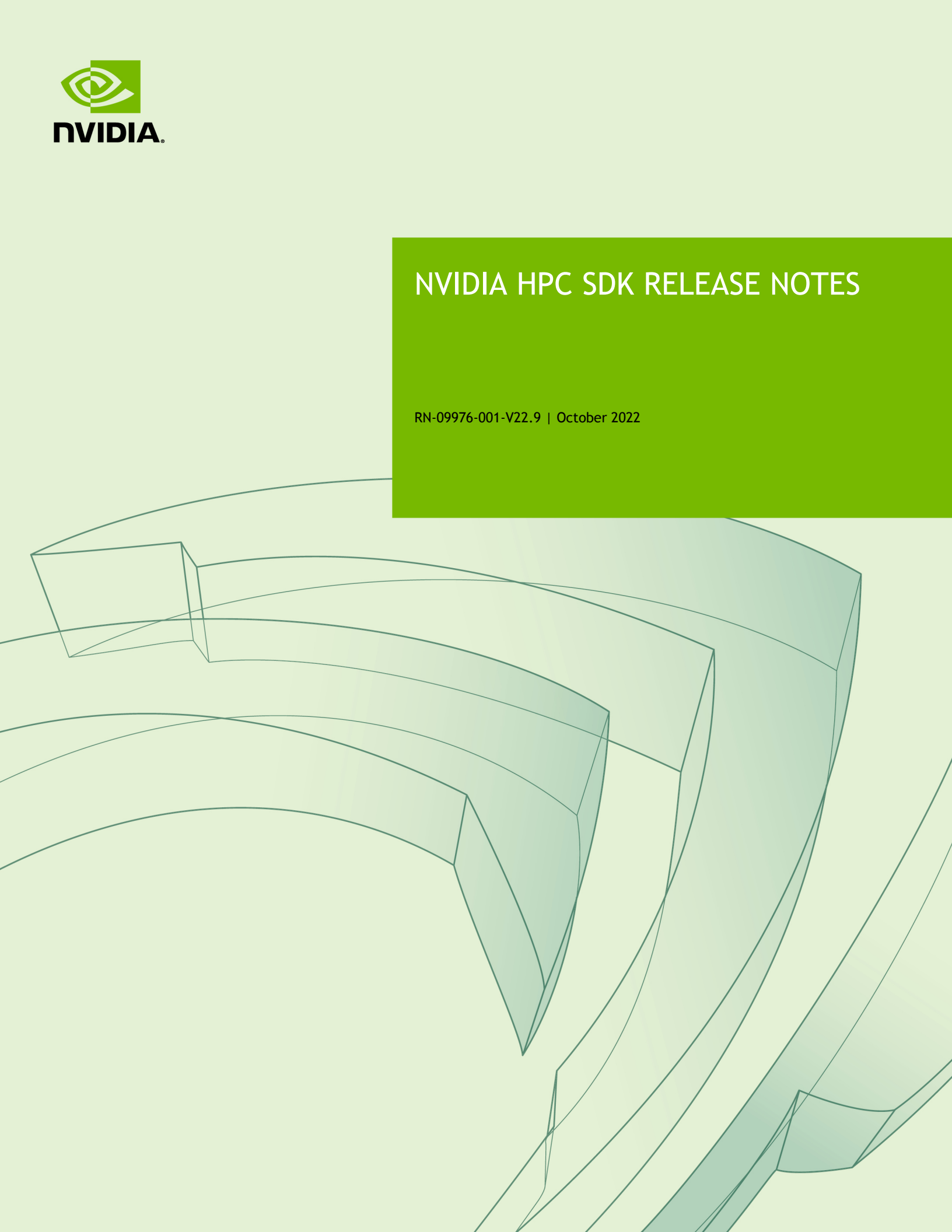

NVIDIA HPC SDK RELEASE NOTES

RN-09976-001-V22.9 | October 2022

TABLE OF CONTENTS

LIST OF TABLES

Chapter 1. WHAT'S NEW

Welcome to the 22.9 version of the NVIDIA HPC SDK, a comprehensive suite of compilers and libraries enabling developers to program the entire HPC platform, from the GPU foundation to the CPU and out through the interconnect. The 22.9 release of the HPC SDK is primarily a maintenance and bugfix release containing important functionality and performance improvements.

The HPC Compilers added support for PTX JIT compilation for relocatable device code mode which is the compilers' default when generating code for NVIDIA GPUs. For more information about using this feature, please refer to the HPC Compilers documentation.

The HPC SDK now offers signed packages for download. Be sure to refer to the Developer Zone download page for further instructions on using these packages.

