



NSIGHT COMPUTE

v2019.3.1 | May 2019

Release Notes



TABLE OF CONTENTS

Chapter 1. Release Notes	1
1.1. Updates in 2019.3.1.....	1
1.2. Updates in 2019.3.....	1
1.3. Updates in 2019.1.....	2
Chapter 2. Known Issues	4
Chapter 3. Support	5
3.1. Platform Support.....	5
3.2. GPU Support.....	6

LIST OF TABLES

Table 1	Platforms supported by NVIDIA Nsight Compute	5
Table 2	GPU architectures supported by NVIDIA Nsight Compute	6

Chapter 1.

RELEASE NOTES

1.1. Updates in 2019.3.1

NVIDIA Nsight Compute

- ▶ Added ability to send bug reports and suggestions for features using *Send Feedback* in the *Help* menu

Resolved Issues

- ▶ Fixed calculation of theoretical occupancy for grids with blocks that are not a multiple of 32 threads
- ▶ Fixed intercepting child processes launched through Python's `subprocess.Popen` class
- ▶ Fixed issue of NVTX push/pop ranges not showing up for child threads in NVIDIA Nsight Compute CLI
- ▶ Fixed performance regression for metric lookups on the Source page
- ▶ Fixed description in rule covering the IMC stall reason
- ▶ Fixed cases where baseline values were not correctly calculated in the Memory tables when comparing reports of different architectures
- ▶ Fixed incorrect calculation of baseline values in the Executed Instruction Mix chart
- ▶ Fixed accessing instanced metrics in the NvRules API
- ▶ Fixed a bug that could cause the collection of unnecessary metrics in the Interactive Profile activity
- ▶ Fixed potential crash on exit of the profiled target application
- ▶ Switched underlying metric for **SOL_FB** in the GPU Speed Of Light section to be driven by `dram__throughput.avg.pct_of_peak_sustained_elapsed` instead of `fbpa__throughput.avg.pct_of_peak_sustained_elapsed`

1.2. Updates in 2019.3

General

- ▶ Improved performance
- ▶ Bug fixes
- ▶ Kernel launch context and stream are reported as metrics
- ▶ PC sampling configuration options are reported as metrics
- ▶ The default base port for connections to the target changed
- ▶ Section files support multiple, named Body fields
- ▶ NvRules allows users to query metrics using any convertible data type

NVIDIA Nsight Compute

- ▶ Support for filtering kernel launches using their NVTX context
- ▶ Support for new options to select the connection port range
- ▶ The Profile activity supports configuring PC sampling parameters
- ▶ Sections on the Details page support selecting individual bodies

NVIDIA Nsight Compute CLI

- ▶ Support for stepping to kernel launches from specific NVTX contexts
- ▶ Support for new `--port` and `--max-connections` options
- ▶ Support for new `--sampling-*` options to configure PC sampling parameters
- ▶ Section file errors are reported with `--list-sections`
- ▶ A warning is shown if some section files could not be loaded

Resolved Issues

- ▶ Using the `--summary` option works for reports that include invalid metrics
- ▶ The full process executable filename is reported for QNX targets
- ▶ The project system now properly stores the state of opened reports
- ▶ Fixed PTX syntax highlighting
- ▶ Fixed an issue when switching between manual and auto profiling in NVIDIA Nsight Compute
- ▶ The source page in NVIDIA Nsight Compute now works with results from multiple processes
- ▶ Charts on the NVIDIA Nsight Compute details page uses proper localization for numbers
- ▶ NVIDIA Nsight Compute no longer requires the system locale to be set to English

1.3. Updates in 2019.1

General

- ▶ Support for CUDA 10.1
- ▶ Improved performance
- ▶ Bug fixes
- ▶ Profiling on Volta GPUs now uses the same metric names as on Turing GPUs
- ▶ Section files support descriptions
- ▶ The default sections and rules directory has been renamed to *sections*

NVIDIA Nsight Compute

- ▶ Added new profiling options to the options dialog
- ▶ Details page shows rule result icons in the section headers
- ▶ Section descriptions are shown in the details page and in the sections tool window
- ▶ Source page supports collapsing multiple source files or functions to show aggregated results
- ▶ Source page heatmap color scale has changed
- ▶ Invalid metric results are highlighted in the profiler report
- ▶ Loaded section and rule files can be opened from the sections tool window

NVIDIA Nsight Compute CLI

- ▶ Support for profiling child processes on Linux and Windows x86_64 targets
- ▶ NVIDIA Nsight Compute CLI uses a temporary file if no output file is specified
- ▶ Support for new **--quiet** option
- ▶ Support for setting the GPU clock control mode using new **--clock-control** option
- ▶ Details page output shows the NVTX context when **--nvtx** is enabled
- ▶ Support for filtering kernel launches for profiling based on their NVTX context using new **--nvtx-include** and **--nvtx-exclude** options
- ▶ Added new **--summary** options for aggregating profiling results
- ▶ Added option **--open-in-ui** to open reports collected with NVIDIA Nsight Compute CLI directly in NVIDIA Nsight Compute

Resolved Issues

- ▶ Installation directory scripts use absolute paths
- ▶ OpenACC kernel names are correctly demangled
- ▶ Profile activity report file supports a relative path
- ▶ Source view can resolve all applicable files at once
- ▶ UI font colors are improved
- ▶ Details page layout and label elision issues are resolved
- ▶ Turing metrics are properly reported on the Summary page
- ▶ All byte-based metrics use a factor of 1000 when scaling units to follow SI standards
- ▶ CSV exports properly align columns with empty entries
- ▶ Fixed the metric computation for `double_precision_fu_utilization` on GV11b
- ▶ Fixed incorrect 'selected' PC sampling counter values
- ▶ The SpeedOfLight section uses 'max' instead of 'avg' cycles metrics for Elapsed Cycles

Chapter 2.

