Virtual GPU Software R418 for Microsoft Windows Server

Release Notes

RN-08686-001 _v8.0 through 8.6 | January 2021

Table of Contents

Chapter 1. Release Notes 1
1.1. NVIDIA vGPU Software Driver Versions 1
1.2. Updates in Release 8.6 2
1.3. Updates in Release 8.5 2
1.4. Updates in Release 8.4 2
1.5. Updates in Release 8.3 3
1.6. Updates in Release 8.2 3
1.7. Updates in Release 8.1 3
1.8. Updates in Release 8.0 4
Chapter 2. Validated Platforms 5
2.1. Supported NVIDIA GPUs and Validated Server Platforms 5
2.2. Hypervisor Software Releases 6
2.3. Guest OS Support 6
2.3.1. Windows Guest OS Support 7
2.3.2. Linux Guest OS Support 7
Chapter 3. Security Updates 8
3.1. Restricting Access to GPU Performance Counters 8
3.1.1. Windows: Restricting Access to GPU Performance Counters for One User by Using NVIDIA Control Panel 8
3.1.2. Windows: Restricting Access to GPU Performance Counters Across an Enterprise by Using a Registry Key 9
3.1.3. Linux Guest VMs: Restricting Access to GPU Performance Counters 9
Chapter 4. Known Issues 11
4.1. NVIDIA Control Panel fails to start if launched too soon from a VM without licensing information 11
4.2. DWM crashes randomly occur in Windows VMs 11
4.3. Microsoft DDA fails with some GPUs 12
4.4. NVIDIA vGPU software graphics driver fails after Linux kernel upgrade with DKMS enabled 13
4.5. 8.0 Only: Incorrect NVIDIA vGPU software Windows graphics driver version in the installer 14
4.6. Frame capture while the interactive logon message is displayed returns blank screen 15
4.7. RDS sessions do not use the GPU with some Microsoft Windows Server releases 15
4.8. Resolution is not updated after a VM acquires a license and is restarted 16
4.9. A segmentation fault in DBus code causes nvidia-gridd to exit on Red Hat Enterprise Linux and CentOS 17

4.10. No Manage License option available in NVIDIA X Server Settings by default..... 17
4.11. Licenses remain checked out when VMs are forcibly powered off..... 18

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 Microsoft Windows Server.

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

1.1. NVIDIA vGPU Software Driver Versions

Each release in this release family of NVIDIA vGPU software includes a specific version of the NVIDIA Windows driver and NVIDIA Linux driver.

NVIDIA vGPU Software Version	NVIDIA Windows Driver Version	NVIDIA Linux Driver Version
8.6	427.11	418.181.07
8.5	426.94	418.165.01
8.4	426.72	418.149
8.3	426.52	418.130
8.2	426.26	418.109
8.1	426.04	418.92
8.0	425.31	418.70

Note:

All releases in this release family of NVIDIA vGPU software are compatible with **all** releases of the NVIDIA vGPU software license server.

1.2. Updates in Release 8.6

New Features in Release 8.6

- Security updates - see Security Bulletin: NVIDIA GPU Display Driver - January 2021
- Miscellaneous bug fixes

Feature Support Withdrawn in Release 8.6

- Red Hat Enterprise Linux 7.6 is no longer supported as a guest OS

1.3. Updates in Release 8.5

New Features in Release 8.5

- Security updates- see Security Bulletin: NVIDIA GPU Display Driver - September 2020
- Miscellaneous bug fixes

1.4. Updates in Release 8.4

New Features in Release 8.4

- Miscellaneous bug fixes
- Security updates - see Security Bulletin: NVIDIA GPU Display Driver - June 2020

Hardware and Software Support Introduced in Release 8.4

- Support for the following guest OS releases
  - Red Hat Enterprise Linux 8.2 and 7.8
  - CentOS 7.8

Feature Support Withdrawn in Release 8.4

- The following OS releases are no longer supported as a guest OS:
  - Red Hat Enterprise Linux 7.5
  - CentOS 7.5

1.5. Updates in Release 8.3

New Features in Release 8.3

- Miscellaneous bug fixes
- Security updates - see Security Bulletin: NVIDIA GPU Display Driver - February 2020

Hardware and Software Support Introduced in Release 8.3

- Support for the following guest OS releases
  - Red Hat Enterprise Linux 8.1
  - CentOS Linux 8 (1911)

Feature Support Withdrawn in Release 8.3

- The following OS releases are no longer supported as a guest OS:
  - Windows Server 2008 R2
  - Red Hat Enterprise Linux 7.0-7.4
  - CentOS 7.0-7.4