Chapter 2. RELEASE COMPONENT VERSIONS

The NVIDIA HPC SDK 22.9 release contains the following versions of each component:

Table 1 HPC SDK Release Components

	Linux_x86_64			Linux_ppc64le			Linux_aarch64		
	CUDA 10.2	**CUDA 11.0**	**CUDA 11.7**	**CUDA 10.2**	**CUDA 11.0**	**CUDA 11.7**	**CUDA 10.2**	**CUDA 11.0**	**CUDA 11.7**
nvc++	22.9			22.9			22.9		
nvc	22.9			22.9			22.9		
nvfortran	22.9			22.9			22.9		
nvcc	10.2.89	11.0.221	11.7.99	10.2.89	11.0.221	11.7.99	N/A	11.0.221	11.7.99
NCCL	2.13.4	2.13.4	2.13.4	2.13.4	2.13.4	2.13.4	N/A	2.13.4	2.13.4
NVSHMEM	2.6.0	2.6.0	2.6.0	2.6.0	2.6.0	2.6.0	N/A	N/A	N/A
cuBLAS	10.2.2.89	11.2.0.252	11.10.3.66	10.2.2.89	11.2.0.252	11.10.3.66	N/A	11.2.0.252	11.10.3.66
cuFFT	10.1.2.89	10.2.1.245	10.7.2.91	10.1.2.89	10.2.1.245	10.7.2.91	N/A	10.2.1.245	10.7.2.91
cuFFTMp	N/A	N/A	10.8.1	N/A	N/A	10.8.1	N/A	N/A	N/A
cuRAND	10.1.2.89	10.2.1.245	10.2.10.91	10.1.2.89	10.2.1.245	10.2.10.91	N/A	10.2.1.245	10.2.10.91
cuSOLVER	10.3.0.89	10.6.0.245	11.4.0.1	10.3.0.89	10.6.0.245	11.4.0.1	N/A	10.6.0.245	11.4.0.1
cuSOLVER	N/A	N/A	0.2.1	N/A	N/A	N/A	N/A	N/A	N/A
cuSPARSE	10.3.1.89	11.1.1.245	11.7.4.91	10.3.1.89	11.1.1.245	11.7.4.91	N/A	11.1.1.245	11.7.4.91
cuTENSOR	1.6.0	1.6.0	1.6.0	1.6.0	1.6.0	1.6.0	N/A	1.6.0	1.6.0
Nsight Compute	2022.2.0			2022.2.0			2022.2.0		
Nsight Systems	2022.3.4.34			2022.3.4.34			2022.3.4.34		
OpenMPI	3.1.5			3.1.5			3.1.5		
HPC-X	N/A	2.12	2.12	N/A	N/A	N/A	N/A	2.12	2.12

	Linux_x86_64			Linux_ppc64le			Linux_aarch64		
	CUDA 10.2	CUDA 11.0	CUDA 11.7	CUDA 10.2	CUDA 11.0	CUDA 11.7	CUDA 10.2	CUDA 11.0	CUDA 11.7
UCX	N/A	1.14.0	1.14.0	N/A	N/A	N/A	N/A	1.14.0	1.14.0
OpenBLAS	0.3.20			0.3.20			0.3.20		
Scalapack	2.2.0			2.2.0			2.2.0		
Thrust	1.9.7	1.9.9	1.15.0	1.9.7	1.9.9	1.15.0	N/A	1.9.10	1.15.0
CUB	N/A	1.9.9	1.15.0	N/A	1.9.9	1.15.0	N/A	1.9.9	1.15.0
libcu++	N/A	1.0.0	1.8.0	N/A	1.0.0	1.8.0	N/A	1.0.0	1.8.0

Chapter 3. SUPPORTED PLATFORMS

3.1. Platform Requirements for the HPC SDK

Table 2 HPC SDK Platform Requirements

Architecture	Linux Distributions	Minimum gcc/ glibc Toolchain	Minimum CUDA Driver
x86_64	CentOS 7.2, 7.3, 7.4, 7.5, 7.6, 7.7, 7.8, 7.9 CentOS 8.0, 8.1, 8.2 Fedora 29, 30, 31, 32, 33, 34 OpenSUSE Leap 15.0, 15.1, 15.2 RHEL 7.0, 7.1, 7.2, 7.3, 7.4, 7.5, 7.6, 7.7, 7.8, 7.9 RHEL 8.0, 8.1, 8.4, 8.5, 8.6 SLES 12SP4, 12SP5, 15, 15SP1, 15SP2, 15SP3 Ubuntu 18.04, 20.04, 22.04 Rocky Linux 8.0	C99: 4.8 C11: 4.9 C++03: 4.8 C++11: 4.9 C++14: 5.1 C++17: 7.1 C++20: 10.1	440.33
ppc64le	RHEL 7.3, 7.4, 7.5, 7.6, 7.7, 8.0, 8.1, 8.3, 8.4 RHEL Pegas 7.5, 7.6 Ubuntu 18.04	C99: 4.8 C11: 4.9 C++03: 4.8 C++11: 4.9 C++14: 5.1 C++17: 7.1	440.33

