NVIDIA DGX H100/H200 Firmware Update Guide

NVIDIA Corporation

Aug 20, 2025

Contents

1 About Firmware Updates **1**
1.1 Firmware Updatable Components . . . 1
1.2 Firmware Update Prerequisites . . . 2
1.3 Firmware Update Methods . . . 2
1.4 Firmware Update Activation . . . 3

2 About the nvfwupd Command **5**
2.1 Requirements . . . 5
2.2 Syntax . . . 5
2.3 Understanding Background Processing . . . 7

3 DGX H100/H200 System Firmware Update Guide Version 25.06.4 **9**
3.1 Highlights . . . 9
3.1.1 Added Support . . . 9
3.1.2 BMC Fixes . . . 9
3.1.3 SBIOS Fixes . . . 9
3.1.4 The nvfwupd Command Updates . . . 9
3.2 Firmware Package Details . . . 9
3.3 Firmware Update Procedure . . . 12

4 Firmware Changes for NVIDIA DGX H100/H200 Systems **13**
4.1 BMC Changes for DGX H100/H200 Systems . . . 13
4.1.1 Changes in 25.06.27 . . . 13
4.1.2 Changes in 25.02.12 . . . 13
4.1.3 Changes in 24.09.17 . . . 13
4.1.4 Changes in 24.01.05 . . . 14
4.1.5 Changes in 23.09.20 . . . 14
4.2 SBIOS Changes for DGX H100/H200 Systems . . . 15
4.2.1 Changes in v1.6.7 . . . 15
4.2.2 Changes in v1.05.03 . . . 15
4.2.3 Changes in v1.01.03 . . . 15
4.2.4 Changes in v1.01.01 . . . 15
4.3 nvfwupd Command Changes . . . 15
4.3.1 Changes in 2.0.5 . . . 15
4.3.2 Changes in v2.0.4 . . . 16
4.3.3 Changes in v2.0.1 . . . 16
4.3.4 Changes in v2.0.0 . . . 16
4.3.5 Changes in v1.1.3 . . . 16
4.3.6 Changes in v1.1.1 . . . 16

5 Firmware Update Steps **17**
5.1 Before You Begin . . . 17
5.2 Update Duration . . . 17
5.3 Update Steps . . . 17

5.3.1 Firmware Update Output of mlxfwmanager . . . 21

6 Known Issues **27**
6.1 Functional Issues . . . 27
6.2 Redfish Service on BMC Might Experience Intermittent Unresponsiveness . . . 27
6.2.1 Issue . . . 27
6.2.2 Workaround . . . 27
6.3 USB1 Port Missing on Occasion after a BMC Cold Reset . . . 29
6.3.1 Issue . . . 29
6.3.2 Explanation . . . 29
6.3.3 Workaround . . . 29
6.4 Misleading Messages During Firmware Update . . . 29
6.4.1 Issue . . . 29
6.4.2 Workaround . . . 30
6.5 Sensors Endpoint for the Redfish API Does Not Support $expand . . . 30
6.5.1 Issue . . . 30
6.5.2 Workaround . . . 30
6.6 Firmware Upgrade or Downgrade Can Fail . . . 30
6.6.1 Issue . . . 30
6.6.2 Workaround . . . 30
6.7 Firmware Inventory Can Be Invalid During Boot . . . 30
6.7.1 Issue . . . 31
6.7.2 Workaround . . . 31
6.8 BMC Slow Startup After AC Power Cycle . . . 31
6.8.1 Issue . . . 31
6.8.2 Workaround . . . 31
6.9 Temperature Sensors Can Report No Reading . . . 31
6.9.1 Issue . . . 31
6.9.2 Workaround . . . 32

7 Resolved Issues **33**
7.1 Issues with ConnectX-7 Network (Cluster) Card Firmware . . . 33
7.1.1 Issue . . . 33
7.1.2 Resolution . . . 33
7.2 Platform DGX H200 Not Supported . . . 33
7.2.1 Issue . . . 34
7.2.2 Explanation . . . 34
7.2.3 Status . . . 34
7.3 The ipmitool dcmi power reading Command Returns 0 Power Reading Value . . . 34
7.3.1 Issue . . . 34
7.3.2 Status . . . 34
7.4 GPUs Show Exclamation Mark in BMC Web Interface . . . 34
7.4.1 Issue . . . 35
7.4.2 Explanation . . . 35
7.4.3 Status . . . 35
7.5 BMC LDAP Fields Do Not Support Space or Slash Characters . . . 35
7.5.1 Issue . . . 35
7.5.2 Status . . . 35
7.6 NVMe Information Not Visible in BCM Web Interface . . . 35
7.6.1 Issue . . . 35
7.6.2 Status . . . 35

8 Getting Started with nvdebug **37**
8.1 Requirements . . . 38

8.2 The nvdebug Command-Line Interface . . . 38
8.2.1 Syntax . . . 39
8.2.2 The Configuration Files . . . 40
8.3 HGX H100/H200 8-GPU Example . . . 40
8.4 DGX H100/H200 Example . . . 42
8.5 DGX Platform Example . . . 44
8.6 Redfish Collectors . . . 46
8.7 IPv6 Configuration . . . 46

9 Viewing the Installed Firmware and Package Versions 47

10 Updating the BMC 51

11 Firmware Update of Motherboard Tray: All Components 53

12 Updating BMC EROT on the Motherboard Tray 55

13 Updating SBIOS EROT on the Motherboard Tray 57

14 Updating the BIOS on the Motherboard Tray 59

15 Updating the CPLDs on the Motherboard Tray 61

16 Updating the Midplane CPLDs on the Motherboard Tray 63

17 Updating the NVMe Firmware 65

18 Updating the Power Supply Units on the Motherboard Tray 67

19 Updating the PCIe Switches on the Motherboard Tray 69

20 Updating the PCIe Retimers on the Motherboard Tray 71

21 Updating the ConnectX-7 Firmware 73

22 Updating the Intel NIC Firmware 75

23 Firmware Update of GPU Tray: All Components 77

24 Updating Multiple Systems 79
24.1 About Updating Multiple Systems . . . 79
24.2 Procedure . . . 79

25 Performing a BMC Factory Reset 81

26 Forcing a Firmware Downgrade 83
26.1 Prerequisites . . . 83
26.2 Procedure . . . 83

27 Troubleshooting an Unsuccessful Firmware Update 85
27.1 Firmware Update Terminates due to Component Not Found . . . 85
27.2 No Devices Where Detected for Handle ID 0 . . . 85
27.3 Wait for Firmware Update Started ID . . . 86

28 Version 25.04.3 87
28.1 Highlights . . . 87
28.1.1 BMC Fixes . . . 87

28.1.2 SBIOS Fixes . . . 87
28.1.3 The nvfwupd Command Updates . . . 87
28.2 Firmware Package Details . . . 88
28.3 Firmware Update Procedure . . . 90

29 Version 24.09.1 **91**
29.1 Highlights . . . 91
29.1.1 Added Support . . . 91
29.1.2 BMC Fixes . . . 91
29.1.3 SBIOS Fixes . . . 92
29.1.4 The nvfwupd Command Updates . . . 92
29.2 Firmware Package Details . . . 92
29.3 Firmware Update Procedure . . . 95

30 Version 1.1.3 **97**
30.1 Highlights . . . 97
30.2 BMC Fixes . . . 97
30.3 Firmware Package Details . . . 98
30.4 Firmware Update Procedure . . . 100

31 Version 1.1.1 **101**
31.1 Features . . . 101
31.2 BMC Fixes . . . 101
31.3 SBIOS Fixes . . . 102
31.4 Firmware Package Details . . . 102
31.5 Firmware Update Procedure . . . 104

32 Version 1.0.0 **105**
32.1 Firmware Package Details . . . 105

33 Third-Party License Notices **107**
33.1 Micron msecli . . . 107
33.2 Mellanox (OFED) . . . 108

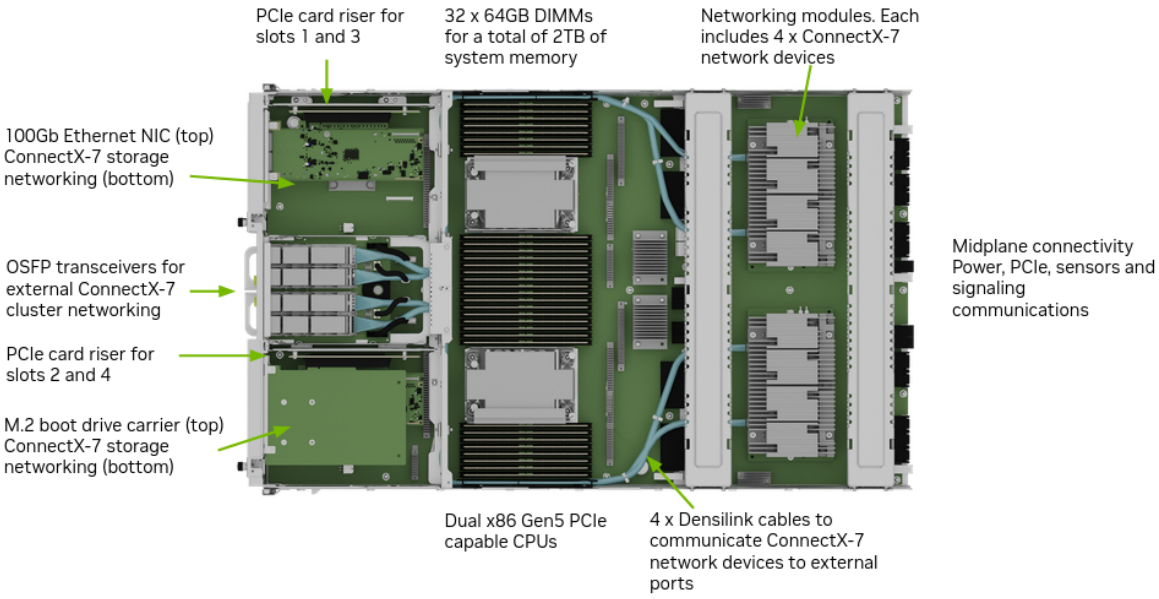
34 Notices **109**
34.1 Notice . . . 109
34.2 Trademarks . . . 110

Chapter 1. About Firmware Updates

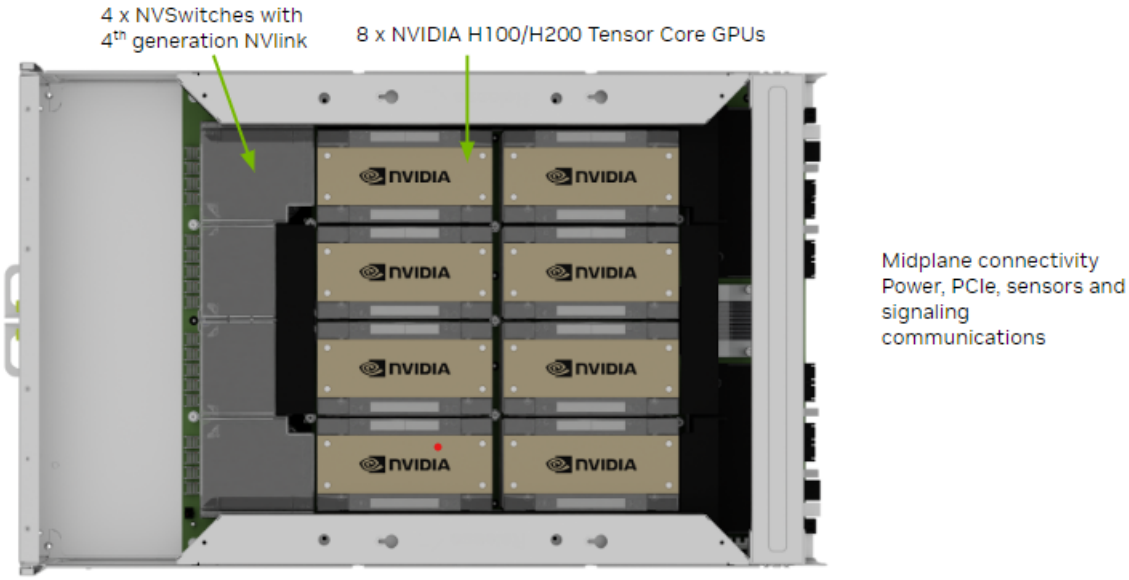
1.1. Firmware Updatable Components

The NVIDIA DGX™ H100/H200 System has several firmware updatable components. Some of the components are on the following two trays in the system:

- The motherboard tray has components, such as the CPUs, PCH, BMC as shown in the following figure:



- The GPU tray has components, such as the GPUs, NVSwitches, HMC as shown in the following figure:



You can update the firmware on the NVIDIA DGX H100/H200 System components out-of-band (OOB) by using Redfish APIs or from the host operating system by using the command-line interface (CLI) commands.

1.2. Firmware Update Prerequisites

- You can download firmware packages from the NVIDIA Enterprise Support Portal at https://enterprise-support.nvidia.com/s/.
- You must know the BMC IP address, a user name, and a password. The sample commands in this document show `admin` for both the user name and the password.
- You must have the `nvfwupd` executable or know how to use the Redfish API.

1.3. Firmware Update Methods

Most of the sample commands in this document show how to update firmware by using the `nvfwupd` command. You can download the executable from the NVIDIA Enterprise Support Portal. Refer to *About the nvfwupd Command* for more information about the command.

You can run the `nvfwupd` command interactively to update systems. Most command examples in this document show this interactive approach. If you have several systems to update, you can create a JSON file that identifies the systems to update. Refer to *Updating Multiple Systems* for more information.

An alternative to the `nvfwupd` command is to update firmware by using the Redfish API. The BMC network interface provides remote management with Redfish APIs.

The *Known Issues* for updating firmware and the *firmware update steps* still apply when you use the Redfish API.

Refer to Redfish APIs Support in the *NVIDIA DGX H100/H200 System User Guide* for more information and sample commands. The sample commands show how to update firmware with the `curl` command.

1.4. Firmware Update Activation

After the firmware update, you must perform one or more of the following tasks to activate the firmware update, depending on the components being updated:

- BMC component

  Reset the BMC by running the following command:

  ```
  sudo ipmitool mc reset cold
  ```

- PCIe Switch, PCIe Retimer, BIOS, and HGX (GPU Tray) components

  Perform a cold reset on the system using the following command:

  ```
  sudo ipmitool chassis power cycle
  ```

- EROT and CPLD components

  Perform an AC power cycle on the system by unplugging all the power supplies and then reconnecting them either manually or through an external PDU device.

  Note

  The AC power cycle will activate firmware for all updated components.

Chapter 2. About the nvfwupd Command

2.1. Requirements

The `nvfwupd` executable runs on the Linux operating system and is available for `x86_64` or `arm64` architecture machines. You can run the `x86_64` executable locally on the DGX system or use a remote Linux system.

If you run `nvfwupd` locally on the DGX system, instead of connecting to the BMC IP address, as shown in the sample commands, you can connect to the host Redfish interface IP address.

To download the latest version of the `nvfwupd` executable, log in to the NVIDIA Enterprise Support Portal.

2.2. Syntax

```
nvfwupd version 2.0.5

Usage: nvfwupd [ global options ] <command>

Global options:
    -t --target ip=<BMC IP> user=<BMC login id> password=<BMC password>
↪servertype=<Type of server>
          BMC target comprising BMC IP address and BMC login credentials.
↪The servertype sub-option is optional.
          Valid value for servertype is one of [DGX, HGX, MGX, GH200, NVOS,
↪HGXB100, GB200, MGX-NVL, GB200Switch].

    -c --config Path for config file (optional).
          Configure tool behavior.

    -v --verbose Chosen path for logfile (optional). Default path is current
↪working directory.
          Increase verbosity.

Commands:
     help       Show tool help.
```

(continues on next page)

(continued from previous page)

```
    version    Show tool version.

    show_pkg_content [ options... ]
        -p  --package        PLDM firmware package.

    unpack [ options... ]
        -p  --package        PLDM firmware package.
        -o  --outdir         Directory path to save unpacked firmware files
↪(optional).
                             Default path is current working directory of
↪tool.

    <Global options...> show_version [ options... ]
        -p  --package                PLDM firmware package.
        -j  --json                   Show output in JSON.

    <Global options...> update_fw [ options... ]
        -p  --package                PLDM firmware package.
        -y  --yes                    Bypass firmware update confirmation
↪prompt.
        -b  --background             Exit without waiting for the update
↪process to finish.
        -t  --timeout                API request timeout value in seconds.
        -s  --special                Special Update json file.
        -d  --details                Show update progress in table format.
        -j  --json                   Show output in JSON. Must be paired
↪with the -b
                                     background option, and always bypasses
↪update confirmation prompt.

    <Global options...> activate_fw [ options... ]
        -c  --cmd                    Activation command name.
                                     List of supported commands ['PWR_STATUS
↪', 'PWR_OFF', 'PWR_ON',
                                     'PWR_CYCLE', 'RESET_COLD', 'RESET_WARM',
↪ 'NVUE_PWR_CYCLE',
                                     'RF_AUX_PWR_CYCLE'].

    <Global options...> force_update [ options... ]
        enable|disable|status        Enable, disable, or check current force
↪update value on target.
        -j  --json                   Show output in JSON.

    <Global options...> show_update_progress [ options... ]
        -i  --id                     List of Task IDs delimited by space.
        -j  --json                   Show output in JSON.

    <Global options...> perform_factory_reset

    <Global options...> install_license

    <Global options...> make_upd_targets [ options... ]
```

(continues on next page)

(continued from previous page)

```
        -o  --outdir                    Directory path to create update target
↪files (optional).
                                        Default path is current working
↪directory of tool.
```

2.3. Understanding Background Processing

By default, the `nvfwupd update_fw` command communicates with the Redfish API, prints the firmware update progress to the console, and then exits when the command runs to completion.

Alternatively, you can specify the `--background` or `-b` argument so that the command communicates with the Redfish API, prints the task ID to the console, and then exits. In this case, you can periodically run the `nvfwupd show_update_progress` command with the task ID to monitor the update progress.

