# TABLE OF CONTENTS

**Chapter 1. Release Notes**
- 1.1. Updates in Release 9.0
- 1.2. Updates in Release 9.1

**Chapter 2. Validated Platforms**
- 2.1. Supported NVIDIA GPUs and Validated Server Platforms
- 2.2. Hypervisor Software Releases
- 2.3. Guest OS Support
  - 2.3.1. Windows Guest OS Support
  - 2.3.2. Linux Guest OS Support
- 2.4. NVIDIA CUDA Toolkit Version Support
- 2.6. Multiple vGPU Support
- 2.7. Peer-to-Peer CUDA Transfers over NVLink Support

**Chapter 3. Known Product Limitations**
- 3.1. Issues may occur with graphics-intensive OpenCL applications on vGPU types with limited frame buffer
- 3.2. In pass through mode, all GPUs connected to each other through NVLink must be assigned to the same VM
- 3.3. vGPU profiles with 512 Mbytes or less of frame buffer support only 1 virtual display head on Windows 10
- 3.4. NVENC requires at least 1 Gbyte of frame buffer
- 3.5. VM failures or crashes on servers with 1 TB or more of system memory
- 3.6. VM running older NVIDIA vGPU drivers fails to initialize vGPU when booted
- 3.7. Single vGPU benchmark scores are lower than pass-through GPU
- 3.8. VMs configured with large memory fail to initialize vGPU when booted

**Chapter 4. Resolved Issues**

**Chapter 5. Known Issues**
- 5.1. Migrating a VM configured with NVIDIA vGPU software release 9.1 to a host running release 9.0 fails
- 5.2. ECC memory with NVIDIA vGPU is not supported on Tesla M60 and Tesla M6
- 5.3. Virtual GPU fails to start if ECC is enabled
- 5.4. VMware vCenter shows GPUs with no available GPU memory
- 5.5. RAPIDS cuDF merge fails on NVIDIA vGPU
- 5.6. 9.0 only: Users' view sessions may become corrupted after migration
- 5.7. Users' sessions may freeze during vMotion migration of VMs configured with vGPU
- 5.8. Migration of VMs configured with vGPU stops before the migration is complete
- 5.9. 9.0 only: nvidia-smi shows the incorrect ECC state for a vGPU
- 5.10. 9.0 only: Incorrect ECC error counts are reported for vGPUs on some GPUs
- 5.11. ECC memory settings for a vGPU cannot be changed by using NVIDIA X Server Settings
- 5.12. Changes to ECC memory settings for a Linux vGPU VM by nvidia-smi might be ignored
- 5.13. 9.0 only: VM crashes after the volatile ECC error count is reset
5.14. 9.0 only: No vComputeServer option available in NVIDIA X Server Settings
5.15. 9.0 only: On Linux VMs, the license directory is not deleted when the guest driver is
uninstalled
5.16. Black screens observed when a VMware Horizon session is connected to four displays
5.17. Quadro RTX 8000 and Quadro RTX 6000 GPUs can’t be used with VMware vSphere ESXi 6.5.35
5.18. Vulkan applications crash in Windows 7 guest VMs configured with NVIDIA vGPU
5.19. Host core CPU utilization is higher than expected for moderate workloads
5.20. H.264 encoder falls back to software encoding on 1Q vGPUs with a 4K display
5.21. H.264 encoder falls back to software encoding on 2Q vGPUs with 3 or more 4K displays
5.22. Frame capture while the interactive logon message is displayed returns blank screen
5.23. RDS sessions do not use the GPU with some Microsoft Windows Server releases
5.24. VMware vMotion fails gracefully under heavy load
5.25. View session freezes intermittently after a Linux VM acquires a license
5.26. Even when the scheduling policy is equal share, unequal GPU utilization is reported
5.27. When the scheduling policy is fixed share, GPU utilization is reported as higher than
expected
5.28. nvidia-smi reports that vGPU migration is supported on all hypervisors
5.29. GPU resources not available error during VMware instant clone provisioning
5.30. VMs with 32 GB or more of RAM fail to boot with GPUs requiring 64 GB of MMIO space
5.31. Module load failed during VIB downgrade from R390 to R384
5.32. Resolution is not updated after a VM acquires a license and is restarted
5.33. Tesla P40 cannot be used in pass-through mode
5.34. On Linux, 3D applications run slowly when windows are dragged
5.35. A segmentation fault in DBus code causes nvidia-gridd to exit on Red Hat Enterprise
Linux and CentOS
5.36. No Manage License option available in NVIDIA X Server Settings by default
5.37. Licenses remain checked out when VMs are forcibly powered off
5.38. Memory exhaustion can occur with vGPU profiles that have 512 Mbytes or less of frame
buffer
5.39. vGPU VM fails to boot in ESXi 6.5 if the graphics type is Shared
5.40. ESXi 6.5 web client shows high memory usage even when VMs are idle
5.41. VMs configured with NVIDIA vGPU must not be on a host in a VMware DRS cluster
5.42. GNOME Display Manager (GDM) fails to start on Red Hat Enterprise Linux 7.2 and CentOS
7.0
5.43. NVIDIA Control Panel fails to start and reports that “you are not currently using a display
that is attached to an Nvidia GPU”
5.44. VM configured with more than one vGPU fails to initialize vGPU when booted
5.45. A VM configured with both a vGPU and a passthrough GPU fails to start the passthrough
GPU
5.46. vGPU allocation policy fails when multiple VMs are started simultaneously
5.47. Before Horizon agent is installed inside a VM, the Start menu’s sleep option is available
5.48. vGPU-enabled VMs fail to start, nvidia-smi fails when VMs are configured with too high
a proportion of the server’s memory
5.49. On reset or restart VMs fail to start with the error VMIOP: no graphics device is available for vGPU.................................................................................................................................................. 57
5.50. nvidia-smi shows high GPU utilization for vGPU VMs with active Horizon sessions......... 57
Chapter 1.
RELEASE NOTES

These Release Notes summarize current status, information on validated platforms, and known issues with NVIDIA vGPU software and associated hardware on VMware vSphere.

The most current version of the documentation for this release of NVIDIA vGPU software can be found online at NVIDIA Virtual GPU Software Documentation.

The releases in this release family of NVIDIA vGPU software include the software listed in the following table:

<table>
<thead>
<tr>
<th>Software</th>
<th>9.0</th>
<th>9.1</th>
</tr>
</thead>
<tbody>
<tr>
<td>NVIDIA Virtual GPU Manager for the VMware vSphere releases listed in Hypervisor Software Releases</td>
<td>430.27</td>
<td>430.46</td>
</tr>
<tr>
<td>NVIDIA Windows driver</td>
<td>431.02</td>
<td>431.79</td>
</tr>
<tr>
<td>NVIDIA Linux driver</td>
<td>430.30</td>
<td>430.46</td>
</tr>
</tbody>
</table>

Caution

If you install the wrong NVIDIA vGPU software packages for the version of VMware vSphere you are using, NVIDIA Virtual GPU Manager will fail to load.

The releases of the vGPU Manager and guest VM drivers that you install must be compatible. Different versions of the vGPU Manager and guest VM driver from within the same main release branch can be used together. For example, you can use the vGPU Manager from release 9.1 with guest VM drivers from release 9.0. However, versions of the vGPU Manager and guest VM driver from different main release branches cannot be used together. For example, you cannot use the vGPU Manager from release 9.1 with guest VM drivers from release 7.2.

See VM running older NVIDIA vGPU drivers fails to initialize vGPU when booted.
This requirement does not apply to the NVIDIA vGPU software license sever. All releases of NVIDIA vGPU software are compatible with all releases of the license server.

1.1. Updates in Release 9.0

New Features in Release 9.0

- NVIDIA vComputeServer vGPUs for artificial intelligence, deep learning, and high-performance computing workloads
- Support for multiple vGPUs in a single VM (requires release 6.7 Update 3)
- Error correcting code (ECC) memory support
- Page retirement support
- Configurable times slices for equal share schedulers and fixed share schedulers
- New configuration parameter to specify host ID of a licensed client
- Miscellaneous bug fixes

Hardware and Software Support Introduced in Release 9.0

- Support for Windows 10 May 2019 Update (1903) as a guest OS
- Support for the following OS releases as a guest OS:
  - Red Hat Enterprise Linux 8.0
  - CentOS 8.0
- Support for VMware Horizon 7.9

1.2. Updates in Release 9.1

New Features in Release 9.1

- Support for NVIDIA vComputeServer vGPUs on the following GPUs:
  - Quadro RTX 6000
  - Quadro RTX 8000
- Security updates
- Miscellaneous bug fixes
Chapter 2.
VALIDATED PLATFORMS

This release family of NVIDIA vGPU software provides support for several NVIDIA GPUs on validated server hardware platforms, VMware vSphere hypervisor software versions, and guest operating systems. It also supports the version of NVIDIA CUDA Toolkit that is compatible with R430 drivers.

2.1. Supported NVIDIA GPUs and Validated Server Platforms

This release of NVIDIA vGPU software provides support for the following NVIDIA GPUs on VMware vSphere, running on validated server hardware platforms:

- GPUs based on the NVIDIA Maxwell™ graphic architecture:
  - Tesla M6 (vComputeServer is not supported.)
  - Tesla M10 (vComputeServer is not supported.)
  - Tesla M60 (vComputeServer is not supported.)

- GPUs based on the NVIDIA Pascal™ architecture:
  - Tesla P4
  - Tesla P6
  - Tesla P40
  - Tesla P100 PCIe 16 GB (vSGA, vMotion with vGPU, and suspend-resume with vGPU are not supported.)
  - Tesla P100 SXM2 16 GB (vSGA, vMotion with vGPU, and suspend-resume with vGPU are not supported.)
  - Tesla P100 PCIe 12GB (vSGA, vMotion with vGPU, and suspend-resume with vGPU are not supported.)