KNOWN ISSUES

- ▶ The Visual Studio 2017 redistributable is not automatically installed by the NVIDIA Nsight Compute installer. The workaround is to install the x64 version of the 'Microsoft Visual C++ Redistributable for Visual Studio 2017' manually. The installer is linked on the main download page for Visual Studio at <https://www.visualstudio.com/downloads/> or download directly from <https://go.microsoft.com/fwlink/?LinkId=746572>.
- ▶ Launching applications on remote targets/platforms is not supported for several combinations. See [Platform Support](#) for details. Manually launch the application using command line `nv-nsight-cu-cli --mode=launch` on the remote system and connect using the UI or CLI afterwards.
- ▶ Real texture traffic is not captured in *First-Level Cache* table for Pascal chips.
- ▶ On platforms other than Windows, NVIDIA Nsight Compute must not be installed in a directory containing spaces or other whitespace characters.
- ▶ In the NVIDIA Nsight Compute connection dialog, a remote system can only be specified for one target platform. Remove a connection from its current target platform in order to be able to add it to another.
- ▶ The installer might not show all patch-level version numbers during installation.
- ▶ For GV100 GPUs, the Shared Memory Configuration Size (`launch__shared_mem_config_size`) might be reported incorrectly.
- ▶ Terminating a remote application profiled via Remote Launch might not work, but NVIDIA Nsight Compute only disconnects from the remote process.
- ▶ Reports collected on Windows might show invalid characters for file and process names when opened in NVIDIA Nsight Compute on Linux.
- ▶ Applications calling blocking functions on std input/output streams can result in the profiler to stop, until the blocking function call is resolved.
- ▶ The Block and Warp Durations histograms in the Launch Statistics section are unavailable for Volta and Turing architectures.
- ▶ The API Statistics filter in NVIDIA Nsight Compute does not support units.
- ▶ PerfWorks metrics on Volta and above that represent a constant value cannot be collected on their own. Selecting any non-constant PerfWorks metric for the same kernel launch resolves the issue.
- ▶ Profiling kernels executed on a device that is part of an SLI group is not supported.

Chapter 3.

SUPPORT

Information on supported platforms and GPUs.

3.1. Platform Support

Host denotes the UI can run on that platform. Target means that we can instrument applications on that platform for data collection. Applications launched with instrumentation on a target system can be connected to from most host platforms. The reports collected on one system can be opened on any other system.

Table 1 Platforms supported by NVIDIA Nsight Compute

	Host	Targets
Windows x86_64	Yes	Windows x86_64*, Linux x86_64
Linux x86_64	Yes	Windows x86_64*, Linux x86_64
Linux x86_64 (Drive SDK)	Yes	Windows x86_64*, Linux x86_64, DRIVE OS Linux, DRIVE OS QNX
MacOSX 10.13+	Yes	Windows x86_64*, Linux x86_64*
DRIVE OS Linux	No	DRIVE OS Linux
DRIVE OS QNX	No	DRIVE OS QNX

Target platforms marked with * do not support remote launch from the respective host. Remote launch means that the application can be launched on the target system from the host UI. Instead, the application must be launched from the target system.

On all Linux platforms, NVIDIA Nsight Compute requires GLIBC version 2.15 or higher.

3.2. GPU Support

Table 2 GPU architectures supported by NVIDIA Nsight Compute

Architecture	Support	Metrics*
Kepler	No	
Maxwell	No	
Pascal GP100	No	
Pascal GP10x	Yes	Group A
Volta GV100	Yes	Group B
Volta GV11b	Yes	Group B
Turing TU10x	Yes	Group B

* NVIDIA Nsight Compute uses different sets of metric names for the different GPU architectures. This is due to the underlying measurement libraries that are used on those architectures. Within each metric name group (Group A, Group B), names are identical, with the exception of some metrics being only available on some specific architectures. The metrics of Group B are identical to those of the [PerfWorks Metrics API](#). A comparison between the metrics used in nvprof and their equivalent in NVIDIA Nsight Compute can be found in the [NVIDIA Nsight Compute CLI User Manual](#).

When using the default sections and rules installed with NVIDIA Nsight Compute, the difference in metric names is handled automatically. When manually selecting metric names for profiling or writing your own sections or rules, the correct metric group must be picked for the respective target architecture.

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.

Trademarks

NVIDIA and the NVIDIA logo 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

© 2018-2019 NVIDIA Corporation. All rights reserved.

This product includes software developed by the Syncro Soft SRL (<http://www.sync.ro/>).