1. Create a component-specific `updparameters.json` file.
2. Update the firmware in the background:

```
nvfwupd -t ip=<bmc-ip-address> user=<bmc-user-id> password=<bmc-password>
↪update_fw \
  -p <firmware-package-file> -y -s updparameters.json -b
```

Example output:

```
FW recipe: ['<firmware-package-file>']
{"@odata.type": "#UpdateService.v1_6_0.UpdateService", ...}
FW update started, Task Id: 1
```

Note

If the `Platform dgxhxxx not supported` error appears, choose either of the following methods to resolve the unidentified platform issue:

- Update the `nvfwupd` tool to version 2.0.4 or later for automatic DGX platform detection.
- Specify the `servertype=DGX` sub-option with the `--target` option and try again. For example,

```
nvfwupd -t ip=<bmc-ip-address> user=<bmc-user-id> password=<bmc-
↪password> \
servertype=DGX update_fw -p <firmware-package-file> -y -s
↪parameters.json
```

3. Display the update progress for the task:

```
nvfwupd -t ip=<bmc-ip-address> user=<bmc-user-id> password=<bmc-password>
↪show_update_progress -i 1
```

Example output:

```
{'id': ['1']}
Task Info for Id: 1
 StartTime: 2023-03-11T01:10:01-0000
 TaskState: Running
 PercentComplete: 58
 TaskStatus: OK
 EndTime: 2023-03-11T01:10:00+00:00
 TaskStatus: Task /redfish/v1/UpdateService/upload is running normally.
```

Eventually, the percent complete field reports `100` and the task state field reports `Completed`.

4. To activate the firmware update, refer to *Firmware Update Activation* for more information.

Chapter 3. DGX H100/H200 System Firmware Update Guide Version 25.06.4

3.1. Highlights

3.1.1. Added Support

- Provides an updated GPU tray package, as shown in *Firmware Package Details*.

3.1.2. BMC Fixes

- Resolved the issue of high fan speed in zone 1 during system idle.
- Fixed an issue that intermittently fetched an invalid HMC configuration path.
- Resolved intermittent BMC Redfish unresponsiveness.

3.1.3. SBIOS Fixes

- No SBIOS fixes for this release.

3.1.4. The nvfwupd Command Updates

- No updates are available.

3.2. Firmware Package Details

This firmware release supports the following systems:

- NVIDIA DGX H100
- NVIDIA DGX H200

This firmware release supports the following operating systems:

- NVIDIA DGX OS 7.1.0, 7.0.2, 6.2.1, 6.1, 6.0.11, and higher
- NVIDIA DGX Software EL9-24.12, EL9-24.06, EL9-23.12, and EL9-23.08
- NVIDIA DGX Software EL8-24.07, EL8-24.01, and EL8-23.08

For more information about the operating systems, refer to the NVIDIA Base OS documentation.

You can download firmware packages from the NVIDIA Enterprise Support Portal.

The following table shows the firmware package files:

Components	Sample File Name
Combined archive	`DGXH100_H200_25.06.4.tar.gz` The combined archive includes the firmware for the system components and the firmware for the GPU tray.
▶ Motherboard tray package ▶ GPU tray package	▶ `nvfw_DGX_250629.1.0.fwpkg` ▶ `nvfw_HGX_DGXH100-H200x8_250319.1.0.fwpkg`

If you are updating from version 25.04.3, the total update time is approximately

- 93 minutes for the CPU tray using sequential updating.
- 36 minutes for the CPU tray using parallel updating.
- 12.5 minutes for the GPU tray using parallel updating.

The following table shows the information about component firmware versions and update time breakdown.

Component	Version	Update Time from 25.04.3 (Minutes)
Host BMC	**25.06.27** Refer to *BMC Changes for DGX H100/H200 Systems* for the list of changes.	32
Host BMC ERoT	04.0058	0.75
SBIOS ERoT	04.0058	0.75
SBIOS	1.6.7 Refer to *SBIOS Changes for DGX H100/H200 Systems* for the list of changes.	8
Motherboard CPLD	0.2.1.9	18.4
Midplane CPLD	0.2.1.3	13.4
PSU (Delta ECD16020137)	Primary 0204 Secondary 0201 Community 0204	PSU_0: 2.2 PSU_1: 2.2 PSU_2: 2.8 PSU_3: 2.2 PSU_4: 2.2 PSU_5: 2.2
Broadcom Gen5 PCIe Switch (PEX89072-B01)	Switch 0: 0.0.7 Switch 1: 1.0.7	Switch 0: 0.75 Switch 1: 0.75
Astera Labs Gen5 PCIe Retimer (PT5161L)	2.07.19	Retimer 0: 2.1 Retimer 1: 2.1
Network (Cluster) Card - ConnectX-7	28.43.2026	
Network (Storage) Card - ConnectX-7	28.43.2026	
Network Card - BlueField-3	32.43.2024	
VBIOS	▶ H100: **96.00.D0.00.02** ▶ H200: **96.00.D0.00.03**	GPU Tray (total): 12.5
NVSwitch (GPU Tray)	96.10.6D.00.01	
ERoT (GPU Tray)	00.02.0192	
HMC (GPU Tray)	**HGX-22.10-1-rc80**	
FPGA (GPU Tray)	2.53	

3.3. Firmware Update Procedure

Refer to *Firmware Update Steps*.

Chapter 4. Firmware Changes for NVIDIA DGX H100/H200 Systems

4.1. BMC Changes for DGX H100/H200 Systems

4.1.1. Changes in 25.06.27

- Resolved the issue of high fan speed in zone 1 during system idle.
- Fixed an issue that intermittently fetched an invalid HMC configuration path.
- Resolved intermittent BMC Redfish unresponsiveness.

4.1.2. Changes in 25.02.12

- Improved handling for firmware update of PSUs.
- Improved reporting on ERoT status.
- Provided multiple stability fixes.
- Improved user management in Redfish.
- Enhanced stability in communication between the BMC and the HMC.
- Corrected power supply sensors where no readings were shown.
- Fixed the inability to query GPU firmware version via Redfish.
- Addressed an issue where some telemetry information from the GPU tray was unavailable.
- Resolved an issue related to chassis power cycling using Redfish.
- Corrected fan readings so the fan will reflect 0 if it is not present or stops spinning.
- Fixed an issue where NIC information was allocated to the incorrect device in Redfish.

4.1.3. Changes in 24.09.17

- Fixed where BMC configuration might reset after upgrading.
- Added Redfish API support for creating, modifying, and deleting power policies.

- Support for deploying firmware update using the Web UI.
- Redfish Disable Host Interface: keeps redfish functional from BIOS to BMC but prevents the direct path from OS to BMC.
- Added ability to specify intermediate certificate authorities in a provisioned certificate chain.
- Included additional Redfish metrics reports.
- Fixed SNMP, syslog, and rsyslog issues.
- Added per BMC AES key for encrypting user/password files during the configuration save and restore process.
- Fixed invalid domain issues in the LDAP/AD settings.
- Enhanced Redfish diagnostics.
- General performance improvements in Redfish APIs and IPMI.
- Added support for ConnectX-7 temperature sensors.
- Improved resolution for energy counters.
- Enhanced Remote Media with support for port numbers and domain names.
- General improvements to the Web UI.

4.1.4. Changes in 24.01.05

- Fixed where SEL logs might fill up for NVMe drives
- Fixed low occurrence where HMC might not be visible in the BMC after BMC reboot
- Ability to control IPMI visibility for Host (Allow All, Limited Command, Hide)
- Higher resolution for CPU and GPU energy telemetry via Redfish
- Improved reliability of Redfish inventory
- Improved overall stability of telemetry collection and handling invalid/missing values
- General improvements to WebUI

4.1.5. Changes in 23.09.20

- WebUI enhancements
- Enabled GPU Info in WebUI
- Enabled NVRAM clear via Redfish
- Disabled RMCP / MD5 Auth Support after factory reset
- Enabled EROT background copy
- Enabled default SNMPv3 MIB
- The BMC update includes software security enhancements. Refer to the NVIDIA DGX H100 - August 2023 Security Bulletin for details.

4.2. SBIOS Changes for DGX H100/H200 Systems

4.2.1. Changes in v1.6.7

- Fixed an issue where the total memory size from the `free -m` command output is smaller than installed.
- Updated to display full device path for HDD devices under the Redfish BootOptions page.
- Fixed an issue to mandate administrator password after version 24.09.1 if SED drive is encrypted.

4.2.2. Changes in v1.05.03

- DIMM that experienced uncorrectable errors at runtime will be mapped out on the next boot.
- Exposed the `C1AutoDemotion`, `C1AutoUnDemotion`, and `C6Enable` setup options.
- Moved the CPU setup options page to under the Advanced page in the setup UI.
- Added a setup option to restrict host access via IPMI.
- Provided the `NvramVarsProtectionInOs` setup option to prevent the OS from changing the NVRAM at runtime.
- Implemented uncorrectable error rate limiting, disabled CSMI (correctable system management interrupts) on error flooding and on the core that reported MLC (middle-level cache) yellow state, and SEL logging when ANF (advisory non-fatal error) threshold was crossed.
- Changed the SncEn default setting to `disable`.

4.2.3. Changes in v1.01.03

- Added support for securing KCS

4.2.4. Changes in v1.01.01

- Fixed Boot options labeling for NIC ports
- Fix for U.2 bay slot numbering
- Set RestoreROWritePerf option to expert mode only
- Expose TDX and IFS options in expert user mode only

4.3. nvfwupd Command Changes

4.3.1. Changes in 2.0.5

- Added support for parallel firmware updates through the YAML configuration file.

- Added the `--json` option to the `update_fw`, `show_update_progress`, and `force_update` commands.
- Added IPv6 support.
- Deprecated the `targets` sub-option for multi-target input. Use `config.yaml` input instead.

4.3.2. Changes in v2.0.4

- Enhanced automatic server type detection for DGX platforms.
- No support for the `activate_fw` command.

4.3.3. Changes in v2.0.1

- Improved log sanitization to mask the IP address and login credentials by default.
- Added support for the `--target` and `--package` override from the command-line interface (CLI) using a configuration file.
- Enhanced the `--target` option with the `servertype` sub-option to resolve unidentified platform errors.

4.3.4. Changes in v2.0.0

- Support for using a platform-agnostic configuration file.

4.3.5. Changes in v1.1.3

- Support for abbreviated firmware update package names.
- Enhanced the `show_update_progress` output to provide a full status report for Redfish.
- Support for custom log file path.
- The command exits with an error code 1 for any update failure or tool failure.

4.3.6. Changes in v1.1.1

- You can update all the system components on the motherboard tray at one time. Previously, you had to update the components individually.
- You can create a JSON file with network addresses and credentials for multiple systems and automatically update multiple systems serially. Refer to *Updating Multiple Systems* for more information.

Chapter 5. Firmware Update Steps

5.1. Before You Begin

- Stop all unnecessary system activity before you begin the firmware update.
- Stop all GPU activity, including running the `nvidia-smi` command. GPU activity and running the command can prevent the VBIOS update.
- Do not add additional loads on the system, such as user jobs, diagnostics, or monitoring services, while an update is in progress. A high workload can disrupt the firmware update process and result in an unusable component.
- When you begin the firmware update, the update software assists in determining the activity state of the DGX system and provides a warning if it detects that activity levels are above a predetermined threshold. If you encounter the warning, take action to reduce the workload before proceeding with the firmware update.
- Fan speeds can increase during the BMC firmware update. This increase in speed is a normal part of the BMC firmware update process.
- If you plan to upgrade from version 1.0.0 (BMC 23.05.11) or 1.1.1 (BMC 23.09.20) to version 24.09.1 (BMC 24.09.17), you must first upgrade to version 1.1.3 (BMC 24.01.05) and then to version 24.09.1 to include all critical security changes. If you upgrade directly from version 1.0.0 or 1.1.1 to version 24.09.1, you are required to perform a factory reset to restore the default settings.

5.2. Update Duration

Updating the firmware on the motherboard tray components and the GPU tray components requires approximately 90 minutes. Updating the firmware on the ConnectX-7 devices requires approximately 30 minutes.

5.3. Update Steps

1. View the installed versions compared with the newly available firmware.

```
nvfwupd --target ip=<bmc-ip-address> user=<bmc-username> password=<bmc-
↪password> \
  show_version -p nvfw_DGX_250629.1.0.fwpkg \
  nvfw_HGX_DGXH100-H200x8_250319.1.0.fwpkg
```

2. Update the BMC.
   1. Create a file, such as `update_bmc.json`, with the following contents:

```
{
    "Targets" :["/redfish/v1/UpdateService/FirmwareInventory/HostBMC_0
↪"]
}
```

   2. Run the following command to update the BMC:

```
nvfwupd -t ip=<bmc-ip-address> user=<bmc-username> password=<bmc-
↪password> update_fw \
  -p nvfw_DGX_250629.1.0.fwpkg -y -s update_bmc.json
```

3. Reboot the BMC.

```
$ ipmitool mc reset cold
```

   Wait a couple of minutes and then confirm the BMC is back online.

   - Use the system shell:

```
$ ipmitool mc info
```

   - Alternatively, you can access the Web UI through a browser.

4. Update the components on the motherboard tray.

   For a one-shot firmware update, the BMC will perform a firmware update on all components in the provided bundle, for example, `nvfw_DGX_xxxxxx.x.x.fwpkg`, which includes the Host BMC (if the `force_update` option is specified), Host BIOS, EROT, PCIe Retimer, PCIe Switch, PSU, Motherboard CPLD, and Midplane CPLD.

   Ensure the system is powered on before updating the firmware.

   1. Create a file, such as `mb_tray.json`, with empty braces:

```
{}
```

   2. Update the firmware:

```
nvfwupd -t ip=<bmc-ip-address> user=<bmc-username> password=<bmc-
↪password> update_fw \
  -p nvfw_DGX_250629.1.0.fwpkg -y -s mb_tray.json
```

5. Update the components on the GPU tray.
   1. Create a `gpu_tray.json` file with the following contents:

```
{
    "Targets" :["/redfish/v1/UpdateService/FirmwareInventory/HGX_0"]
}
```

   2. Update the firmware:

```
nvfwupd -t ip=<bmc-ip-address> user=<bmc-username> password=<bmc-
↪password> update_fw \
  -p nvfw_HGX_DGXH100-H200x8_250319.1.0.fwpkg -y -s gpu_tray.json
```

This step performs parallel updates on all the components contained in the GPU tray, such as VBIOS, NVSwitch, EROTs, and FPGA.

3. Verify that the background copy has been completed successfully by looking for `"BackgroundCopyStatus": "Completed"` in the following command output:

```
curl -s -k -u <bmc-user>:<bmc-password> -H content-type:application/
↪json \
    -X GET https://<bmc-ip-address>/redfish/v1/Chassis/HGX_ERoT_BMC_
↪0 | jq
```

4. Perform a cold reset to restart the system:

```
ipmitool chassis power cycle
```

6. Confirm the firmware update is complete by viewing the installed versions again.

   After the system is operational again, repeat the following command to confirm all firmware has been updated:

```
nvfwupd --target ip=<bmc-ip-address> user=<bmc-username> password=<bmc-
↪password> \
  show_version -p nvfw_DGX_250629.1.0.fwpkg \
  nvfw_HGX_DGXH100-H200x8_250319.1.0.fwpkg
```

7. Execute background copy commands for the BMC and the system BIOS.

   1. BMC:

      Background copy Redfish API request:

```
curl -k -u <bmc-user>:<password> --request POST --location 'https://
↪<bmc-ip-address>/redfish/v1/UpdateService/Actions/Oem/
↪NvidiaUpdateService.CommitImage' \
    --header 'Content-Type: application/json' \
    --data '{
            "Targets": ["/redfish/v1/UpdateService/FirmwareInventory/
↪HostBMC_0"]
           }'
```

      Example response:

```
{
   "@odata.type":"#UpdateService.v1_11_0.UpdateService",
   "Messages":[
      {
         "@odata.type":"#Message.v1_0_8.Message",
         "Message":"A new task /redfish/v1/TaskService/Tasks/1 was
↪created.",
         "MessageArgs":[
            "/redfish/v1/TaskService/Tasks/1"
         ],
         "MessageId":"Task.1.0.New",
         "Resolution":"None",
         "Severity":"OK"
      },
```

(continues on next page)

(continued from previous page)

```
        {
            "@odata.type":"#Message.v1_0_8.Message",
            "Message":"ActivateFirmware Action is initiated.",
            "MessageId":"UpdateService.1.0.StartActivateFirmware",
            "Resolution":"None",
            "Severity":"OK"
        }
    ]
}
```

Query the update status using the task ID, which is 1, as shown in the output response:

```
nvfwupd -t ip=<bmc-ip-address> user=<bmc-username> password=<bmc-
↪password> show_update_progress -i 1
```

When the status indicates 100% complete, proceed with the next step.

2. SBIOS:

   Background copy Redfish API request:

```
curl -k -u <bmc-user>:<password> --request POST --location 'https://
↪<bmc-ip-address>/redfish/v1/UpdateService/Actions/Oem/
↪NvidiaUpdateService.CommitImage' \
    --header 'Content-Type: application/json' \
    --data '{
            "Targets": ["/redfish/v1/UpdateService/FirmwareInventory/
↪HostBIOS_0"]
            }'
```

   Find the task ID from the response, which is usually 2, to query the update status:

```
nvfwupd -t ip=<bmc-ip-address> user=<bmc-username> password=<bmc-
↪password> show_update_progress -i 2
```

   When the status indicates 100% complete, proceed with the next step.

3. Perform an AC power cycle on the system by unplugging all power supplies for five to six minutes to allow power drain, then reconnecting them manually or through an external PDU.

   Before performing the AC power cycle, ensure the chassis is powered off from the BMC.

8. Update the firmware on the network cards and NVMe drives.

Note

During the update, the `mlxfwmanager` command will report the ConnectX-7 device identified as `/dev/mst/mt4129_pciconf0` cannot be updated as shown in the following error message:

```
-E- Failed to query /dev/mst/mt4129_pciconf0 device, error : MFE_ICMD_
↪BAD_PARAM
```

This behavior is expected because this device is not one of the networking cards used to cluster the system, but a bridge device used internally and updated using a separate process.

1. To update the ConnectX®-7 cards and NVIDIA® BlueField®-3 cards, navigate to the `network` directory and run the `mlxfwmanager` command:

   ```
   cd network
   sudo mlxfwmanager -u -D .
   ```

   When prompted to update all 10 ConnectX-7 cards and BlueField-3 cards, type Y to confirm.

   For a complete output, refer to *Firmware Update Output of mlxfwmanager*.