- GPUs based on the NVIDIA Volta architecture:
  - Tesla V100 SXM2 (vSGA is not supported.)
  - Tesla V100 SXM2 32GB (vSGA is not supported.)
Validated Platforms

- Tesla V100 PCIe (vSGA is not supported.)
- Tesla V100 PCIe 32GB (vSGA is not supported.)
- Tesla V100 FHHL (vSGA is not supported.)
- GPUs based on the NVIDIA Turing™ architecture:
  - Tesla T4
  - Quadro RTX 6000 in displayless mode (GRID Virtual PC and GRID Virtual Applications are not supported. vComputeServer is supported only since release 9.1.)
  - Quadro RTX 8000 in displayless mode (GRID Virtual PC and GRID Virtual Applications are not supported. vComputeServer is supported only since release 9.1.)

In displayless mode, local physical display connectors are disabled.

For a list of validated server platforms, refer to NVIDIA GRID Certified Servers.

Tesla M60 and M6 GPUs support compute mode and graphics mode. NVIDIA vGPU requires GPUs that support both modes to operate in graphics mode.

Recent Tesla M60 GPUs and M6 GPUs are supplied in graphics mode. However, your GPU might be in compute mode if it is an older Tesla M60 GPU or M6 GPU, or if its mode has previously been changed.

To configure the mode of Tesla M60 and M6 GPUs, use the `gpumodeswitch` tool provided with NVIDIA vGPU software releases.

Even in compute mode, Tesla M60 and M6 GPUs do not support NVIDIA vComputeServer vGPU types.

Requirements for Using C-Series vComputeServer vGPUs

Because C-Series vComputeServer vGPUs have large BAR memory settings, using these vGPUs has some restrictions on VMware ESXi:

- The guest OS must be a 64-bit OS.
- 64-bit MMIO and EFI boot must be enabled for the VM.
- The guest OS must be able to be installed in EFI boot mode.
- The VM’s MMIO space must be increased to 64 GB as explained in VMware Knowledge Base Article: VMware vSphere VMDirectPath I/O: Requirements for Platforms and Devices (2142307).
- Because the VM’s MMIO space must be increased to 64 GB, vComputeServer requires ESXi 6.0 Update 3 and later, or ESXi 6.5 and later.
Requirements for Using vGPU on GPUs Requiring 64 GB of MMIO Space with Large-Memory VMs

Any GPU that has 16 GB or more of frame buffer requires 64 GB of MMIO space. When a vGPU on a GPU that requires 64 GB of MMIO space is assigned to a VM with 32 GB or more of memory on ESXi 6.0 Update 3 and later, or ESXi 6.5 and later updates, the VM’s MMIO space must be increased to 64 GB. For more information, see VMware Knowledge Base Article: VMware vSphere VMDirectPath I/O: Requirements for Platforms and Devices (2142307).

With ESXi 6.7, no extra configuration is needed.

The following GPUs require 64 GB of MMIO space:

- Tesla P6
- Tesla P40
- Tesla P100 (all variants)
- Tesla V100 (all variants)

Requirements for Using GPUs Based on the Pascal and Volta Architectures in Pass-Through Mode

- The Tesla V100, Tesla P100, and Tesla P6 GPUs require 32 GB of MMIO space in pass-through mode.
- The Tesla P40 GPU requires 64 GB of MMIO space in pass-through mode.
- Pass through of GPUs with large BAR memory settings has some restrictions on VMware ESXi:
  - The guest OS must be a 64-bit OS.
  - 64-bit MMIO and EFI boot must be enabled for the VM.
  - The guest OS must be able to be installed in EFI boot mode.
  - The Tesla V100, Tesla P100, and Tesla P6 require ESXi 6.0 Update 1 and later, or ESXi 6.5 and later.
  - Because it requires 64 GB of MMIO space, the Tesla P40 requires ESXi 6.0 Update 3 and later, or ESXi 6.5 and later.

As a result, the VM’s MMIO space must be increased to 64 GB as explained in VMware Knowledge Base Article: VMware vSphere VMDirectPath I/O: Requirements for Platforms and Devices (2142307).
2.2. Hypervisor Software Releases

Supported VMware vSphere Hypervisor (ESXi) Releases

This release is supported on the VMware vSphere Hypervisor (ESXi) releases listed in the table.

Support for NVIDIA vGPU software requires the Enterprise Plus Edition of VMware vSphere Hypervisor (ESXi). For details, see Compare VMware vSphere Editions (PDF). Updates to a base release of VMware vSphere Hypervisor (ESXi) are compatible with the base release and can also be used with this version of NVIDIA vGPU software unless expressly stated otherwise.

<table>
<thead>
<tr>
<th>Software</th>
<th>Release Supported</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>VMware vSphere Hypervisor (ESXi) 6.7</td>
<td>6.7 and compatible updates</td>
<td>All NVIDIA GPUs that support NVIDIA vGPU software are supported. Starting with release 6.7 U3, the assignment of multiple vGPUs to a single VM is supported. Starting with release 6.7 U1, vMotion with vGPU and suspend and resume with vGPU are supported on suitable GPUs as listed in Supported NVIDIA GPUs and Validated Server Platforms. Release 6.7 supports only suspend and resume with vGPU. vMotion with vGPU is not supported on release 6.7.</td>
</tr>
<tr>
<td>VMware vSphere Hypervisor (ESXi) 6.5</td>
<td>6.5 and compatible updates</td>
<td>All NVIDIA GPUs that support NVIDIA vGPU software are supported. The following features of NVIDIA vGPU software are not supported. Assignment of multiple vGPUs to a single VM</td>
</tr>
</tbody>
</table>
## Supported Management Software and Virtual Desktop Software Releases

This release supports the management software and virtual desktop software releases listed in the table.

<table>
<thead>
<tr>
<th>Software</th>
<th>Releases Supported</th>
</tr>
</thead>
<tbody>
<tr>
<td>VMware Horizon</td>
<td>7.9 and compatible 7.9.x updates</td>
</tr>
<tr>
<td></td>
<td>7.8 and compatible 7.8.x updates</td>
</tr>
<tr>
<td></td>
<td>7.7 and compatible 7.7.x updates</td>
</tr>
<tr>
<td></td>
<td>7.6 and compatible 7.6.x updates</td>
</tr>
</tbody>
</table>
2.3. Guest OS Support

NVIDIA vGPU software supports several Windows releases and Linux distributions as a guest OS. The supported guest operating systems depend on the hypervisor software version.

Use only a guest OS release that is listed as supported by NVIDIA vGPU software with your virtualization software. To be listed as supported, a guest OS release must be supported not only by NVIDIA vGPU software, but also by your virtualization software. NVIDIA cannot support guest OS releases that your virtualization software does not support.

NVIDIA vGPU software supports only 64-bit guest operating systems. No 32-bit guest operating systems are supported.

2.3.1. Windows Guest OS Support

NVIDIA vGPU software supports only the 64-bit Windows releases listed in the table as a guest OS on VMware vSphere. The releases of VMware vSphere for which a Windows release is supported depend on whether NVIDIA vGPU or pass-through GPU is used.

If a specific release, even an update release, is not listed, it’s not supported.

VMware vMotion with vGPU and suspend-resume with vGPU are supported on supported Windows guest OS releases.
### 2.3.2. Linux Guest OS Support

NVIDIA vGPU software supports **only** the Linux distributions listed in the table as a guest OS on VMware vSphere. The releases of VMware vSphere for which a Linux release is supported depend on whether NVIDIA vGPU or pass-through GPU is used.

If a specific release, even an update release, is not listed, it’s **not** supported.

VMware vMotion with vGPU and suspend-resume with vGPU are supported on supported Linux guest OS releases.
2.4. NVIDIA CUDA Toolkit Version Support

The releases in this release family of NVIDIA vGPU software support NVIDIA CUDA Toolkit 10.1 Update 1.

For more information about NVIDIA CUDA Toolkit, see CUDA Toolkit 10.1 Documentation.

If you are using NVIDIA vGPU software with CUDA on Linux, avoid conflicting installation methods by installing CUDA from a distribution-independent runfile package. Do not install CUDA from distribution-specific RPM or Deb package.

To ensure that the NVIDIA vGPU software graphics driver is not overwritten when CUDA is installed, deselect the CUDA driver when selecting the CUDA components to install.

For more information, see NVIDIA CUDA Installation Guide for Linux.

2.5. vGPU Migration Support

vGPU migration, which includes vMotion and suspend-resume, is supported only on a subset of supported GPUs, VMware vSphere Hypervisor (ESXi) releases, and guest operating systems.

Supported GPUs:
- Tesla M6
- Tesla M10
- Tesla M60
- Tesla P4
- Tesla P6
- Tesla P40
- Tesla V100 SXM2
- Tesla V100 SXM2 32GB
Validated Platforms

- Tesla V100 PCIe
- Tesla V100 PCIe 32GB
- Tesla V100 FHHL
- Tesla T4
- Quadro RTX 6000
- Quadro RTX 8000

Supported VMware vSphere Hypervisor (ESXi) releases:

- Release 6.7 U1 and compatible updates support vMotion with vGPU and suspend-resume with vGPU.
- Release 6.7 supports only suspend-resume with vGPU.
- Releases earlier than 6.7 do not support any form of vGPU migration.

Supported guest OS releases: Windows and Linux.

