 tips to improve the performance of both vSGA and vDGA.

Configuring Adequate Virtual Machine Resources

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 3D workloads are two vCPUs and 4GB of RAM.

Optimizing PCoIP

 Occasionally, PCoIP custom configurations can contribute to poor performance. By default, PCoIP is set to allow a maximum of 30fps. 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 120fps.

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.

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 new feature in the Windows Horizon Client 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.

With the introduction of vSGA and vDGA, 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.

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 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.
Virtual Machines Using VMXNET3

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

1. Start the Registry Editor (Regedit32.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 changing this registry setting. 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.

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.

Important: This change can negatively affect other applications. Use with caution and only if you are experiencing the symptoms mentioned above.
Resource Monitoring

The following 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.

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

```bash
# gpuvm
```

For example, the utility might display the following output:

```bash
# 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.

```bash
# nvidia-smi
```

You can also issue the following command:

```bash
# 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 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 metric can be helpful in troubleshooting poor performance. (See Figure 2.)

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 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 or vDGA

Any issue with vSGA or vDGA, or with Xorg failing to start, can be solved by one or several of the following solutions. These solutions are not listed in any particular order.

Solution: Verify That the GPU VIB Is Installed

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

AMD-based GPUs:

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

NVIDIA-based GPUs:

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

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

```
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:

AMD-based GPUs:

```
# esxcli system module load –m fglrx
```

NVIDIA-based GPUs:

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

If the driver loads correctly, you should see output like the example below:

```
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 figure in quotes, `4096` here, must match the amount of configured virtual machine memory. The example is for a virtual machine with 4GB of RAM.
About the Authors and Contributors

The following individuals updated this document:

• Stephane Asselin, EUC Architect, VMware
• Alexander West, Technical Writer, End-User Computing, VMware

Mike Coleman, formerly of VMware, updated a previous version of this document.

Simon Long, Cloud Architect, VMware, is the author of the original version of this document.

Contributors to this document include:

• Aaron Blasius, Product Line Manager, ESXi Platform, VMware
• Chris ‘Kif’ Davis, Security Product Manager, VMware
• Rasmus Jensen, EUC Architect, VMware
• Giridhar Kommisetty, formerly of VMware
• Pat Lee, Senior Director, Remote Experience, VMware
• Joel Lindberg, EUC Specialist, VMware
• Warren Ponder, Director, Product Management, VMware
• Josh Spencer, Senior Systems Engineer, End-User-Computing Specialist, VMware
• Tommy Walker, Chief Field Technologist, End-User Computing, VMware
• Vincent Wu, Senior Systems Engineer, VMware

To comment on this paper, contact us at twitter.com/vmwarehorizon.