2. For firmware update on the Intel E810-C Ethernet Network Adapters, refer to *Updating the Intel NIC Firmware*.

3. For firmware update on the NVMe drives, refer to *Updating the NVMe Firmware*.

5.3.1. Firmware Update Output of mlxfwmanager

```
$ sudo mlxfwmanager -u -D .
Querying Mellanox devices firmware ...

Device #1:
----------

Device Type:      ConnectX7
Part Number:      --
Description:
PSID:
PCI Device Name:  /dev/mst/mt4129_pciconf0
Base GUID:        N/A
Base MAC:         N/A
Versions:         Current        Available
   FW:            --
Status:           Failed to open device

Device #2:
----------

Device Type:      ConnectX7
Part Number:      MCX750500B-0D00_Ax_Bx
Description:      Nvidia adapter card with four ConnectX-7; each up to 400Gb/
↪s IB (default mode) or 400GbE; PCIe 5.0 x32; PCIe switch; crypto disabled;
↪secure boot enabled
PSID:             MT_0000000891
PCI Device Name:  /dev/mst/mt4129_pciconf1
Base GUID:        b8e924030081db74
Versions:         Current        Available
   FW:            28.42.1000     28.43.2026
   PXE:           3.7.0500       N/A
   UEFI:          14.35.0015     N/A
Status:           Update required

Device #3:
----------
```

(continues on next page)

(continued from previous page)

```
Device Type:      ConnectX7
Part Number:      MCX750500B-0D00_Ax_Bx
Description:      Nvidia adapter card with four ConnectX-7; each up to 400Gb/
 ↪s IB (default mode) or 400GbE; PCIe 5.0 x32; PCIe switch; crypto disabled;
 ↪secure boot enabled
PSID:             MT_0000000891
PCI Device Name:  /dev/mst/mt4129_pciconf2
Base GUID:        b8e924030081db7c
Versions:         Current        Available
   FW:            28.42.1000     28.43.2026
   PXE:           3.7.0500       N/A
   UEFI:          14.35.0015     N/A
Status:           Update required

Device #4:
----------

Device Type:      ConnectX7
Part Number:      MCX750500B-0D00_Ax_Bx
Description:      Nvidia adapter card with four ConnectX-7; each up to 400Gb/
 ↪s IB (default mode) or 400GbE; PCIe 5.0 x32; PCIe switch; crypto disabled;
 ↪secure boot enabled
PSID:             MT_0000000891
PCI Device Name:  /dev/mst/mt4129_pciconf3
Base GUID:        b8e924030081db78
Versions:         Current        Available
   FW:            28.42.1000     28.43.2026
   PXE:           3.7.0500       N/A
   UEFI:          14.35.0015     N/A
Status:           Update required

Device #5:
----------

Device Type:      ConnectX7
Part Number:      MCX750500B-0D00_Ax_Bx
Description:      Nvidia adapter card with four ConnectX-7; each up to 400Gb/
 ↪s IB (default mode) or 400GbE; PCIe 5.0 x32; PCIe switch; crypto disabled;
 ↪secure boot enabled
PSID:             MT_0000000891
PCI Device Name:  /dev/mst/mt4129_pciconf4
Base GUID:        b8e924030081db70
Versions:         Current        Available
   FW:            28.42.1000     28.43.2026
   PXE:           3.7.0500       N/A
   UEFI:          14.35.0015     N/A
Status:           Update required

Device #6:
----------

Device Type:      ConnectX7
```

(continues on next page)

(continued from previous page)

```
Part Number:      MCX750500B-0D00_Ax_Bx
Description:      Nvidia adapter card with four ConnectX-7; each up to 400Gb/
↪s IB (default mode) or 400GbE; PCIe 5.0 x32; PCIe switch; crypto disabled;
↪secure boot enabled
PSID:             MT_0000000891
PCI Device Name:  /dev/mst/mt4129_pciconf5
Base GUID:        b8e924030081d954
Versions:         Current        Available
   FW:            28.42.1000     28.43.2026
   PXE:           3.7.0500       N/A
   UEFI:          14.35.0015     N/A
Status:           Update required

Device #7:
----------

Device Type:      ConnectX7
Part Number:      MCX750500B-0D00_Ax_Bx
Description:      Nvidia adapter card with four ConnectX-7; each up to 400Gb/
↪s IB (default mode) or 400GbE; PCIe 5.0 x32; PCIe switch; crypto disabled;
↪secure boot enabled
PSID:             MT_0000000891
PCI Device Name:  /dev/mst/mt4129_pciconf6
Base GUID:        b8e924030081d95c
Versions:         Current        Available
   FW:            28.42.1000     28.43.2026
   PXE:           3.7.0500       N/A
   UEFI:          14.35.0015     N/A
Status:           Update required

Device #8:
----------

Device Type:      ConnectX7
Part Number:      MCX750500B-0D00_Ax_Bx
Description:      Nvidia adapter card with four ConnectX-7; each up to 400Gb/
↪s IB (default mode) or 400GbE; PCIe 5.0 x32; PCIe switch; crypto disabled;
↪secure boot enabled
PSID:             MT_0000000891
PCI Device Name:  /dev/mst/mt4129_pciconf7
Base GUID:        b8e924030081d958
Versions:         Current        Available
   FW:            28.42.1000     28.43.2026
   PXE:           3.7.0500       N/A
   UEFI:          14.35.0015     N/A
Status:           Update required

Device #9:
----------

Device Type:      ConnectX7
Part Number:      MCX750500B-0D00_Ax_Bx
```

(continues on next page)

(continued from previous page)

```
Description:      Nvidia adapter card with four ConnectX-7; each up to 400Gb/
 ↪s IB (default mode) or 400GbE; PCIe 5.0 x32; PCIe switch; crypto disabled;
 ↪secure boot enabled
PSID:             MT_0000000891
PCI Device Name:  /dev/mst/mt4129_pciconf8
Base GUID:        b8e924030081d950
Versions:         Current        Available
   FW:            28.42.1000     28.43.2026
   PXE:           3.7.0500       N/A
   UEFI:          14.35.0015     N/A
Status:           Update required

Device #10:
----------

Device Type:      BlueField3
Part Number:      900-9D3B6-00CN-A_Ax
Description:      NVIDIA BlueField-3 B3240 P-Series Dual-slot FHHL DPU;
 ↪400GbE / NDR IB (default mode); Dual-port QSFP112; PCIe Gen5.0 x16 with x16
 ↪PCIe extension option; 16 Arm cores; 32GB on-board DDR; integrated BMC;
 ↪Crypto Enabled
PSID:             MT_0000000883
PCI Device Name:  /dev/mst/mt41692_pciconf0
Base GUID:        b8e9240300a65e18
Base MAC:         b8e924a65e18
Versions:         Current        Available
   FW:            32.42.1000     32.43.2024
   PXE:           3.7.0500       N/A
   UEFI:          14.35.0015     N/A
   UEFI Virtio blk:   22.4.0013      N/A
   UEFI Virtio net:   21.4.0013      N/A
Status:           Update required

Device #11:
----------

Device Type:      BlueField3
Part Number:      900-9D3B6-00CN-A_Ax
Description:      NVIDIA BlueField-3 B3240 P-Series Dual-slot FHHL DPU;
 ↪400GbE / NDR IB (default mode); Dual-port QSFP112; PCIe Gen5.0 x16 with x16
 ↪PCIe extension option; 16 Arm cores; 32GB on-board DDR; integrated BMC;
 ↪Crypto Enabled
PSID:             MT_0000000883
PCI Device Name:  /dev/mst/mt41692_pciconf1
Base GUID:        b8e9240300a667be
Base MAC:         b8e924a667be
Versions:         Current        Available
   FW:            32.42.1000     32.43.2024
   PXE:           3.7.0500       N/A
   UEFI:          14.35.0015     N/A
   UEFI Virtio blk:   22.4.0013      N/A
   UEFI Virtio net:   21.4.0013      N/A
```

(continues on next page)

(continued from previous page)

```
Status:           Update required


---------
-E- Failed to query /dev/mst/mt4129_pciconf0 device, error : MFE_ICMD_BAD_
↪PARAM
Found 10 device(s) requiring firmware update...

Perform FW update? [y/N]: y
Device #1: Device query failed
Device #2: Updating FW ...
FSMST_INITIALIZE -   OK
Writing Boot image component -   OK
Done
Device #3: Updating FW ...
FSMST_INITIALIZE -   OK
Writing Boot image component -   OK
Done
Device #4: Updating FW ...
FSMST_INITIALIZE -   OK
Writing Boot image component -   OK
Done
Device #5: Updating FW ...
FSMST_INITIALIZE -   OK
Writing Boot image component -   OK
Done
Device #6: Updating FW ...
FSMST_INITIALIZE -   OK
Writing Boot image component -   OK
Done
Device #7: Updating FW ...
FSMST_INITIALIZE -   OK
Writing Boot image component -   OK
Done
Device #8: Updating FW ...
FSMST_INITIALIZE -   OK
Writing Boot image component -   OK
Done
Device #9: Updating FW ...
FSMST_INITIALIZE -   OK
Writing Boot image component -   OK
Done
Device #10: Updating FW ...
FSMST_INITIALIZE -   OK
Writing Boot image component -   OK
Done
Device #11: Updating FW ...
FSMST_INITIALIZE -   OK
Writing Boot image component -   OK
Done

Restart needed for updates to take effect.
```

(continues on next page)

(continued from previous page)

```
-E- One or more errors were encountered.
```

Chapter 6. Known Issues

6.1. Functional Issues

- You cannot update firmware of the individual components of the DGX H100/H200 GPU tray. For example, you can not individually update the firmware for the GPU only. You must update the firmware by flashing the entire DGX H100/H200 GPU tray.
- Firmware download is not automatic. You must download the firmware manually from the NVIDIA Enterprise Support Portal.
- For systems running DGX OS 6.0, the `nvfwupd` command-line utility that is shown in sample commands is not automatically installed. You must download the utility from the NVIDIA Enterprise Support Portal. For systems running DGX OS version 6.1 or later, the `nvfwupd` command-line utility is included with the operating system.

6.2. Redfish Service on BMC Might Experience Intermittent Unresponsiveness

6.2.1. Issue

An intermittent issue affecting several nodes where the BMC Redfish service might become unresponsive. When this happens, any attempt to query Redfish returns the following error:

```
"code": "Base.1.12.ServiceInUnknownState",
"message": "The operation failed because the service is in an unknown state
↪and can no longer take incoming requests."
```

6.2.2. Workaround

For systems that have exhibited the `"code": "Base.1.12.ServiceInUnknownState"` failure, follow these steps as part of updating the new BMC firmware:

Option 1: Reinitialize the Redis database using the Redfish API

1. Perform a Redis database reset by invoking the `BMC/Actions/Oem/AMIManager.RedfishDBReset` action.

```
curl -k -u <username>:<password> --request POST --location 'https://
↪$BMCIP/redfish/v1/Managers/BMC/Actions/Oem/AMIManager.RedfishDBReset' --
↪header 'Content-Type: application/json' --data '{"RedfishDBResetType":
↪"ResetAll"}' | jq
```

   Example response:

```
{
  "@odata.context": "/redfish/v1/$metadata#Task.Task",
  "@odata.id": "/redfish/v1/TaskService/Tasks/1",
  "@odata.type": "#Task.v1_4_2.Task",
  "Description": "Task for RedfishDBReset Task",
  "Id": "1",
  "Name": "RedfishDBReset Task",
  "TaskState": "New"
}
```

2. Wait approximately 3-4 minutes for the Redfish service to recover and stabilize.
3. Run the following `curl` command to reboot the BMC.

```
curl -k -u <bmc-user>:<password> --request POST --location 'https://<bmc-
↪ip-address>/redfish/v1/Managers/BMC/Actions/Manager.Reset' --header
↪'Content-Type: application/json' --data '{"ResetType": "ForceRestart"}'
```

Option 2: Restore to defaults using IPMItool while preserving all configuration settings except Redfish

To preserve all configuration settings except the Redfish configuration using IPMItool:

1. Get the current preserve configuration settings.

```
$ ipmitool -H $BMCIP -I lanplus -U <username> -P <password> raw 0x32 0xBB
00 00 00 ff 77
```

2. Set all preserve configuration settings except Redfish (byte 2, bit 6).

```
$ ipmitool -H $BMCIP -I lanplus -U <username> -P <password> raw 0x32 0xBA
↪0xff 0x37
```

3. Get the current preserve configuration settings.

```
$ ipmitool -H $BMCIP -I lanplus -U <username> -P <password> raw 0x32 0xBB
00 ff 37 ff 77
```

4. Initiate a restore to defaults, which will cause the BMC to reboot.

```
$ ipmitool -H $BMCIP -I lanplus -U <username> -P <password> raw 0x32 0x66
```

5. After the BMC finishes rebooting, restore all settings to their initial state.

```
$ ipmitool -H $BMCIP -I lanplus -U <username> -P <password> raw 0x32 0xBA
↪0x00 0x00
```

6. Get the current preserve configuration settings.

```
$ ipmitool -H $BMCIP -I lanplus -U <username> -P <password> raw 0x32 0xBB
00 00 00 ff 77
```

6.3. USB1 Port Missing on Occasion after a BMC Cold Reset

6.3.1. Issue

On the DGX system with BMC version 24.09.17 and HMC version rc67, the USB1 port becomes unreachable on occasion as shown on the BMC console after running a BMC cold reset:

```
ipmitool -H <bmc-ip-address> -I lanplus -U <bmc-username> -P <bmc-password>
↪mc reset cold
```

6.3.2. Explanation

On a cold reset, the HMC might reset as well, resulting in a short delay in baseboard telemetry.

6.3.3. Workaround

Periodically issue the following command to determine if the HMC is up. When the command returns a response, the HMC is operating.

```
curl -k -u <bmc-user>:<password> --request PATCH 'https://<bmc-ip-address>/
↪redfish/v1/Chassis/HGX_BMC_0'
```

6.4. Misleading Messages During Firmware Update

6.4.1. Issue

During the process of the ConnectX-7 firmware update, upon completion of applying the update, a reboot is required as suggested by these messages: `To load new FW, run mlxfwreset or reboot machine.` and `Please reboot machine to load new configurations.` However, rebooting the system does not load the firmware update or new configurations properly for the ConnectX-7 firmware versions 28.36.1010 and later.

6.4.2. Workaround

For the firmware update and new configurations to load successfully, perform an AC power cycle on the system instead of rebooting.

6.5. Sensors Endpoint for the Redfish API Does Not Support $expand

6.5.1. Issue

An HTTP GET request to the sensors endpoint with an $expand argument like the following fails.

```
/redfish/v1/Chassis/DGX/Sensors?$expand=.($levels=3)
```

6.5.2. Workaround

You can request sensor data from the Redfish API by requesting one sensor at a time. You can use the IPMI tool to request sensor data.

6.6. Firmware Upgrade or Downgrade Can Fail

6.6.1. Issue

When you perform a firmware upgrade or downgrade, the change can fail with a message like the following example:

```
...
[Sat 19 Aug 2023 08:20:50 AM CST] Firmware update task ended with state
↪Exception, percentComplete: [98]
[Sat 19 Aug 2023 08:20:50 AM CST] Update RC: 1
[Sat 19 Aug 2023 08:20:50 AM CST] Collect RF task
[Sat 19 Aug 2023 08:21:01 AM CST] Update failed with [nvfw_DGX-H100_0005_
↪230615.1.0_dbg-signed.fwpkg]:[/redfish/v1/UpdateService/FirmwareInventory/
↪EROT_BMC_0]
```

6.6.2. Workaround

Retry the firmware upgrade or downgrade.

6.7. Firmware Inventory Can Be Invalid During Boot

6.7.1. Issue

In rare instances, polling the firmware inventory endpoint of the BMC Redfish API can report an inaccurate firmware versions for the HGX_0 component.

6.7.2. Workaround

Query the firmware inventory after the system completes the boot sequence to retrieve the current firmware inventory.

6.8. BMC Slow Startup After AC Power Cycle

6.8.1. Issue

After an AC power cycle, the BMC can require approximately 10 minutes before it is available for communication. The BMC is typically available within three minutes.

6.8.2. Workaround

No workaround is available.

6.9. Temperature Sensors Can Report No Reading

6.9.1. Issue

The following sensors can report `No Reading` rather than a temperature value:

- TEMP_PSU4
- TEMP_PSU5
- PWR_PSU5
- SPD_FAN_PSU5_R
- SPD_FAN_PSU5_R
- STATUS_PSU0
- STATUS_PSU1
- STATUS_PSU2
- STATUS_PSU3
- STATUS_PSU4
- STATUS_PSU5
- STATUS_HMC

- TEMP_PCIE_SW_1
- TEMP_Cedar_OSFP0
- TEMP_Cedar_OSFP1
- TEMP_Cedar_OSFP2
- TEMP_Cedar_OSFP3
- TEMP_PCIE_CX7_1
- TEMP_PCIE_CX7_2
- TEMP_CX7_QSFP0
- TEMP_CX7_QSFP1
- TEMP_CX7_QSFP2
- TEMP_CX7_QSFP3
- TEMP_Intel_NIC
- TEMP_NIC_QSFP0
- TEMP_NIC_QSFP1

6.9.2. Workaround

Polling the sensors again can resolve the issue.

Chapter 7. Resolved Issues

The following issues that were previously identified as known issues have been resolved.

7.1. Issues with ConnectX-7 Network (Cluster) Card Firmware

7.1.1. Issue

If the NVIDIA® ConnectX®-7 Network (Cluster) Card firmware version 28.39.3560 is currently installed on your DGX H100/H200 system, you might encounter the following issues:

- After a long runtime on a DGX H100/H200 system, one or more GPUs might fall off the bus, and the `nvidia-smi` command fails to run. After a power cycle, the system will recover, and all GPUs will be operational. The system will continue to run again without any issues for a long time.
- After a reboot or power cycle, one or more OSFP ports on the DGX system might remain in the Down state.

7.1.2. Resolution

To prevent these issues, NVIDIA recommends updating the firmware of the following ConnectX-7 network cards to version 28.42.1000:

NVIDIA ConnectX-7 Card	Version for the 24.09.1 Release	Recommended Version
Network (cluster) card	28.39.3560	28.42.1000
Network (storage) card	28.39.3560	28.42.1000

For more information, refer to DGX H100/H200 - Update for ConnectX-7 Networking Cards Available.