2.6. Multiple vGPU Support

To support applications and workloads that are compute or graphics intensive, multiple vGPUs can be added to a single VM. The assignment of more than one vGPU to a VM is supported only on a subset of vGPUs and VMware vSphere Hypervisor (ESXi) releases.

Supported vGPUs

Only Q-series and C-series vGPUs that are allocated all of the physical GPU’s frame buffer are supported.

<table>
<thead>
<tr>
<th>GPU Architecture</th>
<th>Board</th>
<th>vGPU</th>
</tr>
</thead>
<tbody>
<tr>
<td>Turing</td>
<td>Tesla T4</td>
<td>T4-16Q</td>
</tr>
<tr>
<td></td>
<td></td>
<td>T4-16C</td>
</tr>
<tr>
<td></td>
<td>Quadro RTX 6000</td>
<td>RTX6000-24Q</td>
</tr>
<tr>
<td></td>
<td>Quadro RTX 8000</td>
<td>RTX8000-48Q</td>
</tr>
<tr>
<td>Volta</td>
<td>Tesla V100 SXM2 32GB</td>
<td>V100DX-32Q</td>
</tr>
<tr>
<td></td>
<td></td>
<td>V100D-32C</td>
</tr>
<tr>
<td></td>
<td>Tesla V100 PCIe 32GB</td>
<td>V100D-32Q</td>
</tr>
<tr>
<td></td>
<td></td>
<td>V100D-32C</td>
</tr>
<tr>
<td></td>
<td>Tesla V100 SXM2</td>
<td>V100X-16Q</td>
</tr>
<tr>
<td></td>
<td></td>
<td>V100X-16C</td>
</tr>
<tr>
<td></td>
<td>Tesla V100 PCIe</td>
<td>V100-16Q</td>
</tr>
<tr>
<td></td>
<td></td>
<td>V100-16C</td>
</tr>
<tr>
<td>GPU Architecture</td>
<td>Board</td>
<td>vGPU</td>
</tr>
<tr>
<td>-------------------</td>
<td>----------------</td>
<td>------------</td>
</tr>
<tr>
<td>Pascal</td>
<td>Tesla V100 FHHL</td>
<td>V100L-16Q</td>
</tr>
<tr>
<td></td>
<td></td>
<td>V100L-16C</td>
</tr>
<tr>
<td></td>
<td>Tesla P100 SXM2</td>
<td>P100X-16Q</td>
</tr>
<tr>
<td></td>
<td></td>
<td>P100X-16C</td>
</tr>
<tr>
<td></td>
<td>Tesla P100 PCIe 16GB</td>
<td>P100-16Q</td>
</tr>
<tr>
<td></td>
<td></td>
<td>P100-16C</td>
</tr>
<tr>
<td></td>
<td>Tesla P100 PCIe 12GB</td>
<td>P100C-12Q</td>
</tr>
<tr>
<td></td>
<td></td>
<td>P100C-12C</td>
</tr>
<tr>
<td></td>
<td>Tesla P40</td>
<td>P40-24Q</td>
</tr>
<tr>
<td></td>
<td></td>
<td>P40-24C</td>
</tr>
<tr>
<td></td>
<td>Tesla P6</td>
<td>P6-16Q</td>
</tr>
<tr>
<td></td>
<td></td>
<td>P6-16C</td>
</tr>
<tr>
<td></td>
<td>Tesla P4</td>
<td>P4-8Q</td>
</tr>
<tr>
<td></td>
<td></td>
<td>P4-8C</td>
</tr>
<tr>
<td>Maxwell</td>
<td>Tesla M60</td>
<td>M60-8Q</td>
</tr>
<tr>
<td></td>
<td>Tesla M10</td>
<td>M10-8Q</td>
</tr>
<tr>
<td></td>
<td>Tesla M6</td>
<td>M6-8Q</td>
</tr>
</tbody>
</table>

**Maximum vGPUs per VM**

NVIDIA vGPU software supports up to a maximum of vGPUs per VM on VMware vSphere Hypervisor (ESXi).

**Supported Hypervisor Releases**

VMware vSphere Hypervisor (ESXi) release 6.7 U3 and later compatible updates only.

**2.7. Peer-to-Peer CUDA Transfers over NVLink Support**

Peer-to-peer CUDA transfers enable device memory between vGPUs on different GPUs that are assigned to the same VM to be accessed from within the CUDA kernels. NVLink is a high-bandwidth interconnect that enables fast communication between such vGPUs.
Peer-to-Peer CUDA Transfers over NVLink is supported only on a subset of vGPUs, VMware vSphere Hypervisor (ESXi) releases, and guest OS releases.

**Supported vGPUs**

Only Q-series and C-series vGPUs that are allocated all of the physical GPU’s frame buffer on physical GPUs that support NVLink are supported.

<table>
<thead>
<tr>
<th>GPU Architecture</th>
<th>Board</th>
<th>vGPU</th>
</tr>
</thead>
<tbody>
<tr>
<td>Turing</td>
<td>Quadro RTX 6000</td>
<td>RTX6000-24Q</td>
</tr>
<tr>
<td></td>
<td>Quadro RTX 8000</td>
<td>RTX8000-48Q</td>
</tr>
<tr>
<td>Volta</td>
<td>Tesla V100 SXM2 32GB</td>
<td>V100DX-32Q</td>
</tr>
<tr>
<td></td>
<td></td>
<td>V100DX-32C</td>
</tr>
<tr>
<td></td>
<td>Tesla V100 SXM2</td>
<td>V100X-16Q</td>
</tr>
<tr>
<td></td>
<td></td>
<td>V100X-16C</td>
</tr>
<tr>
<td>Pascal</td>
<td>Tesla P100 SXM2</td>
<td>P100X-16Q</td>
</tr>
<tr>
<td></td>
<td></td>
<td>P100X-16C</td>
</tr>
</tbody>
</table>

**Supported Hypervisor Releases**

Peer-to-Peer CUDA Transfers over NVLink are supported on all hypervisor releases that support the assignment of more than one vGPU to a VM. For details, see **Multiple vGPU Support**.

**Supported Guest OS Releases**

Linux only. Peer-to-Peer CUDA Transfers over NVLink are **not** supported on Windows.

**Limitations**

- Only direct connections are supported. NVSwitch is not supported.
- PCIe is not supported.
- SLI is not supported.
Known product limitations for this release of NVIDIA vGPU software are described in the following sections.

3.1. Issues may occur with graphics-intensive OpenCL applications on vGPU types with limited frame buffer

Description

Issues may occur when graphics-intensive OpenCL applications are used with vGPU types that have limited frame buffer. These issues occur when the applications demand more frame buffer than is allocated to the vGPU.

For example, these issues may occur with the Adobe Photoshop and LuxMark OpenCL Benchmark applications:

- When the image resolution and size are changed in Adobe Photoshop, a program error may occur or Photoshop may display a message about a problem with the graphics hardware and a suggestion to disable OpenCL.
- When the LuxMark OpenCL Benchmark application is run, XID error 31 may occur.

Workaround

For graphics-intensive OpenCL applications, use a vGPU type with more frame buffer.
3.2. In pass through mode, all GPUs connected to each other through NVLink must be assigned to the same VM

Description
In pass through mode, all GPUs connected to each other through NVLink must be assigned to the same VM. If a subset of GPUs connected to each other through NVLink is passed through to a VM, unrecoverable error XID 74 occurs when the VM is booted. This error corrupts the NVLink state on the physical GPUs and, as a result, the NVLink bridge between the GPUs is unusable.

Workaround
Restore the NVLink state on the physical GPUs by resetting the GPUs or rebooting the hypervisor host.

3.3. vGPU profiles with 512 Mbytes or less of frame buffer support only 1 virtual display head on Windows 10

Description
To reduce the possibility of memory exhaustion, vGPU profiles with 512 Mbytes or less of frame buffer support only 1 virtual display head on a Windows 10 guest OS.

The following vGPU profiles have 512 Mbytes or less of frame buffer:

- Tesla M6-0B, M6-0Q
- Tesla M10-0B, M10-0Q
- Tesla M60-0B, M60-0Q

Workaround
Use a profile that supports more than 1 virtual display head and has at least 1 Gbyte of frame buffer.
3.4. NVENC requires at least 1 Gbyte of frame buffer

Description
Using the frame buffer for the NVIDIA hardware-based H.264/HEVC video encoder (NVENC) may cause memory exhaustion with vGPU profiles that have 512 Mbytes or less of frame buffer. To reduce the possibility of memory exhaustion, NVENC is disabled on profiles that have 512 Mbytes or less of frame buffer. Application GPU acceleration remains fully supported and available for all profiles, including profiles with 512 MBytes or less of frame buffer. NVENC support from both Citrix and VMware is a recent feature and, if you are using an older version, you should experience no change in functionality.

The following vGPU profiles have 512 Mbytes or less of frame buffer:

- Tesla M6-0B, M6-0Q
- Tesla M10-0B, M10-0Q
- Tesla M60-0B, M60-0Q

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

3.5. VM failures or crashes on servers with 1 TB or more of system memory

Description
Support for vGPU and vSGA is limited to servers with less than 1 TB of system memory. On servers with 1 TB or more of system memory, VM failures or crashes may occur. For example, when Citrix Virtual Apps and Desktops is used with a Windows 7 guest OS, a blue screen crash may occur. However, support for vDGA is not affected by this limitation.

This limitation applies only to systems with supported GPUs based on the Maxwell architecture: Tesla M6, Tesla M10, and Tesla M60.

Resolution
1. Limit the amount of system memory on the server to 1 TB minus 16 GB by setting memmapMaxRAMMB to 1032192, which is equal to 1048576 minus 16384.
2. Reboot the server.

