



NVIDIA CUDA TOOLKIT 9.2.88

RN-06722-001 _v9.2 | April 2018

Release Notes for Windows, Linux, and Mac OS



TABLE OF CONTENTS

Chapter 1. CUDA Toolkit Major Components.....	1
Chapter 2. Release Notes.....	3

Chapter 1.

CUDA TOOLKIT MAJOR COMPONENTS

This section provides an overview of the major components of the CUDA Toolkit and points to their locations after installation.

Compiler

The CUDA-C and CUDA-C++ compiler, **nvcc**, is found in the **bin/** directory. It is built on top of the NVVM optimizer, which is itself built on top of the LLVM compiler infrastructure. Developers who want to target NVVM directly can do so using the Compiler SDK, which is available in the **nvvm/** directory.

Please note that the following files are compiler-internal and subject to change without any prior notice.

- ▶ header files in **include/crt** and their corresponding wrapper files in **include/**
- ▶ stub files in **bin/crt**
- ▶ **nvvm/bin/cicc**
- ▶ **bin/cudafe++**, **bin/bin2c**, **bin/fatbinary**

Tools

The following development tools are available in the **bin/** directory (except for Nsight Visual Studio Edition (VSE) which is installed as a plug-in to Microsoft Visual Studio).

- ▶ IDEs: **nsight** (Linux, Mac), Nsight VSE (Windows)
- ▶ Debuggers: **cuda-memcheck**, **cuda-gdb** (Linux), Nsight VSE (Windows)
- ▶ Profilers: **nvprof**, **nvvp**, Nsight VSE (Windows)
- ▶ Utilities: **cuobjdump**, **nvdiasm**, **gpu-library-advisor**

Libraries

The scientific and utility libraries listed below are available in the **lib/** directory (DLLs on Windows are in **bin/**), and their interfaces are available in the **include/** directory.

- ▶ **cublas** (BLAS)
- ▶ **cublas_device** (BLAS Kernel Interface)
- ▶ **cuda_occupancy** (Kernel Occupancy Calculation [header file implementation])
- ▶ **cudadevrt** (CUDA Device Runtime)
- ▶ **cudart** (CUDA Runtime)

- ▶ **cufft** (Fast Fourier Transform [FFT])
- ▶ **cupti** (Profiling Tools Interface)
- ▶ **curand** (Random Number Generation)
- ▶ **cusolver** (Dense and Sparse Direct Linear Solvers and Eigen Solvers)
- ▶ **cusparse** (Sparse Matrix)
- ▶ **npp** (NVIDIA Performance Primitives [image and signal processing])
- ▶ **nvblas** ("Drop-in" BLAS)
- ▶ **nvcuvid** (CUDA Video Decoder [Windows, Linux])
- ▶ **nvgraph** (CUDA nvGRAPH [accelerated graph analytics])
- ▶ **nvml** (NVIDIA Management Library)
- ▶ **nVRTC** (CUDA Runtime Compilation)
- ▶ **nvtx** (NVIDIA Tools Extension)
- ▶ **thrust** (Parallel Algorithm Library [header file implementation])

CUDA Samples

Code samples that illustrate how to use various CUDA and library APIs are available in the **samples/** directory on Linux and Mac, and are installed to **C:\ProgramData\NVIDIA Corporation\CUDA Samples** on Windows. On Linux and Mac, the **samples/** directory is read-only and the samples must be copied to another location if they are to be modified. Further instructions can be found in the *Getting Started Guides* for Linux and Mac.

Documentation

The most current version of these release notes can be found online at <http://docs.nvidia.com/cuda/cuda-toolkit-release-notes/index.html>. Also, the **version.txt** file in the root directory of the toolkit will contain the version and build number of the installed toolkit.

Documentation can be found in PDF form in the **doc/pdf/** directory, or in HTML form at **doc/html/index.html** and online at <http://docs.nvidia.com/cuda/index.html>.

CUDA-GDB Sources

CUDA-GDB sources are available as follows:

- ▶ For CUDA Toolkit 7.0 and newer, in the installation directory **extras/**. The directory is created by default during the toolkit installation unless the **.rpm** or **.deb** package installer is used. In this case, the **cuda-gdb-src** package must be manually installed.
- ▶ For CUDA Toolkit 6.5, 6.0, and 5.5, at <https://github.com/NVIDIA/cuda-gdb>.
- ▶ For CUDA Toolkit 5.0 and earlier, at <ftp://download.nvidia.com/CUDAOpen64/>.
- ▶ Upon request by sending an e-mail to <mailto:oss-requests@nvidia.com>.

Chapter 2. RELEASE NOTES

The release notes for the CUDA Toolkit can be found online at <http://docs.nvidia.com/cuda/cuda-toolkit-release-notes/index.html>.

Acknowledgments

NVIDIA extends thanks to Professor Mike Giles of Oxford University for providing the initial code for the optimized version of the device implementation of the double-precision `exp()` function found in this release of the CUDA toolkit.

NVIDIA acknowledges Scott Gray for his work on small-tile GEMM kernels for Pascal. These kernels were originally developed for OpenAI and included since cuBLAS 8.0.61.2.

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

© 2007-2018 NVIDIA Corporation. All rights reserved.
www.nvidia.com