7.2. Platform DGX H200 Not Supported

7.2.1. Issue

On DGX H200 systems with `nvfwupd` version 2.0.1 installed, the following error message might appear when you update the firmware using the `nvfwupd` command.

```
Platform dgxh200 not supported.
```

7.2.2. Explanation

Starting with `nvfwupd` version 2.0.1, the server type is required to update the firmware on new DGX platforms. An enhanced solution to automatically detect the server type for DGX platforms will be available in a future release.

7.2.3. Status

Resolved in `nvfwupd` version 2.0.4.

7.3. The ipmitool dcmi power reading Command Returns 0 Power Reading Value

7.3.1. Issue

When you use the `ipmitool dcmi power reading` command to report the power consumption data, the command reports 0 Watts for the power reading value as shown in the following example:

```
$ sudo ipmitool -I lanplus -H IPaddress -U user -P password dcmi power reading
Instantaneous power reading:                     0 Watts
Minimum during sampling period:                  0 Watts
Maximum during sampling period:                7852 Watts
Average power reading over sample period:      1885 Watts
IPMI timestamp:                           Jan 12 09:20:45 2024
Sampling period:                          00000005 Seconds
Power reading state is:                    activated
```

7.3.2. Status

Resolved in version 24.09.1.

7.4. GPUs Show Exclamation Mark in BMC Web Interface

7.4.1. Issue

When you view the GPUs from the BMC web interface, the GPUs are shown with an exclamation mark ( ).

7.4.2. Explanation

The icon is a false positive. You can view the results of the `nvsm show health` command to confirm that the GPU status is healthy.

7.4.3. Status

Resolved in version 1.1.3.

7.5. BMC LDAP Fields Do Not Support Space or Slash Characters

7.5.1. Issue

The BMC LDAP settings do not support the space or slash characters as part of the bind DN or search base. The following DN results in a failure:

```
DC=Echo Studios,DC=com
```

7.5.2. Status

Resolved in version 24.09.1.

7.6. NVMe Information Not Visible in BCM Web Interface

7.6.1. Issue

In some cases, the NVMe information is not visible in the BMC web interface.

7.6.2. Status

Resolved in version 24.09.1.

Chapter 8. Getting Started with nvdebug

The NVIDIA® NVDebug tool, `nvdebug`, runs on server platforms or from remote client machines. This binary tool, which is available for x86_64 or arm64-SBSA architecture systems, collects the following information:

- Out-of-band (OOB) BMC logs and information for troubleshooting server issues
- Logs from the host

8.1. Requirements

Table 1: Requirements for Client Host and Server Host

Requirement	Client Host	Server Host
Linux-based operating system: Linux kernel 4.4 or later supported (version 4.15 or later recommended)	X	X
GNU C Library glibc-2.7 or later	X	X
OS: Ubuntu 18.04 or later supported (Ubuntu 22.04 recommended)	X	X
Python 3.10	X	X
`ipmitool` 1.8.18 or later	X	X
The `sshpass` command	X	X
A server device under test (DUT) accessible by the BMC from the client host using Redfish and IPMI-over-LAN.	X	X
The `nvme-cli` tool		X
BMC Management and Server Host Management networks are in the same subnet.		X

NVSwitch tray host requires NVOS version 2.

8.2. The nvdebug Command-Line Interface

The high-level syntax of the `nvdebug` command supports the collection of debug logs over OOB.

You can run the tool in either of the following ways:

- From a remote machine with access to the BMC and host.
- Directly on the host machine if the host can access the BMC.

If the host IP is passed through the configuration file or the command-line interface (CLI) using `-I/--hostip`, `nvdebug` assumes the tool runs on a remote machine. Otherwise, `nvdebug` assumes the tool runs on the host and collects the host logs locally.

8.2.1. Syntax

```
$ nvdebug -i <BMCIP> -u <BMCUSER> -p <BMCPASS> -t <PLATFORM>

Mandatory options:
    -i/--ip is the BMC IP address.
    -u/--user is BMC username with administrative privileges.
    -p/--password is BMC administrative user password.
    -t/--platform is the platform type of the DUT, and it accepts DGX, HGX-
↪HMC, arm64, x86_64, and NVSwitch.

Additional credentials:
    -r/--sshuser is BMC SSH username.
    -w/--sshpass is BMC SSH password.
    -R/--rfuser is BMC Redfish username.
    -W/--rfpass is BMC Redfish password.

Host options:
    -I/--hostip is the Host IP Address.
     If the IP address is not provided, the tool assumes it is running on the
↪host machine.
    -U/--hostuser is the Host username with administrative privileges.
    -H/--hostpass is the Host password.

Additional options:
    -b/--baseboard <baseboard> is the baseboard type, such as Hopper-HGX-8-
↪GPU and Blackwell-HGX-8-GPU.
    -C/--config <file path> is the path to the config file. The default is ./
↪config.yaml.
    -d/--dutconfig <dut config path> is the path to the DUT specific config
↪file.
     The default path is ./dut_config.yaml.
    -c/--common collects the common logs using the included common.json file.
    -v/--verbose displays the detailed output and error messages.
    -o/--outdir <output dir> the output directory where the output is
↪generated.
     The default location is /tmp.
    -P/--port <fw_port> is the port number that will be used for forwarding.
     The --port variable applies only to HGX-Baseboard based platforms,
     and the default value is 18888.
    --local enables Local Execution mode.
    -z/--skipzip skips zipping individual DUT folders.


Log collection options:
    -S/--cids CID [CID ...] runs the log collectors that correspond to the
↪CIDs that were passed.
    -g/--loggroup <Redfish|IPMI|SSH|Host|HealthCheck> runs all log collectors
↪of a specific type
     that is supported on the current platform.
     Only one collector group can be specified.
    -j/--vendor_file <vendor.json> is a vendor-defined JSON file that uses
↪proprietary methods
```

(continues on next page)

(continued from previous page)

```
     and tools as defined by the user.
     The -S and –g options cannot be used together.

Utility options:
    -h/--help and --version are standalone options, and –l/--list requires
↪the platform
     type to be specified using –t/--platform.
    --parse <log dump> parses an nvdebug log dump and decodes the binary data.
    -h/--help provides information about tool usage.
    --version displays the current version of the tool.
    -l/--list [Redfish|IPMI|SSH|Host|HealthCheck] lists log collectors that
↪are supported by platform
     with their collector IDs (CID). If a type is passed, it will only list
↪log collectors
     of that type. The -l/--list options require the target platform type to
↪be specified with –t/--platform.

    By default, if option -c is not included, the nvdebug tool will collect
↪logs based on the common.json
    and platform_xyz.json files. At the end of the run, the tool will
↪generate the output log xyz.zip
    file in the directory specified by the –o option. If no directory is
↪provided, the log
    will be generated in the /tmp directory.
```

8.2.2. The Configuration Files

The NVDebug tool has two configuration files in the same folder as the executable:

- The DUT configuration file: The default is `dut_config.yaml`.
- The NVDebug-specific configuration file: The default is `config.yaml`.

These files can be used to provide additional (but optional) configuration data. If an argument is provided by both the CLI and the configuration file, the value provided through the CLI takes precedence.

8.3. HGX H100/H200 8-GPU Example

To communicate with the HGX baseboard, you need the BMC SSH credentials to set up SSH tunneling through the BMC. By default, the SSH credentials are assumed to be the same as the BMC credentials. To use different credentials, specify the `–r` and `–w` CLI options for the SSH username and password, respectively.

```
nvdebug –i $BMCIP –u $BMCUSER –p $BMCPASS –r SSHUSER –w SSHPASS –t HGX-HMC –P
↪port_num
```

Example output:

```
Log directory created at /tmp/nvdebug_logs_30_09_2024_12_27_46
Starting a collection for DUT dut-1
```

(continues on next page)

(continued from previous page)

```
hgx-h100-node2: [12:28:13] Identified system as Model: P2312-A04, Partno: 692-
↪22312-0001-000, Serialno:1324623011823
hgx-h100-node2: [12:28:13] User provided platform type: HGX-HMC
hgx-h100-node2: [12:28:13] BMC IP: XXXX

Log collection has started for dut-1
hgx-h100-node2: [12:45:43] Log collection is now complete
hgx-h100-node2: [12:45:43] Log collection took 17m 30.29s
DUT hgx-h100-node2 completed.

The log zip file (:literal:`nvdebug_logs_30_09_2024_12_27_46.zip`) will be
↪created in the :literal:`/tmp` directory.
```

The SSH tunnel is set up automatically by the tool using the specified port, and the default value is 18888. To use an existing SSH tunnel, do not set up SSH tunneling in the configuration file, as shown in the following `dut_config` file:

```
hgx-h100-node2:
  <<: *dut_defaults
  BMC_IP: "bmc_ip"
  BMC_USERNAME: "bmc_user"
  BMC_PASSWORD: "bmc_pass"
  BMC_SSH_USERNAME: "ssh_user"
  BMC_SSH_PASSWORD: "ssh_pass"
  TUNNEL_TCP_PORT: "port_num"

  SETUP_PORT_FORWARDING: false
```

After configuring the NVDebug tool, run the `nvdebug` command:

> **Note**
>
> The Host BMC needs to support port forwarding.

Example output:

```
$ nvdebug

Log directory created at /tmp/nvdebug_logs_30_09_2024_12_27_46
Starting a collection for DUT hgx-h100-node2
hgx-h100-node2: [12:28:13] Identified system as Model: P2312-A04, Partno: 692-
↪22312-0001-000, Serialno:1324623011823
hgx-h100-node2: [12:28:13] User provided platform type: HGX-HMC
hgx-h100-node2: [12:28:13] BMC IP: XXXX

Log collection has started for hgx-h100-node2
hgx-h100-node2: [12:45:43] Log collection is now complete
hgx-h100-node2: [12:45:43] Log collection took 17m 30.29s
DUT hgx-h100-node2 completed.

The log zip file (nvdebug_logs_30_09_2024_12_27_46.zip) will be created in
↪the /tmp directory.
```

8.4. DGX H100/H200 Example

To run the tool using the command-line parameters, but not the configuration file:

```
nvdebug -i <bmc ip> -u <bmc user> -p <bmc pass> -t DGX -I <host IP> -U <host
↪user> -H <host pass> -r <bmc ssh username> -w <bmc ssh password>
```

The `I`, `-U`, `-H`, `-r`, and `-w` options are optional.

To run the tool using the configuration file, provide all the required parameter settings including `BMC_IP`, `BMC_USERNAME`, `BMC_PASSWORD`, and `PLATFORM`. You can set the output directory by providing a value for the `OUTPUT_DIR` parameter (or by using the `–o/--outdir` CLI option). The following example shows the BMC credentials in the configuration setup:

```
DUT_Defaults: &dut_defaults
  NodeType: "Compute"
  ipmi_cipher: "-C17"

# Create a dut object and inherit the default values.
# For any specific configuration details, add them below.
dgx-h100-node2:
  <<: *dut_defaults
  BMC_IP: "bmc_ip"
  BMC_USERNAME: "bmc_user"
  BMC_PASSWORD: "bmc_pass"
  BMC_SSH_USERNAME: "ssh_user"
  BMC_SSH_PASSWORD: "ssh_pass"
  RF_User: "redfish_user"
  RF_Pass: "redfish_pass"
```

After configuring the NVDebug tool, run the `nvdebug` command.

```
$ nvdebug -o /tmp/dgx/sys1
```

To show the verbose output of all log collections and the elapsed time for each log, specify the –v option.

```
$ nvdebug -o /tmp/dgx/sys1 -v
```

Example output:

```
Read DUT config file details from ..../dut_config.yaml
Read config file details from ..../config.yaml
Log directory created at /tmp/dgx/sys1/nvdebug_logs_30_09_2024_12_53_27
Starting a collection for DUT dgx-h100-node2
dgx-h100-node2: [12:53:33] All preflight checks passed
dgx-h100-node2: [12:53:35] Identified system as Model: DGXH100, Partno: 965-
24387-0002-003, Serialno:1660224000069
dgx-h100-node2: [12:53:35] User provided platform type: DGX
dgx-h100-node2: [12:53:35] BMC IP: XXXX
Log collection has started for dgx-h100-node2
dgx-h100-node2: [12:53:35]
dgx-h100-node2: [12:53:35] ####################################
dgx-h100-node2: [12:53:35]
```

(continues on next page)

(continued from previous page)

```
dgx-h100-node2: [12:53:35] Collecting redfish logs:
dgx-h100-node2: [12:53:35]
dgx-h100-node2: [12:53:35] Log collection was initiated for: r8_firmware_
↪inventory
dgx-h100-node2: [12:53:45] Log collection for r8_firmware_inventory took 0m
↪10.32s
dgx-h100-node2: [12:53:45] Log collection was initiated for: r9_firmware_
↪inventory_expand_query
dgx-h100-node2: [12:56:49] Log collection for r9_firmware_inventory_expand_
↪query took 3m 4.07s
dgx-h100-node2: [12:56:49] Log collection was initiated for: r10_chassis_info
dgx-h100-node2: [12:56:51] Log collection for r10_chassis_info took 0m 1.32s
dgx-h100-node2: [12:56:51] Log collection was initiated for: r11_chassis_
↪expand_query
dgx-h100-node2: [12:57:06] Log collection for r11_chassis_expand_query took
↪0m 15.13s

...

dgx-h100-node2: [12:59:32] ####################################
dgx-h100-node2: [12:59:32]
dgx-h100-node2: [12:59:32] Collecting IPMI logs:
dgx-h100-node2: [12:59:32]
dgx-h100-node2: [12:59:32] Log collection was initiated for: i1_mc_info
dgx-h100-node2: [12:59:32] Log collection for i1_mc_info took 0m 0.2s
dgx-h100-node2: [12:59:32] Log collection was initiated for: i2_lan_info
dgx-h100-node2: [12:59:33] Log collection for i2_lan_info took 0m 0.85s

...

dgx-h100-node2: [13:02:59] ####################################
dgx-h100-node2: [13:02:59]
dgx-h100-node2: [13:02:59] Collecting host logs:
dgx-h100-node2: [13:02:59]
dgx-h100-node2: [13:02:59] Log collection was initiated for: h7_node_crash_
↪dump
dgx-h100-node2: [13:03:00] Log collection for h7_node_crash_dump took 0m 1.02s
dgx-h100-node2: [13:03:00] Log collection was initiated for: h2_node_lspci
dgx-h100-node2: [13:03:04] Log collection for h2_node_lspci took 0m 3.57s

...

dgx-h100-node2: [13:15:42] Task state is Running. Rechecking in 30 seconds...
dgx-h100-node2: [13:16:18] Log collection for r17_dgx_manager_oem_log_dump
↪took 12m 34.53s
dgx-h100-node2: [13:16:18] Log collection is now complete
dgx-h100-node2: [13:16:18] Log collection took 22m 44.08s
DUT dgx-h100-node2 completed.
Log zip created at /tmp/dgx/sys1/nvdebug_logs_30_09_2024_12_53_27.zip
```

8.5. DGX Platform Example

To list the collectors that are available on a DGX platform, specify the `-l` option and the `-t DGX` option for log collectors and the DGX platform, respectively:

```
$ nvdebug -l -t DGX
```

Example output:

```
Redfish
  CID    Collector Name                          Log Location
   R8    firmware_inventory                      Redfish_R8_firmware_
↪inventory.json
   R9    firmware_inventory_expand_query         Redfish_R9_firmware_
↪inventory_expand_query.json
  R10    chassis_info                            Redfish_R10_chassis_info.json
  R11    chassis_expand_query                    Redfish_R11_chassis_expand_
↪query.json
  R12    system_info                             Redfish_R12_system_info.json
  R13    system_expand_query                     Redfish_R13_system_expand_
↪query.json
  R14    manager_info                            Redfish_R14_manager_info.json
  R15    manager_expand_query                    Redfish_R15_manager_expand_
↪query.json
  R17    dgx_manager_oem_log_dump                Redfish_R17_dgx_oem_dump_
↪{manager_id}_{task_id}.tar.xz
  R18    telemetry_metric_reports                Redfish_R18_report_{metric_
↪report}.json
  R19    chassis_thermal_metrics                 Redfish_R19_chassis_{chassis}
↪_thermal_metrics.json
  R20    firmware_inventory_table                Redfish_R20_firmware_
↪inventory_table.txt
  R22    task_details                            Redfish_R22_task_{task_id}.
↪json
  R23    nvlink_oob_logs                         Redfish_R23_NVLINK_OOB_Log_
↪{id}.json
  R25    additional_oob_logs                     Redfish_R25_OOB_Log_{id}.json
  R26    chassis_certificates                    Redfish_R26_chassis_{chassis_
↪id}_certificate.json
  R29    background_copy_status                  Redfish_R29_{chassis_id}_
↪copy_status.json
  R30    software_inventory                      Redfish_R30_software_
↪inventory
  R32    system_post_codes                       Redfish_R32_system_post_codes

IPMI
  CID    Collector Name                          Log Location
   I1    mc_info                                 IPMI_I1_mc_info.txt
   I2    lan_info                                IPMI_I2_lan_info.txt
   I3    session_info                            IPMI_I3_session_info.txt
   I4    fru_info                                IPMI_I4_fru_info.txt
   I5    sdr_info                                IPMI_I5_sdr_info.txt
   I6    sel_info                                IPMI_I6_sel_info.txt
```

(continues on next page)