If the problem persists, contact your server vendor for the recommended system memory configuration with NVIDIA GPUs.

### 3.6. VM running older NVIDIA vGPU drivers fails to initialize vGPU when booted

**Description**

A VM running a version of the NVIDIA guest VM drivers from a previous main release branch, for example release 4.4, will fail to initialize vGPU when booted on a VMware vSphere platform running the current release of Virtual GPU Manager.

In this scenario, the VM boots in standard VGA mode with reduced resolution and color depth. The NVIDIA virtual GPU is present in Windows Device Manager but displays a warning sign, and the following device status:

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

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

- A version mismatch between guest and host drivers:

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

- A signature mismatch:

  ```
  ```

**Resolution**

Install the current NVIDIA guest VM driver in the VM.

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

**Description**

A single vGPU configured on a physical GPU produces lower benchmark scores than the physical GPU run in pass-through mode.

Aside from performance differences that may be attributed to a vGPU's smaller frame buffer size, vGPU incorporates a performance balancing feature known as Frame Rate
Limiter (FRL). On vGPUs that use the best-effort scheduler, FRL is enabled. On vGPUs that use the fixed share or equal share scheduler, FRL is disabled.

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

Resolution

FRL is controlled by an internal vGPU setting. On vGPUs that use the best-effort scheduler, NVIDIA does not validate vGPU with FRL disabled, but for validation of benchmark performance, FRL can be temporarily disabled by adding the configuration parameter `pciPassthru0.cfg.frame_rate_limiter` in the VM’s advanced configuration options.

This setting can only be changed when the VM is powered off.

1. Select **Edit Settings**.
2. In **Edit Settings** window, select the **VM Options** tab.
3. From the **Advanced** drop-down list, select **Edit Configuration**.
4. In the **Configuration Parameters** dialog box, click **Add Row**.
5. In the **Name** field, type the parameter name `pciPassthru0.cfg.frame_rate_limiter`, in the **Value** field type 0, and click **OK**.
With this setting in place, the VM's vGPU will run without any frame rate limit. The FRL can be reverted back to its default setting by setting `pciPassthru0.cfg.frame_rate_limiter` to 1 or by removing the parameter from the advanced settings.

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

#### Description

When starting multiple VMs configured with large amounts of RAM (typically more than 32GB per VM), a VM may fail to initialize vGPU. In this scenario, the VM boots in VMware SVGA mode and doesn’t load the NVIDIA driver. The NVIDIA vGPU software GPU is present in **Windows Device Manager** but displays a warning sign, and the following device status:

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

The VMware vSphere VM's log file contains these error messages:
vthread10|E105: NVOS status 0x29
vthread10|E105: Assertion Failed at 0x7620fd4b:179
vthread10|E105: 8 frames returned by backtrace
... vthread10|E105: VGPU message 12 failed, result code: 0x29
... vthread10|E105: NVOS status 0x8
vthread10|E105: Assertion Failed at 0x7620c8df:280
vthread10|E105: 8 frames returned by backtrace
... vthread10|E105: VGPU message 26 failed, result code: 0x8

Resolution

vGPU reserves a portion of the VM’s framebuffer for use in GPU mapping of VM system memory. The reservation is sufficient to support up to 32GB of system memory, and may be increased to accommodate up to 64GB by adding the configuration parameter pciPassthru0.cfg.enable_large_sys_mem in the VM’s advanced configuration options.

This setting can only be changed when the VM is powered off.

1. Select Edit Settings.
2. In Edit Settings window, select the VM Options tab.
3. From the Advanced drop-down list, select Edit Configuration.
4. In the Configuration Parameters dialog box, click Add Row.
5. In the Name field, type the parameter name pciPassthru0.cfg.enable_large_sys_mem, in the Value field type 1, and click OK.

With this setting in place, less GPU framebuffer is available to applications running in the VM. To accommodate system memory larger than 64GB, the reservation can be further increased by adding pciPassthru0.cfg.extra_fb_reservation in the VM’s advanced configuration options, and setting its value to the desired reservation size in megabytes. The default value of 64M is sufficient to support 64 GB of RAM. We recommend adding 2 M of reservation for each additional 1 GB of system memory. For example, to support 96 GB of RAM, set pciPassthru0.cfg.extra_fb_reservation to 128.

The reservation can be reverted back to its default setting by setting pciPassthru0.cfg.enable_large_sys_mem to 0, or by removing the parameter from the advanced settings.
Chapter 4.
RESOLVED ISSUES

Only resolved issues that have been previously noted as known issues or had a noticeable user impact are listed. The summary and description for each resolved issue indicate the effect of the issue on NVIDIA vGPU software before the issue was resolved.

Issues Resolved in Release 9.0

<table>
<thead>
<tr>
<th>Bug ID</th>
<th>Summary and Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Virtual GPU fails to start if ECC is enabled</td>
</tr>
<tr>
<td></td>
<td>NVIDIA vGPU does not support error correcting code (ECC) memory. If ECC memory is enabled, NVIDIA vGPU fails to start.</td>
</tr>
<tr>
<td></td>
<td>Starting with NVIDIA vGPU software release 9.0, NVIDIA vGPU supports ECC memory on GPUs and hypervisor software versions that support ECC.</td>
</tr>
<tr>
<td>200269717</td>
<td>On Tesla P40, P6, and P4 GPUs, the default ECC setting prevents NVIDIA vGPU from starting.</td>
</tr>
<tr>
<td></td>
<td>Starting with NVIDIA vGPU software release 9.0, NVIDIA vGPU supports ECC memory on GPUs and hypervisor software versions that support ECC.</td>
</tr>
<tr>
<td>2285306</td>
<td>Cloned VMs configured with a vGPU type different than the type in the master image fail to start.</td>
</tr>
<tr>
<td></td>
<td>Cloned VMs configured with a vGPU type different than the type in the master image fail to start.</td>
</tr>
<tr>
<td></td>
<td>When a Windows 10 VM is booted, the VM becomes stuck in a loop and alternately displays <strong>Getting devices ready: 50% and Preparation in progress.</strong></td>
</tr>
</tbody>
</table>
## Issues Resolved in Release 9.1

<table>
<thead>
<tr>
<th>Bug ID</th>
<th>Summary and Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>200526633</td>
<td><strong>9.0 only: VM crashes after the volatile ECC error count is reset</strong></td>
</tr>
<tr>
<td></td>
<td>After the command <code>nvidia-smi -p 0</code> is run from a guest VM to reset the volatile ECC error count, the VM crashes.</td>
</tr>
<tr>
<td>200525006</td>
<td><strong>9.0 only: Incorrect ECC error counts are reported for vGPUs on some GPUs</strong></td>
</tr>
<tr>
<td></td>
<td>Incorrect ECC error counts are reported for vGPUs on some GPUs when the command <code>nvidia-smi -q</code> is run from a guest VM.</td>
</tr>
<tr>
<td>200524555</td>
<td><strong>9.0 only: On Linux VMs, the license directory is not deleted when the guest driver is uninstalled</strong></td>
</tr>
<tr>
<td></td>
<td>On Linux guest VMs, the license directory <code>/etc/nvidia/license</code> is not deleted when the NVIDIA vGPU software graphics driver is uninstalled.</td>
</tr>
<tr>
<td>200524348</td>
<td><strong>9.0 only: nvidia-smi shows the incorrect ECC state for a vGPU</strong></td>
</tr>
<tr>
<td></td>
<td><code>nvidia-smi vgpu -q</code> shows the incorrect ECC state of a vGPU when ECC is enabled on the physical GPU but disabled on the vGPU from the vGPU VM. This issue occurs because data for the physical GPU host is not being reset and is being reused even after reboot.</td>
</tr>
<tr>
<td>200522255</td>
<td><strong>9.0 only: No vComputeServer option available in NVIDIA X Server Settings</strong></td>
</tr>
<tr>
<td></td>
<td>The <code>vComputeServer</code> option is missing from the Manage License section in the NVIDIA X Server Settings window.</td>
</tr>
<tr>
<td>200434909</td>
<td><strong>9.0 only: Users' view sessions may become corrupted after migration</strong></td>
</tr>
<tr>
<td></td>
<td>When a VM configured with vGPU under heavy load is migrated to another host, users' view sessions may become corrupted after the migration.</td>
</tr>
</tbody>
</table>
5.1. Migrating a VM configured with NVIDIA vGPU software release 9.1 to a host running release 9.0 fails

Description

This issue occurs only with the following combination of releases of guest VM graphics driver, vGPU manager on the source host, and vGPU manager on the destination host:

<table>
<thead>
<tr>
<th>Guest VM Graphics Driver</th>
<th>Source vGPU Manager</th>
<th>Destination vGPU Manager</th>
</tr>
</thead>
<tbody>
<tr>
<td>9.1</td>
<td>9.1</td>
<td>9.0</td>
</tr>
</tbody>
</table>

Workaround

Tesla M10 GPUs do not support this workaround. Even after applying this workaround to a system on which this issue occurs, vGPU migration with Tesla M10 GPUs fails with the following error:

Unexpected migration data block encountered.

1. On the host that is running vGPU Manager 9.1, set the registry key `RMSetVGPUVersionMax` to 0x30001.
2. Start the VM.
3. Confirm that the vGPU version in the log files is 0x30001.

The VM can now be migrated.
Known Issues

Status
Not a bug

Ref. #
200533827

5.2. ECC memory with NVIDIA vGPU is not supported on Tesla M60 and Tesla M6

Description
Error-correcting code (ECC) memory with NVIDIA vGPU is not supported on Tesla M60 and Tesla M6 GPUs. The effect of starting NVIDIA vGPU when it is configured on a Tesla M60 or Tesla M6 GPU on which ECC memory is enabled depends on your NVIDIA vGPU software release.