1.6. Updates in Release 8.2

New Features in Release 8.2

- Miscellaneous bug fixes
- Security updates - see Security Updates

1.7. Updates in Release 8.1

New Features in Release 8.1

- Security updates
- Miscellaneous bug fixes

Hardware and Software Support Introduced in Release 8.1

- Support for Red Hat Enterprise Linux 7.7 as a guest OS

1.8. Updates in Release 8.0

New Features in Release 8.0

- Security updates
- Miscellaneous bug fixes

Hardware and Software Support Introduced in Release 8.0

- Support for the following GPUs:
  - Quadro RTX 6000
  - Quadro RTX 8000
- Support for Windows Server 2019 with Hyper-V role
- Support for the following OS releases as a guest OS:
  - Windows 10 October 2018 Update (1809)
  - Windows Server 2019

Chapter 2. Validated Platforms

This release family of NVIDIA vGPU software provides support for several NVIDIA GPUs on validated server hardware platforms, Microsoft Windows Server hypervisor software versions, and guest operating systems.

2.1. Supported NVIDIA GPUs and Validated Server Platforms

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

- GPUs based on the NVIDIA Maxwell™ graphic architecture:
  - Tesla M6
  - Tesla M10
  - Tesla M60
- GPUs based on the NVIDIA Pascal™ architecture:
  - Tesla P4
  - Tesla P6
  - Tesla P40
  - Tesla P100 PCIe 16 GB
  - Tesla P100 SXM2 16 GB
  - Tesla P100 PCIe 12GB
- GPUs based on the NVIDIA Volta architecture:
  - Tesla V100 SXM2
  - Tesla V100 SXM2 32GB
  - Tesla V100 PCIe
  - Tesla V100 PCIe 32GB
  - Tesla V100 FHHL
- GPUs based on the NVIDIA Turing architecture:

- Tesla T4
- Quadro RTX 6000 in displayless mode
- Quadro RTX 8000 in displayless mode

In displayless mode, local physical display connectors are disabled.

Note: These GPUs are supported as a secondary device in a bare-metal deployment. Tesla M6 is also supported as the primary display device in a bare-metal deployment.

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

2.2. Hypervisor Software Releases

This release supports **only** the hypervisor software versions listed in the table.

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

Software	Version Supported
Microsoft Windows Server	Windows Server 2019 with Hyper-V role
	Windows Server 2016 1803 with Hyper-V role
	Windows Server 2016 1709 with Hyper-V role
	Windows Server 2016 1607 with Hyper-V role

2.3. Guest OS Support

NVIDIA vGPU software supports several Windows releases and Linux distributions as a guest OS using GPU pass-through.

Microsoft Windows Server with Hyper-V role supports GPU pass-through over Microsoft Virtual PCI bus. This bus is supported through paravirtualized drivers.

Note:

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 as a guest OS on Microsoft Windows Server.

> **Note:**
> If a specific release, even an update release, is not listed, it's **not** supported.

- Windows Server 2019
- Windows Server 2016 1607, 1709
- Windows Server 2012 R2 with patch `Windows8.1-KB3133690-x64.msu`
- Windows 10 October 2018 Update (1809) and all Windows 10 releases supported by Microsoft up to and including this release

2.3.2. Linux Guest OS Support

NVIDIA vGPU software supports **only** the 64-bit Linux distributions listed as a guest OS on Microsoft Windows Server.

> **Note:**
> If a specific release, even an update release, is not listed, it's **not** supported.

- **Since 8.4:** Red Hat Enterprise Linux 7.6-7.8
- **8.3 only:** Red Hat Enterprise Linux 7.5-7.7
- **8.1, 8.2 only:** Red Hat Enterprise Linux 7.0-7.7
- **8.0:** Red Hat Enterprise Linux 7.0-7.6
- **Since 8.4:** CentOS 7.6-7.8
- **8.3 only:** CentOS 7.5, 7.6
- **8.0-8.2 only:** CentOS 7.0-7.6
- Ubuntu 16.04 LTS
- SUSE Linux Enterprise Server 12 SP2

Chapter 3. Security Updates

3.1. Restricting Access to GPU Performance Counters

The NVIDIA graphics driver contains a vulnerability (CVE-2018-6260) that may allow access to application data processed on the GPU through a side channel exposed by the GPU performance counters. To address this vulnerability, update the driver and restrict access to GPU performance counters to allow access only by administrator users and users who need to use CUDA profiling tools.