(continued from previous page)

```
   I7    sensor_list                          IPMI_I7_sensor_list.txt
   I8    sel_list                             IPMI_I8_sel_list.txt
   I9    sel_raw_dump                         IPMI_I9_sel_raw_dump.txt
  I10    chassis_status                       IPMI_I10_chassis_status.txt
  I11    chassis_restart_cause                IPMI_I11_chassis_restart_
↪cause.txt
  I12    user_list                            IPMI_I12_user_list.txt
  I13    channel_info                         IPMI_I13_channel_info.txt
  I14    sdr_elist                            IPMI_I14_sdr_elist.txt

SSH
  CID    Collector Name                       Log Location
   S2    bmc_dmesg                            BMC_SSH_S2_bmc_dmesg.txt
   S3    network_info                         BMC_SSH_S3_network_info/...
   S5    bmc_list_kernel_modules              BMC_SSH_S5_bmc_list_kernel_
↪modules.txt
   S8    bmc_mem_cpu_utilization              BMC_SSH_S8_bmc_mem_cpu_
↪utilization/...
  S11    uptime                               BMC_SSH_S11_uptime.txt
  S12    fpga_register_table                  BMC_SSH_S12_fpga_register_
↪table.txt
  S13    hmc_boot_status                      BMC_SSH_S13_hmc_boot_status.
↪txt
  S15    bmc_power_status                     BMC_SSH_S15_bmc_power_status/
↪...

Host
  CID    Collector Name                       Log Location
   H1    node_dmesg                           Host_H1_node_dmesg.tar.gz
   H2    node_lspci                           Host_H2_node_lspci*.txt
   H3    node_smbios                          Host_H3_dmidecode*.txt
   H4    node_lshw                            Host_H4_lshw*.txt
   H5    node_nvidia_smi                      Host_H5_nvidia-smi*.txt
   H6    node_kern_log                        Host_H6_node_kern_log.tar.gz
   H7    node_crash_dump                      Host_H7_node_crash_dump.tar.
↪gz
   H8    node_nvme_list                       Host_H8_nvme_list_-v.txt
   H9    node_fabric_manager_log              Host_H9_fabricmanager.log
  H10    node_nvflash_log                     Host_H10_nvflash_--check_-i_
↪{num}.txt
  H11    nvidia_bug_report                    Host_H11_nvidia_bug_report_
↪op.log.gz
  H15    node_subnet_manager                  Host_H15_node_subnet_manager/
  H16    one_diag_dump                        Host_H16_one_diag_dump/
  H17    node_nvme_log_dump                   Host_H17_nvos_tech_support_
↪dump/

HealthCheck
  CID    Collector Name                       Log Location
   C1    out_of_band_health_check             HealthCheck_C1_out_of_band_
↪health_check.json
```

8.6. Redfish Collectors

To collect only specific collectors, specify the -S option for firmware inventory, system information, and ipmi manager information.

```
nvdebug -i <bmc_ip> -u <bmc_user> -p <bmc_pass> ... -t DGX -v -S R8 I1 R12
```

Example output:

```
Log directory created at /tmp/nvdebug_logs_06_11_2024_15_40_27
Starting a collection for DUT dut-1
dut-1: [15:40:34] All preflight checks passed
dut-1: [15:40:34] Identified system as Model: DGXH100, Partno: 965-24387-0002-
↪003, Serialno:1660224000069
dut-1: [15:40:34] User provided platform type: DGX
dut-1: [15:40:34] BMC IP: XXXX
Log collection has started for dut-1
dut-1: [15:40:34]
dut-1: [15:40:34] ####################################
dut-1: [15:40:34]
dut-1: [15:40:34] Collecting custom logs:
dut-1: [15:40:34]
dut-1: [15:40:34] Log collection was initiated for: r8_firmware_inventory
dut-1: [15:40:36] Log collection for r8_firmware_inventory took 0m 1.71s
dut-1: [15:40:36] Log collection was initiated for: r12_system_info
dut-1: [15:40:36] Log collection for r12_system_info took 0m 0.06s
dut-1: [15:40:36] Log collection was initiated for: i1_mc_info
dut-1: [15:40:36] Log collection for i1_mc_info took 0m 0.14s
dut-1: [15:40:36] Log collection is now complete
dut-1: [15:40:36] Log collection took 0m 2.16s
DUT dut-1 completed.
Log zip created at /tmp/nvdebug_logs_06_11_2024_15_40_27.zip
```

To run the Redfish log collectors, specify the -g option for the Redfish log group:

```
$ nvdebug -i $BMC_IP -u $BMC_USER -p $BMC_PASS -t DGX -g Redfish
```

8.7. IPv6 Configuration

By default, the nvdebug tool uses IPv4. For IPv6, set IP_NETWORK to ipv6 in the DUT configuration. When providing IPv6 addresses for the BMC/Host, do not use square brackets.

Chapter 9. Viewing the Installed Firmware and Package Versions

Perform the following steps to view the firmware versions that are installed on the system and the versions in the firmware update packages.

- Compare the installed firmware versions with the versions available in the packages:

```
nvfwupd --target ip=<bmc-ip-address> user=<bmc-username> password=<bmc-
↪password> \
  show_version -p nvfw_DGX_250629.1.0.fwpkg \
  nvfw_HGX_DGXH100-H200x8_250319.1.0.fwpkg
```

  Replace the firmware package file names with the file names that you downloaded.

> **Tip**
>
> If you run the `nvfwupd show_version` command without any arguments, the command displays the currently installed firmware versions.

  Example Output

```
System Model: DGXH100
Part number: xxx-xxxxx-xxxx-xxx
Serial number: xxxxxxxxxxxx
BMC IP: 192.168.1.1

Firmware Devices:
AP Name                         Sys Version                     Pkg Version
↪                Up-To-Date
-------                         -----------                     -----------
↪                ----------
CPLDMB_0                        0.2.1.6                         0.2.1.6
↪                  Yes
CPLDMID_0                       0.2.0.7                         0.2.0.7
↪                  Yes
EROT_BIOS_0                     00.04.0011.0000_n00             00.04.0018.
↪0000_n00           No
```

(continues on next page)

(continued from previous page)

```
EROT_BMC_0                      00.04.0011.0000_n00             00.04.0018.
↪0000_n00            No
HGX_FW_BMC_0                    HGX-22.10-1-rc31                HGX-23.03-
↪09-rc01              No
HGX_FW_ERoT_BMC_0               00.02.0120.0000_n00             00.02.0114.
↪0001_n00            Yes
HGX_FW_ERoT_FPGA_0              00.02.0120.0000_n00             00.02.0114.
↪0001_n00            Yes
HGX_FW_ERoT_GPU_SXM_1           00.02.0120.0000_n00             00.02.0114.
↪0001_n00            Yes
HGX_FW_ERoT_GPU_SXM_2           00.02.0120.0000_n00             00.02.0114.
↪0001_n00            Yes
HGX_FW_ERoT_GPU_SXM_3           00.02.0120.0000_n00             00.02.0114.
↪0001_n00            Yes
HGX_FW_ERoT_GPU_SXM_4           00.02.0120.0000_n00             00.02.0114.
↪0001_n00            Yes
HGX_FW_ERoT_GPU_SXM_5           00.02.0120.0000_n00             00.02.0114.
↪0001_n00            Yes
HGX_FW_ERoT_GPU_SXM_6           00.02.0120.0000_n00             00.02.0114.
↪0001_n00            Yes
HGX_FW_ERoT_GPU_SXM_7           00.02.0120.0000_n00             00.02.0114.
↪0001_n00            Yes
HGX_FW_ERoT_GPU_SXM_8           00.02.0120.0000_n00             00.02.0114.
↪0001_n00            Yes
HGX_FW_ERoT_NVSwitch_0          00.02.0120.0000_n00             00.02.0114.
↪0001_n00            Yes
HGX_FW_ERoT_NVSwitch_1          00.02.0120.0000_n00             00.02.0114.
↪0001_n00            Yes
HGX_FW_ERoT_NVSwitch_2          00.02.0120.0000_n00             00.02.0114.
↪0001_n00            Yes
HGX_FW_ERoT_NVSwitch_3          00.02.0120.0000_n00             00.02.0114.
↪0001_n00            Yes
HGX_FW_ERoT_PCIeSwitch_0        00.02.0120.0000_n00             00.02.0114.
↪0001_n00            Yes
HGX_FW_FPGA_0                   2.0D                            2.09
↪                    Yes
HGX_FW_GPU_SXM_1                96.00.61.00.01                  96.00.5E.00.
↪00                  Yes
HGX_FW_GPU_SXM_2                96.00.61.00.01                  96.00.5E.00.
↪00                  Yes
HGX_FW_GPU_SXM_3                96.00.61.00.01                  96.00.5E.00.
↪00                  Yes
HGX_FW_GPU_SXM_4                96.00.61.00.01                  96.00.5E.00.
↪00                  Yes
HGX_FW_GPU_SXM_5                96.00.61.00.01                  96.00.5E.00.
↪00                  Yes
HGX_FW_GPU_SXM_6                96.00.61.00.01                  96.00.5E.00.
↪00                  Yes
HGX_FW_GPU_SXM_7                96.00.61.00.01                  96.00.5E.00.
↪00                  Yes
HGX_FW_GPU_SXM_8                96.00.61.00.01                  96.00.5E.00.
↪00                  Yes
```

(continues on next page)

(continued from previous page)

```
HGX_FW_NVSwitch_0               96.10.35.00.02                  96.10.2E.00.
↪00                 Yes
HGX_FW_NVSwitch_1               96.10.35.00.02                  96.10.2E.00.
↪00                 Yes
HGX_FW_NVSwitch_2               96.10.35.00.02                  96.10.2E.00.
↪00                 Yes
HGX_FW_NVSwitch_3               96.10.35.00.02                  96.10.2E.00.
↪00                 Yes
HGX_FW_PCIeRetimer_0            1.31.7                          1.31.7
↪                   Yes
HGX_FW_PCIeRetimer_1            1.31.7                          1.31.7
↪                   Yes
HGX_FW_PCIeRetimer_2            1.31.7                          1.31.7
↪                   Yes
HGX_FW_PCIeRetimer_3            1.31.7                          1.31.7
↪                   Yes
HGX_FW_PCIeRetimer_4            1.31.7                          1.31.7
↪                   Yes
HGX_FW_PCIeRetimer_5            1.31.7                          1.31.7
↪                   Yes
HGX_FW_PCIeRetimer_6            1.31.7                          1.31.7
↪                   Yes
HGX_FW_PCIeRetimer_7            1.31.7                          1.31.7
↪                   Yes
HGX_FW_PCIeSwitch_0             1.7.5F                          1.7.5F
↪                   Yes
HGX_InfoROM_GPU_SXM_1           G520.0200.00.01                 N/A
↪                   No
HGX_InfoROM_GPU_SXM_2           G520.0200.00.01                 N/A
↪                   No
HGX_InfoROM_GPU_SXM_3           G520.0200.00.01                 N/A
↪                   No
HGX_InfoROM_GPU_SXM_4           G520.0200.00.01                 N/A
↪                   No
HGX_InfoROM_GPU_SXM_5           G520.0200.00.01                 N/A
↪                   No
HGX_InfoROM_GPU_SXM_6           G520.0200.00.01                 N/A
↪                   No
HGX_InfoROM_GPU_SXM_7           G520.0200.00.01                 N/A
↪                   No
HGX_InfoROM_GPU_SXM_8           G520.0200.00.01                 N/A
↪                   No
HGX_InfoROM_NVSwitch_0          5612.0002.00.01                 N/A
↪                   No
HGX_InfoROM_NVSwitch_1          5612.0002.00.01                 N/A
↪                   No
HGX_InfoROM_NVSwitch_2          5612.0002.00.01                 N/A
↪                   No
HGX_InfoROM_NVSwitch_3          5612.0002.00.01                 N/A
↪                   No
HostBIOS_0                      01.00.04                        01.00.04
↪                   Yes
```

(continues on next page)

(continued from previous page)

```
HostBMC_0                        23.04.18                        44.04.19
↪                    No
PCIeRetimer_0                    1.30.12                         1.30.0
↪                    Yes
PCIeRetimer_1                    1.30.12                         1.30.0
↪                    Yes
PCIeSwitch_0                     0.0.6                           00.06.78
↪                    No
PCIeSwitch_1                     1.0.6                           01.06.78
↪                    No
PSU_0                            0202.0201.0202                  0202.0201.
↪0202                  Yes
PSU_1                            0202.0201.0202                  0202.0201.
↪0202                  Yes
PSU_2                            0202.0201.0202                  0202.0201.
↪0202                  Yes
PSU_3                            0202.0201.0202                  0202.0201.
↪0202                  Yes
PSU_4                            0202.0201.0202                  0202.0201.
↪0202                  Yes
PSU_5                            0202.0201.0202                  0202.0201.
↪0202                  Yes
```

HGX Firmware from the GPU tray reports the `HGX_InfoRom_GPU_SXM_n` and `HGX_InfoRom_NVSwitch_n` in the firmware inventory Redfish output. As a result, it is included in the preceding output as `N/A`. In the `Up-To-Date` column, these entries show `No` because you cannot update them OOB from the GPU or NVSwitch firmware images respectively.

Chapter 10. Updating the BMC

1. Create a `update_bmc.json` file with the following contents:

```
{
    "Targets" :["/redfish/v1/UpdateService/FirmwareInventory/HostBMC_0"]
}
```

2. Update the firmware:

```
nvfwupd -t ip=<bmc-ip-address> user=<bmc-username> password=<bmc-password>
 ↪ update_fw \
  -p nvfw_DGX_250629.1.0.fwpkg -y -s update_bmc.json
```

Example Output

```
FW recipe: ['nvfw_DGX_250629.1.0.fwpkg']
{"@odata.type": "#UpdateService.v1_6_0.UpdateService", "Messages": [{
 ↪"@odata.type": "#Message.v1_0_8.Message", "Message": "A new task /
 ↪redfish/v1/TaskService/Tasks/2 was created.", "MessageArgs": ["/redfish/
 ↪v1/TaskService/Tasks/2"], "MessageId": "Task.1.0.New", "Resolution":
 ↪"None", "Severity": "OK"}, {"@odata.type": "#Message.v1_0_8.Message",
 ↪"Message": "The action UpdateService.MultipartPush was submitted to do
 ↪firmware update.", "MessageArgs": ["UpdateService.MultipartPush"],
 ↪"MessageId": "UpdateService.1.0.StartFirmwareUpdate", "Resolution":
 ↪"None", "Severity": "OK"}]}
 FW update started, Task Id: 2
Wait for FirmwareUpdateStarted in MessageId
Wait for FirmwareUpdateStarted in MessageId
Wait for FirmwareUpdateStarted in MessageId
Wait for FirmwareUpdateStarted in MessageId
Wait for FirmwareUpdateStarted in MessageId
Wait for FirmwareUpdateStarted in MessageId
Wait for FirmwareUpdateStarted in MessageId
Wait for FirmwareUpdateStarted in MessageId
Wait for FirmwareUpdateStarted in MessageId
Wait for FirmwareUpdateStarted in MessageId
PercentageComplete: 6
 TaskState: Running
 PercentComplete: 6
 TaskStatus: OK
 ...
```

(continues on next page)

(continued from previous page)

```
 PercentageComplete: 100
 TaskState: Completed
 PercentComplete: 100
 TaskStatus: OK
Firmware update successful!
 Overall Time Taken: 0:36:11
Refer to 'DGX H100 Firmware Update Document' on activation steps for new
 ↪firmware to take effect.
```

3. Reset the BMC so the BMC boots the new firmware:

```
# If you have a shell on the system
$ sudo ipmitool mc reset cold

# If you are logged in to a different system
$ ipmitool -H <bmc-ip-address> -I lanplus -U <bmc-username> -P <bmc-
 ↪password> mc reset cold
```

4. Reboot the system.

Chapter 11. Firmware Update of Motherboard Tray: All Components

Perform the following steps to update the firmware on all the system components, such as CPLDs, PSUs, PCIe switches, and so on. Ensure the system is powered on before updating the firmware.

This procedure is an alternative to updating each component individually.

1. Create a `mb_tray.json` file with empty braces, like the following example:

   ```
   {}
   ```

2. Update the firmware:

   ```
   nvfwupd -t ip=<bmc-ip-address> user=<bmc-username> password=<bmc-password>
    ↪ update_fw \
     -p nvfw_DGX_250629.1.0.fwpkg -y -s mb_tray.json
   ```

   Example Output

   ```
   FW package: ['nvfw_DGX_250629.1.0.fwpkg']
   Ok to proceed with firmware update? <Y/N>
   y
   {"@odata.type": "#UpdateService.v1_11_0.UpdateService", "Messages": [{
    ↪"@odata.type": "#Message.v1_0_8.Message", "Message": "A new task /
    ↪redfish/v1/TaskService/Tasks/2 was created.", "MessageArgs": ["/redfish/
    ↪v1/TaskService/Tasks/2"], "MessageId": "Task.1.0.New", "Resolution":
    ↪"None", "Severity": "OK"}, {"@odata.type": "#Message.v1_0_8.Message",
    ↪"Message": "The action UpdateService.MultipartPush was submitted to do
    ↪firmware update.", "MessageArgs": ["UpdateService.MultipartPush"],
    ↪"MessageId": "UpdateService.1.0.StartFirmwareUpdate", "Resolution":
    ↪"None", "Severity": "OK"}]}
     FW update started, Task Id: 2
   Wait for Firmware Update to Start...
     TaskState: Running
     PercentComplete: 1
     TaskStatus: OK
     TaskState: Running
     PercentComplete: 20
     TaskStatus: OK
   ```

(continues on next page)

(continued from previous page)

```
  TaskState: Running
  PercentComplete: 40
  TaskStatus: OK
  TaskState: Running
  PercentComplete: 61
  TaskStatus: OK
  TaskState: Running
  PercentComplete: 80
  TaskStatus: OK
  TaskState: Running
  PercentComplete: 99
  TaskStatus: OK
  TaskState: Completed
  PercentComplete: 100
  TaskStatus: OK
  Firmware update successful!
Overall Time Taken: 0:24:38
Refer to the DGX H100/H200 Firmware Update Guide on activation steps for
 ↪the new firmware to take effect.
--------------------------------------------------------------------------
 ↪------------------------------
```

Chapter 12. Updating BMC EROT on the Motherboard Tray

1. Create a `updparameters.json` file with the following contents:

```
{
    "Targets" :["/redfish/v1/UpdateService/FirmwareInventory/EROT_BMC_0"]
}
```

2. Update the firmware:

```
nvfwupd -t ip=<bmc-ip-address> user=admin password=admin update_fw \
  -p nvfw_DGX_250629.1.0.fwpkg -y -s updparameters.json
```

Example Output

```
FW recipe: ['nvfw_DGX_250629.1.0.fwpkg]
{"@odata.type": "#UpdateService.v1_6_0.UpdateService", "Messages": [{
↪"@odata.type": "#Message.v1_0_8.Message", "Message": "A new task /
↪redfish/v1/TaskService/Tasks/1 was created.", "MessageArgs": ["/redfish/
↪v1/TaskService/Tasks/1"], "MessageId": "Task.1.0.New", "Resolution":
↪"None", "Severity": "OK"}, {"@odata.type": "#Message.v1_0_8.Message",
↪"Message": "The action UpdateService.MultipartPush was submitted to do
↪firmware update.", "MessageArgs": ["UpdateService.MultipartPush"],
↪"MessageId": "UpdateService.1.0.StartFirmwareUpdate", "Resolution":
↪"None", "Severity": "OK"}]}
 FW update started, Task Id: 1
Wait for Firmware Update to Start...
Wait for Firmware Update to Start...
 TaskState: Completed
 PercentComplete: 100
 TaskStatus: OK
Firmware update successful!
 Overall Time Taken: 0:00:09
Refer to 'DGX H100 Firmware Update Document' on activation steps for new
↪firmware to take effect.
```