- 9.0 only: The hypervisor host fails.
- Since 9.1: The VM fails to start.

Workaround
Ensure that ECC memory is disabled on Tesla M60 and Tesla M6 GPUs. For more information, see Virtual GPU fails to start if ECC is enabled.

Status
Open

5.3. Virtual GPU fails to start if ECC is enabled

Description
Tesla M60, Tesla M6, and GPUs based on the Pascal GPU architecture, for example Tesla P100 or Tesla P4, support error correcting code (ECC) memory for improved data integrity. Tesla M60 and M6 GPUs in graphics mode are supplied with ECC memory disabled by default, but it may subsequently be enabled using nvidia-smi. GPUs based on the Pascal GPU architecture are supplied with ECC memory enabled.

However, NVIDIA vGPU does not support ECC memory with the following GPUs:

- Tesla M60 GPUs
- Tesla M6 GPUs
If ECC memory is enabled and your GPU does not support ECC, NVIDIA vGPU fails to start.

The following error is logged in the VMware vSphere host's log file:

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

**Resolution**

If you are using Tesla M60 or Tesla M6 GPUs, ensure that ECC is disabled on all GPUs.

Before you begin, ensure that NVIDIA Virtual GPU Manager is installed on your hypervisor.

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

   ```
   # nvidia-smi -q
   ==================NVSMI LOG==================
   Timestamp: Tue Dec 19 18:36:45 2017
   Driver Version: 384.99
   Attached GPUs: 1
   GPU 0000:02:00.0
   [...]
   Ecc Mode
   Current: Enabled
   Pending: Enabled
   [...]
   ```

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

   - If you want to change the ECC status to off for all GPUs on your host machine, run this command:
     ```
     # nvidia-smi -e 0
     ```
   - If you want to change the ECC status to off for a specific GPU, run this command:
     ```
     # nvidia-smi -i id -e 0
     id is the index of the GPU as reported by `nvidia-smi`.
     ```

     This example disables ECC for the GPU with index `0000:02:00.0`.

     ```
     # nvidia-smi -i 0000:02:00.0 -e 0
     ```

3. Reboot the host.

4. Confirm that ECC is now disabled for the GPU.

   ```
   # nvidia-smi -q
   ==================NVSMI LOG==================
   ```
Known Issues

If you later need to enable ECC on your GPUs, run one of the following commands:

- If you want to change the ECC status to on for all GPUs on your host machine, run this command:
  ```
  # nvidia-smi -e 1
  ```

- If you want to change the ECC status to on for a specific GPU, run this command:
  ```
  # nvidia-smi -i id -e 1
  
  id is the index of the GPU as reported by nvidia-smi.
  
  This example enables ECC for the GPU with index 0000:02:00.0.
  ```

  ```
  # nvidia-smi -i 0000:02:00.0 -e 1
  ```

After changing the ECC status to on, reboot the host.

5.4. VMware vCenter shows GPUs with no available GPU memory

Description

VMware vCenter shows some physical GPUs as having 0.0 B of available GPU memory. VMs that have been assigned vGPUs on the affected physical GPUs cannot be booted. The `nvidia-smi` command shows the same physical GPUs as having some GPU memory available.

Workaround

Stop and restart the Xorg service and `nv-hostengine` on the ESXi host.

1. Stop all running VM instances on the host.
2. Stop the Xorg service.
   ```
   [root@esxi:~] /etc/init.d/xorg stop
   ```
   ```
   [root@esxi:~] nv-hostengine -t
   ```
4. Wait for 1 second to allow `nv-hostengine` to stop.
5. Start `nv-hostengine`.

```bash
[root@esxi:~] nv-hostengine -d
```

6. Start the Xorg service.

```bash
[root@esxi:~] /etc/init.d/xorg start
```

**Status**

Not an NVIDIA bug

A fix is available from VMware in VMware vSphere ESXi 6.7 U3. For information about the availability of fixes for other releases of VMware vSphere ESXi, contact VMware.

**Ref. #**

2644794

### 5.5. RAPIDS cuDF `merge` fails on NVIDIA vGPU

**Description**

The `merge` function of the RAPIDS cuDF GPU data frame library fails on NVIDIA vGPU. This function fails because RAPIDS uses the Unified Memory feature of CUDA, which NVIDIA vGPU does not support.

**Status**

Open

**Ref. #**

2642134

### 5.6. 9.0 only: Users' view sessions may become corrupted after migration

**Description**

When a VM configured with vGPU under heavy load is migrated to another host, users' view sessions may become corrupted after the migration.

**Workaround**

Restart the VM.
**Status**
Resolved in NVIDIA vGPU software 9.1

**Ref. #**
200434909

**5.7. Users' sessions may freeze during vMotion migration of VMs configured with vGPU**

**Description**
When vMotion is used to migrate a VM configured with vGPU to another host, users’ sessions may freeze for up to several seconds during the migration.

These factors may increase the length of time for which a session freezes:
- Continuous use of the frame buffer by the workload, which typically occurs with workloads such as video streaming
- A large amount of vGPU frame buffer
- A large amount of system memory
- Limited network bandwidth

**Workaround**
Administrators can mitigate the effects on end users by avoiding migration of VMs configured with vGPU during business hours or warning end users that migration is about to start and that they may experience session freezes.

End users experiencing this issue must wait for their sessions to resume when the migration is complete.

**Status**
Open

**Ref. #**
2569578
5.8. Migration of VMs configured with vGPU stops before the migration is complete

Description
When a VM configured with vGPU is migrated to another host, the migration stops before it is complete. After the migration stops, the VM is no longer accessible.

This issue occurs if the ECC memory configuration (enabled or disabled) on the source and destination hosts are different. The ECC memory configuration on both the source and destination hosts must be identical.

Workaround
Reboot the hypervisor host to recover the VM. Before attempting to migrate the VM again, ensure that the ECC memory configuration on both the source and destination hosts are identical.

Status
Not an NVIDIA bug

Ref. #
200520027

5.9. 9.0 only: nvidia-smi shows the incorrect ECC state for a vGPU

Description
nvidia-smi vgpu -q shows the incorrect ECC state of a vGPU when ECC is enabled on the physical GPU but disabled on the vGPU from the vGPU VM. This issue occurs because data for the physical GPU host is not being reset and is being reused even after reboot.

Status
Resolved in NVIDIA vGPU software 9.1

Ref. #
200524348
5.10. 9.0 only: Incorrect ECC error counts are reported for vGPUs on some GPUs

Description
Incorrect ECC error counts are reported for vGPUs on some GPUs when the command `nvidia-smi -q` is run from a guest VM.

This issue affects only vGPUs that reside on physical GPUs based on the NVIDIA Volta GPU architecture. For vGPUs on GPUs based on other architectures, the ECC error count is correct.

Status
Resolved in NVIDIA vGPU software 9.1

Ref. #
200525006

5.11. ECC memory settings for a vGPU cannot be changed by using NVIDIA X Server Settings

Description
The ECC memory settings for a vGPU cannot be changed from a Linux guest VM by using NVIDIA X Server Settings. After the ECC memory state has been changed on the ECC Settings page and the VM has been rebooted, the ECC memory state remains unchanged.

Workaround
Use the `nvidia-smi` command in the guest VM to enable or disable ECC memory for the vGPU as explained in Virtual GPU Software User Guide.

If the ECC memory state remains unchanged even after you use the `nvidia-smi` command to change it, use the workaround in Changes to ECC memory settings for a Linux vGPU VM by nvidia-smi might be ignored.

Status
Open
5.12. Changes to ECC memory settings for a Linux vGPU VM by `nvidia-smi` might be ignored

Description

After the ECC memory state for a Linux vGPU VM has been changed by using the `nvidia-smi` command and the VM has been rebooted, the ECC memory state might remain unchanged.

This issue occurs when multiple NVIDIA configuration files in the system cause the kernel module option for setting the ECC memory state `RMGuestECCState` in `/etc/modprobe.d/nvidia.conf` to be ignored.

When the `nvidia-smi` command is used to enable ECC memory, the file `/etc/modprobe.d/nvidia.conf` is created or updated to set the kernel module option `RMGuestECCState`. Another configuration file in `/etc/modprobe.d/` that contains the keyword `NVreg_RegistryDwordsPerDevice` might cause the kernel module option `RMGuestECCState` to be ignored.

Workaround

This workaround requires administrator privileges.

1. Move the entry containing the keyword `NVreg_RegistryDwordsPerDevice` from the other configuration file to `/etc/modprobe.d/nvidia.conf`.
2. Reboot the VM.

Status

Open

Ref. #

200505777
5.13. 9.0 only: VM crashes after the volatile ECC error count is reset

Description
After the command `nvidia-smi -p 0` is run from a guest VM to reset the volatile ECC error count, the VM crashes.
This issue does not occur if the EEC state in the VM is set to off.

Status
Resolved in NVIDIA vGPU software 9.1

Ref. #
200526633

5.14. 9.0 only: No `vComputeServer` option available in NVIDIA X Server Settings

Description
The `vComputeServer` option is missing from the Manage License section in the NVIDIA X Server Settings window.
As a result of this missing option, the NVIDIA X Server Settings window incorrectly states that the system is licensed for Quadro vDWS when, in fact, the system is licensed for vComputeServer.
Workaround

If you are licensing a physical GPU for vComputeServer, you must use the configuration file /etc/nvidia/gridd.conf. See Virtual GPU Client Licensing User Guide.