The GPU performance counters that are affected by this vulnerability are the hardware performance monitors used by the CUDA profiling tools such as CUPTI, Nsight Graphics, and Nsight Compute. These performance counters are exposed on the hypervisor host and in guest VMs only as follows:

- On the hypervisor host, they are always exposed. However, the Virtual GPU Manager does not access these performance counters and, therefore, is not affected.
- In Windows and Linux guest VMs, they are exposed **only** in VMs configured for GPU pass through. They are not exposed in VMs configured for NVIDIA vGPU.

3.1.1. Windows: Restricting Access to GPU Performance Counters for One User by Using NVIDIA Control Panel

Perform this task from the guest VM to which the GPU is passed through.

Ensure that you are running **NVIDIA Control Panel** version 8.1.950.

1. Open **NVIDIA Control Panel**:
   - Right-click on the Windows desktop and select **NVIDIA Control Panel** from the menu.
   - Open **Windows Control Panel** and double-click the **NVIDIA Control Panel** icon.
2. In **NVIDIA Control Panel**, select the **Manage GPU Performance Counters** task in the **Developer** section of the navigation pane.

3. Complete the task by following the instructions in the **Manage GPU Performance Counters** > **Developer** topic in the **NVIDIA Control Panel** help.

3.1.2. Windows: Restricting Access to GPU Performance Counters Across an Enterprise by Using a Registry Key

You can use a registry key to restrict access to GPU Performance Counters for all users who log in to a Windows guest VM. By incorporating the registry key information into a script, you can automate the setting of this registry for all Windows guest VMs across your enterprise.

Perform this task from the guest VM to which the GPU is passed through.

> **CAUTION:** Only **enterprise administrators** should perform this task. Changes to the Windows registry must be made with care and system instability can result if registry keys are incorrectly set.

1. Set the `RmProfilingAdminOnly` Windows registry key to 1.

```
[HKLM\SYSTEM\CurrentControlSet\Services\nvlddmkm\Global\NVTweak]
Value: "RmProfilingAdminOnly"
Type: DWORD
Data: 00000001
```

   The data value 1 restricts access, and the data value 0 allows access, to application data processed on the GPU through a side channel exposed by the GPU performance counters.
2. Restart the VM.

3.1.3. Linux Guest VMs: Restricting Access to GPU Performance Counters

On systems where unprivileged users don't need to use GPU performance counters, restrict access to these counters to system administrators, namely users with the `CAP_SYS_ADMIN` capability set. By default, the GPU performance counters are not restricted to users with the `CAP_SYS_ADMIN` capability.

Perform this task from the guest VM to which the GPU is passed through.

This task requires `sudo` privileges.

1. Log in to the guest VM.
2. Set the kernel module parameter `NVreg_RestrictProfilingToAdminUsers` to 1 by adding this parameter to the `/etc/modprobe.d/nvidia.conf` file.
   - If you are setting only this parameter, add an entry for it to the `/etc/modprobe.d/nvidia.conf` file as follows:

```
options nvidia NVreg_RegistryDwords="NVreg_RestrictProfilingToAdminUsers=1"
```

- If you are setting multiple parameters, set them in a single entry as in the following example:

```
options nvidia NVreg_RegistryDwords="RmPVMRL=0x0"
 "NVreg_RestrictProfilingToAdminUsers=1"
```

  If the `/etc/modprobe.d/nvidia.conf` file does not already exist, create it.

3. Restart the VM.

Chapter 4. Known Issues

4.1. **NVIDIA Control Panel** fails to start if launched too soon from a VM without licensing information

Description

If NVIDIA licensing information is not configured on the system, any attempt to start **NVIDIA Control Panel** by right-clicking on the desktop within 30 seconds of the VM being started fails.

Workaround

Wait at least 30 seconds before trying to launch **NVIDIA Control Panel**.

Status

Open

Ref.

200623179

4.2. DWM crashes randomly occur in Windows VMs

Description

Desktop Windows Manager (DWM) crashes randomly occur in Windows VMs, causing a blue-screen crash and the bug check `CRITICAL_PROCESS_DIED`. Computer Management shows problems with the primary display device.

Version

This issue affects Windows 10 1809, 1903 and 1909 VMs.

Status

Not an NVIDIA bug

Ref.

2730037

4.3. Microsoft DDA fails with some GPUs

Description

Microsoft Discrete Device Assignment (DDA) fails with GPUs that have more than 16 GB of GPU memory. After the NVIDIA vGPU software graphics driver is installed in the guest VM, a second display device appears on the GPU and the driver prompts for a reboot. After the reboot, the device disappears and the Microsoft Hyper-V Video device appears.