Chapter 13. Updating SBIOS EROT on the Motherboard Tray

1. Create a `updparameters.json` file with the following contents:

```
{
    "Targets" :["/redfish/v1/UpdateService/FirmwareInventory/EROT_BIOS_0"]
}
```

2. Update the firmware:

```
nvfwupd -t ip=<bmc-ip-address> user=admin password=admin update_fw \
  -p nvfw_DGX_250629.1.0.fwpkg -y -s updparameters.json
```

Example Output

```
FW recipe: ['nvfw_DGX_250629.1.0.fwpkg.fwpkg]
{"@odata.type": "#UpdateService.v1_6_0.UpdateService", "Messages": [{
 ↪"@odata.type": "#Message.v1_0_8.Message", "Message": "A new task /
 ↪redfish/v1/TaskService/Tasks/2 was created.", "MessageArgs": ["/redfish/
 ↪v1/TaskService/Tasks/2"], "MessageId": "Task.1.0.New", "Resolution":
 ↪"None", "Severity": "OK"}, {"@odata.type": "#Message.v1_0_8.Message",
 ↪"Message": "The action UpdateService.MultipartPush was submitted to do
 ↪firmware update.", "MessageArgs": ["UpdateService.MultipartPush"],
 ↪"MessageId": "UpdateService.1.0.StartFirmwareUpdate", "Resolution":
 ↪"None", "Severity": "OK"}]}
 FW update started, Task Id: 2
Wait for Firmware Update to Start...
Wait for Firmware Update to Start...
 TaskState: Completed
 PercentComplete: 100
 TaskStatus: OK
Firmware update successful!
 Overall Time Taken: 0:00:10
Refer to 'DGX H100 Firmware Update Document' on activation steps for new
 ↪firmware to take effect.
```

Chapter 14. Updating the BIOS on the Motherboard Tray

1. Create a `updparameters.json` file with the following contents:

```
{
    "Targets" :["/redfish/v1/UpdateService/FirmwareInventory/HostBIOS_0"]
}
```

2. Update the firmware:

```
nvfwupd -t ip=<bmc-ip-address> user=admin password=admin update_fw \
  -p nvfw_DGX_250629.1.0.fwpkg -y -s updparameters.json
```

Example Output

```
FW recipe: [nvfw_DGX_250629.1.0.fwpkg']
{"@odata.type": "#UpdateService.v1_6_0.UpdateService", "Messages": [{
 ↪"@odata.type": "#Message.v1_0_8.Message", "Message": "A new task /
 ↪redfish/v1/TaskService/Tasks/2 was created.", "MessageArgs": ["/redfish/
 ↪v1/TaskService/Tasks/2"], "MessageId": "Task.1.0.New", "Resolution":
 ↪"None", "Severity": "OK"}, {"@odata.type": "#Message.v1_0_8.Message",
 ↪"Message": "The action UpdateService.MultipartPush was submitted to do
 ↪firmware update.", "MessageArgs": ["UpdateService.MultipartPush"],
 ↪"MessageId": "UpdateService.1.0.StartFirmwareUpdate", "Resolution":
 ↪"None", "Severity": "OK"}]}
 FW update started, Task Id: 2
Wait for FirmwareUpdateStarted in MessageId
Wait for FirmwareUpdateStarted in MessageId
Wait for FirmwareUpdateStarted in MessageId
Wait for FirmwareUpdateStarted in MessageId
Wait for FirmwareUpdateStarted in MessageId
Wait for FirmwareUpdateStarted in MessageId
Wait for FirmwareUpdateStarted in MessageId
Wait for FirmwareUpdateStarted in MessageId
Wait for FirmwareUpdateStarted in MessageId
Wait for FirmwareUpdateStarted in MessageId
PercentageComplete: 6
 TaskState: Running
 PercentComplete: 6
 TaskStatus: OK
 ...
```

(continues on next page)

(continued from previous page)

```
PercentageComplete: 100
 TaskState: Completed
 PercentComplete: 100
 TaskStatus: OK
Firmware update successful!
 Overall Time Taken: 0:36:11
Refer to 'DGX H100 Firmware Update Document' on activation steps for new
 ↪firmware to take effect.
```

Chapter 15. Updating the CPLDs on the Motherboard Tray

1. Create a `updparameters.json` file with the following contents:

```
{
    "Targets" :["/redfish/v1/UpdateService/FirmwareInventory/CPLDMB_0"]
}
```

2. Update the firmware:

```
nvfwupd -t ip=<bmc-ip-address> user=admin password=admin update_fw \
  -p nvfw_DGX_250629.1.0.fwpkg -y -s updparameters.json
```

Example Output

```
FW recipe: ['nvfw_DGX_250629.1.0.fwpkg]
{"@odata.type": "#UpdateService.v1_6_0.UpdateService", "Messages": [{
 ↪"@odata.type": "#Message.v1_0_8.Message", "Message": "A new task /
 ↪redfish/v1/TaskService/Tasks/4 was created.", "MessageArgs": ["/redfish/
 ↪v1/TaskService/Tasks/4"], "MessageId": "Task.1.0.New", "Resolution":
 ↪"None", "Severity": "OK"}, {"@odata.type": "#Message.v1_0_8.Message",
 ↪"Message": "The action UpdateService.MultipartPush was submitted to do
 ↪firmware update.", "MessageArgs": ["UpdateService.MultipartPush"],
 ↪"MessageId": "UpdateService.1.0.StartFirmwareUpdate", "Resolution":
 ↪"None", "Severity": "OK"}]}
 FW update started, Task Id: 4
Wait for Firmware Update to Start...
Wait for Firmware Update to Start...
 TaskState: Completed
 PercentComplete: 100
 TaskStatus: OK
Firmware update successful!
 Overall Time Taken: 0:00:08
Refer to 'DGX H100 Firmware Update Document' on activation steps for new
 ↪firmware to take effect.
```

Chapter 16. Updating the Midplane CPLDs on the Motherboard Tray

1. Create a `updparameters.json` file with the following contents:

```
{
    "Targets" :["/redfish/v1/UpdateService/FirmwareInventory/CPLDMID_0"]
}
```

2. Update the firmware:

```
nvfwupd -t ip=<bmc-ip-address> user=admin password=admin update_fw \
  -p nvfw_DGX_250629.1.0.fwpkg -y -s updparameters.json
```

Example Output

```
FW recipe: ['nvfw_DGX_250629.1.0.fwpkg']
{"@odata.type": "#UpdateService.v1_6_0.UpdateService", "Messages": [{
 ↪"@odata.type": "#Message.v1_0_8.Message", "Message": "A new task /
 ↪redfish/v1/TaskService/Tasks/5 was created.", "MessageArgs": ["/redfish/
 ↪v1/TaskService/Tasks/5"], "MessageId": "Task.1.0.New", "Resolution":
 ↪"None", "Severity": "OK"}, {"@odata.type": "#Message.v1_0_8.Message",
 ↪"Message": "The action UpdateService.MultipartPush was submitted to do
 ↪firmware update.", "MessageArgs": ["UpdateService.MultipartPush"],
 ↪"MessageId": "UpdateService.1.0.StartFirmwareUpdate", "Resolution":
 ↪"None", "Severity": "OK"}]}
 FW update started, Task Id: 5
Wait for Firmware Update to Start...
Wait for Firmware Update to Start...
  TaskState: Completed
 PercentComplete: 100
 TaskStatus: OK
Firmware update successful!
 Overall Time Taken: 0:00:09
Refer to 'DGX H100 Firmware Update Document' on activation steps for new
 ↪firmware to take effect.
```

Chapter 17. Updating the NVMe Firmware

The following instructions describe how to update the firmware on an NVMe device using the NVME-CLI command-line interface. The `nvme-cli` tool is preinstalled on all DGX H100/H200 systems.

1. List the devices and check the firmware versions.

```
$ sudo nvme list

Node             SN                   Model                                    Namespace
 ↪Usage                      Format           FW Rev
---------------- -------------------- ---------------------------------------- --------- ---
 ↪------------------------- ---------------- --------
/dev/nvme0n1     S666NE0T395778       SAMSUNG MZ1L21T9HCLS-00A07               1
 ↪1.18  TB /   1.92  TB    512   B +  0 B   GDC7302Q
/dev/nvme1n1     S666NE0T395802       SAMSUNG MZ1L21T9HCLS-00A07               1
 ↪1.18  TB /   1.92  TB    512   B +  0 B   GDC7302Q
/dev/nvme2n1     22L0A031T2N8        KCM6DRUL3T84                             1
 ↪27.04  GB /   3.84  TB    512   B +  0 B   0107
/dev/nvme3n1     22W0A02AT2N8        KCM6DRUL3T84                             1
 ↪27.57  GB /   3.84  TB    512   B +  0 B   0107
/dev/nvme4n1     22W0A02HT2N8        KCM6DRUL3T84                             1
 ↪26.19  GB /   3.84  TB    512   B +  0 B   0107
/dev/nvme5n1     22L0A01WT2N8        KCM6DRUL3T84                             1
 ↪25.44  GB /   3.84  TB    512   B +  0 B   0107
/dev/nvme6n1     22L0A025T2N8        KCM6DRUL3T84                             1
 ↪24.91  GB /   3.84  TB    512   B +  0 B   0107
/dev/nvme7n1     22L0A02CT2N8        KCM6DRUL3T84                             1
 ↪25.17  GB /   3.84  TB    512   B +  0 B   0107
/dev/nvme8n1     22L0A01FT2N8        KCM6DRUL3T84                             1
 ↪24.65  GB /   3.84  TB    512   B +  0 B   0107
/dev/nvme9n1     22L0A01YT2N8        KCM6DRUL3T84                             1
 ↪24.37  GB /   3.84  TB    512   B +  0 B   0107
```

   From the output, you can find the device names and firmware versions, such as `/dev/nvme0n1` and `GDC7302Q`.

2. Download the firmware you want to upgrade for the NVMe device.

   Based on the command output in step 1, determine the firmware file using the following table and include the correct path to download the firmware:

```
$ sudo nvme fw-download $DeviceName --fw=***(specify the firmware file
↪for upgrade)
```

This table lists the firmware file names for the supported Samsung and Kioxia models:

Manufacturer	Model Number	Firmware File
Samsung	MZ1L21T9HCLS-00A07	General_PM9A3_M.2_GDC7502Q_Noformat.bin
Samsung	MZWLO3T8HCLS-00A07	General_PM1743_U.2_0PPA4B5Q.bin
Kioxia	KCM6DRUL3T84	CM6-SED-0107.std
Kioxia	KCM7DRUL3T84	CM6-SED-0107.std

3. Commit and activate the downloaded firmware immediately without reset by setting the `action` argument to 3.

```
$ sudo nvme fw-commit $DeviceName --action=3
```

4. Verify that the correct firmware version is updated.

```
$ sudo nvme id-ctrl $DeviceName
```

Chapter 18. Updating the Power Supply Units on the Motherboard Tray

1. Create a `updparameters.json` file with the following contents:

```
{
    "Targets" :["/redfish/v1/UpdateService/FirmwareInventory/PSU_0"]
}
```

> **Important**
>
> Repeat this procedure for PSU_1 through PSU_5.

2. Update the firmware:

```
nvfwupd -t ip=<bmc-ip-address> user=admin password=admin update_fw \
  -p nvfw_DGX_250629.1.0.fwpkg -y -s updparameters.json
```

Example Output

```
FW recipe: ['nvfw_DGX_250629.1.0.fwpkg']
{"@odata.type": "#UpdateService.v1_6_0.UpdateService", "Messages": [{
 ↪"@odata.type": "#Message.v1_0_8.Message", "Message": "A new task /
 ↪redfish/v1/TaskService/Tasks/6 was created.", "MessageArgs": ["/redfish/
 ↪v1/TaskService/Tasks/6"], "MessageId": "Task.1.0.New", "Resolution":
 ↪"None", "Severity": "OK"}, {"@odata.type": "#Message.v1_0_8.Message",
 ↪"Message": "The action UpdateService.MultipartPush was submitted to do
 ↪firmware update.", "MessageArgs": ["UpdateService.MultipartPush"],
 ↪"MessageId": "UpdateService.1.0.StartFirmwareUpdate", "Resolution":
 ↪"None", "Severity": "OK"}]}
 FW update started, Task Id: 6
Wait for Firmware Update to Start...
Wait for Firmware Update to Start...
 TaskState: Completed
 PercentComplete: 100
 TaskStatus: OK
Firmware update successful!
 Overall Time Taken: 0:00:08
```

(continues on next page)

(continued from previous page)

```
Refer to 'DGX H100 Firmware Update Document' on activation steps for new
↪firmware to take effect.
```

Chapter 19. Updating the PCIe Switches on the Motherboard Tray

1. Create a `updparameters.json` file with the following contents:

```
{
    "Targets" :["/redfish/v1/UpdateService/FirmwareInventory/PCIeSwitch_0
↪"]
}
```

> **Important**
>
> Repeat this procedure for `PCIeSwitch_1`.

2. Update the firmware:

```
nvfwupd -t ip=<bmc-ip-address> user=admin password=admin update_fw \
  -p nvfw_DGX_250629.1.0.fwpkg -y -s updparameters.json
```

Example Output

```
FW recipe: ['nvfw_DGX_250629.1.0.fwpkg]
{"@odata.type": "#UpdateService.v1_6_0.UpdateService", "Messages": [{
↪"@odata.type": "#Message.v1_0_8.Message", "Message": "A new task /
↪redfish/v1/TaskService/Tasks/7 was created.", "MessageArgs": ["/redfish/
↪v1/TaskService/Tasks/7"], "MessageId": "Task.1.0.New", "Resolution":
↪"None", "Severity": "OK"}, {"@odata.type": "#Message.v1_0_8.Message",
↪"Message": "The action UpdateService.MultipartPush was submitted to do
↪firmware update.", "MessageArgs": ["UpdateService.MultipartPush"],
↪"MessageId": "UpdateService.1.0.StartFirmwareUpdate", "Resolution":
↪"None", "Severity": "OK"}]}
 FW update started, Task Id: 7
Wait for Firmware Update to Start...
Wait for Firmware Update to Start...
 TaskState: Completed
 PercentComplete: 100
 TaskStatus: OK
Firmware update successful!
```

(continues on next page)

(continued from previous page)

```
 Overall Time Taken: 0:00:09
Refer to 'DGX H100 Firmware Update Document' on activation steps for new
↪firmware to take effect.
```

Chapter 20. Updating the PCIe Retimers on the Motherboard Tray

1. Create a `updparameters.json` file with the following contents:

```
{
    "Targets" :["/redfish/v1/UpdateService/FirmwareInventory/PCIeRetimer_0
 ↪"]
}
```

> **Important**
>
> Repeat this procedure for `PCIeRetimer_1`.

2. Update the firmware:

```
nvfwupd -t ip=<bmc-ip-address> user=admin password=admin update_fw \
  -p nvfw_DGX_250629.1.0.fwpkg -y -s updparameters.json
```

Example Output

```
FW recipe: ['nvfw_DGX_250629.1.0.fwpkg]
{"@odata.type": "#UpdateService.v1_6_0.UpdateService", "Messages": [{
 ↪"@odata.type": "#Message.v1_0_8.Message", "Message": "A new task /
 ↪redfish/v1/TaskService/Tasks/8 was created.", "MessageArgs": ["/redfish/
 ↪v1/TaskService/Tasks/8"], "MessageId": "Task.1.0.New", "Resolution":
 ↪"None", "Severity": "OK"}, {"@odata.type": "#Message.v1_0_8.Message",
 ↪"Message": "The action UpdateService.MultipartPush was submitted to do
 ↪firmware update.", "MessageArgs": ["UpdateService.MultipartPush"],
 ↪"MessageId": "UpdateService.1.0.StartFirmwareUpdate", "Resolution":
 ↪"None", "Severity": "OK"}]}
 FW update started, Task Id: 8
Wait for Firmware Update to Start...
Wait for Firmware Update to Start...
 TaskState: Completed
 PercentComplete: 100
 TaskStatus: OK
Firmware update successful!
 Overall Time Taken: 0:00:09
Refer to 'DGX H100 Firmware Update Document' on activation steps for new
 ↪firmware to take effect.
```

Chapter 21. Updating the ConnectX-7 Firmware

After replacing or installing the ConnectX-7 cards, make sure the firmware on the cards is up to date.

Refer to the *component firmware versions table* to find the most recent firmware version.