Status

Resolved in NVIDIA vGPU software 9.1

Ref. #

200522255
5.15. 9.0 only: On Linux VMs, the license directory is not deleted when the guest driver is uninstalled

**Description**

On Linux guest VMs, the license directory `/etc/nvidia/license` is not deleted when the NVIDIA vGPU software graphics driver is uninstalled.

The following error message is written to the `nvidia-uninstaller` log file:

```
Failed to delete the directory '/etc/nvidia' (Directory not empty).
```

**Workaround**

As root, remove the `/etc/nvidia/license` directory after the NVIDIA vGPU software graphics driver is uninstalled.

**Status**

Resolved in NVIDIA vGPU software 9.1

**Ref. #**

200524555

5.16. Black screens observed when a VMware Horizon session is connected to four displays

**Description**

When a VMware Horizon session with Windows 7 is connected to four displays, a black screen is observed on one or more displays.

This issue occurs because a VMware Horizon session does not support connections to four 4K displays with Windows 7.

**Status**

Not an NVIDIA bug

**Ref. #**

200503538
5.17. Quadro RTX 8000 and Quadro RTX 6000 GPUs can't be used with VMware vSphere ESXi 6.5

Description
Quadro RTX 8000 and Quadro RTX 6000 GPUs can't be used with VMware vSphere ESXi 6.5. If you attempt to use the Quadro RTX 8000 or Quadro RTX 6000 GPU with VMware vSphere ESXi 6.5, a purple-screen crash occurs after you install the NVIDIA Virtual GPU Manager.

Version
VMware vSphere ESXi 6.5

Status
Open

Ref. #
200491080

5.18. Vulkan applications crash in Windows 7 guest VMs configured with NVIDIA vGPU

Description
In Windows 7 guest VMs configured with NVIDIA vGPU, applications developed with Vulkan APIs crash or throw errors when they are launched. Vulkan APIs require sparse texture support, but in Windows 7 guest VMs configured with NVIDIA vGPU, sparse textures are not enabled.

In Windows 10 guest VMs configured with NVIDIA vGPU, sparse textures are enabled and applications developed with Vulkan APIs run correctly in these VMs.

Status
Open

Ref. #
200381348
5.19. Host core CPU utilization is higher than expected for moderate workloads

Description
When GPU performance is being monitored, host core CPU utilization is higher than expected for moderate workloads. For example, host CPU utilization when only a small number of VMs are running is as high as when several times as many VMs are running.

Workaround
Disable monitoring of the following GPU performance statistics:
- vGPU engine usage by applications across multiple vGPUs
- Encoder session statistics
- Frame buffer capture (FBC) session statistics
- Statistics gathered by performance counters in guest VMs

Status
Open

Ref. #
2414897

5.20. H.264 encoder falls back to software encoding on 1Q vGPUs with a 4K display

Description
On 1Q vGPUs with a 4K display, a shortage of frame buffer causes the H.264 encoder to fall back to software encoding.

Workaround
Use a 2Q or larger virtual GPU type to provide more frame buffer for each vGPU.

Status
Open
5.21. H.264 encoder falls back to software encoding on 2Q vGPUs with 3 or more 4K displays

Description
On 2Q vGPUs with three or more 4K displays, a shortage of frame buffer causes the H.264 encoder to fall back to software encoding.

This issue affects only vGPUs assigned to VMs that are running a Linux guest OS.

Workaround
Use a 4Q or larger virtual GPU type to provide more frame buffer for each vGPU.

Status
Open

Ref. #
200457177

5.22. Frame capture while the interactive logon message is displayed returns blank screen

Description
Because of a known limitation with NvFBC, a frame capture while the interactive logon message is displayed returns a blank screen.

An NvFBC session can capture screen updates that occur after the session is created. Before the logon message appears, there is no screen update after the message is shown and, therefore, a black screen is returned instead. If the NvFBC session is created after this update has occurred, NvFBC cannot get a frame to capture.

Workaround
Press Enter or wait for the screen to update for NvFBC to capture the frame.
Status
Not a bug

Ref. #
2115733

5.23. RDS sessions do not use the GPU with some Microsoft Windows Server releases

Description
When some releases of Windows Server are used as a guest OS, Remote Desktop Services (RDS) sessions do not use the GPU. With these releases, the RDS sessions by default use the Microsoft Basic Render Driver instead of the GPU. This default setting enables 2D DirectX applications such as Microsoft Office to use software rendering, which can be more efficient than using the GPU for rendering. However, as a result, 3D applications that use DirectX are prevented from using the GPU.

Version
- Windows Server 2016
- Windows Server 2012

Solution
Change the local computer policy to use the hardware graphics adapter for all RDS sessions.

2. Set the Use the hardware default graphics adapter for all Remote Desktop Services sessions option.

5.24. VMware vMotion fails gracefully under heavy load

Description
Migrating a VM configured with vGPU fails gracefully if the VM is running an intensive workload.
The error stack in the task details on the vSphere web client contains the following error message:

```
The migration has exceeded the maximum switchover time of 100 second(s). ESX has preemptively failed the migration to allow the VM to continue running on the source. To avoid this failure, either increase the maximum allowable switchover time or wait until the VM is performing a less intensive workload.
```

**Workaround**

Increase the maximum switchover time by increasing the `vmotion.maxSwitchoverSeconds` option from the default value of 100 seconds.

For more information, see VMware Knowledge Base Article: vMotion or Storage vMotion of a VM fails with the error: The migration has exceeded the maximum switchover time of 100 second(s) (2141355).

**Status**

Not an NVIDIA bug

**Ref. #**

200416700

5.25. View session freezes intermittently after a Linux VM acquires a license

**Description**

In a Linux VM, the view session can sometimes freeze after the VM acquires a license.

**Workaround**

Resize the view session.

**Status**

Open

**Ref. #**

200426961
5.26. Even when the scheduling policy is equal share, unequal GPU utilization is reported

Description

When the scheduling policy is equal share, unequal GPU engine utilization can be reported for the vGPUs on the same physical GPU.

For example, GPU engine usage for three P40-8Q vGPUs on a Tesla P40 GPU might be reported as follows:

```
[root@localhost:~] nvidia-smi vgpu
Wed Jun 27 10:33:18 2018
+-----------------------------------------------------------------------------+
| NVIDIA-SMI 390.59                 Driver Version: 390.59                    |
|-------------------------------+--------------------------------+------------|
| GPU  Name                     | Bus-Id                         | GPU-Util   |
|      vGPU ID    Name          | VM ID    VM Name               | vGPU-Util  |
|===============================+================================+============|
|   0  Tesla P40                | 00000000:81:00.0               |  52%       |
|      2122661    GRID P40-8Q   | 2122682  centos7.4-xmpl-211... |     19%    |
|      2122663    GRID P40-8Q   | 2122692  centos7.4-xmpl-211... |      0%    |
|      2122659    GRID P40-8Q   | 2122664  centos7.4-xmpl-211... |     25%    |
|   1  Tesla P40                | 00000000:85:00.0               |  58%       |
|      2122662    GRID P40-8Q   | 2122689  centos7.4-xmpl-211... |      0%    |
|      2122658    GRID P40-8Q   | 2122667  centos7.4-xmpl-211... |     59%    |
|      2122660    GRID P40-8Q   | 2122670  centos7.4-xmpl-211... |      0%    |
+-------------------------------+--------------------------------+------------+
```

The vGPU utilization of the vGPU 2122658 is reported as 59%. However, the expected vGPU utilization should not exceed 33%.

This behavior is a result of the mechanism that is used to measure GPU engine utilization.

Status

Open

Ref. #

2175888
5.27. When the scheduling policy is fixed share, GPU utilization is reported as higher than expected

Description

When the scheduling policy is fixed share, GPU engine utilization can be reported as higher than expected for a vGPU.

For example, GPU engine usage for six P40-4Q vGPUs on a Tesla P40 GPU might be reported as follows:

```
[root@localhost:~] nvidia-smi vgpu
Mon Aug 20 10:33:18 2018
+-----------------------------------------------------------------------------+
| NVIDIA-SMI 390.42                 Driver Version: 390.42                    |
|-------------------------------+--------------------------------+------------|
| GPU  Name                     | Bus-Id                         | GPU-Util   |
|      vGPU ID    Name          | VM ID    VM Name               | vGPU-Util  |
|===============================+================================+============|
|   0  Tesla P40                | 00000000:81:00.0               |  99%       |
|      85109      GRID P40-4Q   | 85110    win7-xmpl-146048-1    |     32%    |
|      87195      GRID P40-4Q   | 87196    win7-xmpl-146048-2    |     39%    |
|      88095      GRID P40-4Q   | 88096    win7-xmpl-146048-3    |     26%    |
|      89170      GRID P40-4Q   | 89171    win7-xmpl-146048-4    |      0%    |
|      90475      GRID P40-4Q   | 90476    win7-xmpl-146048-5    |      0%    |
|      93363      GRID P40-4Q   | 93364    win7-xmpl-146048-6    |      0%    |
|   1  Tesla P40                | 00000000:85:00.0               |  0%       |
```

The vGPU utilization of vGPU 85109 is reported as 32%. For vGPU 87195, vGPU utilization is reported as 39%. And for 88095, it is reported as 26%. However, the expected vGPU utilization of any vGPU should not exceed approximately 16.7%.

This behavior is a result of the mechanism that is used to measure GPU engine utilization.

Status

Open

Ref. #

2227591
5.28. **nvidia-smi** reports that vGPU migration is supported on all hypervisors

**Description**

The command `nvidia-smi vgpu -m` shows that vGPU migration is supported on all hypervisors, even hypervisors or hypervisor versions that do not support vGPU migration.