Architecture	Linux Distributions	Minimum gcc/ glibc Toolchain	Minimum CUDA Driver
		C++20: 10.1	
aarch64	CentOS 8.0, 8.1, 8.2, 8.3, 8.4 RHEL 8.1, 8.2, 8.6 Ubuntu 18.04, 20.04, 22.04 SLES 15SP3 Amazon Linux 2	C99: 4.8 C11: 4.9 C++03: 4.8 C++11: 4.9 C++14: 5.1 C++17: 7.1 C++20: 10.1	450.36

Programs generated by the HPC Compilers for x86_64 processors require a minimum of AVX instructions, which includes Sandy Bridge and newer CPUs from Intel, as well as Bulldozer and newer CPUs from AMD. POWER 8 and POWER 9 CPUs from the POWER architecture are supported. For the Arm architecture, the minimum required version is Arm v8.1.

The HPC Compilers are compatible with gcc and g++ and use the GCC C and C++ libraries; the minimum compatible versions of GCC are listed in Table 2. The minimum system requirements for CUDA and NVIDIA Math Library requiremetns are available in the NVIDIA CUDA Toolkit documentation.

3.2. Supported CUDA Toolchain Versions

The NVIDIA HPC SDK uses elements of the CUDA toolchain when building programs for execution with NVIDIA GPUs. Every HPC SDK installation package puts the required CUDA components into an installation directory called `[install-prefix]/[arch]/[nvhpc-version]/cuda`.

An NVIDIA CUDA GPU device driver must be installed on a system with a GPU before you can run a program compiled for the GPU on that system. The NVIDIA HPC SDK does not contain CUDA Drivers. You must download and install the appropriate CUDA Driver from NVIDIA , including the CUDA Compatibility Platform if that is required.

The `nvaccelinfo` tool prints the CUDA Driver version in its output. You can use it to find out which version of the CUDA Driver is installed on your system.

The NVIDIA HPC SDK 22.9 includes the following CUDA toolchain versions:

- CUDA 10.2
- CUDA 11.0
- CUDA 11.7

The minimum required CUDA driver versions are listed in the table in Section 3.1.

Chapter 4. KNOWN LIMITATIONS

- Prior to using HPC-X, users should take care to source the hpcx-init.sh script: $. /[install-path]/Linux_x86_64/dev/comm_libs/hpcx/hpcx-2.11/hpcx-init.sh Then, run the hpcx_load function defined by this script: $ hpcx_load These actions will set important environment variables that are needed when running HPC-X. Also, if you see the following warning from HPC-X while running an MPI job: WARNING: Open MPI tried to bind a process but failed. This is a warning only; your job will continue, though performance may be degraded. This is a known issue, and may be worked around as follows: export OMPI_MCA_hwloc_base_binding_policy=""
- Derived type objects with zero-size derived type allocatable components that are used in sourced allocation or allocatable assignment may result in a runtime segmentation violation.
- When using -stdpar to accelerate C++ parallel algorithms, the algorithm calls cannot include virtual function calls or function calls through a function pointer, cannot use C++ exceptions, can only dereference pointers that point to the heap, and must use random access iterators (raw pointers as iterators work best).

Chapter 5. DEPRECATIONS AND CHANGES

- cudaDeviceSynchronize() in CUDA Fortran has been deprecated and will be removed in a future release.
- Starting with the 21.11 version of the NVIDIA HPC SDK, the HPC-X package is no longer shipped as part of the packages made available for the POWER architecture.
- Starting with the 21.5 version of the NVIDIA HPC SDK, the -cuda option for NVC++ and NVFORTRAN no longer automatically links the NVIDIA GPU math libraries. Please refer to the -cudalib option.
- HPC Compiler support for the Kepler architecture of NVIDIA GPUs was deprecated starting with the 21.3 version of the NVIDIA HPC SDK.
- Support for the KNL architecture of multicore CPUs in the NVIDIA HPC SDK was removed in the HPC SDK version 21.3.

Notice

Trademarks

Copyright