This issue occurs because less memory-mapped input/output (MMIO) space is configured for the operating system than the device requires.

Workaround

Perform this workaround in a **Windows Power Shell** window on the hypervisor host.

Set the upper MMIO space to the amount that the device requires to allow all of the MMIO to be mapped. Upper MMIO space starts at approximately 64 GB in address space.

```
Set-VM –HighMemoryMappedIoSpace mmio-space –VMName vm-name
```

mmio-space

The amount of MMIO space that the device requires, appended with the appropriate unit of measurement, for example, **64GB** for 64 GB of MMIO space.

The required amount of MMIO space depends on the amount of BAR1 memory on the installed GPUs and the number of GPUs assigned to the VM as follows:

mmio-space = 2 # *gpu-bar1-memory* # *assigned-gpus*

gpu-bar1-memory

The amount of BAR1 memory on one of the installed GPUs. For example, in a server in which eight GPUs are installed and each GPU has 32 GB of BAR1 memory, *gpu-bar1-memory* is 32 GB.

assigned-gpus

The number of GPUs assigned to the VM.

vm-name

The name of the VM to which the GPU is assigned.

The following example sets the upper MMIO space to 64 GB for the VM named `mygpuvm`, to which one GPU with 32 GB of BAR1 memory is assigned.

```
Set-VM -HighMemoryMappedIoSpace 64GB -VMName mygpuvm
```

For more information, see Deploy graphics devices using Discrete Device Assignment on the Microsoft technical documentation site.

Status

Not an NVIDIA bug

Ref.

2812853

4.4. NVIDIA vGPU software graphics driver fails after Linux kernel upgrade with DKMS enabled

Description

After the Linux kernel is upgraded (for example by running `sudo apt full-upgrade`) with Dynamic Kernel Module Support (DKMS) enabled, the `nvidia-smi` command fails to run. If DKMS is enabled, an upgrade to the Linux kernel triggers a rebuild of the NVIDIA vGPU software graphics driver. The rebuild of the driver fails because the compiler version is incorrect. Any attempt to reinstall the driver fails because the kernel fails to build.

When the failure occurs, the following messages are displayed:

```
-> Installing DKMS kernel module:
        ERROR: Failed to run `/usr/sbin/dkms build -m nvidia -v  418.70 -k 5.3.0-28-
generic`:
        Kernel preparation unnecessary for this kernel. Skipping...
        Building module:
        cleaning build area...
        'make' -j8 NV_EXCLUDE_BUILD_MODULES='' KERNEL_UNAME=5.3.0-28-generic
 IGNORE_CC_MISMATCH='' modules...(bad exit status: 2)
        ERROR (dkms apport): binary package for nvidia:  418.70 not found
        Error! Bad return status for module build on kernel: 5.3.0-28-generic
 (x86_64)
        Consult /var/lib/dkms/nvidia/ 418.70/build/make.log for more information.
        -> error.
        ERROR: Failed to install the kernel module through DKMS. No kernel module
 was installed;
        please try installing again without DKMS, or check the DKMS logs for more
 information.
        ERROR: Installation has failed. Please see the file '/var/log/nvidia-
installer.log' for details.
```

```
       You may find suggestions on fixing installation problems in the README
available on the Linux driver download page at www.nvidia.com.
```

Workaround

When installing the NVIDIA vGPU software graphics driver with DKMS enabled, specify the `--no-cc-version-check` option.

Status

Not a bug.

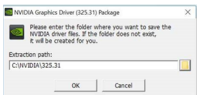
Ref.

2836271

4.5. 8.0 Only: Incorrect NVIDIA vGPU software Windows graphics driver version in the installer

Description

The NVIDIA vGPU software Windows graphics driver version in the installer is incorrect. The driver version incorrectly appears as 325.31 instead of 425.31 in the **Extraction path** field and the title of the **NVIDIA Graphics Driver (325.31) Package Window**.



Version

NVIDIA vGPU software Windows graphics driver version 425.31 in NVIDIA vGPU software release 8.0

Workaround

To simplify future administration of your system, you can correct the driver version in the folder name when the installer prompts you for the extraction path. However, if you do not change the name of the folder in the extraction path, the installation succeeds and the driver functions correctly.