**Status**

Closed

**Ref. #**

200407230

5.29. **GPU resources not available** error during VMware instant clone provisioning

**Description**

A **GPU resources not available** error might occur during VMware instant clone provisioning. On Windows VMs, a **Video TDR failure - NVLDDMKM.sys** error causes a blue screen crash.

This error occurs when options for VMware Virtual Shared Graphics Acceleration (vSGA) are set for a VM that is configured with NVIDIA vGPU. VMware vSGA is a feature of VMware vSphere that enables multiple virtual machines to share the physical GPUs on ESXi hosts and can be used as an alternative to NVIDIA vGPU.

Depending on the combination of options set, one of the following error messages is seen when the VM is powered on:

- Module 'MKS' power on failed.
  
  This message is seen when the following options are set:
  
  - Enable 3D support is selected.
  - 3D Renderer is set to Hardware
  - The graphics type of all GPUs on the ESXi host is Shared Direct.

- Hardware GPU resources are not available. The virtual machine will use software rendering.
Known Issues

This message is seen when the following options are set:

- Enable 3D support is selected.
- 3D Renderer is set to Automatic.
- The graphics type of all GPUs on the ESXi host is Shared Direct.

Resolution

If you want to use NVIDIA vGPU, unset any options for VMware vSGA that are set for the VM.

1. Ensure that the VM is powered off.
2. Open the vCenter Web UI.
3. In the vCenter Web UI, right-click the VM and choose Edit Settings.
4. Click the Virtual Hardware tab.
5. In the device list, expand the Video card node and de-select the Enable 3D support option.
6. Start the VM.

Status

Not a bug

Ref. #

2369683

5.30. VMs with 32 GB or more of RAM fail to boot with GPUs requiring 64 GB of MMIO space

Description

VMs with 32 GB or more of RAM fail to boot with GPUs that require 64 GB of MMIO space. VMs boot successfully with RAM allocations of less than 32 GB.

The following GPUs require 64 GB of MMIO space:

- Tesla P6
- Tesla P40

Version

This issue affects the following versions of VMware vSphere ESXi:

- 6.0 Update 3 and later updates
- 6.5 and later updates
Workaround

If you want to use a VM with 32 GB or more of RAM with GPUs that require 64 GB of MMIO space, use this workaround:

1. Create a VM to which less than 32 GB of RAM is allocated.
2. Choose VM Options > Advanced and set
   `pciPassthru.use64bitMMIO="TRUE"`.
3. Allocate the required amount of RAM to the VM.

For more information, see VMware Knowledge Base Article: VMware vSphere VMDirectPath I/O: Requirements for Platforms and Devices (2142307).

Status

Not an NVIDIA bug

Resolved in VMware vSphere ESXi 6.7

Ref. #

2043171

5.31. Module load failed during VIB downgrade from R390 to R384

Description

Some registry keys are available only with the R390 Virtual GPU Manager, for example, `NVreg_IgnoreMMIOCheck`. If any keys that are available only with the R390 Virtual GPU Manager are set, the NVIDIA module fails to load after a downgrade from R390 to R384.

When `nvidia-smi` is run without any arguments to verify the installation, the following error message is displayed:

```
NVIDIA-SMI has failed because it couldn't communicate with the NVIDIA driver.
Make sure that the latest NVIDIA driver is installed and running.
```

Workaround

Before uninstalling the R390 VIB, clear all parameters of the `nvidia` module to remove any registry keys that are available only for the R390 Virtual GPU Manager.

```
# esxcli system module parameters set -p "" -m nvidia
```
**Known Issues**

**Status**
Not an NVIDIA bug

**Ref. #**
200366884

### 5.32. Resolution is not updated after a VM acquires a license and is restarted

**Description**
In a Red Enterprise Linux 7.3 guest VM, an increase in resolution from 1024×768 to 2560×1600 is not applied after a license is acquired and the gridd service is restarted. This issue occurs if the multimonitor parameter is added to the xorg.conf file.

**Version**
Red Enterprise Linux 7.3

**Status**
Open

**Ref. #**
200275925

### 5.33. Tesla P40 cannot be used in pass-through mode

**Description**
Pass-through mode on Tesla P40 GPUs and other GPUs based on the Pascal architecture does not work as expected. In some situations, after the VM is powered on, the guest OS crashes or fails to boot.

**Workaround**
Ensure that your GPUs are configured as described in Requirements for Using GPUs Based on the Pascal and Volta Architectures in Pass-Through Mode
5.34. On Linux, 3D applications run slowly when windows are dragged

Description
When windows for 3D applications on Linux are dragged, the frame rate drops substantially and the application runs slowly.
This issue does not affect 2D applications.

5.35. A segmentation fault in DBus code causes nvidia-gridd to exit on Red Hat Enterprise Linux and CentOS

Description
On Red Hat Enterprise Linux 6.8 and 6.9, and CentOS 6.8 and 6.9, a segmentation fault in DBus code causes the nvidia-gridd service to exit.

The nvidia-gridd service uses DBus for communication with NVIDIA X Server Settings to display licensing information through the Manage License page. Disabling the GUI for licensing resolves this issue.

To prevent this issue, the GUI for licensing is disabled by default. You might encounter this issue if you have enabled the GUI for licensing and are using Red Hat Enterprise Linux 6.8 or 6.9, or CentOS 6.8 and 6.9.
Version
Red Hat Enterprise Linux 6.8 and 6.9
CentOS 6.8 and 6.9

Status
Open

Ref. #
- 200358191
- 200319854
- 1895945

5.36. No Manage License option available in NVIDIA X Server Settings by default

Description
By default, the Manage License option is not available in NVIDIA X Server Settings. This option is missing because the GUI for licensing on Linux is disabled by default to work around the issue that is described in A segmentation fault in DBus code causes nvidia-gridd to exit on Red Hat Enterprise Linux and CentOS.

Workaround
This workaround requires sudo privileges.

Do not use this workaround with Red Hat Enterprise Linux 6.8 and 6.9 or CentOS 6.8 and 6.9. To prevent a segmentation fault in DBus code from causing the nvidia-gridd service from exiting, the GUI for licensing must be disabled with these OS versions.

If you are licensing a physical GPU for vComputeServer, you must use the configuration file /etc/nvidia/gridd.conf.

1. If NVIDIA X Server Settings is running, shut it down.
2. If the /etc/nvidia/gridd.conf file does not already exist, create it by copying the supplied template file /etc/nvidia/gridd.conf.template.
3. As root, edit the /etc/nvidia/gridd.conf file to set the EnableUI option to TRUE.
4. Start the nvidia-gridd service.
When NVIDIA X Server Settings is restarted, the Manage License option is now available.

**Status**

Open

### 5.37. Licenses remain checked out when VMs are forcibly powered off

**Description**

NVIDIA vGPU software licenses remain checked out on the license server when non-persistent VMs are forcibly powered off.

The NVIDIA service running in a VM returns checked out licenses when the VM is shut down. In environments where non-persistent licensed VMs are not cleanly shut down, licenses on the license server can become exhausted. For example, this issue can occur in automated test environments where VMs are frequently changing and are not guaranteed to be cleanly shut down. The licenses from such VMs remain checked out against their MAC address for seven days before they time out and become available to other VMs.

**Resolution**

If VMs are routinely being powered off without clean shutdown in your environment, you can avoid this issue by shortening the license borrow period. To shorten the license borrow period, set the LicenseInterval configuration setting in your VM image. For details, refer to Virtual GPU Client Licensing User Guide.

**Status**

Closed

**Ref. #**

1694975
5.38. Memory exhaustion can occur with vGPU profiles that have 512 Mbytes or less of frame buffer

Description

Memory exhaustion can occur with vGPU profiles that have 512 Mbytes or less of frame buffer.

This issue typically occurs in the following situations:

- Full screen 1080p video content is playing in a browser. In this situation, the session hangs and session reconnection fails.
- Multiple display heads are used with Citrix Virtual Apps and Desktops or VMware Horizon on a Windows 10 guest VM.
- Higher resolution monitors are used.
- Applications that are frame-buffer intensive are used.
- NVENC is in use.

To reduce the possibility of memory exhaustion, NVENC is disabled on profiles that have 512 Mbytes or less of frame buffer.

When memory exhaustion occurs, the NVIDIA host driver reports Xid error 31 and Xid error 43 in the VMware vSphere log file `vmware.log` in the guest VM's storage directory.

The following vGPU profiles have 512 Mbytes or less of frame buffer:

- Tesla M6-0B, M6-0Q
- Tesla M10-0B, M10-0Q
- Tesla M60-0B, M60-0Q

The root cause is a known issue associated with changes to the way that recent Microsoft operating systems handle and allow access to overprovisioning messages and errors. If your systems are provisioned with enough frame buffer to support your use cases, you should not encounter these issues.

Workaround

- Use an appropriately sized vGPU to ensure that the frame buffer supplied to a VM through the vGPU is adequate for your workloads.
- Monitor your frame buffer usage.
- If you are using Windows 10, consider these workarounds and solutions:
  - Use a profile that has 1 Gbyte of frame buffer.
Optimize your Windows 10 resource usage.

To obtain information about best practices for improved user experience using Windows 10 in virtual environments, complete the NVIDIA GRID vGPU Profile Sizing Guide for Windows 10 download request form.

Additionally, you can use the VMware OS Optimization Tool to make and apply optimization recommendations for Windows 10 and other operating systems.

**Status**

Open

**Ref. #**

- 200130864
- 1803861

**5.39. vGPU VM fails to boot in ESXi 6.5 if the graphics type is Shared**

**Description**

If vSGA is being used, this issue shouldn't be encountered and changing the default graphics type is not necessary.