1. Download the firmware from https://network.nvidia.com/support/firmware/connectx7ib/.

   Download the firmware for both OPN options.
2. Transfer the firmware ZIP file to the DGX system and extract the archive.
3. Update the firmware on the cards that are used for cluster communication:

   Note

   If MLNX_OFED is installed, use the `mlxfwmanager` tool to update the firmware. Otherwise, use the `mstflint` tool and specify the `burn` (or `b`) command, for example: `sudo mstflint -d <device> -i <firmware-image> b`.

```
sudo mlxfwmanager -d 5e:00.0 -i fw-ConnectX7-rel-28_43_2026-MCX750500B-
↪0D00_Ax_Bx-UEFI-14.36.21-FlexBoot-3.7.500.signed.bin
sudo mlxfwmanager -d dc:00.0 -i fw-ConnectX7-rel-28_43_2026-MCX750500B-
↪0D00_Ax_Bx-UEFI-14.36.21-FlexBoot-3.7.500.signed.bin
sudo mlxfwmanager -d c0:00.0 -i fw-ConnectX7-rel-28_43_2026-MCX750500B-
↪0D00_Ax_Bx-UEFI-14.36.21-FlexBoot-3.7.500.signed.bin
sudo mlxfwmanager -d 18:00.0 -i fw-ConnectX7-rel-28_43_2026-MCX750500B-
↪0D00_Ax_Bx-UEFI-14.36.21-FlexBoot-3.7.500.signed.bin
sudo mlxfwmanager -d 40:00.0 -i fw-ConnectX7-rel-28_43_2026-MCX750500B-
↪0D00_Ax_Bx-UEFI-14.36.21-FlexBoot-3.7.500.signed.bin
sudo mlxfwmanager -d 4f:00.0 -i fw-ConnectX7-rel-28_43_2026-MCX750500B-
↪0D00_Ax_Bx-UEFI-14.36.21-FlexBoot-3.7.500.signed.bin
sudo mlxfwmanager -d ce:00.0 -i fw-ConnectX7-rel-28_43_2026-MCX750500B-
↪0D00_Ax_Bx-UEFI-14.36.21-FlexBoot-3.7.500.signed.bin
sudo mlxfwmanager -d 9a:00.0 -i fw-ConnectX7-rel-28_43_2026-MCX750500B-
↪0D00_Ax_Bx-UEFI-14.36.21-FlexBoot-3.7.500.signed.bin
```

4. Update the firmware on the cards that are used for storage communication:

```
sudo mlxfwmanager -d aa:00.0 -i fw-ConnectX7-rel-28_43_2026-MCX755206AS-
↪NEA_Ax-UEFI-14.36.21-FlexBoot-3.7.500.signed.bin
```

(continues on next page)

(continued from previous page)

```
sudo mlxfwmanager -d 29:00.0 -i fw-ConnectX7-rel-28_43_2026-MCX755206AS-
↪NEA_Ax-UEFI-14.36.21-FlexBoot-3.7.500.signed.bin
```

5. Perform an AC power cycle on the system for the firmware update to take effect.

   Wait for the operating system to boot.

6. After the system starts, log in and confirm the firmware versions are all the same:

```
$ cat /sys/class/infiniband/mlx5_*/fw_ver
```

Chapter 22. Updating the Intel NIC Firmware

The following instructions describe how to update the firmware on the Intel Ethernet Network Adapter E810-C using interactive mode.

1. Download the update package to a temporary directory.
2. Start the update by running the Intel Ethernet NVM Update Tool `nvmupdate64e`.

   ```
   sudo ./nvmupdate64e
   ```

3. Follow the prompts to update the NVM image on the network adapter.

   Example output:

```
$ sudo ./nvmupdate64e

Intel(R) Ethernet NVM Update Tool
NVMUpdate version 1.41.3.3
Copyright(C) 2013 - 2024 Intel Corporation.


WARNING: To avoid damage to your device, do not stop the update or reboot
 ↪or power off the system during this update.
Inventory in progress. Please wait [***.......]


Num Description                          Ver.(hex)  DevId S:B    Status
=== ==================================== ============ ===== ======
 ↪==============
01) Intel(R) Ethernet Controller          N/A(N/A)   1563 00:011 Update
 ↪not
    X550-T2                                                     available
02) Intel(R) Ethernet Network Adapter   2.80(2.50)   1592 00:130 Update
    E810-C-Q2                                                   available

Options: Adapter Index List (comma-separated), [A]ll, e[X]it
Enter selection: 02
Would you like to back up the NVM images? [Y]es/[N]o: Y
Update in progress. This operation may take several minutes.
[*+........]
```

(continues on next page)

(continued from previous page)

```
Num Description                          Ver.(hex)  DevId S:B    Status
=== ================================== ============ ===== ======
 ↪==============
01) Intel(R) Ethernet Controller          N/A(N/A)   1563 00:011 Update
 ↪not
    X550-T2                                                    available
02) Intel(R) Ethernet Network Adapter   2.80(2.50)   1592 00:130 Update
    E810-C-Q2
 ↪successful

A reboot is required to complete the update process.

Tool execution completed with the following status: All operations
 ↪completed successfully.
Press any key to exit.
```

4. Perform an AC power cycle on the system for the firmware update to take effect.

   Wait for the operating system to boot.

Chapter 23. Firmware Update of GPU Tray: All Components

Perform the following steps to update the firmware on all the components in the GPU tray, such as GPUs, NVSwitches, and so on. Ensure the system is powered on before updating the firmware.

1. Create a `gpu_tray.json` file with the following contents:

```
{
    "Targets" :["/redfish/v1/UpdateService/FirmwareInventory/HGX_0"]
}
```

2. Update the firmware:

```
nvfwupd -t ip=<bmc-ip-address> user=<bmc-username> password=<bmc-password>
 ↪ update_fw \
  -p nvfw_HGX_DGXH100-H200x8_250319.1.0.fwpkg -y -s gpu_tray.json
```

Example Output

```
FW recipe: ['nvfw_HGX100x8_002_230705.1.1_prod-signed.fwpkg']
{"@odata.type": "#UpdateService.v1_6_0.UpdateService", "Messages": [{
 ↪"@odata.type": "#Message.v1_0_8.Message", "Message": "A new task /
 ↪redfish/v1/TaskService/Tasks/3 was created.", "MessageArgs": ["/redfish/
 ↪v1/TaskService/Tasks/3"], "MessageId": "Task.1.0.New", "Resolution":
 ↪"None", "Severity": "OK"}, {"@odata.type": "#Message.v1_0_8.Message",
 ↪"Message": "The action UpdateService.MultipartPush was submitted to do
 ↪firmware update.", "MessageArgs": ["UpdateService.MultipartPush"],
 ↪"MessageId": "UpdateService.1.0.StartFirmwareUpdate", "Resolution":
 ↪"None", "Severity": "OK"}]}
 FW update started, Task Id: 3
Wait for Firmware Update to Start...
Wait for Firmware Update to Start...
 Started Updating: HGX_0
 TaskState: Running
 PercentComplete: 20
 TaskStatus: OK
 TaskState: Running
 PercentComplete: 40
 TaskStatus: OK
 TaskState: Completed
 PercentComplete: 100
```

(continues on next page)

(continued from previous page)

```
 TaskStatus: OK
Firmware update successful!
 Overall Time Taken: 0:09:14
Refer to the DGX H100/H200 Firmware Update Guide on activation steps for
 ↪new firmware to take effect.
```

Chapter 24. Updating Multiple Systems

24.1. About Updating Multiple Systems

You can use the `nvfwupd` command with the `-t targets=<json-file>` argument and a JSON file to update the firmware on multiple systems. The update is performed serially on the systems.

Refer to the following sample `targets.json` file:

```
[
   {"ip": "192.168.1.10", "user": "admin", "password": "admin"},
   {"ip": "192.168.1.20", "user": "admin", "password": "admin"},
   {"ip": "192.168.1.30", "user": "admin", "password": "admin"}
]
```

You can specify a host name instead of the IP address in the `ip` field.

When you specify the `-t targets=<json-file>` argument, the following arguments to the `nvfwupd` command are not supported:

- `--background`
- `show_update_progress`

24.2. Procedure

- Run the `nvfwupd` command and specify the `-t targets=<json-file>` argument:

```
$ nvfwupd -t targets=../targets.json update_fw -s updparams.json -y -p \
    nvfw_DGXH100_xxxx_xxxxxx.x.x.fwpkg
```

Example Output

```
Updating ip address: ip=192.168.1.10

FW recipe: ['nvfw_DGXH100_xxxx_xxxxxx.x.x_custom_prod-signed.fwpkg']
{"@odata.type": "#UpdateService.v1_6_0.UpdateService", "Messages": [
{"@odata.type": "#Message.v1_0_8.Message", "Message": "A new task
/redfish/v1/TaskService/Tasks/6 was created.", "MessageArgs":
["/redfish/v1/TaskService/Tasks/6"], "MessageId": "Task.1.0.New",
"Resolution": "None", "Severity": "OK"}, {"@odata.type": "#Message.v1_0_8.
↪Message",
```

(continues on next page)

(continued from previous page)

```
"Message": "The action UpdateService.MultipartPush was submitted to do
 ↪firmware update.",
"MessageArgs": ["UpdateService.MultipartPush"],
"MessageId": "UpdateService.1.0.StartFirmwareUpdate", "Resolution": "None
 ↪", "Severity": "OK"}]}
FW update started, Task Id: 6
Wait for Firmware Update to Start...
Wait for Firmware Update to Start...
 TaskState: Completed
 PercentComplete: 100
 TaskStatus: OK
Firmware update successful!
 Overall Time Taken: 0:00:08
Refer to 'DGX H100 Firmware Update Document' on activation steps for new
 ↪firmware to take effect.
------------------------------------------------------------------------
 ↪-----------------------

Updating ip address: ip=192.168.1.20
...
```

Chapter 25. Performing a BMC Factory Reset

- Reset the BMC to factory default values:

```
nvfwupd --target ip=<bmc-ip-address> user=admin password=admin perform_
↪factory_reset
```

Example Output

```
Factory Reset request successful
Task State:
{"@odata.context": "/redfish/v1/$metadata#Task.Task", "@odata.id": "/
↪redfish/v1/TaskService/Tasks/2", "@odata.type": "#Task.v1_4_2.Task",
↪"Description": "Task for Manager ResetToDefaults", "Id": "2", "Name":
↪"Manager ResetToDefaults", "TaskState": "New"}
```

Chapter 26. Forcing a Firmware Downgrade

26.1. Prerequisites

- Refer to *Viewing the Installed Firmware and Package Versions* to confirm that the firmware package has the firmware version that you want.

26.2. Procedure

1. Enable the `ForceUpdate` flag on the BMC:

```
nvfwupd --target ip=<bmc-ip-address> user=admin password=admin force_
↪update enable
```

Example Output

```
ForceUpdate flag was successfully set True on the system.
```

2. Confirm the `ForceUpdate` flag status:

```
nvfwupd --target ip=<bmc-ip-address> user=admin password=admin force_
↪update status
```

Example Output

```
ForceUpdate is set to True
```

3. Perform the firmware update.
4. Disable the `ForceUpdate` flag on the BMC:

```
nvfwupd --target ip=<bmc-ip-address> user=admin password=admin force_
↪update disable
```

Example Output

```
ForceUpdate flag was successfully set False on the system.
```

5. Confirm the `ForceUpdate` flag status:

```
nvfwupd --target ip=<bmc-ip-address> user=admin password=admin force_
↪update status
```

Example Output

```
ForceUpdate is set to False
```

Chapter 27. Troubleshooting an Unsuccessful Firmware Update

27.1. Firmware Update Terminates due to Component Not Found

When performing a firmware update of the GPU tray with the motherboard firmware package, the firmware update stops with the following output message:

```
...
{
  "@odata.type": "#Message.v1_0_8.Message",
  "Message": "Given PLDMBundle Status Message : Requested component was not
↪found in the firmware bundle.",
  "MessageArgs": [
    "Requested component was not found in the firmware bundle."
  ],
  "MessageId": "UpdateService.1.0.FwUpdateStatusMessage",
  "Resolution": "None",
  "Severity": "Warning"
},
...
```

The message indicates that the firmware file specified by the `-p` argument of the `nvfwupd` command is invalid. Retry the update and specify the firmware file that matches the component. For example, use the GPU firmware file, which contains the HGX string, for the GPU tray update. Refer to *Version 24.09.1* for the firmware file names and components.

27.2. No Devices Where Detected for Handle ID 0

When performing a firmware update with the Redfish API, the following output message indicates that the firmware file specified in the `-F UpdateFile=` argument is not the correct file for the component specified in the JSON file.

```
...
{
  "@odata.type": "#Message.v1_0_8.Message",
  "Message": "Given PLDMBundle Status Message : No devices where detected for
↪handle id 0.",
  "MessageArgs": [
    "No devices where detected for handle id 0"
  ],
  "MessageId": "UpdateService.1.0.FwUpdateStatusMessage",
  "Resolution": "None",
  "Severity": "Warning"
},
...
```

Retry the update and specify the firmware file that matches the component. Refer to *Version 1.1.3* for the firmware file names and components. Refer to Redfish APIs Support in the *NVIDIA DGX H100 and H200 System User Guide* for information about using the Redfish API.

27.3. Wait for Firmware Update Started ID

The output for an unsuccessful firmware update using the `nvfwupd` command can look like the following example:

```
FW recipe: ['nvfw_DGXH100_xxxx_xxxxxx.x.x.fwpkg']
{"@odata.type": "#UpdateService.v1_6_0.UpdateService", "Messages": [{"@odata.
↪type": "#Message.v1_0_8.Message", "Message": "A new task /redfish/v1/
↪TaskService/Tasks/4 was created.", "MessageArgs": ["/redfish/v1/TaskService/
↪Tasks/4"], "MessageId": "Task.1.0.New", "Resolution": "None", "Severity":
↪"OK"}, {"@odata.type": "#Message.v1_0_8.Message", "Message": "The action
↪UpdateService.MultipartPush was submitted to do firmware update.",
↪"MessageArgs": ["UpdateService.MultipartPush"], "MessageId": "UpdateService.
↪1.0.StartFirmwareUpdate", "Resolution": "None", "Severity": "OK"}]}
FW update started, Task Id: 4

Wait for FirmwareUpdateStarted Id in Messages
Wait for FirmwareUpdateStarted Id in Messages
 Task Message: Task /redfish/v1/UpdateService/upload has stopped due to an
↪exception condition.
Firmware update failed, retry the firmware update
```

Retry the firmware update, as indicated in the command output.

Chapter 28. Version 25.04.3

28.1. Highlights

- Improved handling for firmware update of PSUs.
- Improved reporting on ERoT status.
- Provided multiple stability fixes.
- Improved user management in Redfish.
- Enhanced stability in communication between the BMC and the HMC.

28.1.1. BMC Fixes

- Corrected power supply sensors where no readings were shown.
- Fixed the inability to query GPU firmware version via Redfish.
- Addressed an issue where some telemetry information from the GPU tray was unavailable.
- Resolved an issue related to chassis power cycling using Redfish.
- Corrected fan readings so the fan will reflect 0 if it is not present or stops spinning.
- Fixed an issue where NIC information was allocated to the incorrect device in Redfish.

28.1.2. SBIOS Fixes

- Fixed an issue where the total memory size from the `free -m` command output is smaller than installed.
- Updated to display full device path for HDD devices under the Redfish BootOptions page.
- Fixed an issue to mandate administrator password after version 24.09.1 if SED drive is encrypted.

28.1.3. The nvfwupd Command Updates

- Added support for parallel firmware updates through the YAML configuration file.
- Added the `--json` option to the `update_fw`, `show_update_progress`, and `force_update` commands.
- Added IPv6 support.

- Deprecated the `targets` sub-option for multi-target input. Use `config.yaml` input instead.

28.2. Firmware Package Details

This firmware release supports the following systems:

- NVIDIA DGX H100
- NVIDIA DGX H200

This firmware release supports the following operating systems:

- NVIDIA DGX OS 7.0.2, 6.2.1, 6.1, 6.0.11, and higher
- NVIDIA DGX Software EL9-24.12, EL9-24.06, EL9-23.12, and EL9-23.08
- NVIDIA DGX Software EL8-24.07, EL8-24.01, and EL8-23.08

For more information about the operating systems, refer to the NVIDIA Base OS documentation.

You can download firmware packages from the NVIDIA Enterprise Support Portal.

The following table shows the firmware package files:

Components	Sample File Name
Combined archive	`DGXH100_H200_25.04.3.tar` The combined archive includes the firmware for the system components and the firmware for the GPU tray.
- Motherboard tray package - GPU tray package	- `nvfw_DGX_250220.1.0.fwpkg` - `nvfw_HGX_DGXH100-H200x8_250131.1.0.fwpkg`

If you are updating from version 24.09.1, the total update time is approximately

- 91 minutes for the CPU tray using sequential updating.
- 34 minutes for the CPU tray using parallel updating.
- 15 minutes for the GPU tray using parallel updating.

The following table shows the information about component firmware versions and update time breakdown.

Component	Version	Update Time from 24.09.1 (Minutes)
Host BMC	**25.02.12** Refer to *BMC Changes for DGX H100/H200 Systems* for the list of changes.	28
Host BMC ERoT	**04.0058**	2.75
SBIOS ERoT	**04.0058**	2.75
SBIOS	**1.6.7** Refer to *SBIOS Changes for DGX H100/H200 Systems* for the list of changes.	8.75
Motherboard CPLD	**0.2.1.9**	18
Midplane CPLD	**0.2.1.3**	13.25
PSU (Delta ECD16020137)	Primary 0204 Secondary 0201 Community **0204**	PSU_0: 2 PSU_1: 2 PSU_2: 1.87 PSU_3: 1.87 PSU_4: 1.87 PSU_5: 1.87
Broadcom Gen5 PCIe Switch (PEX89072-B01)	Switch 0: 0.0.7 Switch 1: 1.0.7	Switch 0: 0.75 Switch 1: 0.75
Astera Labs Gen5 PCIe Retimer (PT5161L)	2.07.19	Retimer 0: 2.5 Retimer 1: 2
Network (Cluster) Card - ConnectX-7	**28.43.2026**	
Network (Storage) Card - ConnectX-7	**28.43.2026**	
Network Card - BlueField-3	**32.43.2024**	
VBIOS	► H100: **96.00.CF.00.01** ► H200: **96.00.CF.00.02**	GPU Tray (total): 15
NVSwitch (GPU Tray)	**96.10.6D.00.01**	
ERoT (GPU Tray)	**02.0192**	
HMC (GPU Tray)	**HGX-22.10-1-rc79**	
FPGA (GPU Tray)	2.53	

28.3. Firmware Update Procedure

Refer to *Firmware Update Steps*.

Chapter 29. Version 24.09.1

Note

Starting with this release, the versioning scheme of the DGX H100/H200 documentation has changed to a 5-digit version. For the new version, the first two digits are the current year followed by two digits of the month and one digit of the build number; for example, version 24.09.1 was the first build released in September 2024.

29.1. Highlights

29.1.1. Added Support

- Introducing support for the NVIDIA DGX H200 System.
- Enabled 3 + 3 power limiting feature to provide continual power source in the event of power distribution unit failure, but at a reduced performance level.
- Added Redfish API support for creating, modifying, and deleting power policies.
- Support for deploying firmware update using the Web UI.
- Redfish Disable Host Interface: keeps redfish functional from BIOS to BMC but prevents the direct path from OS to BMC.
- Added ability to specify intermediate certificate authorities in a provisioned certificate chain.
- Incorporates updated firmware for GPU tray, network, and NVMe drives.