Status

Resolved in NVIDIA vGPU software release 8.1

4.6. 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

4.7. 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 2019
- Windows Server 2016
- Windows Server 2012

Solution

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

1. Choose **Local Computer Policy** > **Computer Configuration** > **Administrative Templates** > **Windows Components** > **Remote Desktop Services** > **Remote Desktop Session Host** > **Remote Session Environment** .
2. Set the **Use the hardware default graphics adapter for all Remote Desktop Services sessions** option.

4.8. 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

4.9. 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

4.10. 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.

> **Note:** 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.

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.

```
# sudo service nvidia-gridd start
```

When **NVIDIA X Server Settings** is restarted, the **Manage License** option is now available.

Status

Open

4.11. 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

Notice

This document is provided for information purposes only and shall not be regarded as a warranty of a certain functionality, condition, or quality of a product. NVIDIA Corporation ("NVIDIA") makes no representations or warranties, expressed or implied, as to the accuracy or completeness of the information contained in this document and assumes no responsibility for any errors contained herein. NVIDIA shall have no liability for the consequences or use of such information or for any infringement of patents or other rights of third parties that may result from its use. This document is not a commitment to develop, release, or deliver any Material (defined below), code, or functionality.

NVIDIA reserves the right to make corrections, modifications, enhancements, improvements, and any other changes to this document, at any time without notice.

Customer should obtain the latest relevant information before placing orders and should verify that such information is current and complete.

NVIDIA products are sold subject to the NVIDIA standard terms and conditions of sale supplied at the time of order acknowledgement, unless otherwise agreed in an individual sales agreement signed by authorized representatives of NVIDIA and customer ("Terms of Sale"). NVIDIA hereby expressly objects to applying any customer general terms and conditions with regards to the purchase of the NVIDIA product referenced in this document. No contractual obligations are formed either directly or indirectly by this document.

NVIDIA products are not designed, authorized, or warranted to be suitable for use in medical, military, aircraft, space, or life support equipment, nor in applications where failure or malfunction of the NVIDIA product can reasonably be expected to result in personal injury, death, or property or environmental damage. NVIDIA accepts no liability for inclusion and/or use of NVIDIA products in such equipment or applications and therefore such inclusion and/or use is at customer's own risk.

NVIDIA makes no representation or warranty that products based on this document will be suitable for any specified use. Testing of all parameters of each product is not necessarily performed by NVIDIA. It is customer's sole responsibility to evaluate and determine the applicability of any information contained in this document, ensure the product is suitable and fit for the application planned by customer, and perform the necessary testing for the application in order to avoid a default of the application or the product. Weaknesses in customer's product designs may affect the quality and reliability of the NVIDIA product and may result in additional or different conditions and/or requirements beyond those contained in this document. NVIDIA accepts no liability related to any default, damage, costs, or problem which may be based on or attributable to: (i) the use of the NVIDIA product in any manner that is contrary to this document or (ii) customer product designs.

No license, either expressed or implied, is granted under any NVIDIA patent right, copyright, or other NVIDIA intellectual property right under this document. Information published by NVIDIA regarding third-party products or services does not constitute a license from NVIDIA to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property rights of the third party, or a license from NVIDIA under the patents or other intellectual property rights of NVIDIA.

Reproduction of information in this document is permissible only if approved in advance by NVIDIA in writing, reproduced without alteration and in full compliance with all applicable export laws and regulations, and accompanied by all associated conditions, limitations, and notices.

THIS DOCUMENT AND 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. TO THE EXTENT NOT PROHIBITED BY LAW, IN NO EVENT WILL NVIDIA BE LIABLE FOR ANY DAMAGES, INCLUDING WITHOUT LIMITATION ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL, PUNITIVE, OR CONSEQUENTIAL DAMAGES, HOWEVER CAUSED AND REGARDLESS OF THE THEORY OF LIABILITY, ARISING OUT OF ANY USE OF THIS DOCUMENT, EVEN IF NVIDIA HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. Notwithstanding any damages that customer might incur for any reason whatsoever, NVIDIA's aggregate and cumulative liability towards customer for the products described herein shall be limited in accordance with the Terms of Sale for the product.

VESA DisplayPort

DisplayPort and DisplayPort Compliance Logo, DisplayPort Compliance Logo for Dual-mode Sources, and DisplayPort Compliance Logo for Active Cables are trademarks owned by the Video Electronics Standards Association in the United States and other countries.

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, NVIDIA GRID vGPU, NVIDIA Maxwell, NVIDIA Pascal, NVIDIA Turing, NVIDIA Volta, GPUDirect, 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

NVIDIA Corporation | 2788 San Tomas Expressway, Santa Clara, CA 95051
http://www.nvidia.com