On VMware vSphere Hypervisor (ESXi) 6.5, after vGPU is configured, VMs to which a vGPU is assigned may fail to start and the following error message may be displayed:

>The amount of graphics resource available in the parent resource pool is insufficient for the operation.

The vGPU Manager VIB provides vSGA and vGPU functionality in a single VIB. After this VIB is installed, the default graphics type is Shared, which provides vSGA functionality. To enable vGPU support for VMs in VMware vSphere 6.5, you must change the default graphics type to Shared Direct. If you do not change the default graphics type you will encounter this issue.

**Version**

VMware vSphere Hypervisor (ESXi) 6.5

**Workaround**

Change the default graphics type to Shared Direct as explained in Virtual GPU Software User Guide.
5.40. ESXi 6.5 web client shows high memory usage even when VMs are idle

Description
On VMware vSphere Hypervisor (ESXi) 6.5, the web client shows a memory usage alarm with critical severity for VMs to which a vGPU is attached even when the VMs are idle. When memory usage is monitored from inside the VM, no memory usage alarm is shown. The web client does not show a memory usage alarm for the same VMs without an attached vGPU.

Version
VMware vSphere Hypervisor (ESXi) 6.5

Workaround
Avoid using the VMware vSphere Hypervisor (ESXi) 6.5 web client to monitor memory usage for VMs to which a vGPU is attached.

Status
Not an NVIDIA bug

Ref. #
200191065

5.41. VMs configured with NVIDIA vGPU must not be on a host in a VMware DRS cluster

Description
The ESXi host on which VMs configured with NVIDIA vGPU reside must not be a member of a VMware Distributed Resource Scheduler (DRS) cluster. The installer for the NVIDIA driver for NVIDIA vGPU software cannot locate the NVIDIA vGPU software
Known Issues

GPU card on a host in a VMware DRS Cluster. Any attempt to install the driver on a VM on a host in a DRS cluster fails with the following error:

```
NVIDIA Installer cannot continue
This graphics driver could not find compatible graphics hardware.
```

Furthermore, you **cannot** overcome this limitation by configuring a VM with NVIDIA vGPU and installing the driver on the VM on a host outside a DRS cluster and moving the host into the DRS cluster after configuring it.

**Workaround**

Move each VM configured with NVIDIA vGPU to a host outside the DRS cluster.

1. Remove NVIDIA Virtual GPU Manager from the host in the DRS cluster.
2. Create a cluster of VMware ESXi hosts outside the DRS domain.
3. Install the NVIDIA Virtual GPU Manager on an ESXi host in the cluster that you created in the previous step.
4. Create a vSphere VM for use with NVIDIA vGPU.
5. Configure the vSphere VM with NVIDIA vGPU.
6. Boot the vSphere VM and install the NVIDIA driver for NVIDIA vGPU.

For instructions for performing these tasks, refer to *Virtual GPU Software User Guide*.

**Status**

Open

**Ref. #**

1933449

5.42. GNOME Display Manager (GDM) fails to start on Red Hat Enterprise Linux 7.2 and CentOS 7.0

**Description**

GDM fails to start on Red Hat Enterprise Linux 7.2 and CentOS 7.0 with the following error:

```
Oh no! Something has gone wrong!
```

**Workaround**

Permanently enable permissive mode for Security Enhanced Linux (SELinux).

1. **As root**, edit the `/etc/selinux/config` file to set `SELINUX` to `permissive`.}

```
SELINUX=permissive
```
2. Reboot the system.

```bash
~]# reboot
```

For more information, see Permissive Mode in Red Hat Enterprise Linux 7 SELinux User’s and Administrator’s Guide.

**Status**

Not an NVIDIA bug

**Ref. #**

200167868

5.43. NVIDIA Control Panel fails to start and reports that “you are not currently using a display that is attached to an Nvidia GPU”

**Description**

When you launch NVIDIA Control Panel on a VM configured with vGPU, it fails to start and reports that you are not using a display attached to an NVIDIA GPU. This happens because Windows is using VMware’s SVGA device instead of NVIDIA vGPU.

**Fix**

Make NVIDIA vGPU the primary display adapter.

Use Windows screen resolution control panel to make the second display, identified as “2” and corresponding to NVIDIA vGPU, to be the active display and select the Show desktop only on 2 option. Click Apply to accept the configuration.

You may need to click on the Detect button for Windows to recognize the display connected to NVIDIA vGPU.

> If the VMware Horizon/View agent is installed in the VM, the NVIDIA GPU is automatically selected in preference to the SVGA device.

**Status**

Open
5.44. VM configured with more than one vGPU fails to initialize vGPU when booted

Description
Using the current VMware vCenter user interface, it is possible to configure a VM with more than one vGPU device. When booted, the VM boots in VMware SVGA mode and doesn’t load the NVIDIA driver. The additional vGPU devices are present in Windows Device Manager but display a warning sign, and the following device status:

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

Workaround
NVIDIA vGPU currently supports a single virtual GPU device per VM. Remove any additional vGPUs from the VM configuration before booting the VM.

Status
Open

5.45. A VM configured with both a vGPU and a passthrough GPU fails to start the passthrough GPU

Description
Using the current VMware vCenter user interface, it is possible to configure a VM with a vGPU device and a passthrough (direct path) GPU device. This is not a currently supported configuration for vGPU. The passthrough GPU appears in Windows Device Manager with a warning sign, and the following device status:

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

Workaround
Do not assign vGPU and passthrough GPUs to a VM simultaneously.
5.46. vGPU allocation policy fails when multiple VMs are started simultaneously

Description
If multiple VMs are started simultaneously, vSphere may not adhere to the placement policy currently in effect. For example, if the default placement policy (breadth-first) is in effect, and 4 physical GPUs are available with no resident vGPUs, then starting 4 VMs simultaneously should result in one vGPU on each GPU. In practice, more than one vGPU may end up resident on a GPU.

Workaround
Start VMs individually.

Status
Not an NVIDIA bug

Ref. #
200042690

5.47. Before Horizon agent is installed inside a VM, the Start menu’s sleep option is available

Description
When a VM is configured with a vGPU, the Sleep option remains available in the Windows Start menu. Sleep is not supported on vGPU and attempts to use it will lead to undefined behavior.

Workaround
Do not use Sleep with vGPU.

Installing the VMware Horizon agent will disable the Sleep option.
**Status**
Closed

**Ref. #**
200043405

### 5.48. vGPU-enabled VMs fail to start, `nvidia-smi` fails when VMs are configured with too high a proportion of the server’s memory.

**Description**
If vGPU-enabled VMs are assigned too high a proportion of the server’s total memory, the following errors occur:

- One or more of the VMs may fail to start with the following error:

  ```
  The available Memory resources in the parent resource pool are insufficient for the operation
  ```

- When run in the host shell, the `nvidia-smi` utility returns this error:

  ```
  -sh: can't fork
  ```

For example, on a server configured with 256G of memory, these errors may occur if vGPU-enabled VMs are assigned more than 243G of memory.

**Workaround**
Reduce the total amount of system memory assigned to the VMs.

**Status**
Closed

**Ref. #**
200060499
5.49. On reset or restart VMs fail to start with the error VMIOP: no graphics device is available for vGPU...

Description
On a system running a maximal configuration, that is, with the maximum number of vGPU VMs the server can support, some VMs might fail to start post a reset or restart operation.

Fix
Upgrade to ESXi 6.0 Update 1.

Status
Closed

Ref. #
200097546

5.50. nvidia-smi shows high GPU utilization for vGPU VMs with active Horizon sessions

Description
vGPU VMs with an active Horizon connection utilize a high percentage of the GPU on the ESXi host. The GPU utilization remains high for the duration of the Horizon session even if there are no active applications running on the VM.

Workaround
None

Status
Open

Partially resolved for Horizon 7.0.1:
- For Blast connections, GPU utilization is no longer high.
- For PCoIP connections, utilization remains high.
Ref. #
1735009
Notice

ALL NVIDIA DESIGN SPECIFICATIONS, REFERENCE BOARDS, FILES, DRAWINGS, DIAGNOSTICS, LISTS, AND OTHER DOCUMENTS (TOGETHER AND SEPARATELY, "MATERIALS") ARE BEING PROVIDED "AS IS." NVIDIA MAKES NO WARRANTIES, EXPRESSED, IMPLIED, STATUTORY, OR OTHERWISE WITH RESPECT TO THE MATERIALS, AND EXPRESSLY DISCLAIMS ALL IMPLIED WARRANTIES OF NONINFRINGEMENT, MERCHANTABILITY, AND FITNESS FOR A PARTICULAR PURPOSE.

Information furnished is believed to be accurate and reliable. However, NVIDIA Corporation assumes no responsibility for the consequences of use of such information or for any infringement of patents or other rights of third parties that may result from its use. No license is granted by implication of otherwise under any patent rights of NVIDIA Corporation. Specifications mentioned in this publication are subject to change without notice. This publication supersedes and replaces all other information previously supplied. NVIDIA Corporation products are not authorized as critical components in life support devices or systems without express written approval of NVIDIA Corporation.

HDMI

HDMI, the HDMI logo, and High-Definition Multimedia Interface are trademarks or registered trademarks of HDMI Licensing LLC.

OpenCL

OpenCL is a trademark of Apple Inc. used under license to the Khronos Group Inc.

Trademarks

NVIDIA, the NVIDIA logo, NVIDIA GRID, vGPU, Pascal, Quadro, and Tesla are trademarks or registered trademarks of NVIDIA Corporation in the U.S. and other countries. Other company and product names may be trademarks of the respective companies with which they are associated.

Copyright

© 2013-2019 NVIDIA Corporation. All rights reserved.