29.1.2. BMC Fixes

- Fixed where BMC configuration might reset after upgrading.
- Included additional Redfish metrics reports.
- Fixed SNMP, syslog, and rsyslog issues.
- Added per BMC AES key for encrypting user/password files during the configuration save and restore process.
- Fixed invalid domain issues in the LDAP/AD settings.
- Enhanced Redfish diagnostics.
- General performance improvements in Redfish APIs and IPMI.

- Added support for ConnectX-7 temperature sensors.
- Improved resolution for energy counters.
- Enhanced Remote Media with support for port numbers and domain names.
- General improvements to the Web UI.

29.1.3. SBIOS Fixes

- DIMM that experienced uncorrectable errors at runtime will be mapped out on the next boot.
- Exposed the `C1AutoDemotion`, `C1AutoUnDemotion`, and `C6Enable` setup options.
- Moved the CPU setup options page to under the Advanced page in the setup UI.
- Added a setup option to restrict host access via IPMI.
- Provided the `NvramVarsProtectionInOs` setup option to prevent the OS from changing the NVRAM at runtime.
- Implemented uncorrectable error rate limiting, disabled CSMI (correctable system management interrupts) on error flooding and on the core that reported MLC (middle-level cache) yellow state, and SEL logging when ANF (advisory non-fatal error) threshold was crossed.
- Changed the SncEn default setting to `disable`.

29.1.4. The nvfwupd Command Updates

- Enhanced automatic server type detection for DGX platforms.
- No support for the `activate_fw` command.

29.2. Firmware Package Details

This firmware release supports the following systems:

- NVIDIA DGX H100
- NVIDIA DGX H200

This firmware release supports the following operating systems:

- NVIDIA DGX OS 6.2.1, 6.1, 6.0.11, and higher
- NVIDIA DGX Software EL9-24.06, EL9-23.12, and EL9-23.08
- NVIDIA DGX Software EL8-24.07, EL8-24.01, and EL8-23.08

For more information about the operating systems, refer to the NVIDIA Base OS documentation.

You can download firmware packages from the NVIDIA Enterprise Support Portal.

The following table shows the firmware package files:

Components	Sample File Name
Combined archive	`DGXH100_H200_24.09.1.tar` The combined archive includes the firmware for the system components and the firmware for the GPU tray.
▶ Motherboard tray package ▶ GPU tray package	▶ `nvfw_DGX_240918.1.0.fwpkg` ▶ `nvfw_HGX_DGXH100-H200x8_240603.1.0.fwpkg`

If you are updating from version 1.1.3, the total update time is approximately

- 92 minutes for the CPU tray using sequential updating.
- 34 minutes for the CPU tray using parallel updating.
- 12 minutes for the GPU tray using parallel updating.

The following table shows the information about component firmware versions and update time breakdown.

Component	Version	Update Time from 1.1.3 (Minutes)
Host BMC	**24.09.17** Refer to *BMC Changes for DGX H100/H200 Systems* for the list of changes.	25
Host BMC ERoT	**04.0052**	2
SBIOS ERoT	**04.0052**	2
SBIOS	**1.05.03** Refer to *SBIOS Changes for DGX H100/H200 Systems* for the list of changes.	7
Motherboard CPLD	0.2.1.8	19
Midplane CPLD	0.2.1.1	13
PSU (Delta ECD16020137)	Primary **0204** Secondary **0201** Community **0203**	PSU_0: 2.75 PSU_1: 2.75 PSU_2: 2.75 PSU_3: 2.75 PSU_4: 2.75 PSU_5: 2.75
Broadcom Gen5 PCIe Switch (PEX89072-B01)	Switch 0: 0.0.7 Switch 1: 1.0.7	Switch 0: 1 Switch 1: 1
Astera Labs Gen5 PCIe Retimer (PT5161L)	2.07.19	Retimer 0: 3 Retimer 1: 2.5
Network (Cluster) Card - ConnectX-7	**28.39.3560**	
Network (Storage) Card - ConnectX-7	**28.39.3560**	
Network Card - BlueField-3	**32.40.1000**	
▶ VBIOS (H100 80GB) ▶ VBIOS (H200 141GB)	▶ **96.00.A5.00.01** ▶ **96.00.A5.00.03**	GPU Tray (total): 12
NVSwitch (GPU Tray)	**96.10.57.00.01**	
ERoT (GPU Tray)	**02.0182**	
HMC (GPU Tray)	**HGX-22.10-1-rc67**	
FPGA (GPU Tray)	**2.53**	
PCIe Switch (GPU Tray)	**1.9.5F**	

29.3. Firmware Update Procedure

Refer to *Firmware Update Steps*.

Chapter 30. Version 1.1.3

30.1. Highlights

- Added support
  - Support for Gen5 NVME drives.
  - U.2 drive temperature sensor fix.
  - Updated power supply firmware.
  - Included the latest GPU tray firmware.
  - Included the latest network (cluster and storage) card firmware.
  - Added support for securing KCS.
- The `nvfwupd` command is updated with the following enhancements:
  - Support for abbreviated firmware update package names.
  - Enhanced the `show_update_progress` output to provide a full status report for Redfish.
  - Support for custom log file path.
  - The command exits with an error code 1 for any update failure or tool failure.

30.2. BMC Fixes

- Fixed where SEL logs might fill up for NVMe drives.
- Fixed low occurrence where HMC might not be visible in the BMC after BMC reboot.
- Added ability to control IPMI visibility for Host (Allow All, Limited Command, Hide).
- Higher resolution for CPU and GPU energy telemetry via Redfish.
- Improved reliability of Redfish inventory.
- Improved overall stability of telemetry collection and handling invalid/missing values.
- General improvements to WebUI.

30.3. Firmware Package Details

This firmware release supports the following hardware:

- NVIDIA DGX H100

This firmware release supports the following operating systems:

- NVIDIA DGX OS 6.1, 6.0.11, and higher
- NVIDIA DGX Software for EL9.2, 23.12 and 23.08
- NVIDIA DGX Software for EL8 23.08

Refer to the NVIDIA Base OS documentation for more information about the operating systems.

You can download firmware packages from the NVIDIA Enterprise Support Portal at https://enterprise-support.nvidia.com/s/.

Download two firmware package files:

Components	Sample File Name
Combined Archive	`DGXH100_1.1.3.tar` The combined archive includes the firmware for the system components, firmware for the GPU tray, and the *nvfwupd* executable.
Motherboard Tray	`nvfw_DGXH100_231206.1.0.fwpkg`
GPU Tray	`nvfw_HGX_DGXH100_231101.1.0.fwpkg`

If you are updating from 1.1.1, the total update time is approximately

- 88 minutes for the CPU tray using sequential updates.
- 33 minutes for the CPU tray using parallel updates.
- 11 minutes for the GPU tray using parallel updates.

The following table shows the information about component firmware versions and update time breakdown.

Component	Version	Update time from 1.1.1 (minutes)
Host BMC	**24.01.05** Refer to *DGX H100 System BMC Changes* for the list of changes.	25
Host BMC EROT	04.0026	2
SBIOS EROT	04.0026	0
SBIOS	**v1.01.03** Refer to *DGX H100 System SBIOS Changes* for the list of changes.	7
Motherboard CPLD	0.2.1.8	18
Midplane CPLD	**0.2.1.1**	14
PSU (Delta ECD16020137)	Primary **2.4** Secondary **2.1** Community **2.2**	PSU_0: 2 PSU_1: 2 PSU_2: 2 PSU_3: 2 PSU_4: 2 PSU_5: 2
Broadcom Gen5 PCIe Switch (PEX89072-B01)	Switch 0: **v0.0.7** Switch 1: **v1.0.7**	Switch 0: 1 Switch 1: 1
Astera Labs Gen5 PCIe Retimer (PT5161L)	**v2.07.19**	Retimer 0: 3 Retimer 1: 3
Network (Cluster) Card - ConnectX-7	**v28.39.1002**	
Network (Storage) Card - ConnectX-7	**v28.39.1002**	
VBIOS (H100 80GB)	**96.00.89.00.01**	GPU Tray (total): 11
NVSwitch (GPU Tray)	**96.10.4A.00.01**	
EROT (GPU Tray)	**02.0150**	
HMC (GPU Tray)	**HGX-22.10-1-rc57**	
FPGA (GPU Tray)	**2.37**	
PCIe Switch (GPU Tray)	1.7.5F	
30.3. Firmware Package Details	**2.07.19**	**99**
Astera Labs Gen5 PCIe Retimer (GPU Tray) (PT5161L)		

30.4. Firmware Update Procedure

Refer to *Firmware Update Steps*.

Chapter 31. Version 1.1.1

31.1. Features

- The `nvfwupd` command is updated with the following enhancements:
  - You can update all the system components on the motherboard tray at one time. Previously, you had to update the components individually.
  - You can create a JSON file with network addresses and credentials for multiple systems and automatically update multiple systems serially. Refer to *Updating Multiple Systems* for more information.
- Various performance enhancements.
- Enhancements to Redfish and new documentation to clear the BIOS and reset the BIOS to factory defaults. Refer to the *NVIDIA DGX H100/H200 User Guide* for more information.
- Sensor enhancements
- SEL logging improvements
- Improved firmware update times
- KVM enhancements

31.2. BMC Fixes

- WebUI enhancements
- Enabled GPU Info in WebUI
- Enabled NVRAM clear via Redfish
- Disabled RMCP / MD5 Auth Support after factory reset
- Enabled EROT background copy
- Enabled default SNMPv3 MIB
- The BMC update includes software security enhancements. Refer to the NVIDIA DGX H100 - August 2023 Security Bulletin for details.

31.3. SBIOS Fixes

- Fixed Boot options labeling for NIC ports
- Fix for U.2 bay slot numbering
- Set RestoreROWritePerf option to expert mode only
- Expose TDX and IFS options in expert user mode only

31.4. Firmware Package Details

This firmware release supports the following hardware:

- NVIDIA DGX H100

This firmware release supports the following operating systems:

- NVIDIA DGX OS 6.0.11 and higher

You can download firmware packages from the NVIDIA Enterprise Support Portal at https://enterprise-support.nvidia.com/s/.

Download two firmware package files:

Components	Sample File Name
Combined Archive	`DGXH100_1.1.1.tar` The combined archive includes the firmware for the system components, firmware for the GPU tray, and the *nvfwupd* executable.
Motherboard Tray	`nvfw_DGXH100_230920.1.0.fwpkg`
GPU Tray	`nvfw_HGX100x8_002_230705.1.1_prod-signed.fwpkg`

Refer to the following table for information about component firmware versions:

Component	Version
Host BMC	23.09.20 Refer to *DGX H100 System BMC Changes* for the list of changes.
Host BMC EROT	04.0026
SBIOS EROT	04.0026 Refer to *DGX H100 System SBIOS Changes* for the list of changes.
SBIOS	v1.01.01
Motherboard CPLD	0.2.1.8
Midplane CPLD	0.2.1.0
PSU (Delta ECD16020137)	Primary 2.2 Secondary 2.1 Community 2.2
Broadcom Gen5 PCIe Switch (PEX89072-B01)	Switch 0: v0.0.6 Switch 1: v1.0.6
Astera Labs Gen5 PCIe Retimer (PT5161L)	v1.30.12
Network Controller	ConnectX-7 Package: 5.9-0.5.6.0.127 Firmware: v28.36.2050
VBIOS (H100 80GB)	96.00.74.00.01
NVSwitch (GPU Tray)	96.10.3F.00.01
EROT (GPU Tray)	02.0134
HMC (GPU Tray)	HGX-22.10-1-rc44
FPGA (GPU Tray)	2.2C
PCIe Switch (GPU Tray)	1.7.5F
Astera Labs PCIe Retimer (GPU Tray) (PT5161L)	2.7.9
Intel 10G Ethernet	v3.60
Intel 50G Ethernet	v2.5
M.2 NVMe	GDC7502Q

31.5. Firmware Update Procedure

Refer to *Firmware Update Steps*.

Chapter 32. Version 1.0.0

Version 1.0.0 is the initial firmware release for NVIDIA DGX H100 Systems.

32.1. Firmware Package Details

This firmware release supports the following hardware:

- NVIDIA DGX H100

This firmware release supports the following operating systems:

- NVIDIA DGX OS 6.0.11 and higher

You can download firmware packages from the NVIDIA Enterprise Support Portal at https://enterprise-support.nvidia.com/s/.

Download two firmware package files:

Components	Sample File Name
System Components	`nvfw_DGX-H100_0003_230310.1.0_custom_prod-signed.fwpkg`
GPU Tray	`nvfw_DGX-HGX-H100x8_0000_230310.1.0_prod-signed.fwpkg`

Refer to the following table for information about component firmware versions:

Component	Version
Host BMC	23.05.11
Host BMC EROT	04.0015
SBIOS EROT	04.0015
SBIOS	v1.00.07
Motherboard CPLD	0.2.1.6
Midplane CPLD	0.2.0.7
PSU (Delta ECD16020137)	Primary 2.2 Secondary 2.1 Community 2.2
Broadcom Gen5 PCIe Switch (PEX89072-B01)	Switch 0: v0.0.6 Switch 1: v1.0.6
Astera Labs Gen5 PCIe Retimer (PT5161L)	v1.30.12
Network Controller	ConnectX-7 Package: 5.9-0.5.6.0.113 Firmware: v28.36.2024
VBIOS (H100 80GB)	96.00.61.00.01
NVSwitch (GPU Tray)	96.10.35.00.01
EROT (GPU Tray)	02.0134
HMC (GPU Tray)	HGX-22.10-1-rc34
FPGA (GPU Tray)	2.11
PCIe Switch (GPU Tray)	1.7.5F
Astera Labs PCIe Retimer (GPU Tray) (PT5161L)	2.4.7
Intel 10G Ethernet	v3.60
Intel 50G Ethernet	v2.5
M.2 NVMe (Samsung PM9A3)	GDC7302Q
U.2 Kioxia CM6	1.0.7

Chapter 33. Third-Party License Notices

This NVIDIA product contains third party software that is being made available to you under their respective open source software licenses. Some of those licenses also require specific legal information to be included in the product. This section provides such information.

33.1. Micron msecli

The msecli utility is provided under the following terms:

Micron Technology, Inc. Software License Agreement PLEASE READ THIS LICENSE AGREEMENT ("AGREEMENT") FROM MICRON TECHNOLOGY, INC. ("MTI") CAREFULLY: BY INSTALLING, COPYING OR OTHERWISE USING THIS SOFTWARE AND ANY RELATED PRINTED MATERIALS ("SOFTWARE"), YOU ARE ACCEPTING AND AGREEING TO THE TERMS OF THIS AGREEMENT. IF YOU DO NOT AGREE WITH THE TERMS OF THIS AGREEMENT, DO NOT INSTALL THE SOFTWARE. LICENSE: MTI hereby grants to you the following rights: You may use and make one (1) backup copy the Software subject to the terms of this Agreement. You must maintain all copyright notices on all copies of the Software. You agree not to modify, adapt, decompile, reverse engineer, disassemble, or otherwise translate the Software. MTI may make changes to the Software at any time without notice to you. In addition MTI is under no obligation whatsoever to update, maintain, or provide new versions or other support for the Software. OWNERSHIP OF MATERIALS: You acknowledge and agree that the Software is proprietary property of MTI (and/or its licensors) and is protected by United States copyright law and international treaty provisions. Except as expressly provided herein, MTI does not grant any express or implied right to you under any patents, copyrights, trademarks, or trade secret information. You further acknowledge and agree that all right, title, and interest in and to the Software, including associated proprietary rights, are and shall remain with MTI (and/or its licensors). This Agreement does not convey to you an interest in or to the Software, but only a limited right to use and copy the Software in accordance with the terms of this Agreement. The Software is licensed to you and not sold.

DISCLAIMER OF WARRANTY:

THE SOFTWARE IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND. MTI EXPRESSLY DISCLAIMS ALL WARRANTIES EXPRESS ORmIMPLIED, INCLUDING BUT NOT LIMITED TO, NONINFRINGEMENT OF THIRD PARTY RIGHTS, AND ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE. MTI DOES NOT WARRANT THAT THE SOFTWARE WILL MEET YOUR REQUIREMENTS, OR THAT THE OPERATION OF THE SOFTWARE WILL BE UNINTERRUPTED OR ERROR-FREE. FURTHERMORE, MTI DOES NOT MAKE ANY REPRESENTATIONS REGARDING THE USE OR THE RESULTS OF THE USE OF THE SOFTWARE IN TERMS OF ITS CORRECTNESS, ACCURACY, RELIABILITY, OR OTHERWISE. THE ENTIRE RISK ARISING OUT OF USE OR PERFORMANCE OF THE SOFTWARE REMAINS WITH YOU. IN NO EVENT SHALL MTI, ITS AFFILIATED COMPANIES OR THEIR SUPPLIERS BE LIABLE FOR ANY DIRECT, INDIRECT, CONSEQUENTIAL, INCIDENTAL, OR SPECIAL DAMAGES (INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF PROFITS, BUSINESS INTERRUPTION, OR LOSS OF INFORMATION) ARISING OUT OF YOUR USE OF OR INABILITY TO USE THE SOFTWARE, EVEN IF MTI

HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. Because some jurisdictions prohibit the exclusion or limitation of liability for consequential or incidental damages, the above limitation may not apply to you.

TERMINATION OF THIS LICENSE: MTI may terminate this license at any time if you are in breach of any of the terms of this Agreement. Upon termination, you will immediately destroy all copies the Software.

GENERAL: This Agreement constitutes the entire agreement between MTI and you regarding the subject matter hereof and supersedes all previous oral or written communications between the parties. This Agreement shall be governed by the laws of the State of Idaho without regard to its conflict of laws rules.

CONTACT: If you have any questions about the terms of this Agreement, please contact MTI's legal department at (208) 368-4500. By proceeding with the installation of the Software, you agree to the terms of this Agreement. You must agree to the terms in order to install and use the Software.

33.2. Mellanox (OFED)

MLNX_OFED <http://www.mellanox.com/> is provided under the following terms:

Copyright (c) 2006 Mellanox Technologies. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Chapter 34. Notices

34.1. 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.

34.2. Trademarks

NVIDIA, the NVIDIA logo, DGX, DGX-1, DGX-2, DGX A100, DGX H100/H200, DGX Station, and DGX Station A100 are trademarks and/or registered trademarks of NVIDIA Corporation in the Unites States and other countries. Other company and product names may be trademarks of the respective companies with which they are associated.

Copyright

©2022-2025, NVIDIA Corporation